4-23-2013 12:00 AM

Object-based Urban Building Footprint Extraction and 3D Building Reconstruction from Airborne LiDAR Data

Ting Zhao
The University of Western Ontario

Supervisor
Dr. Jinfei Wang
The University of Western Ontario

Graduate Program in Geography
A thesis submitted in partial fulfillment of the requirements for the degree in Master of Science

OBJECT-BASED URBAN BUILDING FOOTPRINT EXTRACTION AND 3D BUILDING RECONSTRUCTION FROM AIRBORNE LIDAR DATA

(Thesis format: Integrated Article)

By

Ting Zhao

Graduate Program in Geography

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science

The School of Graduate and Postdoctoral Studies
The University of Western Ontario
London, Ontario, Canada

Abstract

Buildings play an essential role in urban intra-construction, urban planning, climate studies and disaster management. The precise knowledge of buildings not only serves as a primary source for interpreting complex urban characteristics, but also provides decision makers with more realistic and multidimensional scenarios for urban management. In this thesis, the 2D extraction and 3D reconstruction methods are proposed to map and visualize urban buildings. Chapter 2 presents an object-based method for extraction of building footprints using LiDAR derived NDTI (Normalized Difference Tree Index) and intensity data. The overall accuracy of 94.0% and commission error of 6.3% in building extraction is achieved with the Kappa of 0.84. Chapter 3 presents a GIS-based 3D building reconstruction method. The results indicate that the method is effective for generating 3D building models. The 91.4% completeness of roof plane identification is achieved, and the overall accuracy of the flat and pitched roof plane classification is 88.8%, with the user's accuracy of the flat roof plane 97.8% and pitched roof plane 100%.

Keywords

Building Footprints, LiDAR, multiple returns, NDTI, object-based classification, GIS, 3D Reconstruction

Acknowledgments

First and foremost, I would like to thank my supervisor, Dr. Jinfei Wang for her support on my research work and thesis writing. Without her thoughtful advice, constructive criticism and great encouragement, my research would not be completed. Her passion and enthusiasm for research and education inspire me very much and set a great example for my future career.

The research presented in my thesis is partially supported by an NSERC Discovery Grant awarded to. Dr. Wang. The LiDAR data were provided by the University of Western Ontario (UWO) and the Applied Geomatics Research Group, Centre of Geographic Sciences, Nova Scotia Community College (NSCC). Dr. Hopkinson (NSCC) and Dr. Smart (UWO) managed the acquisition of the LiDAR data. The data processing software LiDAR Analyst™ and eCognition 8.0.1 were provided by the State Key Laboratory of Earth Surface Processes and Resource Ecology, Beijing Normal University (BNU), China. Thank Dr. Peijun Shi and Dr. Chunyang He from BNU for their support of the research project.

Thank Chuiqing Zeng for the help of measuring height data in the field. Many thanks to all the people in Dr. Jinfei Wang's research group for their help and valuable suggestions on my research and the anonymous reviewers and audience in the conference for their helpful advice to improve this manuscript.

Table of Contents

List of Tables

List of Figures

List of Appendices

Chapter 1

Introduction

1.1 Research Context

Building footprints and three-dimensional (3D) building models have become one of the essential components in various scientific and engineering applications, such as urban development, urban planning and urban climate (Meng et al., 2009).Though part of the following paragraphs is repeated in Chapters 2 and 3, the importance of mapping and visualizing buildings in urban environments should not be overlooked.

Building footprints serve as one of the fundamental Geographic information System (GIS) data that can be used to estimate energy demand, quality of life, urban population, property taxes (Jensen, 2000) and generate 3D building models for visualization (Zhang et al., 2006). The 3D building models can be applied to digital city visualization, landscape modeling, virtual tourism information systems and wireless telecommunication (Zhang et al., 2009; Elberink and Vosselman, 2011; Kabolizade et al., 2012). Environmental protection and planning, microclimate study such as urban heat island effect and rain-runoff modeling (Davis, 2005; Miliaresis and Kokkas, 2007) or emergency response such as earthquake damage assessment (Kim and Shan, 2011) all require high accuracy 3D building models, considering the raising standards and the improvement of complex prediction systems. The precise knowledge of building footprint and 3D building models can be applied as primary sources for interpreting complex urban characteristics, and provide regional planners with more realistic and multidimensional scenarios for urban management (Zhou et al., 2009).

High resolution optical images (e.g. aerial photos, QuickBird, IKONOS, WorldView) have been and still are one of the preferred data sources to obtain 2D and 3D building information(Zhou and Troy, 2008). High quality images with resolution finer than 5 meters offer good opportunities for building information extraction (Shufelt and Mckeown, 1993; Strassopolou et al., 2000; Lee et al., 2003; Aldred and Wang, 2011). However, the success of building footprint extraction methods can be largely limited due to inadequate image understanding, sun shadows, relief displacement and occlusion effect presented in optical images which are without height information (Zhou and Troy, 2008; Meng et al., 2009, Wang et al., 2011).

Recently-developed airborne Light Detection And Ranging (LiDAR) technology provides a very promising alternative for building footprint measurement and 3D building modeling. LiDAR directly collects an accurately georeferenced set of dense point clouds by recording reflected laser beams emitted by the sensor. Elevation and the amplitude of the reflected energy are both stored to represent three-dimensional objects on the Earth surface (Shan and Toth, 2009). Compared with other data sources, LiDAR measurements are not only free from sun shadows, relief displacement and occlusion effect, but also offer several advantages such as fast data acquisition, high point density, strong capability of measuring both height and planimetric location and canopy penetration (Zhang et al., 2006; Beraldin et al., 2010). Though the cost of acquiring LiDAR data was quite high in the past, prices have dropped and image acquisition for research purposes has become affordable in recent years.

Two types of approaches are utilized to identify building footprints from LiDAR data in previous studies. One is to separate the buildings, trees, roads, grass and other land-use types from LiDAR data simultaneously. Pixel-based classifications (e.g. Maximum likelihood) are mostly seen in this category (Song et al., 2002; Arefi and Hahn, 2005; El-Ashmawy et al., 2011). Nevertheless, these pixel-based methods are not efficient for high resolution images, in which objects are represented by many pixels instead of mixed ones (Blaschke, 2010). The more commonly used way is to separate the ground from non-ground LiDAR measurements first and then identify buildings from non-ground measurements. The initial separation of the ground and non-ground measurements can be achieved by classifying ground points using morphological filters (Weidner, 1997; Morgan and Tempfli, 2000) and surface-based filters (also known as robust interpolation) (Rottensteiner, 2002). The critical step is to classify building and tree objects that dominate non-ground measurements (Zhang et al., 2006). Thresholding (Brunn and Weidner, 1997), morphological filters (Morgan and Tempfli, 2000; Goodwin et al., 2009), texture analysis (Rottensteiner, 2003; Ma, 2005) and the plane-fitting method (Zhang et al., 2006) have been mostly applied in previous studies. However, not all trees are detected since some dense tree canopies can be as smooth as topographic surfaces or roof patches, which lack textural patterns. Though object-based algorithm has been applied in several studies (Brennan and Webster, 2006; Huang et al., 2008), the efficiency of object-based classification in the differentiation between buildings and trees as well as building extraction in more complex urban environments should be examined.

The building footprints can be later used as the input data for constructing 3D building models. Traditionally, 3D building models can be directly built by manual interpretation and hand-drawing using high resolution imagery and auxiliary information such as building heights. Though the accuracy of the derived building models can be very high, it is excessively time-consuming and costly. Currently most reported LiDAR-based 3D building reconstruction approaches use an automatic or semi-automatic strategy of detecting planar facets for the determination of building primitives. The core step is to segment LiDAR data or building reference data into segments representing single planar facets. Various methods have been applied, including region growing (Rottensteiner, 2003), gradient orientation (Forlani et al., 2006), Hough transform (Vosselman and Dijkman, 2001) and RANSAC (RANdomSAmple Consensus) (Tarsha-Kurdi et al., 2007). However, there are some aspects to be improved such as efficiently detecting the disconnection of segmented roofs and locating jump edges of rooftops (Kim and Shan, 2011). Additionally, these approaches always need the assistance of a building's dominant orientation to determine roof primitives and edge directions. Though the aforementioned difficulties can be avoided by directly extracting the building edges within building footprints without segmenting roof planar facets, few studies can identify roof edges from LiDAR data without the assistance of other data sources such as 2D maps and aerial images.

1.2 Research Objectives

The main objectives of this research are to present effective methods for building footprint extraction and 3D building reconstruction, to describe the new techniques and to evaluate their effectiveness for improving building extraction and reconstruction. Specific questions regarding the research are presented as follows:

1. How accurately can building footprints be classified using an object-based method from LiDAR data in the complex urban environment?

2. What object-based parameters are effective for building footprint extraction based on LiDAR data?

3. Can 3D building models be identified and reconstructed from LiDAR data using GIS techniques?

4. What performance can GIS-based rooftop faces identification achieve?

The main objectives of this study are as follows:

1. To develop an object-based classification method based on the LiDAR elevation and intensity data to extract locations and boundaries of building footprints in a complex urban environment; to find effective parameters for building classification and examine their roles in the object-based method; to examine a best combination of rules for object-based classification that can be effectively used in building footprint extraction, and to evaluate the accuracy of the proposed method.

2. To develop a reconstruction method of 3D building models from LiDAR data using GIS techniques, to find a strategy of detecting and processing the edges within building boundaries and classifying the roof shape types, and to assess the performance of the edge detection and roof shape classification.

1.3 Study Area

The study area for Chapter 2 and Chapter 3 is the campus of the University of Western Ontario (Figure 1.1, in red), which is located in London, Ontario, Canada (Figure 1.1, in black). The LiDAR data used for the two chapters are provided by the University of Western Ontario and the Applied Geomatics Research Group, Center of Geographic Sciences, Nova Scotia Community College (NSCC). The LiDAR data cover the Medway Creek region of North London (Figure 1.1, in green). More detailed information about the study area can be found in the "Data Acquisition and Study area" sections of Chapter 2 and Chapter 3.

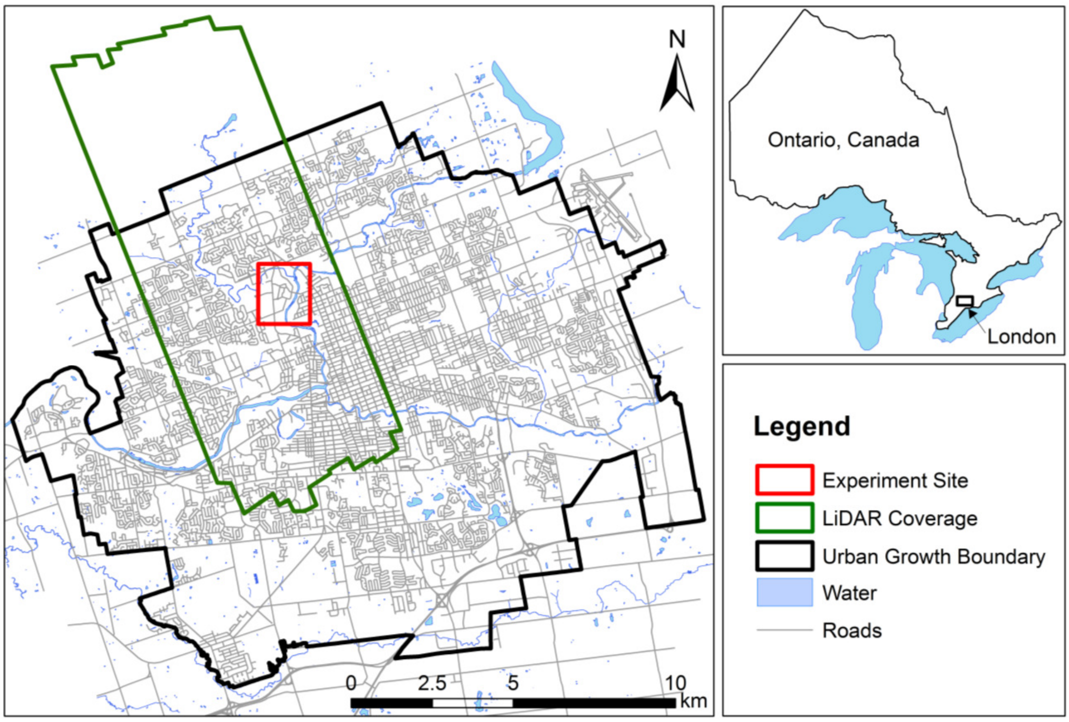


Figure 1.1 **Study area**

1.4 Background

1.4.1 LiDAR

LiDAR (Light Detection and Ranging) is an active remote sensing technique that has gained popularity since the 1990s. Unlike the commonly-used multispectral optical sensors, which take a snap-shot from a certain angle, LiDAR measurements are free from sun shadows, relief displacement and occlusion effect (Zhang et al. 2006). The general principle of LiDAR is summarized in Figure 1.2. A LiDAR sensor emits a laser beam to the target, the remaining energy of the laser, after interactions with the Earth surface, reflects back to the sensor. The time traveled during the process is recorded. The elevation of target objects can be calculated if the speed of laser and the aircraft altitude are both known. Simultaneously the amplitude of reflected energy is recorded as intensity data. Therefore, elevation and reflected intensity information are both stored in LiDAR to represent 3D objects on the Earth surface (Shan and Toth, 2009).The most attractive characteristic of LiDAR is that it has very high point density with at least 1 point / m^2, some LiDAR sensors are even capable of recording more than 40 points within 1 m^2. Moreover, the vertical accuracy of LiDAR is at millimeter level.

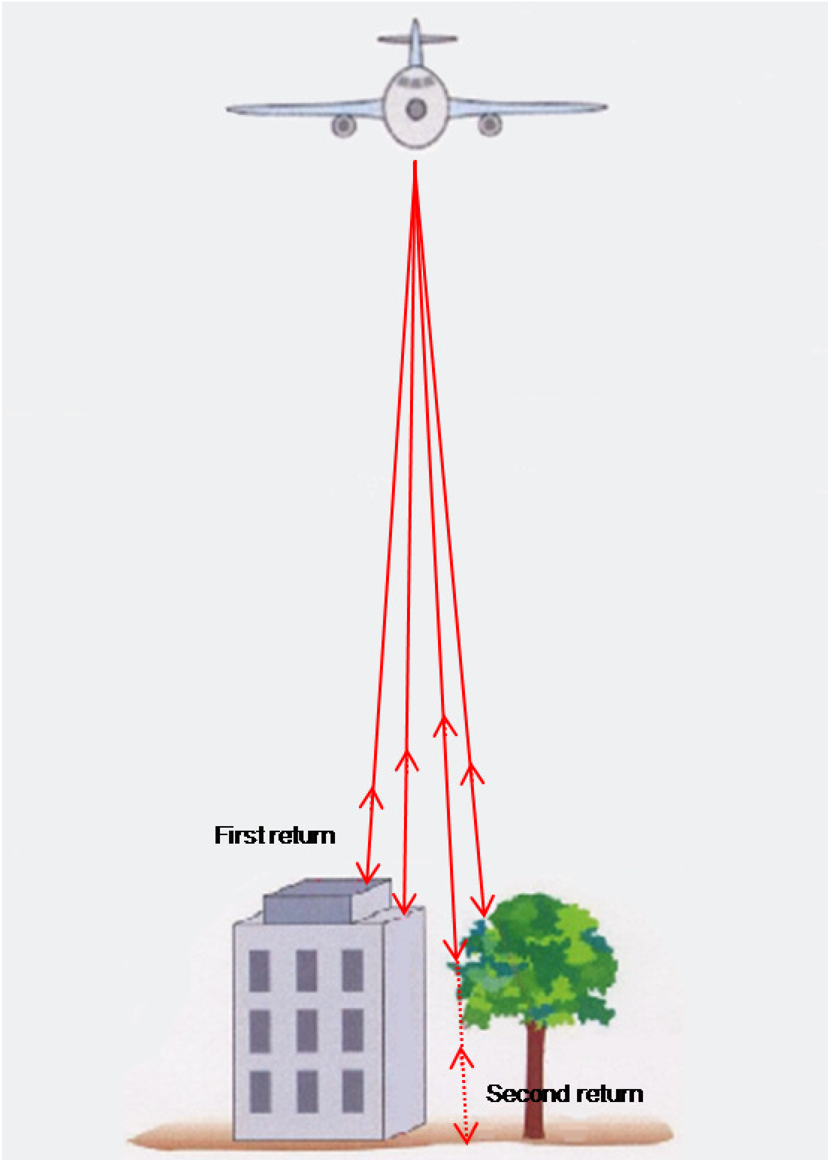


Figure 1.2 **The principle of LiDAR**

Another promising characteristic of LiDAR is the ability of recording multiple returns (elevation) per point. This is useful for differentiating ground and objects above the ground at the same location. In a tall vegetation area, the gaps between tree canopies allow the laser to penetrate trees to some extent and record additional returns at lower elevations (e.g. second return, third return) (Goodwin, et al., 2009). In the ideal situation, the laser can reach the ground and record the last return at surface elevation. Usually there are up to 5 returns in LiDAR data. This multiple-return mechanism is suitable for generating a terrain model and distinguishing buildings from trees.

The LiDAR point clouds can be interpolated as raster data for further processing. In most cases, the DSM (Digital Surface Model) represents the elevation of the earth's surface and includes all objects on it. In contrast to DSM, DTM (Digital Terrain Model) represents the elevation of bare ground surface with all objects like tall vegetation and buildings removed. The difference model of a DSM and a DTM is called nDSM (Normalized Digital Surface Model), which is a representation of the heights of objects on a plane surface (Shan and Toth, 2009). The raw LiDAR point clouds can be separated into “ground” and “non-ground” targets and the ground points are used to construct DTM. The DSM is usually interpolated from first return point clouds, and the nDSM is generated by subtracting DTM from DSM (Figure 1.3). By generating nDSM, the terrain effect is eliminated so that the elevation in the nDSM can represent the heights of objects on the plane surface.

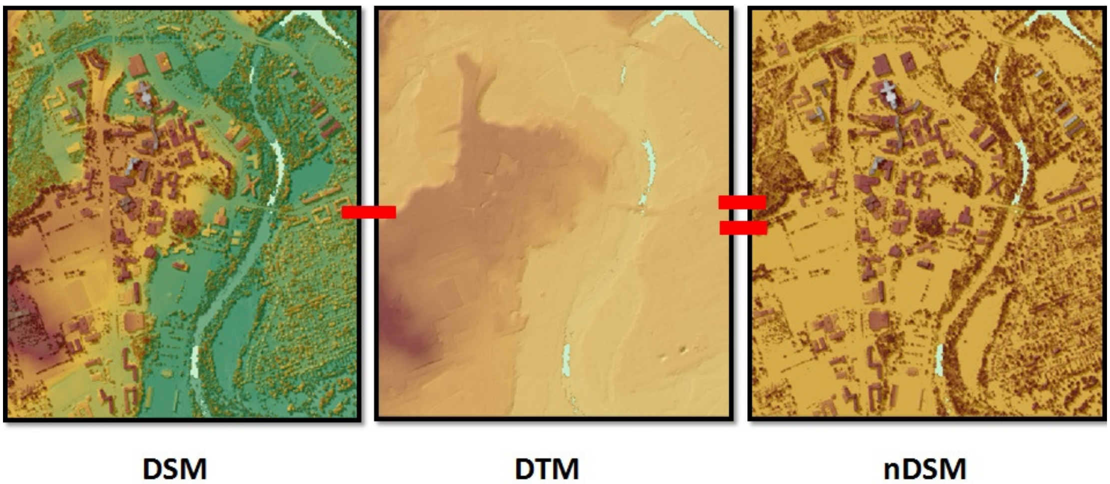


Figure 1.3 **Concepts of DSM, DTM and nDSM**

Specifically, the LiDAR data used in this research was acquired by the Optech ALTM 3100 sensor. The airborne LiDAR data were collected over the Medway Creek of London region on May 20th of 2006. The survey was operated at 1000 m above ground level with a mirror scanning rate of 39 Hz and a laser frequency of 70 kHz. This sensor is capable of recording the first, second, third and last returns and intensity data. There are in total 15 stripes with 710-750 meters in width consisting of the entire scene. The overlap percentage of those stripes is around 50%. The point density of LiDAR is 1 point / m^2. The aircraft was equipped with an inertial measurement unit (IMU) and high-precision global positioning system (GPS) used to locate and orient the laser returns in three-dimensional space (Shan and Toth, 2009). All LiDAR point coordinates and elevations are in the UTM Zone 17 coordinate system relative to the NAD83 datum.

1.4.2 Object-based classification

A detailed introduction of object-based classification is merited here since Chapter 2 uses object-based classification for building footprint extraction. The proposed method is performed in the Definiens's proprietary eCognition 8.0™ software. The general steps of object-based classification include image segmentation and classification. There are mainly two approaches in the classification stage, one is nearest neighbor classification, and the other is rule-based classification, which is used in this study and introduced in the following sections.

1.4.2.1 Image segmentation

Rather than classifying individual pixels into discrete cover types, Object-based classification first segments imagery into small meaningful objects, which are the basic units in the classification (Figure 1.4). An effective segmentation is one in which image objects are large enough to have meaningful geometric or spectral values, but small enough to represent only one class (Lillesand et al., 2008).

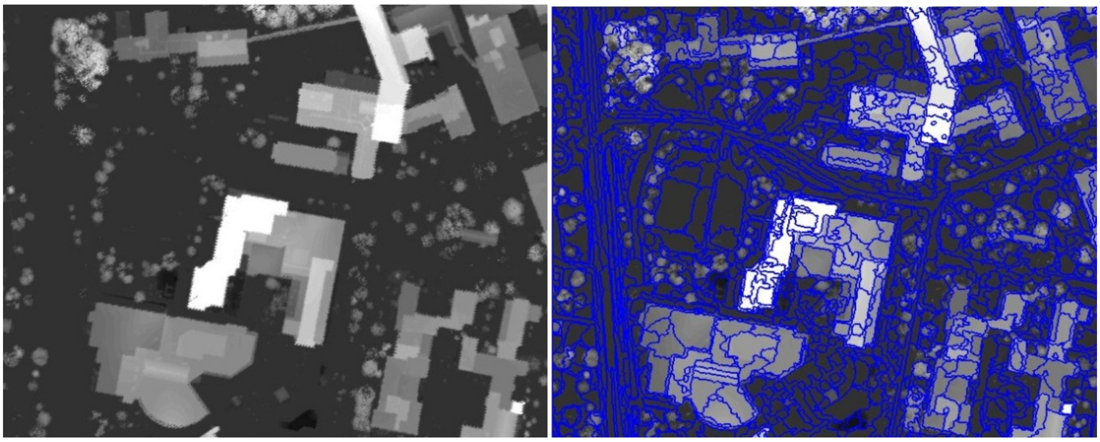


Figure 1.4 **Image before and after segmentation. Left: image before segmentation; Right: image after segmentation**

Image objects start as single pixels, and neighboring pixels are merged together when the increases in heterogeneity of objects are less than a user-defined threshold of heterogeneity usually referred to as a scale parameter (Frauman and Wolff, 2005). Generally, the scale parameter is used to determine the maximum allowed heterogeneity for the resulting image objects. For heterogeneous areas, the resulting objects for a given scale parameter will be smaller than that in more homogeneous areas. By modifying the value of the scale parameter, the size of image objects can be varied as well (Definiens, 2010). An illustration of how scale parameters affect the size of segmented objects is shown in Figure 1.5.

Figure 1.5 **Illustration of the relationship between the scale parameter and the size of segmented objects. Left: scale parameter = 5; Right: scale parameter = 20**

Segmentation is complete when the increase in heterogeneity created by any remaining merger exceeds the threshold. This heterogeneity threshold (H) is calculated using a weighted combination of both spectral (color) and spatial (compactness and border smoothness) characteristics. The following equations are proposed in Baatz and Schape, 2000 and Baatz et al., 2001, the combination of the criteria is shown as follows:

$$H = w_{color} \cdot h_{color} + (1 - w_{color}) \cdot h_{shape} \tag{1}$$

where w_{color} is user-defined weight for color parameter; h_{color} is the spectral heterogeneity of an image object, which is computed as the sum of the standard deviations of spectral values of each layer (σ_k) multiplied by the weight for each layer (w_k):

$$h_{color} = \sum_{k=1}^{m} w_k \cdot \sigma_k \tag{2}$$

h_{shape} represents the spatial heterogeneity of an image object, which is calculated using compactness and smoothness:

$$h_{shape} = w_{cpt} \cdot h_{cpt} + (1 - w_{cpt}) \cdot h_{smooth} \tag{3}$$

Among which w_{cpt} is the user-defined weight for compactness parameter; h_{cpt} is described by the ratio of the pixel perimeter length l and square root of the number of pixels n forming an image object:

$$h_{cpt} = l/\sqrt{n} \tag{4}$$

h_{smooth} is the ratio of the pixel perimeter length l and the shortest possible border length b of a box bounding the image object parallel to the raster:

$$h_{smooth} = l/b \tag{5}$$

These three criteria for heterogeneity can be applied in various combinations. For most cases, the color information is the most important for creating meaningful objects. However, a certain degree of shape homogeneity can improve the performance of image segmentation. This is probably due to the fact that the compactness of image objects is related to the shapes of features presented in the image. Thus the shape criteria (compactness and smoothness) are also helpful in avoiding highly fractured image objects resulting in strongly textured data (e.g. radar data) (Definiens, 2010).

1.4.2.2 Rule-based classification of building footprints

After segmentation, the unclassified image objects can then serve as building candidates for subsequent classification of larger building entities. Object characteristics such as color, shape, texture, as well as contextual information (e.g. connectivity, contiguity, distances and direction) can be used in the rule-set for classification (Zhou and Troy, 2008). In this way, objects with heterogeneous height values in LiDAR elevation data, such as building and trees, can still be separated despite their similarity (Figure 1.6). A rule set is a sequence of processes which are the elementary tools providing a solution to a specific image analysis problem (e.g. assign the image objects with areas less than 100 m^2 to building class) (Definiens, 2010). Those processes are executed in a user-defined order. A detailed combination of processes for object-based classification can be found in Chapter 2. Here some commonly used categories of tools for the rule-based classification are introduced as follows:

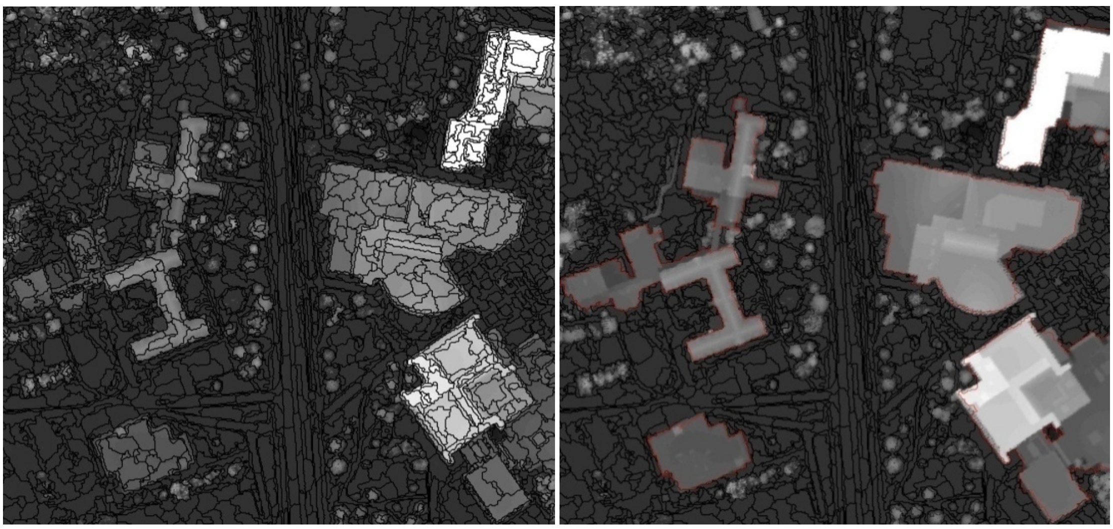


Figure 1.6 **Image before and after classification. Left: image before classification; Right: image after classification**

(1) Layer values

Layer values represent the first and second statistical moment (mean and standard deviation) of an image object's pixel values and the object's relations to other image objects' pixel values (Definiens, 2010). Layer values are the basic tools in the object-based classification since they calculate the spectral information of images. Mean and standard deviation values are commonly used in the rule-based classification. Mean values refer to the average of the layer values of all pixels within an image object. Standard deviation values represent the variation or dispersion from mean values. The layers are the input data from users, which can be different bands of images or their derived calculation images such as Normalized Difference Vegetation Index (NDVI). For example, Lehrbass and Wang, (2010) use NDVI as one of the input data for object-based classification of trees.

(2) Shape

Shape tools evaluate the shapes of image objects in a variety of respects. The basic shape values are calculated based on the object's pixels, such as area and border length. The area of an image object is the true area of one pixel multiplied by the number of pixels within the image object; the border length of an image object is defined as the sum of edges of the image object that are shared with other image objects or are situated on the edge of the entire scene (Definiens, 2010). This category of tools can compensate for the spatial information omitted by layer values. For example in Huang et al., 2008, both spatial and spectral information are needed for the extraction of building footprint since buildings present not only special spectral information due to different surface materials, but also unique spatial characteristics such as square or round shapes.

(3) GLCM Texture

The most commonly used texture tools are GLCM textures. Texture information examines the pattern presented in an image object. GLCM (Grey Level Co-occurrence Matrix) is a tabulation of how often different combinations of pixel brightness values (grey levels) occur in an image, which is regarded as a statistical method of examining texture that considers the spatial relationship of pixels (Haralick et al., 1973). The GLCM functions characterize the texture of an image by calculating how often pairs of pixels with specific values or a specified spatial relationship occur in an image (Haralick, 1979). There are many kinds of GLCM texture measurements used in previous studies (El-Ashmawy et al., 2011), such as Homogeneity, Contrast and Entropy.

1.5 Thesis Format and Outlines

The thesis is in integrated-article format. The goal of the thesis presented here is to use airborne LiDAR data to extract building footprints, based on which 3D building models are reconstructed. Chapter 1 introduces the research contexts, background and objectives of the studies.

Chapter 2 and Chapter 3 are two individual studies aimed for publication in academic journals. Chapter 2 aims at object-based extraction of building footprints from airborne LiDAR data. A simplified version of Chapter 2 was presented at the 33rd Canadian Symposium on Remote Sensing (CSRS) in Ottawa, Ontario in June, 2012.

Chapter 3 benefits from the availability of building footprints extracted in Chapter 2. The focus of Chapter 3 is to reconstruct 3D building models using GIS techniques from LiDAR data.

Finally, general discussions and conclusions, contributions of the studies and possible improvement in future work are presented in Chapter 4.

1.6 References

Aldred, D. and Wang, J. 2011. A method for obtaining and applying classification parameters in object-based urban rooftop extraction from VHR multispectral images. *International Journal of Remote Sensing*, Vol. 32, No.10, pp. 2811-2823.

Arefi, H., Hahn, M., and Lindenberger, J., 2003. LiDAR data classification with remote sensing tools. *Proceeding of the ISPRS Commission IV Joint Workshop: Challenges in Geospatial Analysis, Integration and Visualization II*, 08-09 September, Stuttgart, Germany, pp.131-136.

Baatz, M. and Schape, A., 2000. Multiresolution Segmentation – an optimization approach for high quality multi-scale image segmentation. *In: Angewandte Geographische Informationsverarbeitung XII, Strobl, J., Blaschke, T., Griesebner, G. (eds.) AGIT Symposium, Salzburg, Germany.* Wichmann: Heidelberg, pp. 12-23.

Baatz, M., Heynen, M., Hofmann, P., Lingenfelder, I., Mimier, M., Schape A., Weber, M., and Willhauck, G., 2001. "eCognition User Guide 2.0 : Object Oriented Image Analysis." *Definiens Imaging GmbH, Germany*.

Beraldin, J. A., Blais, F. and Lohr, U. 2010.Laser scanning technology. Chapter 1 in *Airborne and Terrestrial Laser Scanning* (Eds. G. Vosselman and H. G. Maas).Whittles, Caithness, Scotland. 320 pages: 1-42.

Blaschke, T. 2010. Object based image analysis for remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 65, No. 1, pp. 2-16.

Brennan, R. and Webster, T.L. 2006. Object-oriented land cover classification of lidar-derived surfaces. *Canadian Journal of Remote Sensing*, Vol. 32, No. 1, pp. 162-172.

Brunn, A., Weidner, U. 1997. Extracting buildings from digital surface models. *IAPRS International Archives of Photogrammetry and Remote Sensing*, Vol. 32, No.W2, 3D Reconstruction and Modeling of Topographic Objects.

Davis, D. 2005. Evaluation of alternatives in obtaining structural elevation data. *Federal Emergency Management Agency (FEMA) Report*. Vol. 136, pp. 1-2. Available online at <http://www.fema.gov/pdf/nfip/alt_elevations/elevations_final.pdf>.

Definiens, 2010. Definiens Developer 8.0.1 Reference Book. *Definiens AG, München, Germany*.

El-Ashmawy, N., Shaker, A., Yan, W.Y. 2011.Pixel vs object-based image classification techniques for LiDAR intensity data. *ISPRS Workshop on Laser Scanning*, Elsevier, Vol. XXXIII, Calgary, Canada.

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

Forlani, G., Nardinocchi, C., Scaioni, M., Zingaretti, P., 2006, Complete classification of raw LiDAR data and 3D reconstruction of buildings. *Pattern Analysis Application*, Vol. 8, pp. 357-374.

Frauman, E., and Wolff, E. 2005. Segmentation of very high spatial resolutions satellite images in urban areas for segments-based classification. *Proceedings of the 3rd International Symposium Remote Sensing and Data Fusion Over Urban Areas*, pp. 14-16 March 2005, Tempe, Arizona.

Goodwin, N. R., Coops, N. C., Tooke, T. R., Christen, A., and Voogt, J. A., 2009. Characterizing urban surface cover and structure with airborne lidar technology. *Canadian Journal of Remote Sensing*, Vol. 35, No. 3, pp. 297-309.

Haralick,R. M., Shanmugan, K. and Dinstein,I., 1973.Textural Features for Image Classification. *IEEE Transaction on Systems, Man and Cybernetics*, Vol SMC-3, No. 6, pp. 610-621.

Haralick,R. M., 1979. Statistical and Structural Approaches to Texture. *Proceedings of the IEEE*, Vol. 67, No. 5, pp. 786-804.

Huang, M. J., Shyue, S. W., Lee, L. H., and Kao, C. C. 2008.A knowledge-based approach to urban feature classification using aerial imagery with LiDAR data. *Photogrammetric Engineering & Remote Sensing*, Vol. 74, No. 12, pp. 1473-1485.

Jensen, J.R., 2000. Remote Sensing of the Environment. *Prentice-Hall,* Upper Saddle River, New Jersey.

Lee, D.S., Shan, J., and Bethel, J.S. 2003.Class-guided building extraction from Ikonos imagery. *Photogrammetic Engineering and Remote Sensing*, Vol. 69, No. 2, pp. 143–150.

Lehrbass, B., Wang, J. 2010. Techniques for object-based classification of urban tree cover from high-resolution multispectral imagery. *Canada Journal of Remote Sensing*, Vol. 36, No.S2, pp. 287-297.

Lillesand, T.M., Kiefer, R.W., and Chipman, J.W., 2008.Remote Sensing and Image Interpretation. *John Wiley & Sons,* Inc., Hoboken, New Jersey.

Kabolizade, M., Ebadi, H., Mohammadzadeh, A., 2012, Design and implementation of an algorithm for automatic 3D reconstruction of building models using genetic algorithm. *International Journal of Applied Earth Observation and Geoinformation*, Vol. 19, pp. 104-114.

Kim, K.H., Shan, J., 2011, Building roof modeling from airborne laser scanning data based on level set approach. *Journal of Photogrammetry & Remote Sensing*, Vol. 66, 484-497.

Ma, R., 2005, DEM generation and building boundary detection from LiDAR data. *Photogrammetic Engineering and Remote Sensing*, Vol. 71, No. 7, pp. 847-854.

Meng, X., Wang, L., and Currit, N., 2009, Morphology-based building detection from airborne LiDAR data. *Photogrammetric Engineering and Remote Sensing*, Vol. 75, No. 4, pp. 437-442.

Miliaresis, G., Kokkas, N., 2007, Segmentation and object-based classification for the extraction of the building class from LIDAR DEMs. *Computer& Geosciences*, Vol. 33, 1076-1087.

Morgan, M., Tempfli, K., 2000, Automatic building extraction from airborne laser scanning data. in*Proc. 19th ISPRS Congr.*, Amsterdam, The Netherlands, 2000, pp. 616–623, Book 3B.

Rottensteiner, F., Briese, C., 2002, A mew method for building extraction in urban areas from high-resolution LiDAR data. *IAPRS International Archives of Photogrammetry and Remote Sensing*, Vol. 34, No. 3A, pp. 295-301.

Rottensteiner, F.2003. Automatic generation of high quality building models from LiDAR data. *IEEE Computer Graphics and Applications*. Vol. 23, No. 6, pp. 42-50.

Shan, J. and Toth, C.K., 2009, Topographic laser ranging and scanning: principles and processing. *CRC Press*, Taylor & Francis, London, pp. 307-334.

Shufelt, J.A. and Mckeown, D.M., 1993, Fusion of monocular cues to detect man-made structures in aerial imagery. *Computer Vision and Graph Image Understanding*, Vol. 57, No. 3, pp. 307–330.

Song, J.H., Han, S.H., Yu, K.Y., Kim, Y.I. 2002.A study on using LiDAR intensity data for land cover classification. *ISPRS Commission III, Symposium*, Graz, Austria, September 9-13.

Strassopolou, A., Caelli, T., and Ramirez, R., 2000, Automatic extraction of building statistics from digital orthographs. *International Journal of Geographic Information System and Science*, Vol. 14, No. 8, pp. 759–814.

Tarsha-Kurdi, F., Landes, T., Grussenmeyer, P., 2007, Hough-transform and extended RANSAC algorithms for automatic detection of 3D building roof planes from LiDAR data. *Proceedings of the ISPRS Workshop on Laser Scanning 2007 and SilviLaser*, Espoo, Finland, pp. 407-412.

Vosselman, G., Dijkman, S., 2001, 3D building model reconstruction from point clouds and ground plans. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 34, pp. 37-43.

Wang, J., Lehrbass, B., Zeng, C. 2011. Urban building mapping using LiDAR and relief-corrected Color-Infrared aerial images. *Proceedings of the 34th International Symposium on Remote Sensing of Environment*, Sydney, Australia. April 10-15.

Weidner, U., 1997, Digital Surface Models for building extraction. Automatic Extraction of Man-Made Objects from Aerial and Space Images (II), *Birkhauser,* Basel, S.193-202.

Zhang, K., Yan, J., and Chen, S. 2006. Automatic construction of building footprints from airborne LIDAR data. *IEEE Transaction on Geoscience and Remote Sensing*, Vol. 44, No. 9, pp. 2523-2533.

Zhang, K., Yan, J., Chen, S.C., 2009, Automatic 3D building reconstruction from airborne LiDAR measurements. *IEEE Urban Remote Sensing Joint Event*, Shanghai, China.

Zhou, W., and Troy, A. 2008.An object-oriented approach for analyzing and characterizing urban landscape at the parcel level. *International Journal of Remote Sensing*, Vol. 29, No. 11, pp. 3119-3135.

Zhou, W., Huang, G., Troy, A. 2009. Object-based land cover classification of shaded areas in high spatial resolution imagery of urban areas: A comparison study. *Remote Sensing of Environment*, Vol. 113, pp. 1769-1777.

Chapter 2

Use of LiDAR Derived NDTI and Intensity for Rule-based Object-oriented Extraction of Building Footprints

2.1 Introduction

2.1.1 Background

As the dominant objects in urban areas, buildings play an essential role in urban development, urban planning and urban climate (Meng et al., 2009). Building footprints are one of the fundamental geographic information system (GIS) data that can be used to estimate energy demand, quality of life, urban population, property taxes (Jensen, 2000) and three-dimensional (3D) building models for visualization (Zhang et al., 2006). Accurate building footprint data are also essential for the construction of urban landscape models, estimation of natural disaster risk, the study of urban heat island effects and earthquake damage assessment (Davis, 2005; Miliaresis and Kokkas, 2007). The precise knowledge of building footprint can serves as a primary source for interpreting complex urban characteristics (Zhou et al., 2009).

2.1.2 Previous Studies

Various optical remote-sensing data sources with different spatial resolutions have been applied for urban building extraction and mapping for decades (Blaschke, 2010). Since the urban environment is extremely complex and heterogeneous, with multiple sizes of objects (e.g. buildings, roads, vegetation) combined with respective complicated spatial patterns, coarse spatial resolution imagery (e.g. Landsat, MODIS, AVHRR) is insufficient for mapping detailed urban land cover (Zhou and Troy, 2008). The recent availability of high resolution satellite images with resolution finer than 5 meters (e.g. IKONOS, QuickBird), representing a building as an object instead of mixed pixels, offers opportunities for building footprint extraction (Shufelt and Mckeown, 1993; Strassopolou et al., 2000; Lee et al., 2003; Aldred and Wang, 2011). However, the success of the derived methods on multispectral images is largely limited due to the influence of sun shadow, relief displacement and occlusion effect of tall objects (Zhou and Troy, 2008; Meng et al., 2009, Wang et al., 2011).

Recently-developed airborne Light Detection and Ranging (LiDAR) technology provides a very promising alternative for building footprint measurement. LiDAR directly collects an accurately georeferenced set of dense point clouds by recording reflected laser beams emitted by the sensor. Elevation and intensity are both stored to represent three-dimensional objects on the Earth surface (Shan and Toth, 2009). Compared to high resolution multispectral images, which take a snap-shot from a certain angle, LiDAR measurements are free from sun shadows, relief displacement and occlusion effect (Zhang et al., 2006). The most attractive characteristic of LiDAR is its very high vertical accuracy which is suitable for Digital Elevation Model (DEM) generation and building detection (Ma, 2005). Additionally, obvious elevation differences between the first and last returns of LiDAR data in tall vegetation areas are very helpful in distinguishing trees from buildings among non-ground objects (Meng et al., 2009). Some studies also combine LiDAR with high resolution optical imagery for developing building extraction models (Sohn and Dowman, 2007; Lee et al., 2008; Vu et al., 2009; Wang et al., 2012). However, the integrated processes may cause errors introduced by resolution and time difference, shadows and building relief displacement problems (Sohn and Dowman, 2007; Lehrbass and Wang, 2012). Therefore, developing competitive techniques to extract buildings from LiDAR data alone is worthwhile.

Previous studies on identifying building footprints from LiDAR data can be divided into two broad categories; a detailed summary is shown in Table 2.1. One category is to use a non-hierarchical method to separate all land-use types (e.g. buildings, trees, roads, grass) simultaneously; the majority of studies use raster data derived from LiDAR point clouds, and pixel-based classifications are mostly seen in this category (Song et al., 2002; Arefi and Hahn, 2005; El-Ashmawy et al., 2011). Nevertheless, the pixel-based methods are not efficient for high resolution images, since objects on the Earth surface are represented by many pixels instead of single mixed pixels in the imagery (Blaschke, 2010). Although some studies add LiDAR intensity in the pixel-based classifications, the intensity of objects with low heights such as roads and parking lots is very similar with that of buildings, which can lower the efficiency of separating all land-use types simultaneously.

The other category is to separate the non-ground objects (e.g. buildings, trees) from the ground and then identify buildings from the non-ground measurements. Both LiDAR point clouds and the derived raster data have been separately used in those studies. For the initial separation of the ground and non-ground measurements, morphological filters (Weidner, 1997; Brunn and Weidner, 1997; Morgan and Tempfli, 2000) and surface-based filters (also known as robust interpolation) (Rottensteiner and Briese, 2002; Rottensteiner, 2003) are mainly applied to classify ground points, based on which Digital Terrain Model (DTM) can be generated to stand for the ground elevation. The non-ground objects can be presented using the derived Normalized Digital Surface Model (nDSM), which is a representation of non-ground objects' elevation on the plane surface. After the non-ground objects are identified, the critical step is to classify building and tree objects that dominate non-ground measurements (Zhang et al., 2006). The most traditional way to classify building footprint is to use thresholds on the Digital Surface Model (DSM) or nDSM derived from LiDAR data (Weidner, 1996; Brunn and Weidner, 1997). Though both global thresholds and locally adjusted thresholds are used in the method, trees and buildings are still poorly separated since they have similar heights and can be very close to each other. Several (Morgan and Tempfli, 2000; Rottensteiner and Briese, 2002; Rottensteiner, 2003; Ma, 2005; Goodwin et al., 2009) improved the initial results from thresholding by using morphological filters or texture analysis to filter out trees, thus a better result of building footprint can be acquired. However, not all trees are detected since some dense tree canopies do not possess characteristics of typical trees, and can be as smooth as topographic surfaces or roof patches, which lack textural patterns. Instead of using raster data, some studies directly classify building points from LiDAR point clouds by mathematical algorithms such as plane-fitting method (Zhang et al., 2006). Nevertheless, the problem of separating trees that have smooth surface from buildings still exists.

Table 2.1 **Review of the previous studies on LiDAR-based building extraction**

Category	Methods for building extraction	References	Drawbacks
Non-hierarchical classification (e.g. MLC)		Song et al., 2002 Arefi and Hahn, 2005 El-Ashmawyet al., 2011	The non-hierarchical classifications are largely limited by similarities of different land-use types, and are not efficient for high resolution images
Hierarchical classification	Thresholding	Weidner, 1996 Brunn and Weidner, 1997	Trees and buildings are poorly separated
	Morphological filter or texture analysis	Morgan and Tempfli, 2000 Rottensteiner and Briese, 2002 Rottensteiner, 2002 Ma, 2005 Goodwin et al., 2009	Not all trees are detected since some dense tree canopies with smooth surfaces lack evident texture patterns
	Plane-fitting	Zhang et al., 2006 Lin et al., 2011	Similar problem as mentioned above
	Object-based	Brennan and Webster, 2006 Antonarakis et al., 2008 Huang, et al., 2008	Object-based methods hierarchically integrate both spatial and spectral information of LiDAR, can possibly improve building extraction results

The hierarchical object-based classification that combines both spatial and spectral information of LiDAR provides a possible route for improvement of building extraction. Though object-based algorithm has been applied in several studies, only a few have studied urban areas (Huang et al., 2008; Lin, 2011) while the others (Brennan and Webster, 2006; Antonarakis et al., 2008) focused more on vegetation classification.

Moreover, many of the aforementioned studies (Ma, 2005; Zhang et al., 2006; El-Ashmawy et al., 2011; Lin et al., 2011) used simple study areas with most buildings and trees spatially separated to achieve high accuracy. The overall accuracies of building extraction can depend on the complexity of study areas. The efficiency of object-based classification of building extraction in more complex urban environments should be examined.

2.1.3 Objectives

Based on previous studies, the objective of this research is to develop an object-based classification method based on LiDAR elevation and intensity data to extract locations and boundaries of building footprints in a complex urban environment. An index named Normalized Difference Tree Index (NDTI) is used. The roles of the LiDAR intensity data and the NDTI in the object-based classification are evaluated; an integrated segmentation approach and a hierarchical rule-based classification strategy are proposed. Finally the performance of the proposed object-based method is examined by comparing to the reference data.

2.2 Methods

2.2.1 Data Acquisition and Study Area

The LiDAR data are acquired by Applied Geomatics Research Group with an Optech ALTM 3100 sensor. The airborne LiDAR data are collected in North of London, Ontario on May 20th of 2006, allowing for maximum penetration to the ground. This sensor is capable of recording the first, second, third and last returns and intensity data. The aircraft is equipped with an inertial measurement unit (IMU) and high-precision global positioning system (GPS) used to locate and orient the laser returns in three-dimensional space (Brennan and Webster, 2006). All LiDAR point coordinates and elevations are recorded using the UTM zone17 coordinate system relative to the NAD83 datum. A subset of LiDAR point clouds is shown in 3D (Figure 2.1).

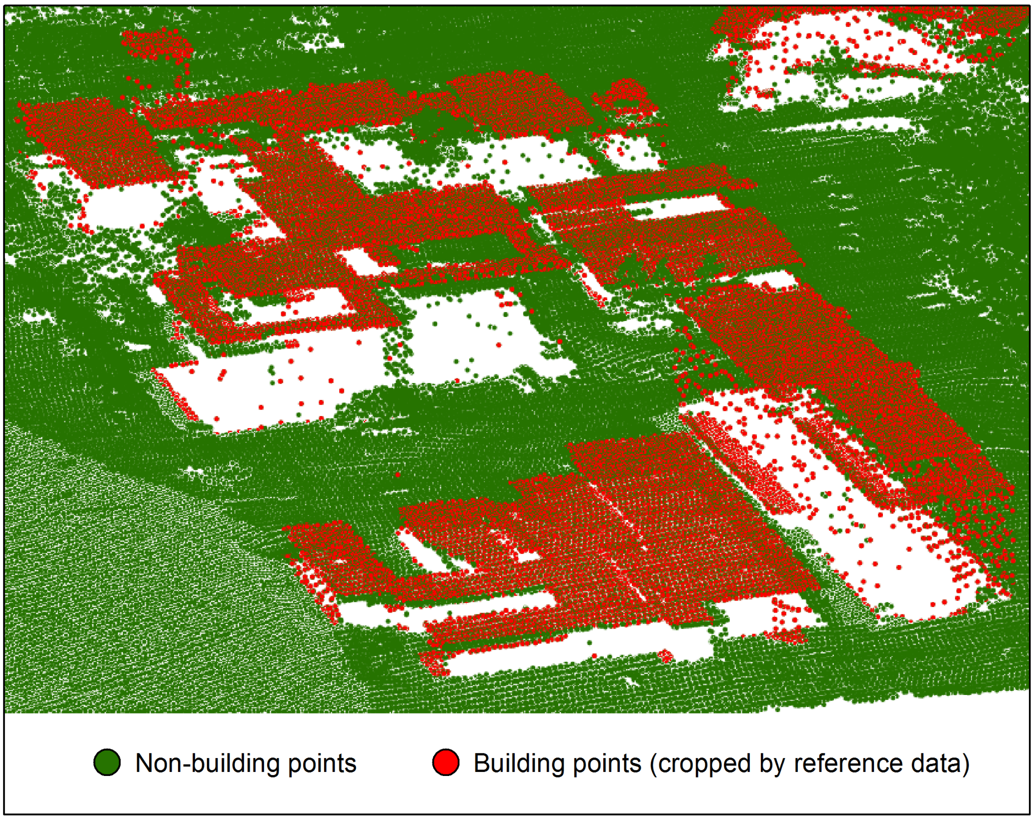


Figure 2.1 **LiDAR point clouds in 3D view**

The vector layer of building footprints in the study area is acquired from the City of London Digital Mapping distribution datasets in 2006, the same year as that of the LiDAR data, avoiding disparity of building boundaries due to time difference. The vector data were delineated using air photo interpretation, and are used as reference data for evaluating the performance of the extracted building footprints.

The experimental site is chosen at the University of Western Ontario, which is located in the north part of London, Ontario. There are many land use and land cover types in the study area such as grass, water, asphalt, trees, buildings, bare ground and agriculture land. A variety of building types exist within the study, such as small buildings with complicated rooftops, tall buildings surrounded by trees, etc. The purpose of choosing the study area with various kinds of buildings is to develop a classification strategy for building footprints in complex urban environments, thus the method has some applicability and can be further applied in other areas as well.

2.2.2 Overview

The proposed object-based classification of building footprints is described in the following sections, with pre-processing and rule-based classification illustrated in Figure 2.2 and Figure 2.5 respectively. The method includes pre-processing to interpolate LiDAR point clouds to raster images and generate LiDAR nDSM, intensity and NDTI data, two-stage segmentation, rule-based classification of building footprints, post-processing and accuracy assessment (Figure 2.2).

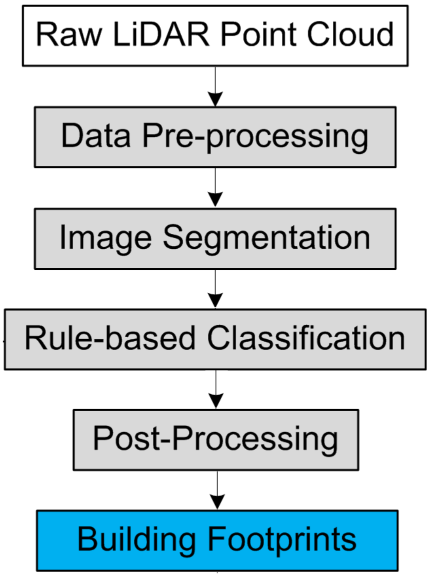


Figure 2.2 **Main steps of the proposed method**

2.2.3 Pre-processing

The DSM represents the elevation of the earth's surface and includes all objects on it. In contrast to DSM, DTM represents the elevation of bare ground surface with all objects like trees and buildings removed. The difference model of a DSM and a DTM is called nDSM, which is a representation of the height of objects on a plane surface (Shan and Toth, 2009). The LiDAR points are used to construct a 0.6 m DTM grid that represents the bare earth (Figure 2.3). The DSMs of the first and last returns are interpolated into 0.6 m raster data from the first-return and the last-return point (Figure 2.4). The nDSM of the first and last returns are generated by subtracting DTM from the respective DSMs.

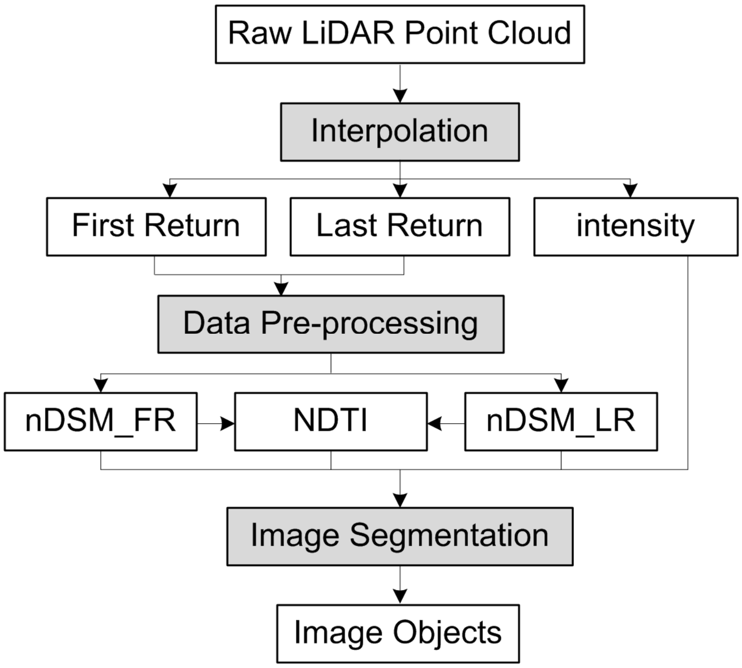


Figure 2.3 **Flowchart of the pre-processing**

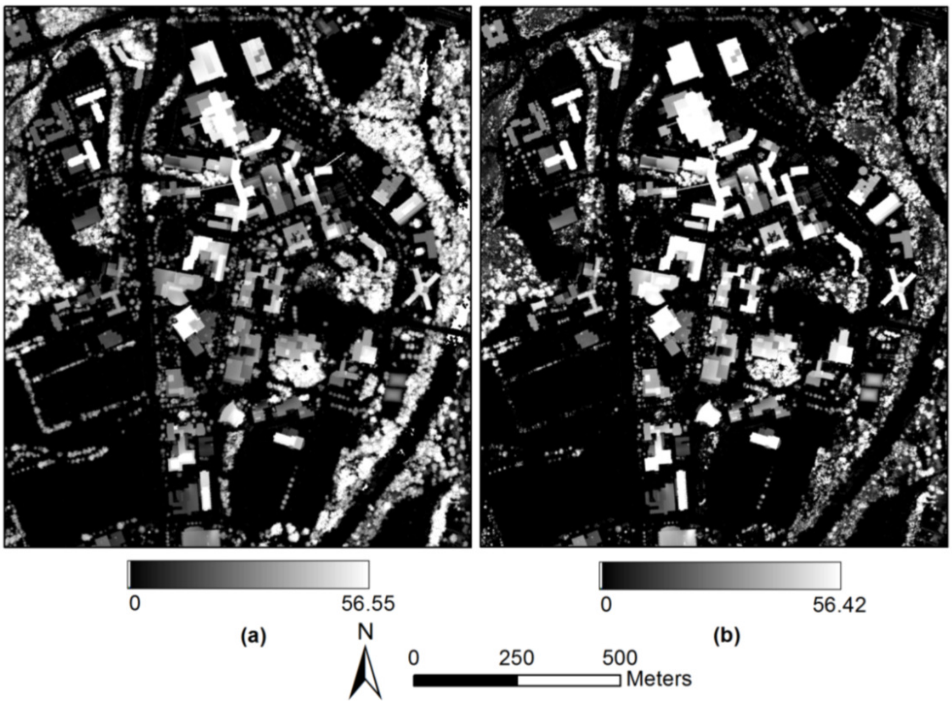


Figure 2.4 **Input data for object-based extraction of building footprints. (a) nDSM of the first return; (b) nDSM of the last return**

The intensity raster layer is computed by interpolating all LiDAR returns based on the intensity attribute of LiDAR points (Brennan and Webster, 2006). LiDAR intensity is usually influenced by surface reflectance, atmospheric transmission, local incidence angle and sensor-to-object distance (Wang and Glenn, 2009). Previous research suggests that local incidence angle has little impact on LiDAR intensity (Kaasalainen et al., 2005), and atmospheric absorption plays a minor role in near-infrared LiDAR intensity due to the wavelength (λ=1064nm) (Mazzarini et al., 2007). Therefore, intensity can be calibrated with respect to the sensor-to-object distance (intensity level changes with the inverse square of the distance) (Mazzarini et al., 2007). However, no intensity calibration is applied in this study since the main objective is to separate buildings and trees with similar heights (therefore similar sensor-to-object distance). The non-calibrated intensity can still be used to distinguish different objects from the surface reflectance.

An index called Normalized Difference Tree index (NDTI) is also used as input data (Figure 2.5). NDTI is modified from the Normalized Difference (ND) used in Arefi et al., 2003 and Huang et al., 2008. The ND can be calculated using the DSM of the first return (DSM_{FR}) and the last return (DSM_{LR}):

$$ND = \frac{DSM_{FR} - DSM_{LR}}{DSM_{FR} + DSM_{LR}} \tag{6}$$

Since the DSM is the sum of the DTM and the nDSM, the above equation can be also calculated as follows:

$$ND = \frac{(nDSM_{FR} + DTM) - (nDSM_{LR} + DTM)}{(nDSM_{FR} + DTM) + (nDSM_{LR} + DTM)} = \frac{nDSM_{FR} - nDSM_{LR}}{nDSM_{FR} + nDSM_{LR} + 2DTM} \tag{7}$$

In contrast, NDTI is calculated using the nDSM of the first return ($nDSM_{FR}$) and the last return ($nDSM_{LR}$) to enlarge height difference between the two returns:

$$NDTI = \frac{nDSM_{FR} - nDSM_{LR}}{nDSM_{FR} + nDSM_{LR}} \tag{8}$$

Although ND can represent the elevation difference between the two returns, the DSMs used in the equation contain the terrain elevation, thus the ND is affected by terrain effect. The value of ND in those areas with higher terrain elevation is smaller than that in areas with lower terrain elevation. Therefore, the thresholds of ND for two areas with different DTM elevations would be different and need to be modified. Compared with ND, NDTI is free from the influence of terrain elevation. Thus the threshold used in one area can be applied to another area as well. The height contrasts presented in NDTI data better represent the actual circumstances. Generally, height difference between the first return and last return is mostly caused by the LiDAR penetration through trees (Meng et al., 2009, Brennan and Webster, 2006). Therefore, NDTI can be used to separate trees from buildings among the non-ground objects. Since building edges can also result in height difference between multiple returns, a median filter with a window size of 5 by 5 is applied to the NDTI layer to reduce the influence of building edges. The LiDAR nDSM of the first and last returns, the intensity data, along with NDTI data are used as input data (spatial resolution is 0.6 meters) for object-based building footprint extraction.

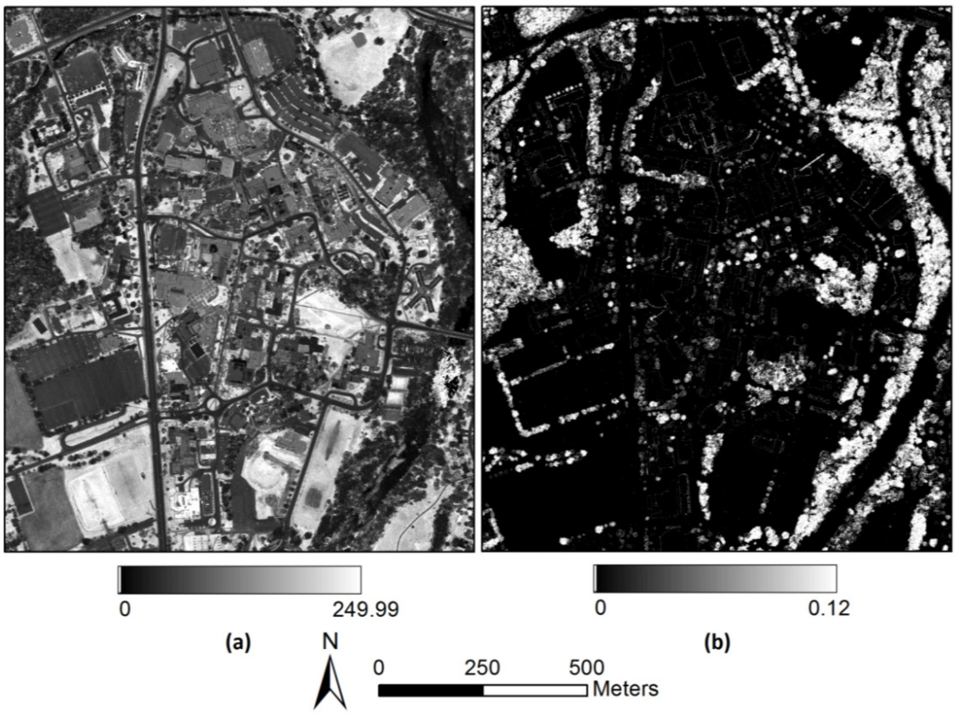
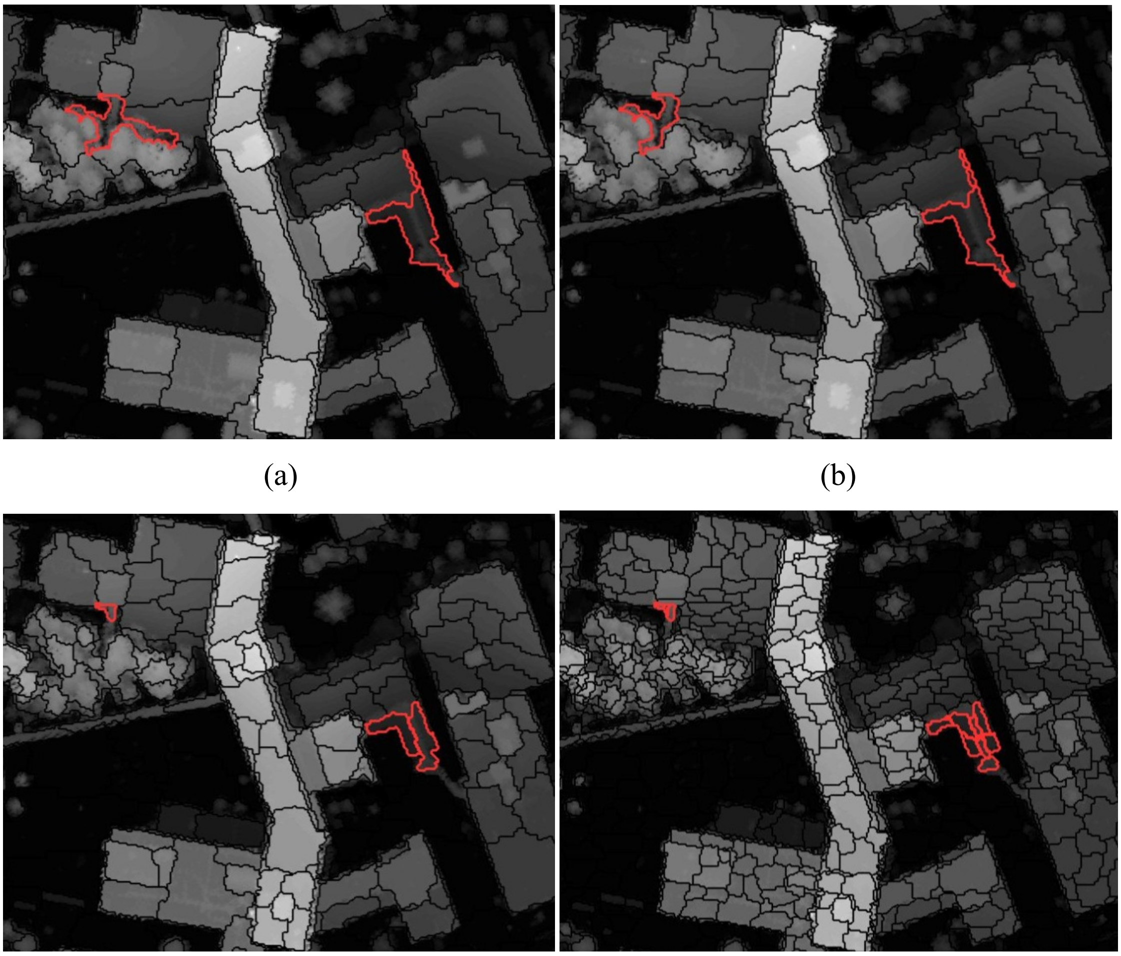


Figure 2.5 **Input data for object-based extraction of building footprints. (a) LiDAR intensity and (b) NDTI data**

2.2.4 Image Segmentation

Rather than classifying individual pixels into discrete classes, the object-oriented classification first segments imagery into small meaningful objects, which are the basic units in the classification. An effective segmentation is the result of image objects that are large enough to have meaningful geometric or spectral values, but small enough that they do not represent more than one feature class (Lillesand et al., 2008).

Two image segmentation approaches are integrated in this study, Multiresolution Segmentation and Spectral Difference Segmentation. Multiresolution Segmentation follows a region-growing algorithm given in Baatz and Schape (2000). This method merges single pixels with their neighboring pixels when increases in spatial and spectral heterogeneity of objects are less than a user-defined threshold of heterogeneity called the scale parameter (Frauman and Wolff, 2005). The segmentation is complete when the increase in heterogeneity created by any remaining merger exceeds the threshold. After attempts with different combinations of layer weights and segmentation scale, the nDSM of the first and last returns and the NDTI are imported as inputs. The image layer weights (equal weights) and segmentation scale (10) are chosen empirically and carefully. The segmentation scale is critical for further classification especially for those buildings that are surrounded by trees. Comparisons among segmentation results using different scale parameters are shown in Figure 2.6. If the scale parameter is set too large, the image objects would be very big thus contain more than one land use types (in this case, buildings and trees). The larger the scale parameter, the bigger chance the segmented objects include mixed classes ((a) and (b) in Figure 2.6). If the scale parameter is set too small, the image objects would be too small to represent a relatively integrated land use type and the number of objects would grow, which can increase the difficulty and the spent time for classification ((d) in Figure 2.6). In this study, the segmentation scale is set to 10, considering the objective is to ensure each segmented object represents one single class and to minimize mixture of pixels of different land use types. However, after Multiresolution Segmentation, the segmented objects are still a little small to represent meaningful characteristics for each class.



(a) (b)

(c) (d)

Figure 2.6 **Comparisons of segmentation results with different scale parameters. (a) the scale parameter is 20; (b) the scale parameter is 15; (c) the scale parameter is 10; (d) the scale parameter is 5.**

To compensate for this deficiency, Spectral Difference Segmentation is used to merge neighboring objects according to their mean layer values. Neighboring image objects should be merged if the difference between their layer mean values is below the value given by the maximum layer mean difference (Definiens, 2010). This algorithm is designed to refine existing segmentation results by integrating image objects with similar values produced by previous segmentations. In this step, the nDSM of the first and last returns, the NDTI and the intensity data with weight ratio of 1:1:2:1 are used to merge image objects. The higher weight for NDTI is intended to separate buildings from trees that are highly adjacent to buildings. A loop is set to iterate the process. The segmentation is completed when the spectral difference among all the neighboring objects are greater than the maximum spectral difference.

The intensity data play a unique role in Spectral Difference Segmentation. The column (b) and (c) in Figure 2.7 are samples of segmentation results with and without intensity data. Without the intensity data, a segmented object may include more than one land use type on the ground, and the shape of the object can be irregular. Compared to the case without use of intensity data, the segmentation results are much better and desirable with the help of the intensity data.

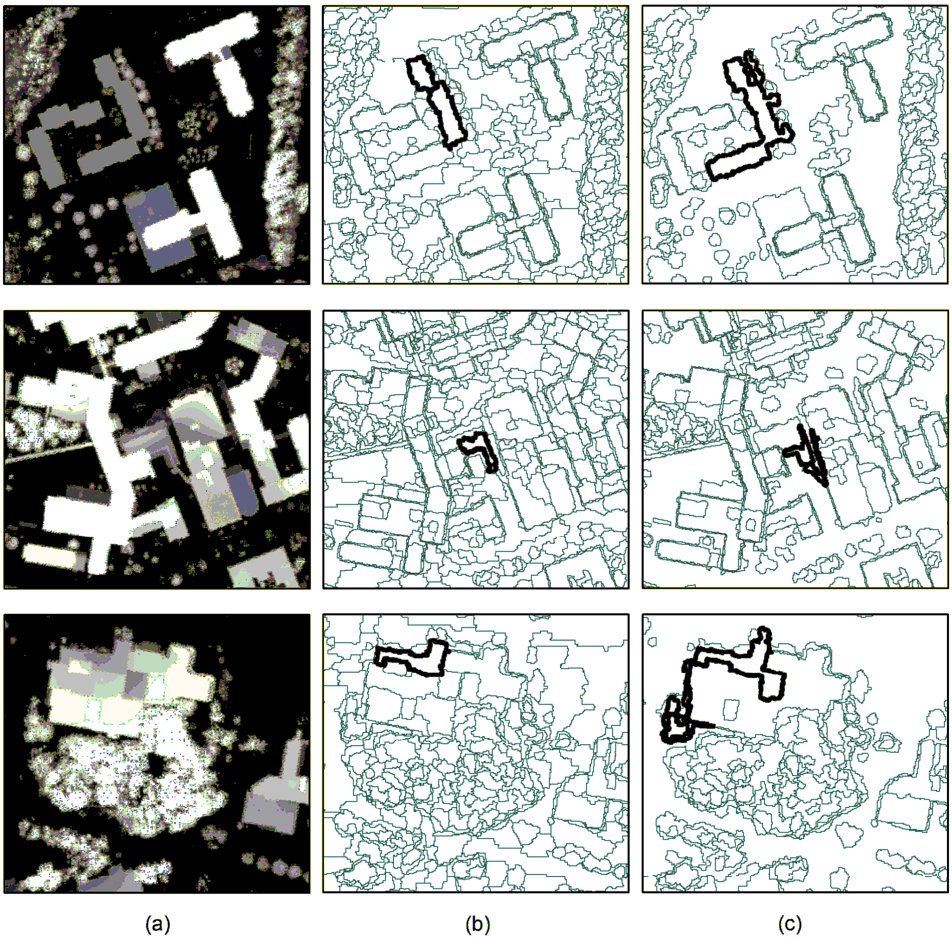


Figure 2.7 **Illustration of the role of LiDAR intensity in the image segmentation. Column (a) are nDSM of the last return as visual reference; Column (b) and (c) are segmentation results with the help of intensity and without using it respectively. The bold objects are examples of the segmented objects in each case.**

2.2.5 Rule-based Classification

Building candidates and unclassified are considered in this object-based classification. Three hierarchical procedures are performed to extract building footprints (The pseudo code is available in Appendix 1). The idea of the rule-set is to identify building objects in multiple hierarchies (Figure 2.8 and Figure 2.9), not to classify all the building footprints simultaneously, which cannot produce accurate results due to the complex characteristics of buildings and their surrounding environments. There are three main steps in the rule-based classification, which are elaborated as follows:

(1) Step one: assign basic building objects

The first step is to classify basic building objects, which assigns typical building objects that have regular shapes and small height difference between the first and last returns as building candidates. Image objects with the NDTI value no less than 0.001 and the nDSM of last return greater than or equal to 2.5 meters are classified as building candidates. The threshold of 0.001 for NDTI is regarded to be generally applicable considering the variety and amount of trees in the study area. The threshold of 2.5 for the nDSM of last return is also suitable for buildings in other areas. Since trees taller than 30 meters are not seen in the study area, tall objects with mean heights more than 30 meters are also classified as potential buildings. The height threshold for trees can be modified according to different situations of study areas. During the process, however, some trees qualify the criteria as well due to eligible height and the NDTI threshold. Thresholds of the Roundness and Area of objects are used to exclude tree objects, since tree canopies usually present relatively rounder shapes and small areas for urban trees that are somewhat isolated. The **Roundness** refers to the difference of enclosing / enclosed ellipse as the radius of the largest enclosed ellipse is subtracted from the radius of the smallest enclosing ellipse (Definiens, 2010). The smaller the value of Roundness is, the rounder the image object is. Thus Roundness is used to distinguish tree objects. Objects that fail to meet the condition are assigned as unclassified and sent to the next step.

(2) Step two: assign adjacent building objects

The second step is to classify objects that are adjacent to building objects. Since most parts of buildings have been properly classified, this step aims at categorizing building objects which have similar NDTI values with that of trees, and yet have higher ratio of the shared border length (Relative border to) with building candidates. The **Relative border to** describes the ratio of the shared border length of an image object with the neighboring image objects assigned to a defined class to the perimeter of the image object (Definiens, 2010). For instance, if the relative border of an image object to image objects of a certain class is 1, it means that the image object is totally embedded in these image objects. If the relative border is 0.5, then the image object is surrounded by half of its border (Definiens, 2010). Since some image objects within building footprints (jump edges between roof planes) are relatively difficult to distinguish by elevation and NDTI thresholds, this parameter can be very helpful in categorizing these image objects by examining the ratio of the shared border length with neighboring image objects that have already been classified as buildings. Again, the thresholds of Roundness and the nDSM of last return are used to avoid classification mixture of trees.

(3) Step three: assign building edge objects

The rest of the unclassified objects are sent to the last step, which is to identify building edge objects. The NDTI values of building edge objects can be very high due to partial penetration of laser beams along building edges. If the heights of trees are similar with that of buildings, and the textures of tree tops are very smooth, it is difficult to differentiate the two classes. Some building edge objects may not have high ratio of the shared border length with building candidates, but have high Roundness values due to the slender shapes resulted from segmentation. Therefore, those objects with Roundness no less than 2.8 or the nDSM of last return greater than or equal to 6 meters, and with the Relative border to building candidates no less than 0.4, are classified as building candidates. The thresholds for the Roundness and the nDSM of last return are chosen based on the knowledge of study area, since the heights and shapes of building edges can vary for different types of buildings. Up to this step, not all building edges are extracted due to the noise within building edge objects caused by the partial penetration of laser beam emitted by LiDAR sensor. Thus the height values in those building edge objects are lower than that in others. The remaining building edge objects can be identified by limiting value range of the nDSM of last return, which is set between 3.5 and 5.3 meters from empirical observation, and the Relative border to building candidates greater than or equal to 0.5. The remaining objects in this step are finally assigned to the unclassified class (Figure 2.8).

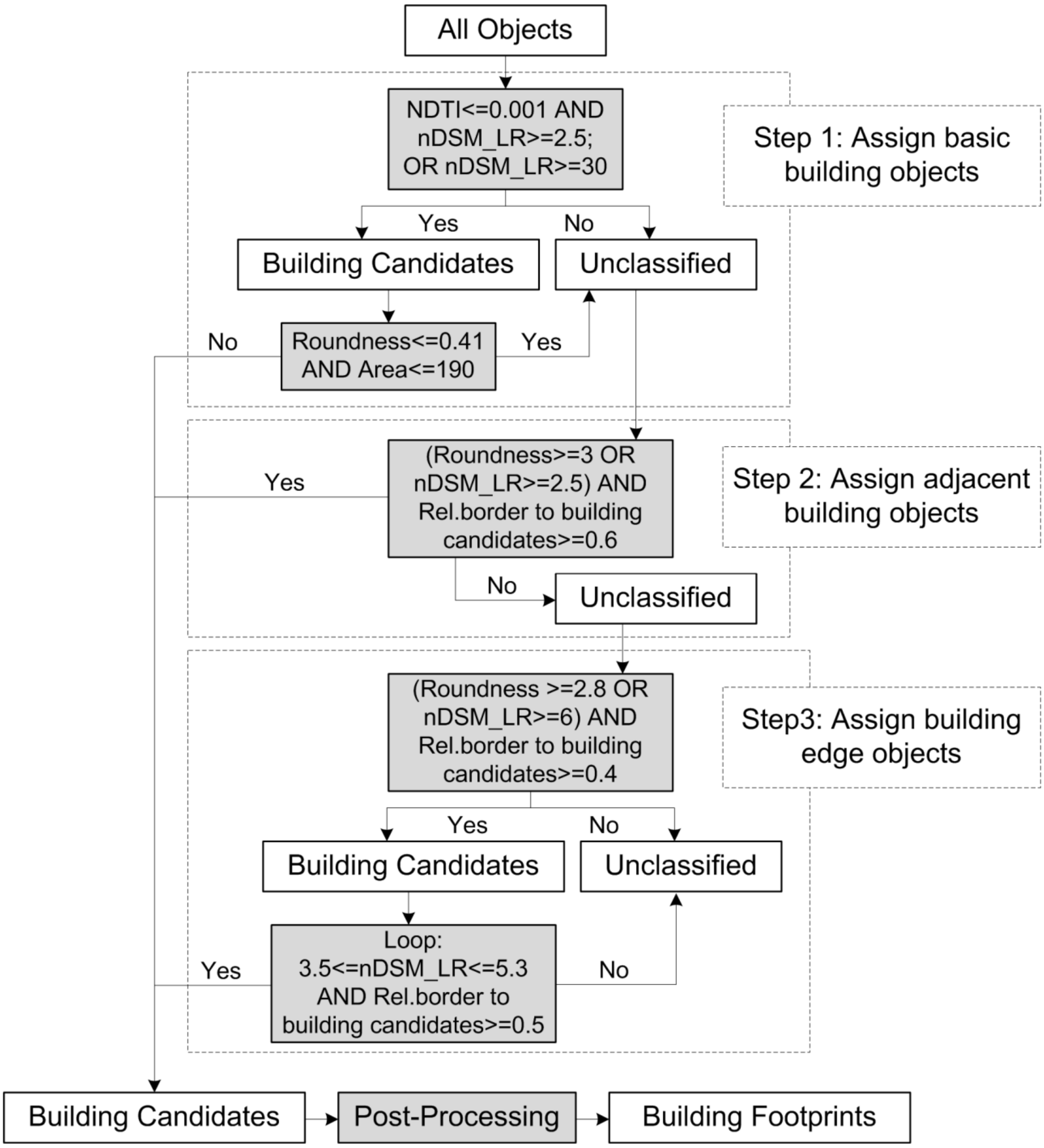


Figure 2.8 **Flowchart of the proposed object-based classification of building footprints**

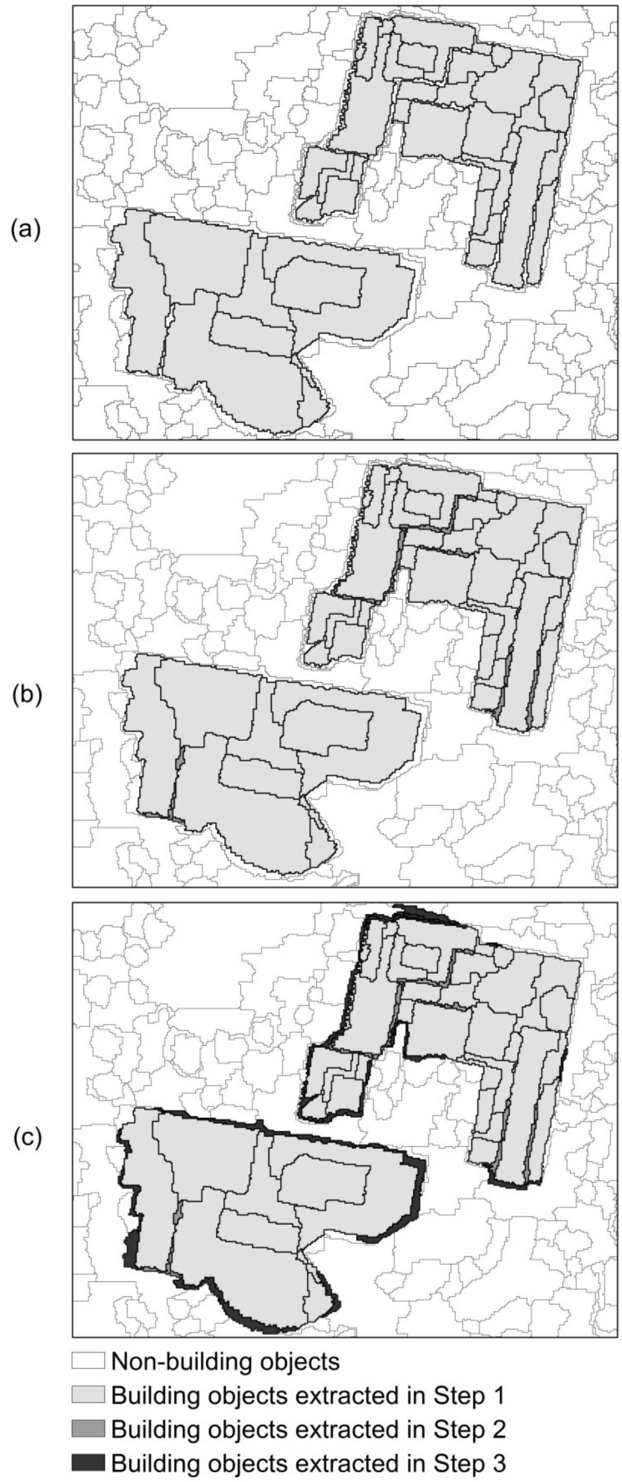


Figure 2.9 **Illustration of the extracted building objects step by step. (a) Identified building objects after step one; (b) Identified building objects after step two; (c) Identified building objects after step three.**

2.2.6 Post-processing

After the rule-based classification, post-processing is necessary to remove small trees that are mistakenly classified as building candidates. Since some building objects have similar sizes with that of tree objects, geometric difference between the two classes are applied. Small tree objects appear to be round while building objects with similar sizes have rectangular shape. Firstly, based upon the empirical knowledge of the study area, building objects with area smaller than a user-defined threshold are reclassified as unclassified. Secondly, the parameter "Rectangular Fit" is used to partition building and tree objects with areas larger than the user-defined threshold. Finally the building footprints are squared up and exported as the final result.

2.2.7 Accuracy Assessment

In this study, the polygon layer of building footprints from City of London in 2006 is used as reference data to assess the performance of the proposed object-based extraction of building footprints. The *Union* overlay between the reference building layer and extracted building layer is performed to generate the required inputs for the accuracy. An illustration of the *Union* overlay is shown in Figure 2.10.

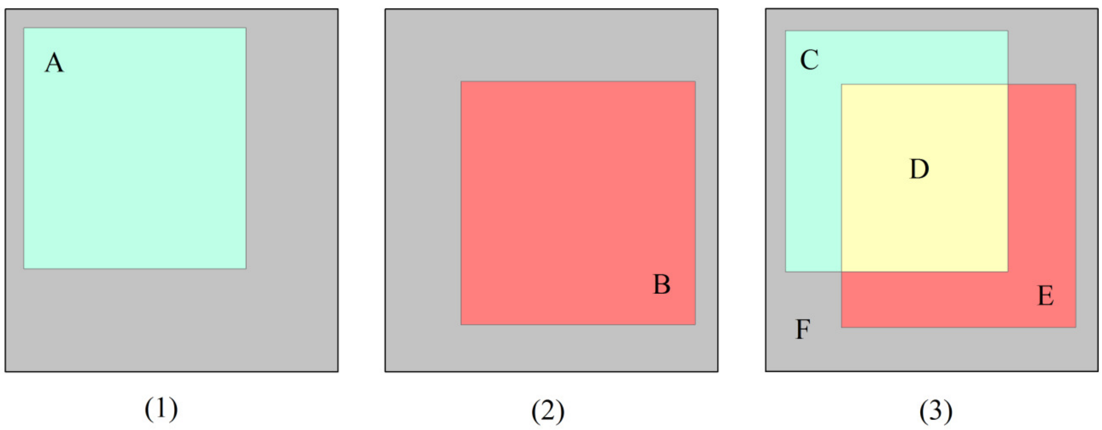


Figure 2.10 **Illustration of the *Union* overlay. (1) Extracted building; (2) reference building; (3) overlay of the extracted and reference buildings.**

Consider Polygon A is the extracted building footprint, and Polygon B is the reference building footprint, the grey areas in (1) and (2) are non-building areas. The result of

Union overlay is shown in (3). Polygon D is the overlaid region between A and B, indicating this part of the extracted building footprint is classified correctly. Likewise, Polygon F represents the non-building region where both A and B lies outside, implying this part of the non-building area is classified correctly as well. Polygon C is actually non-building region but mistakenly classified as buildings; similarly, Polygon E actually represents real building footprint yet is classified as non-building region. The areas of polygons in (3) can be calculated as (c), (d), (e) and (f), which are the necessary variables for the confusion matrix calculations. A confusion matrix for two classes, Building and Non-building can be computed as follows:

Table 2.2 **The confusion matrix for two classes, Building and Non-building**

	Reference data		Row total	User's accuracy
Classified data	Building	Non-building		
Building	(d)	(c)	(c+d)	(d)/(c+d)
Non-building	(e)	(f)	(e+f)	(e)/(e+f)
Column total	(d+e)	(c+f)	(c+d+e+f)	
Producer's accuracy	(d)/(d+e)	(c)/(c+f)		

Overall accuracy = (d+f)/(c+d+e+f)
Kappa = (d+c)*(d+ f)–((d+c)*(d+e)+(e+f)*(c+f))/((c+d+e+f)^2–(d+c)*(d+e)*(e+f)*(c+f))
Omission error = 1 – Producer's accuracy
Commission error = 1 – User's accuracy

(Lillesand et al., 2008)

The producer's accuracy indicates the probability of the reference objects being correctly classified, while the omission error corresponds to those objects belonging to the class of interest that the classification has failed to recognize. The user's accuracy indicates the possibility that the classified objects actually represents that category on the ground, yet the commission error represents those objects from other classes that the classification has assigned to the class of interest. Since the overall accuracy only incorporates the major diagonal information and excludes the omission and commission errors for each class, Kappa is used to integrate off-diagonal elements for the comprehensive statistical calculations. Kappa analysis yields a K_{hat} statistic (an estimate of Kappa) that is a measure of agreement or accuracy (Rosenfield and Fitzpatrick-Lins, 1986; Congalton, 1991). The higher the overall accuracy and the Kappa, as well the lower commission error the results are, the better the classification performs.

2.3 Results

To quantitatively compare the performance of the proposed object-based technique in this study, several attempts with different combinations of parameters for the object-based building footprint extraction are made, and three contrastive object-based classifications are performed: (1) use LiDAR nDSM data only (M1); (2) use LiDAR nDSM and intensity data (M2); (3) use LiDAR nDSM and NDTI data (M3). For the three methods, the same segmentation scale, hierarchical procedures and post-processing afterwards are applied as the proposed object-based classification (M4). In addition, the building extraction algorithm from the commercial software LiDAR Analyst™ (M5) is applied in the same study area. LiDAR Analyst™ has an interface which allows users to set up parameters for building extraction (e.g. height, area, roof gradient). The accuracy of the three object-based classification strategies and LiDAR Analyst™ are all compared with that of the proposed object-based classification.

Both M1 and M2 have over 1% higher overall accuracies than the proposed method, but the respective commission error of 37.1% and 36.7% and a low Kappa of 0.42 and 0.43 indicate buildings and trees are very poorly distinguished; many trees are mistakenly classified as building objects. With the assistance of NDTI, M3 evidently reduces the commission error to only 8.1% and increases the Kappa to 0.79. The proposed method M4 performs the best among the five methods, with the Kappa of 0.84, overall accuracy of 94.0% and 6.3% commission error. The performance of M5 is worse than M4, with 5.7% and 0.17 decrease in overall accuracy and Kappa, and 7.1% increase in commission error (Table 2.4).

The detailed comparison of the extracted results from the five methods is shown in Figure 2.11. M1 performs the worst among the five methods with the low Kappa of 0.42. This method excessively overestimates the size of building footprints, categorizing too many trees into the building class (column b in Figure 2.11). In the circumstance of sample (5b) in Figure 2.11, small buildings are covered by trees above them, making it difficult to extract complete building footprints from LiDAR. Although those building are extracted, M1 fails to separate buildings from trees. The final extraction result of M1 is shown in Figure 2.12. Not only trees that are adjacent to buildings are classified as buildings, those trees with large areas and smooth surface (east of the study area in Figure 2.12) are incorrectly classified. After adding the intensity data to the object-based classification, the result is slightly improved (column c in Figure 2.11). M2 improves both the accuracy and decreases the commission error by 0.4%, with only 0.01 improvements in Kappa. The final extraction result of M2 is shown in Figure 2.13. Not much improvement is made by adding only intensity to the LiDAR nDSM data. There are still many trees (east of the study area in Figure 2.13) mistakenly classified as buildings.

Table 2.3 **Comparison of the five methods of building footprint extraction**

Method	Technique	Overall accuracy	Commission error	Kappa
M1	Object-based classification with only LiDAR nDSM data	95.0%	37.1%	0.42
M2	Object-based classification with LiDAR nDSM and intensity data	95.4%	36.7%	0.43
M3	Object-based classification with LiDAR nDSM and NDTI data	92.5%	8.1%	0.79
M4	Proposed object-based classification with LiDAR nDSM, NDTI and intensity data	94.0%	6.3%	0.84
M5	LiDAR Analyst™	88.3%	13.4%	0.67

By contrast, the NDTI proved to be very effective for building extraction. M3 evidently reduces the commission error from 37.1% to only 8.1%, meanwhile the accuracy drops slightly and the Kappa evidently increases from 0.42 to 0.79 as a result. However, buildings and trees are still confused by this method. Trees that are adjacent to buildings are segmented together (column c in Figure 2.7), resulting in less satisfactory separation of the two classes ((1d) in Figure 2.11). In addition, due to the absence of the intensity data in the segmentation, objects mixing both buildings and trees can be classified as unclassified, which reduces the overall accuracy ((2c), (3c) in Figure 2.7). The final extraction result using M3 is shown in Figure 2.14. Compared with M1 and M2, the result of M3 evidently improves the classification accuracy with most of the trees classified correctly.

Based on NDTI, in comparison, the proposed method M4 is superior to the previous methods with the help of intensity data, both in extracting more complete building footprints and avoiding mistakenly categorizing other features to building class (Table 2.4 and column e in Figure 2.11). The final classification result of M4 overlaid with the reference data is shown in Figure 2.15. Most of building footprints are detected and extracted completely and the trees are successfully separated from buildings as well, except for some omissions of building class. Compared with M4, many building objects are missed in the classification result of M5 (4f, 5f in Figure 2.11). The final extraction result using M5 is shown in Figure 2.16. Though the "Square Up" function of M5 can smooth building boundaries, some buildings are not adequately regularized, which can lower the overall accuracy due to the inflexible direction of straightening boundaries and the removal of details of building structures (2f, 4f in Figure 2.11). However, this method does not make too much confusion between buildings and trees, with a commission error of 13.4%.

Overall, the separation of buildings and trees is successful using the proposed hierarchical strategies with some useful parameters contributing to the process. A common fault, nevertheless, that presents in all the methods is the deficiency in extracting small buildings. Buildings within Region A and B are not completely detected by the proposed methods. The aerial photo for Region A in figure 2.15 is shown in figure 2.17. The aerial photo was taken on May 1st, 2006. The small buildings in this region are partially covered by trees above. The percentage of the overlap would likely increase by the time of LiDAR data acquisition (May 20th, 2006), considering the growth of leaves. The trees overlap with buildings can influence the overall accuracy of building footprint extraction. The omission error of buildings can be also caused by the special surface material of buildings. The field picture for Region B in figure 2.15 is shown in figure 2.18. The blue lines within Region B indicate the viewing direction of the field picture.

The building with three-part structure is greenhouse made of glass. The glass buildings are not detected due to lasers' partial penetration of glass. Though the proposed technique is able to detect some small buildings, the rest are left out due to not only their spatial and spectral disparity from the larger buildings, but the problem of tree overlap with these buildings.

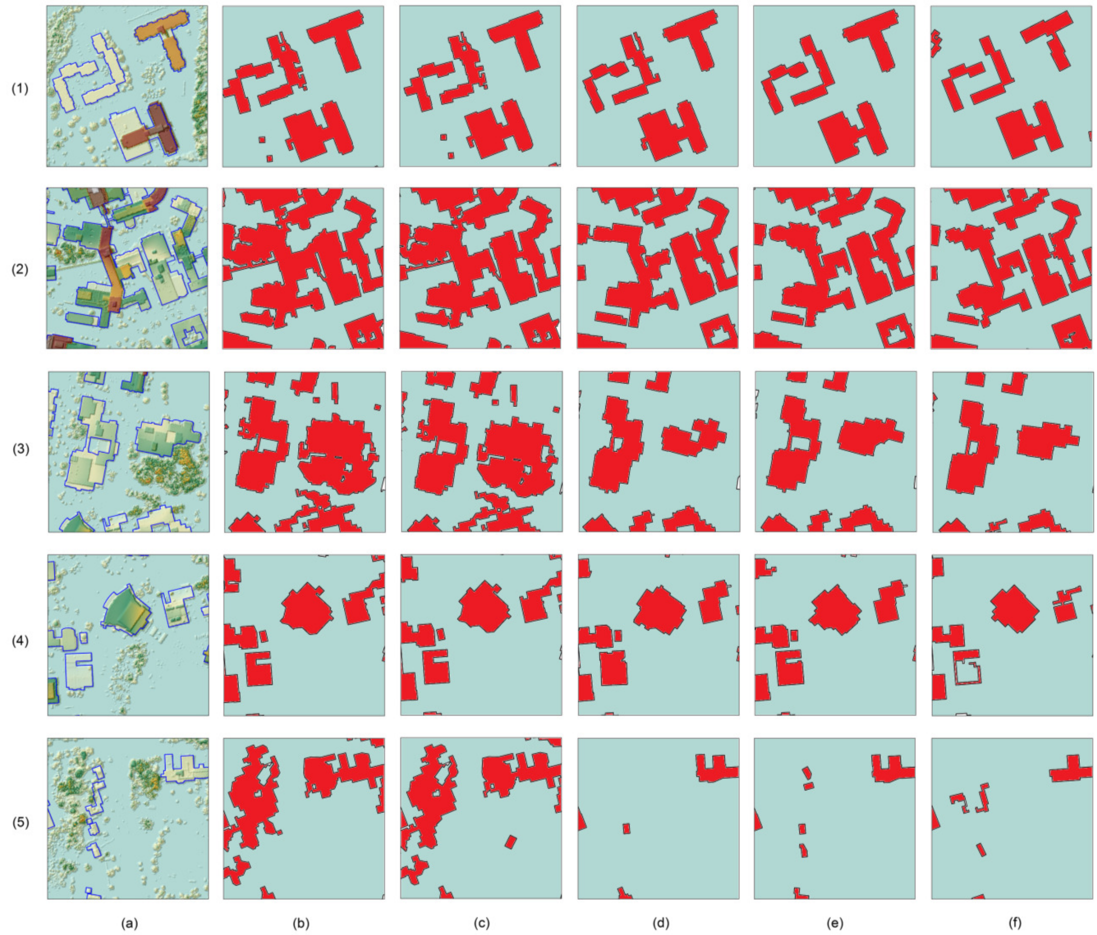


Figure 2.11 **Comparison of the reference buildings and results from the five methods. Five sets of examples for different areas (1), (2), (3), (4) and (5) are illustrated. Column (a) are from the reference building footprints; column (b) are samples of results from the object-based method with LiDAR nDSM data only; column (c) are samples of results from the object-based method with LiDAR nDSM and intensity data; column (d) are from the results of the object-based method with LiDAR nDSM and NDTI data; column (e) are from the results of the object-based**

method with LiDAR nDSM, intensity and NDTI data; and column (f) are samples of results from the LiDAR Analyst™ algorithm.

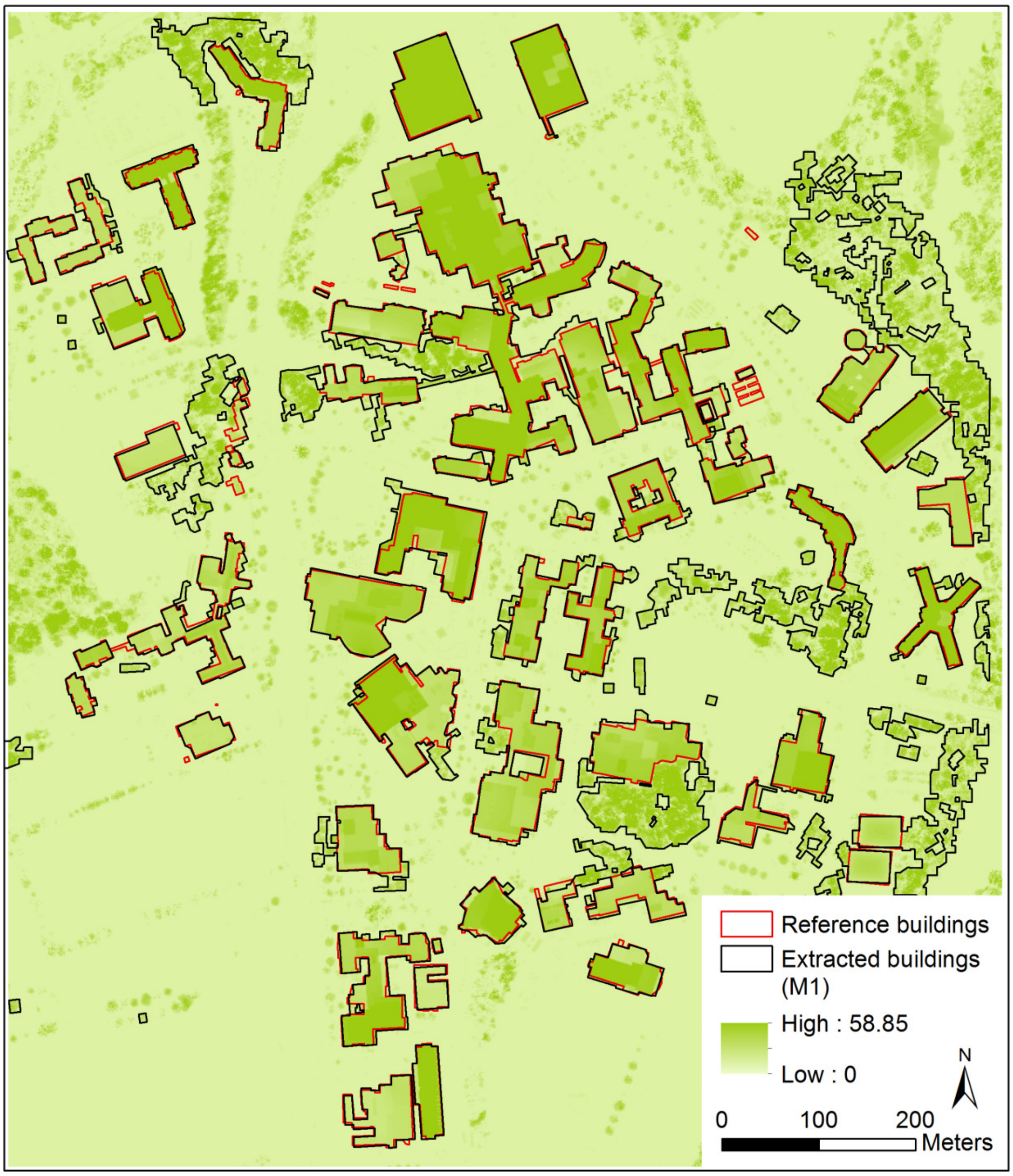


Figure 2.12 **Extraction result of building footprints using the object-based method with LiDAR nDSM data only (M1). The background is the image of nDSM of the last return data.**

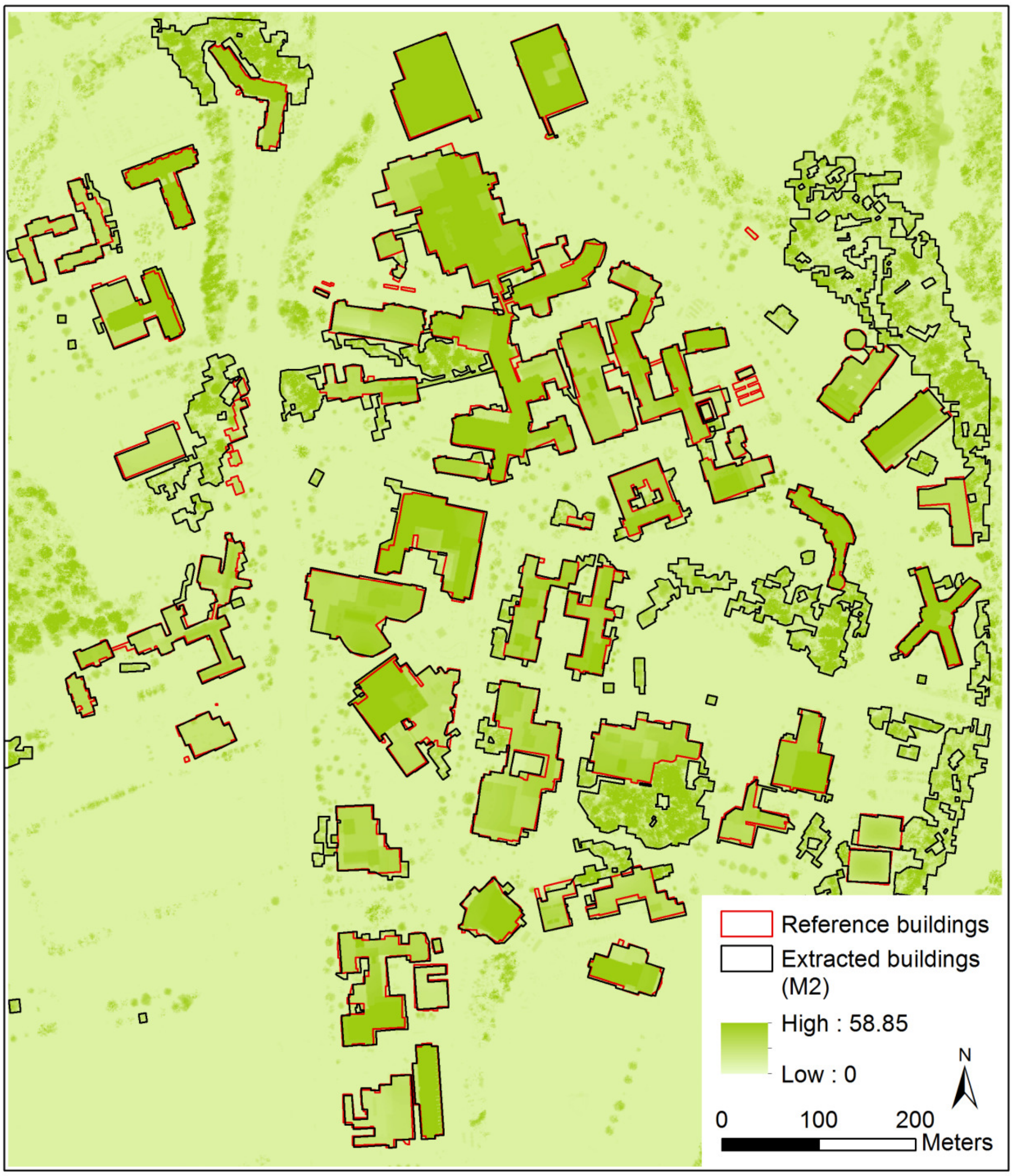


Figure 2.13 **Extraction result of building footprints using the object-based method with LiDAR nDSM and intensity data (M2). The background is the image of nDSM of the last return data.**

Figure 2.14 **Extraction result of building footprints using the object-based method with LiDAR nDSM and NDTI data (M3). The background is the image of nDSM of the last return data.**

Figure 2.15 **Extraction result of building footprints using the proposed method (M4). The background is the image of nDSM of the last return data.**

Figure 2.16 **Extraction result of building footprints using LiDAR Analyst™ (M5). The background is the image of nDSM of the last return data.**

Figure 2.17 **Aerial photo of Region A in figure 2.15.**

Figure 2.18 **Field picture of Region B in figure 2.15.**

2.4 Conclusions and Discussions

Several new techniques are evaluated in this chapter for object-based classification in building footprint extraction using airborne LiDAR data. Though traditional optical imagery are mostly used in building footprint extraction in the past decades, performance of the derived methods are largely limited by sun shadows, relief displacement and occlusion effects presented in the images. LiDAR data are more efficient for building extraction compared with optical imagery, providing highly accurate vertical information with multiple returns and intensity information. In this study, detailed procedures are presented for the pre-processing, the integrated segmentation, the object-based classification strategies and accuracy assessment.

The object-based classification presented here is proved to be a very effective semi-automated method of extracting building footprints from LiDAR imagery in urban environments. The result of the proposed object-based classification outperforms the other methods with the Kappa of 0.84, the overall accuracy of 94.0% and commission error of 6.3%. Compared with LiDAR Analyst™ algorithm, the proposed method has 0.17 higher Kappa, 5.7% higher overall accuracy and 7.1% lower commission error, which means it extracts building more accurately and also performs better to avoid categorizing other land use types to buildings.

The NDTI is used for the object-based building extraction. The results of the three contrastive methods indicate that NDTI plays a very important role in the object-based building extraction. The object-based method with only LiDAR nDSM data has the highest commission error, while adding NDTI successfully reduces the error by 29.0%.

The LiDAR intensity, when combined with NDTI in the object-based method, produces the best result. Though the object-based classification with LiDAR nDSM and intensity data slightly improves the extraction result from the initial one, the intensity data contribute to the final result by improving 1.5% for accuracy and decreasing 1.8% for commission error making the joint effort with NDTI in the proposed method. Though the separation of buildings and trees remains a problem in previous studies, the proposed method in this paper successfully distinguishes trees from buildings.

2.5 References

Aldred, D. and Wang, J. 2011.A method for obtaining and applying classification parameters in object-based urban rooftop extraction from VHR multispectral images. *International Journal of Remote Sensing*, Vol. 32, No.10, pp. 2811-2823.

Antonarakis, A., Richards, K., and Brasington, J. 2008. Object-based land cover classification using airborne LiDAR, *Remote Sensing of Environment*, Vol. 112, 2988-2998.

Arefi, H., Halm, M., and Lindenberger, J. 2003. LiDAR data classification with remote sensing tools. *Proceedings of the ISPRS Commission IV Joint Workshop: Challenges in Geospatial Analysis, Integration and Visualization II*, 08-09 September, Stuttgart, Germany, pp. 131-136.

Arefi, H., Hahn, M., 2005. A morphological reconstruction algorithm for separating off-terrain points from terrain points in laser scanning data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*36 (Part 3/W19), pp. 120–125.

Baatz, M. and Schape, A., 2000. Multiresolution Segmentation – an optimization approach for high quality multi-scale image segmentation. *In: Angewandte Geographische Informationsverarbeitung XII, Strobl, J., Blaschke, T., Griesebner, G. (eds.) AGIT Symposium, Salzburg, Germany.* Wichmann: Heidelberg, pp. 12-23.

Blaschke, T. 2010. Object based image analysis for remote sensing. *ISPRS Journal of photogrammetry and Remote Sensing*, Vol. 65, No. 1, pp. 2-16.

Brennan, R. and Webster, T.L. 2006. Object-oriented land cover classification of lidar-derived surfaces. *Canadian Journal of Remote Sensing*, Vol. 32, No. 1, pp. 162-172.

Brunn, A., Weidner, U. 1997.Extracting buildings from digital surface models. *IAPRS International Archives of Photogrammetry and Remote Sensing*, Vol. 32, No. W2, 3D Reconstruction and Modeling of Topographic Objects.

Congalton, R.G. 1991. A review of assessing the accuracy of classifications of remotely sensed data. *Remote Sensing of Environment*, Vol. 37, pp. 35-46.

Davis, D. 2005. Evaluation of alternatives in obtaining structural elevation data. *Federal Emergency Management Agency (FEMA) Report*. Vol. 136, pp. 1-2. Available online at <http://www.fema.gov/pdf/nfip/alt_elevations/elevations_final.pdf>.

Definiens, 2010.Definiens Developer 8.0.1 Reference Book. Definiens AG, München, Germany.

El-Ashmawy, N., Shaker, A., Yan, W.Y. 2011.Pixel vs object-based image classification techniques for LiDAR intensity data. *ISPRS Workshop on Laser Scanning, Elsevier*, Vol. XXXIII, Calgary, Canada.

Frauman, E., and Wolff, E.2005. Segmentation of very high spatial resolutions satellite images in urban areas for segments-based classification. *Proceedings of the 3rd International Symposium Remote Sensing And Data Fusion Over Urban Areas*, 14-16 March 2005, Tempe, Arizona.

Goodwin, N. R., Coops, N. C., Tooke, T. R., Christen, A., and Voogt, J. A., 2009. Characterizing urban surface cover and structure with airborne lidar technology. *Canadian Journal of Remote Sensing*, Vol. 35, No. 3, pp. 297-309.

Huang, M. J., Shyue, S. W., Lee, L. H., and Kao, C. C. 2008.A knowledge-based approach to urban feature classification using aerial imagery with LiDAR data.*Photogrammetric Engineering & Remote Sensing*, Vol. 74, No. 12, pp. 1473-1485.

Jensen, J.R. 2000.Remote Sensing of the Environment. *Prentice-Hall*, Upper Saddle River, New Jersey.

Kaasalainen, S., Ahokas, E., Hyyppa, J., and Suomalainen, J. 2005. Study of surface brightness from backscattered laser intensity: calibration of laser data. *IEEE Geoscience and Remote Sensing Letters*, Vol. 2, No. 3, pp. 255-259.

Lee, D.S., Shan, J., and Bethel, J.S.2003.Class-guided building extraction from Ikonos imagery. *Photogrammetic Engineering and Remote Sensing*, Vol. 69, No. 2, pp. 143–150.

Lee, D.H., Lee, K.M., Lee, S.U. 2008.Fusion of LiDAR and imagery for reliable building extraction. *Photogrammetric Engineering and Remote Sensing*, Vol. 74,No.2, pp. 215-225.

Lehrbass, B. and Wang, J. 2012. Urban tree cover mapping with relief-corrected aerial imagery and LIDAR. *Photogrammetric Engineering and Remote Sensing*.Vol.78, No. 5, pp.473-484.

Lillesand, T.M., Kiefer, R.W., and Chipman, J.W., 2008.Remote Sensing and Image Interpretation. *John Wiley & Sons,* Inc., Hoboken, New Jersey.

Lin, X.G., Zhang, J.X., and Shen J. 2011.Object-based classification of airborne LiDAR point clouds with multiple echoes. *2011 International Symposium on Image and Data Fusion (ISIDF)*. TengChong, YunNan Province, China, August 9 – 11, 2011.

Ma, R. 2005. DEM generation and building boundary detection from LiDAR data. *Photogrammetic Engineering and Remote Sensing*, Vol. 71, No. 7, pp. 847-854.

Mazzarini, F., Pareschi, M. T., Favalli, M., Isola, I., Tarquini, S., and Boschi, E. 2007. Lava flow identification and aging by means of LiDAR intensity: Mount Etna case. *Journal of Geophysical Research*, Vol. 112, No.B2, pp. B02-201.

Meng, X., Wang, L., and Currit, N. 2009.Morphology-based building detection from airborne LiDAR data. *Photogrammetric Engineering and Remote Sensing*, Vol. 75, No. 4, pp. 437-442.

Miliaresis, G., Kokkas, N. 2007. Segmentation and object-based classification for the extraction of the building class from LIDAR DEMs. *Computer& Geosciences*, Vol. 33, pp. 1076-1087.

Morgan, M., Tempfli, K. 2000.Automatic building extraction from airborne laser scanning data. *in Proceeding of 19th ISPRS Congr., Amsterdam*, The Netherlands, pp. 616–623, Book 3B.

Rosenfield, G.H. and Fitzpatrick-Lins, K. 1986. A coefficient of agreement as a measure of thematic classification accuracy, *Photogrammetric Engineering and Remote sensing*, Vol. 52, No. 2, pp. 223-227.

Rottensteiner, F., Briese, C., 2002, A mew method for building extraction in urban areas from high-resolution LiDAR data. *IAPRS International Archives of Photogrammetry and Remote Sensing*, Vol. 34, No. 3A, pp. 295-301.

Rottensteiner, F., 2003, Automatic generation of high-quality building models from LiDAR data. *IEEE Computer Graphics and Applications*, Vol. 23, No.6, pp. 42-50.

Shan, J. and Toth, C.K., 2009, Topographic laser ranging and scanning: principles and processing. *CRC Press*, Taylor & Francis, London, pp. 307-334.

Shufelt, J.A. and Mckeown, D.M. 1993.Fusion of monocular cues to detect man-made structures in aerial imagery. *Computer Vision and Graphic Image Understanding*, Vol. 57, No. 3, pp. 307-330.

Sohn, G., Dowman, I.2007. Data fusion of high-resolution satellite imagery and LiDAR data for automatic building extraction. *ISPRS Journal of Photogrammetry & Remote Sensing*, Vol. 62, pp. 43-63.

Song, J.H., Han, S.H., Yu, K.Y., Kim, Y.I. 2002.Assessingthe possibility of land cover classification using LiDAR intensity data. *ISPRS Commission III, Symposium,* September 9-13, 2002, Graz, Austria.

Strassopolou, A., Caelli, T., and Ramirez, R. 2000.Automatic extraction of building statistics from digital ortho-graphs. *International Journal of Geographic Information System and Science*, Vol. 14, No. 8, pp. 759–814.

Vu, T.T., Yamazaki, F., Matsuoka, M.2009.Multi-scale solution for building extraction from LiDAR & imagery data. *International Journal of Applied Earth Observation and Geoinformation*, Vol. 11, pp. 281-289.

Wang, C., Glenn, N.F. 2009.Integrating LiDAR intensity and elevation data for terrain characterization in a forested area. *IEEE Geoscience and Remote Sensing Letters*, Vol. 6, No. 3, pp. 463-466.

Wang, J., Lehrbass, B., Zeng, C.2011.Urban building mapping using LiDAR and relief-corrected Colour-Infrared aerial images. *Proceedings of the 34th International Symposium on Remote Sensing of Environment*, Sydney, Australia. April 10-15, 4p.

Wang, J.,Zeng, C., Lerhbass, B. 2012.Building extraction from LiDAR and aerial images and its accuracy evaluation. *Proceedings of the 2012 IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2012),* Munich, Germany. July 22-27, 4p.

Weidner, U. 1997. Digital Surface Models for building extraction. *Automatic Extraction of Man-Made Objects from Aerial and Space Images (II)*, Birkhauser, Basel, S.193-202.

Zhang, K., Yan, J., and Chen, S. 2006.Automatic construction of building footprints from airborne LIDAR data. *IEEE Transaction on Geoscience and Remote Sensing*, Vol. 44, No. 9, pp. 2523-2533.

Zhou, W., and Troy, A. 2008.An object-oriented approach for analyzing and characterizing urban landscape at the parcel level. *International Journal of Remote Sensing*, Vol. 29, No. 11, pp. 3119-3135.

Zhou, W., Huang, G., Troy, A. 2009. Object-based land cover classification of shaded areas in high spatial resolution imagery of urban areas: A comparison study. *Remote Sensing of Environment*, Vol. 113, pp. 1769-1777.

Chapter 3

GIS –based Reconstruction of 3D Building Models from Airborne LiDAR Data

3.1 Introduction

3.1.1 Background

The three dimensional (3D) building model has become one of the essential components in various scientific and engineering applications, such as urban planning, digital city visualization, virtual tourism information systems (Zhang et al., 2009; Elberink and Vosselman, 2011), and wireless telecommunication which need stereo building models as input in spatial databases, virtual reality tools or wave propagation simulators, etc. (Kabolizade et al., 2012). Other applications such as environment protection and planning, microclimate study (e.g. urban heat island effect and rain-runoff modeling), or emergency response (Kim and Shan, 2011) also require high accuracy 3D building models driven by the more strict standards and the improvement of complex prediction systems. Moreover, the accurate reconstruction of building models from remotely sensed images is likely to open up a range of potential GIS applications.

3.1.2 Previous Studies

In the past decades, traditional high resolution optical images (e.g. QuickBird, GeoEye and WorldView) have been and still are widely used to construct Digital Surface Models (DSMs), a representation of elevations of objects on the Earth surface, which 3D building models can be derived from (Shan and Toth, 2009). DSMs are usually obtained by automatic image matching algorithms applied on stereo pair of aerial or high-resolution satellite images (Noronha and Nevatia, 2001; Gong et al., 2002). The results of these algorithms are satisfactory for smooth terrain at small to medium scale. However, the automatic performances decrease for complex scenes in dense urban areas at large-scale, which can be mainly caused by inadequate image understanding, sun shadows, relief displacement, occlusions, poor or repeated textures, poor image quality or lack of man-made objects modeling (Lehrbass and Wang, 2012; Zhou et al., 2004; Zhou et al., 1999).

Even semi-automatic techniques require at least the support of the recognition of very complex buildings by a human operator (Brenner 2005).

As an alternative data source for DSM generation, Light Detection And Ranging (LiDAR) techniques have gained increasing popularity in the recent practices and studies (Satari et al., 2012; Verma et al., 2006; Wang and Chu, 2009). Compared with the other methods of DSM generation, LiDAR measurements are not only free from sun shadows, relief displacement and occlusion effect, but also offer several advantages such as fast data acquisition, high point density, strong capability of measuring both height and planimetric location and canopy penetration (Beraldin et al., 2010). Therefore, detailed roof structures can be determined from LiDAR data, based on which 3D building models can be reconstructed.

Most reported LiDAR-based building reconstruction approaches follow three common processing steps (Figure 3.1). First, the non-ground objects are separated from the ground objects, using LiDAR point clouds or interpolated grid data. Morphological filters (Elaksher and Bethel, 2002; Arefi and Hahn, 2005; Zhang et al., 2009) are commonly applied in this step. Detailed comparisons are presented in Sithole and Vosselman (2004). Building footprints are then detected from the non-ground objects. The next step is to segment LiDAR data or building reference data into segments representing single roof planes. The main objective of this step is to determine roof primitives (e.g. outline, intersected lines and intersected nodes). Various methods have been studied, including region growing (Rottensteiner, 2003), gradient orientation (Forlani et al., 2006), Hough transform (Vosselman and Dijkman, 2001) and RANSAC (RANdom SAmple Consensus) (Tarsha-Kurdi et al., 2007). However, these approaches always need the assistance of dominant building orientation to determine roof primitives such as edge direction. Though some studies segment building reference data instead of LiDAR data (Rottensteiner, 2003), the reference data are not always available and can be outdated.

In the final step, building models are reconstructed, for which two general approaches are applied: the data-driven and the model-driven. For the data-driven method, building models are reconstructed by comparing and assembling the roof segments. However, topological relations among segments are difficult to determine (Elberink and Vosselman, 2009, Sampath and Shan, 2010). For instance, topological relation can refer to "there should be no gaps or overlaps between two adjacent roof planes". The model-driven method defines a database of roof forms or a basic formulation representing many primitives to fit for the detected roof planes. This approach can always reconstruct a topologically consistent model (Kim and Shan, 2011), but the reconstruction process may be less satisfactory if other primitives or complicated models are not considered.

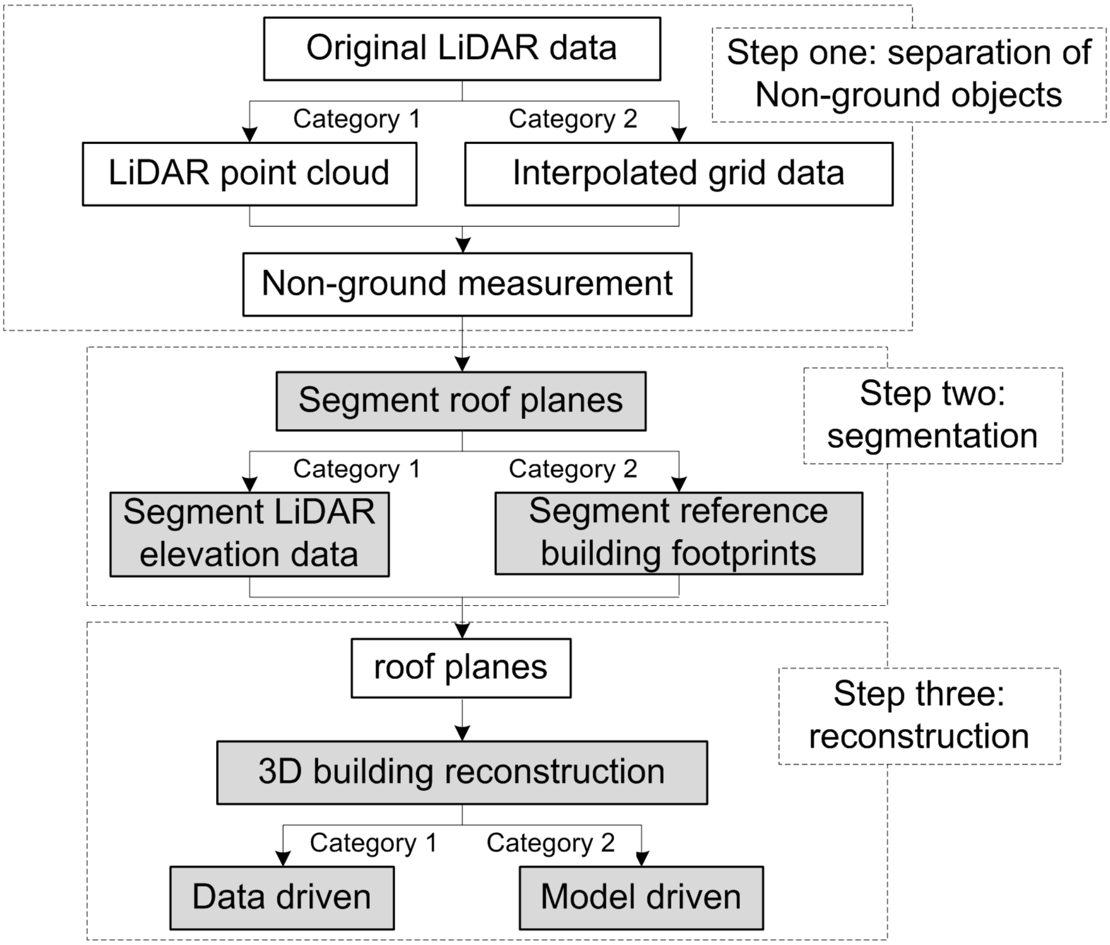


Figure 3.1 **Summary of LiDAR-based building reconstruction methods and steps**

Although many strategies of detecting roof planes for the determination of building primitives are used in the previous research, there are still some unsolved problems such as the disconnection of roof planes and the difficulty locating height discontinuities of roof planes (Kim and Shan, 2011). An alternative solution that can avoid the aforementioned limitations is to extract the intra building edges directly without segmentation. However, few studies can identify edges from LiDAR data without the assistance of other data sources such as incorporating a 2D map (Alexander et al., 2009) or fusing with aerial images (Zhou et al., 2004; Yong, 2011). Therefore, developing techniques of the detection and processing of building edges from LiDAR data directly is worthwhile.

3.1.3 Objectives

Based on previous literature, the objective of this study is to develop a reconstruction strategy of 3D building models from LiDAR data in a GIS environment. In the proposed method, the intra building edges are directly detected and processed from the LiDAR first return data, types of roof shapes (flat and pitched) are classified using the slope of the nDSM data. Different strategies are applied for the visualization of flat roof planes and pitched roof planes, based on which 3D building models are finally reconstructed.

3.2 Methods

3.2.1 Data Acquisition and Study Area

The LiDAR data are acquired by Applied Geomatics Research Group (AGRG) with the Optech ALTM 3100 sensor. The data are collected in the north part of London, Ontario on May 20th of 2006 (Figure 3.2). The aircraft is equipped with an inertial measurement unit (IMU) and high-precision global positioning system (GPS) used to locate and orient the laser returns in three-dimensional space (Brennan and Webster, 2006). The coordinates and elevations of all LiDAR points are recorded using the UTM zone17 coordinate system relative to the NAD83 datum. The experimental site is chosen over the campus of University of Western Ontario, where various kinds of building roof shapes (flat, pitched and complicated) are included to develop 3D reconstruction strategy for building models in a complex urban environment.

The 2D building footprints are extracted using a rule-based object-oriented classification of LiDAR data from Chapter 2, and are used as input data in this study. Thus building footprints extracted directly from LiDAR data can increase the applicability of the proposed method without depending on more data sources.

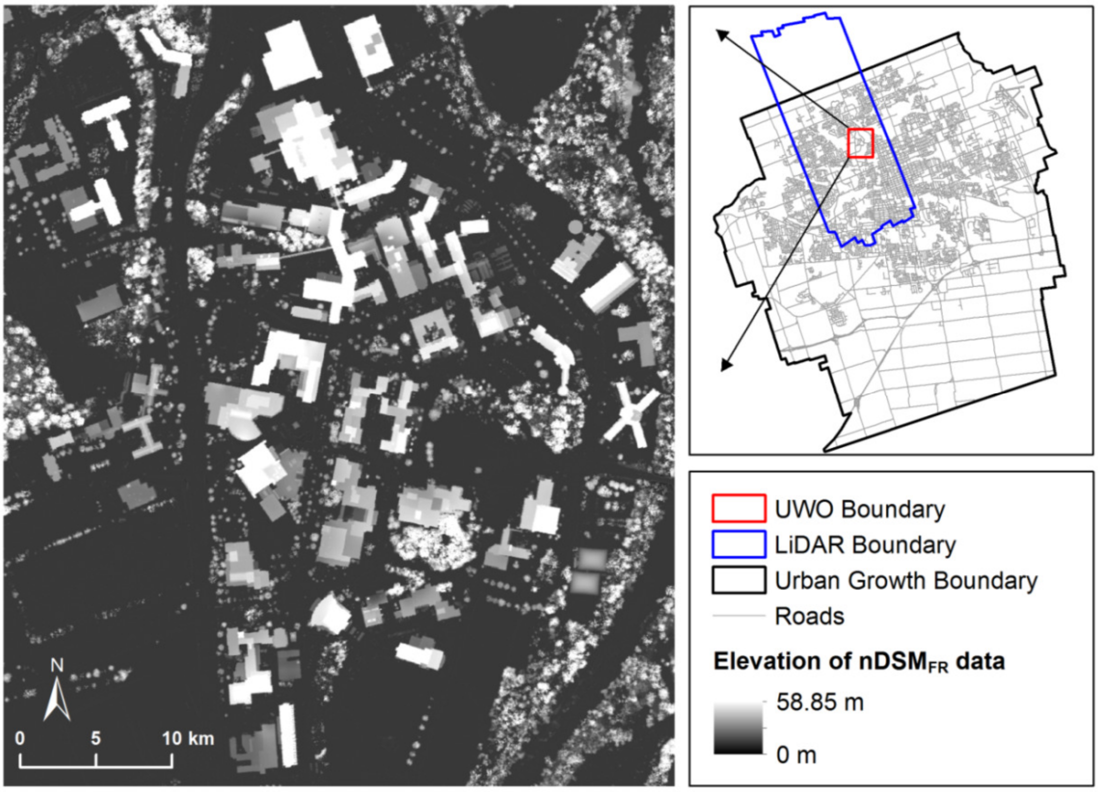


Figure 3.2 **Data and study area**

3.2.2 Overview

The proposed 3D building reconstruction method is illustrated in Figure 3.3, which includes the following steps: pre-processing to interpolate LiDAR point clouds to raster images, edge detection and processing, boundary processing incorporated with closed polygons acquired from the previous step and the 2D building footprints, classification of roof shape types, and 3D building models reconstruction. A detailed description of the reconstruction strategy is shown in Appendix 3.

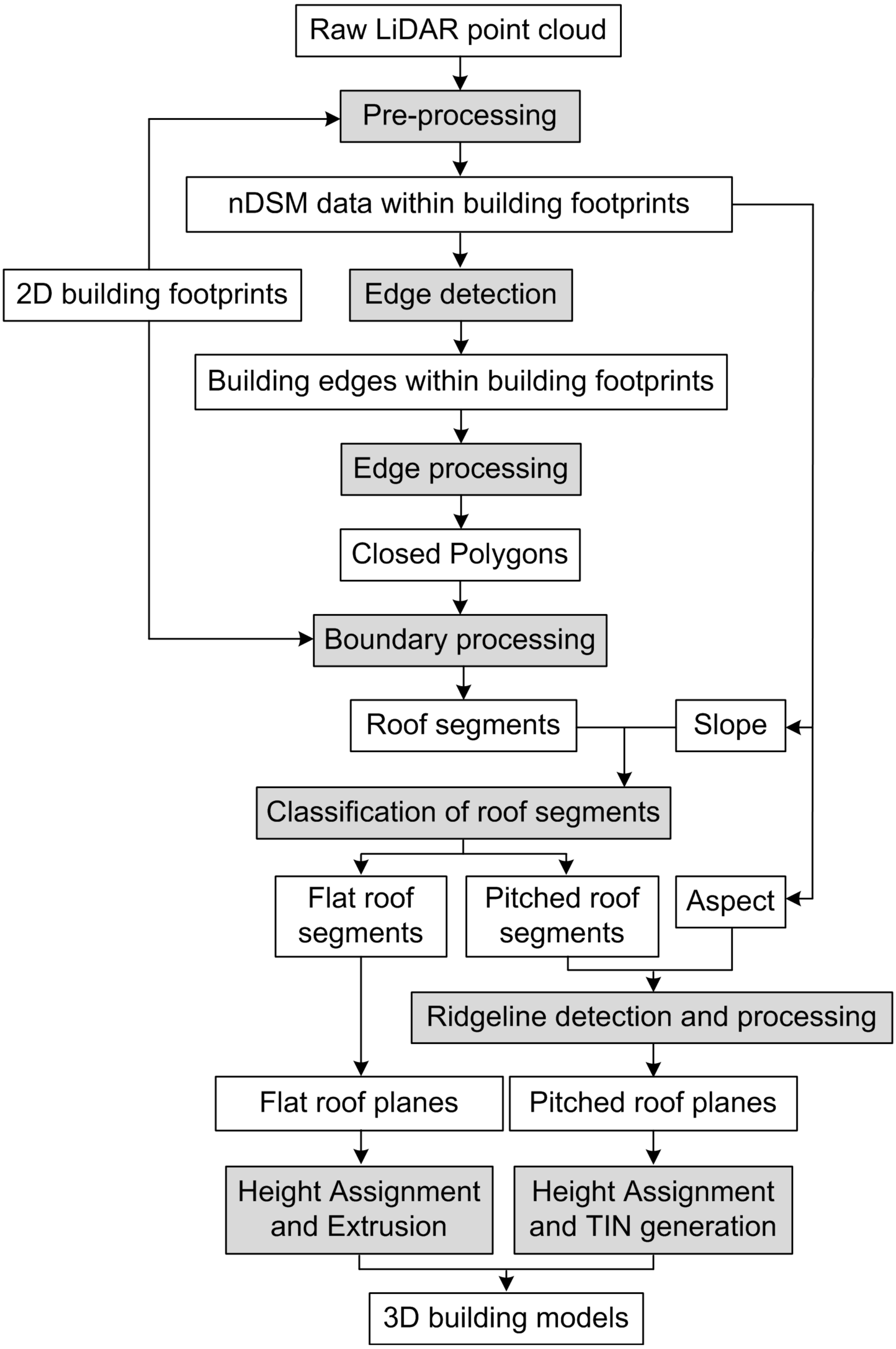


Figure 3.3 **Flowchart of methodology**

3.2.3 Pre-processing

Firstly the raw LiDAR point clouds are interpolated to generate Digital Terrain Model (DTM) data that represent the elevation of the bare earth surface. The DSM of the first return data are interpolated from the first-return point clouds. The normalized DSM (nDSM) of the first return, which is the subtraction between the DSM and DTM, are calculated and used to identify the height of buildings with terrain effect eliminated. The nDSM of building areas are cropped from the extracted 2D building footprints to exclude the influence of trees. The slope and aspect of the cropped nDSM are also generated to classify roof shape types and detect ridgelines respectively in the following steps. Data including the cropped nDSM, the derived slope and aspect data, and the extracted 2D building footprints are used as input for 3D building reconstruction.

3.2.4 Edge Detection

The Canny edge-detection algorithm is selected to extract edge information within buildings. For comparisons of the performance of different edge detection operators, see Appendix 2.The Canny detector was first developed by Canny, (1986) using a multi-stage algorithm to identify a wide range of edges in images.

According to Canny, (1986), three performance criteria should be followed for edge detection: (1) Good detection. The detection should mark as many real edge points as possible and have low probability of falsely identifying non-edge points; (2) Good localization. The distance between the points marked as edge points by the operator and the center of the true edge should be minimized; (3) Multiple response criteria. There should be only one edge point considered true when there are multiple responses of points to the same edge.

Based on the criteria mentioned above, five main steps are included in the Canny edge detection. The first step is to smooth the image with a Gaussian filter to decrease noise in the image. The second step computes the gradient orientation and magnitude in x and y direction respectively using a pair of 3 by 3 convolution masks. In the third step, the directions of the edges are calculated and a set of values (0, 1, 2, or 3) are used to assign to the directions according to the sectors they belong. The next step is to use a non-maximum suppression to form ridges with one-pixel width at the edge points whose gradient magnitudes are the local maximum in the direction of the gradient. The final step is to fill up the gaps along the ridges. Any pixel in the non-maximum suppression image that has a value greater than a user-defined high threshold is marked as an edge pixel immediately. Then any pixels that are connected to this edge pixel and that have a value greater than a user-defined low threshold are also selected as edge pixels. All edges are followed until the value drops below the low threshold (Canny, 1986).

Three parameters need to be specified for the Canny edge detector: sigma, low threshold and high threshold. Sigma is the standard deviation of the Gaussian filter, of which typical values are from 0.6 to 2.5. Low threshold and high threshold are the hysteresis threshold values. The typical value range of low threshold is between 0.2 and 0.5 and that of high threshold is within 0.6 and 0.9. A three-level loop is set to iterate the Canny algorithm with different combination of the three parameters.

Based on empirical observation, the performance of the edge detection is regarded the best when the sigma is set to 0.7, with the high threshold 0.6 and the low threshold 0.4 among more than 300 edge detection images. Three examples are illustrated in Figure 3.4. From the visual interpretation of the hillshade of the nDSM data, A, B, C, D, E, F and G are individual roof segments. The parameter combination used in this study is able to detect the outlines of those segments which are important parts of the roof structures, yet the other parameter combinations fail to recognize. Moreover, B and C are pitched roof segments, the edges of which should be identified first and then used to detect ridgelines.

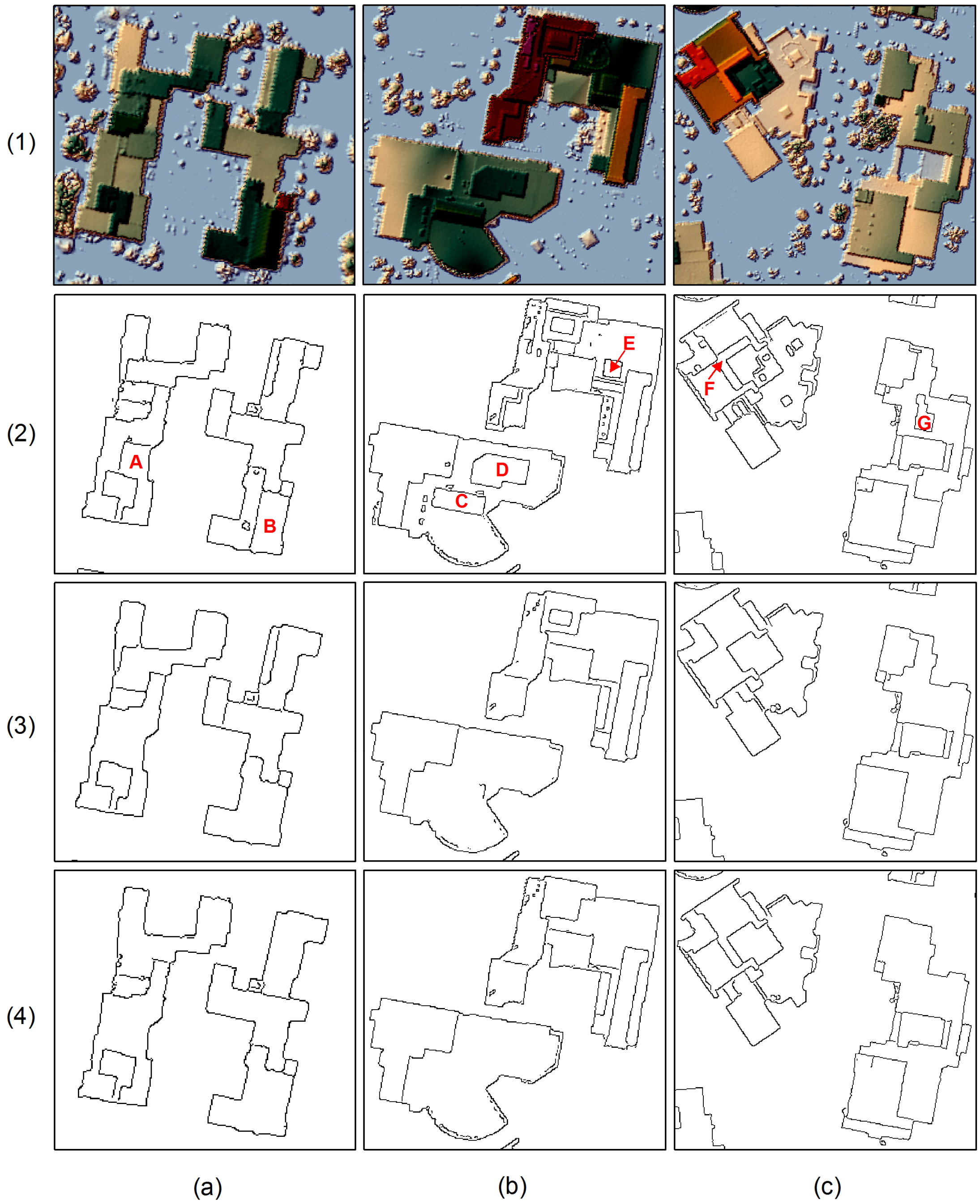


Figure 3.4 **Canny edge detection results from the nDSM of first return data. Three sets of samples (a), (b) and (c) are illustrated with different parameter combinations. (1) The hillshade image of the nDSM data; (2) sigma 0.7, high threshold 0.6 and low threshold 0.4; (3) sigma 1.2, high threshold 0.6 and low threshold 0.4; (4) sigma 0.7, high threshold 0.7 and low threshold 0.3.**

3.2.5 Edge and Boundary Processing

After edge detection, the building edges need to be vectorized for further processing. This step is automatically performed on the binary image of the building edges using the "Vectorization" function from the ArcScan module of ArcGIS 10. The clean-up is followed by removing polylines with length shorter than 5 meters.

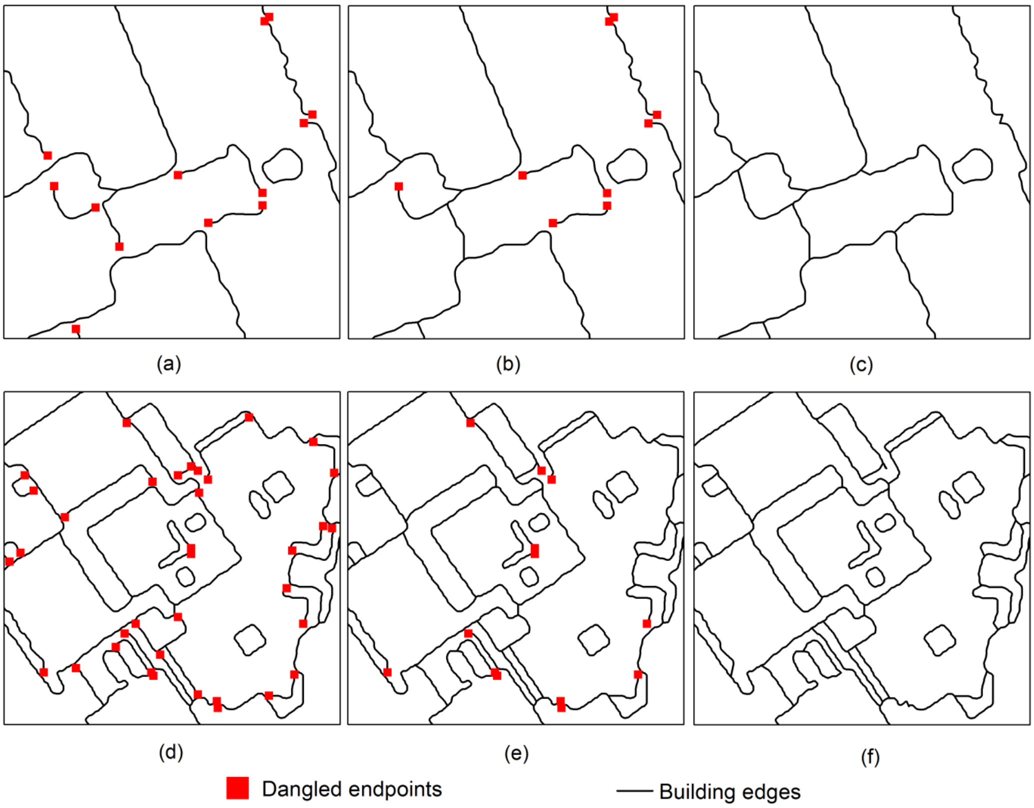


Figure 3.5 **Illustration of edge processing. (a) and (d): before processing; (b) and (e): after extending iteration on the dangled endpoints; (c) and (f): after snapping.**

Since not all the edges are connected in the edge detection process, the endpoints of polylines that "dangle" (Figure 3.5 (a) and (d)) are detected by validating the topology rule "Must Not Have Dangles (Line)" implemented to the vectorized edges in ArcGIS 10. An iteration with different levels of cluster tolerance thresholds (2 meters, 4 meters, 6 meters etc.) based on empirical observation is set for the "Extend" function to extend the dangled endpoints until most of them are joined to the polylines encountered on the way in the extending direction (Figure 3.5 (b) and (e)). The "Snap" function is then used to attach the rest of dangled endpoints to the nearest polylines (Figure 3.5 (c) and (f)). After edge processing, the closed edge polylines are used to generate polygons that represent roof segments, among which small polygons with areas less than 20 m^2 are merged with neighboring polygons by dropping the shared border. The neighboring polygons refer to the ones that have the longest shared border with those small polygons.

In order to demonstrate the benefit of edge processing, roof segments without edge processing are generated to compare with those that have been processed. Since the outlines of some buildings cannot even be generated without the help of edge processing (gaps between detected edges are not eliminated), the later boundary processing is still used to form at least building boundaries. A comparison between the roof segments with and without edge processing is shown in Figure 3.6. The result shows that the roof structures without edge processing are simplified to a great extent. The proposed edge processing better restores the building roof structures.

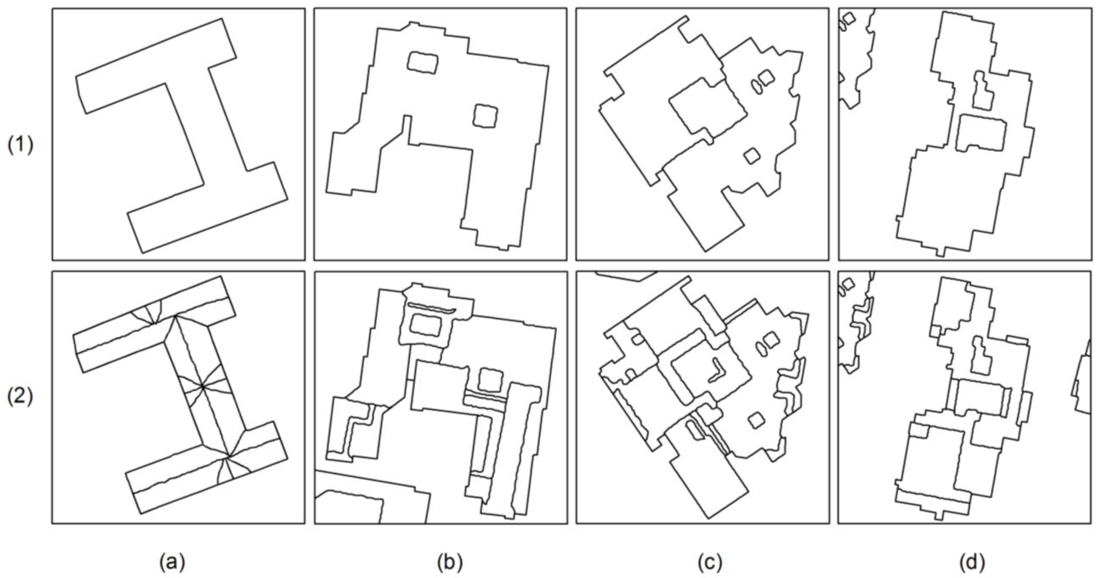


Figure 3.6 **The generated roof segments with and without edge processing. (1) Roof segments without edge processing; (2) Roof segments with edge processing**

Boundary processing is then needed in the following step, considering that the building edges are detected within the extent of the building footprints, the outlines of the polygons generated from the building edges (Figure 3.6 (a)) do not perfectly overlap with the extracted building footprints (Figure 3.7 (b)). The slivers (pink areas shown in Figure 3.7(c)) are identified by overlapping the polygons and the extracted building boundaries. Only the slivers within the building footprints are merged to the neighboring polygons with the longest sharing border. The revised polygons are shown in Figure 3.7(d).

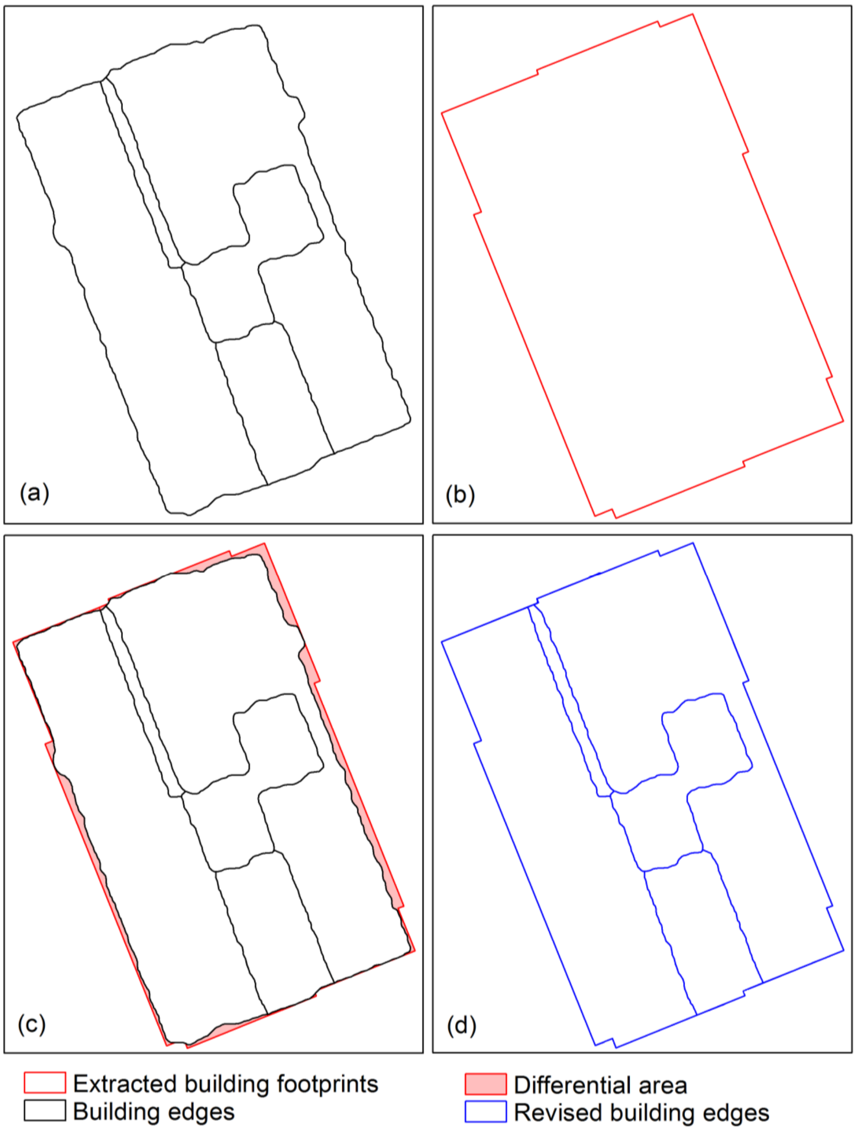


***Figure 3.7* Illustration of building boundary processing. (a) The original building edges; (b) the extracted building footprint; (c) after the union of (a) and (b); (d) the revised building edges**

3.2.6 Classification of Roof Shape

After the roof segments are detected and processed, the roof shape types should be classified to distinguish flat and pitched structures. Firstly, the average heights and slopes are assigned to the corresponding roof segments using Zonal Statistics in ArcGIS 10. The slope of the nDSM data is used to classify the building roof segments, for which two types of roof shape (flat and pitched) are considered.

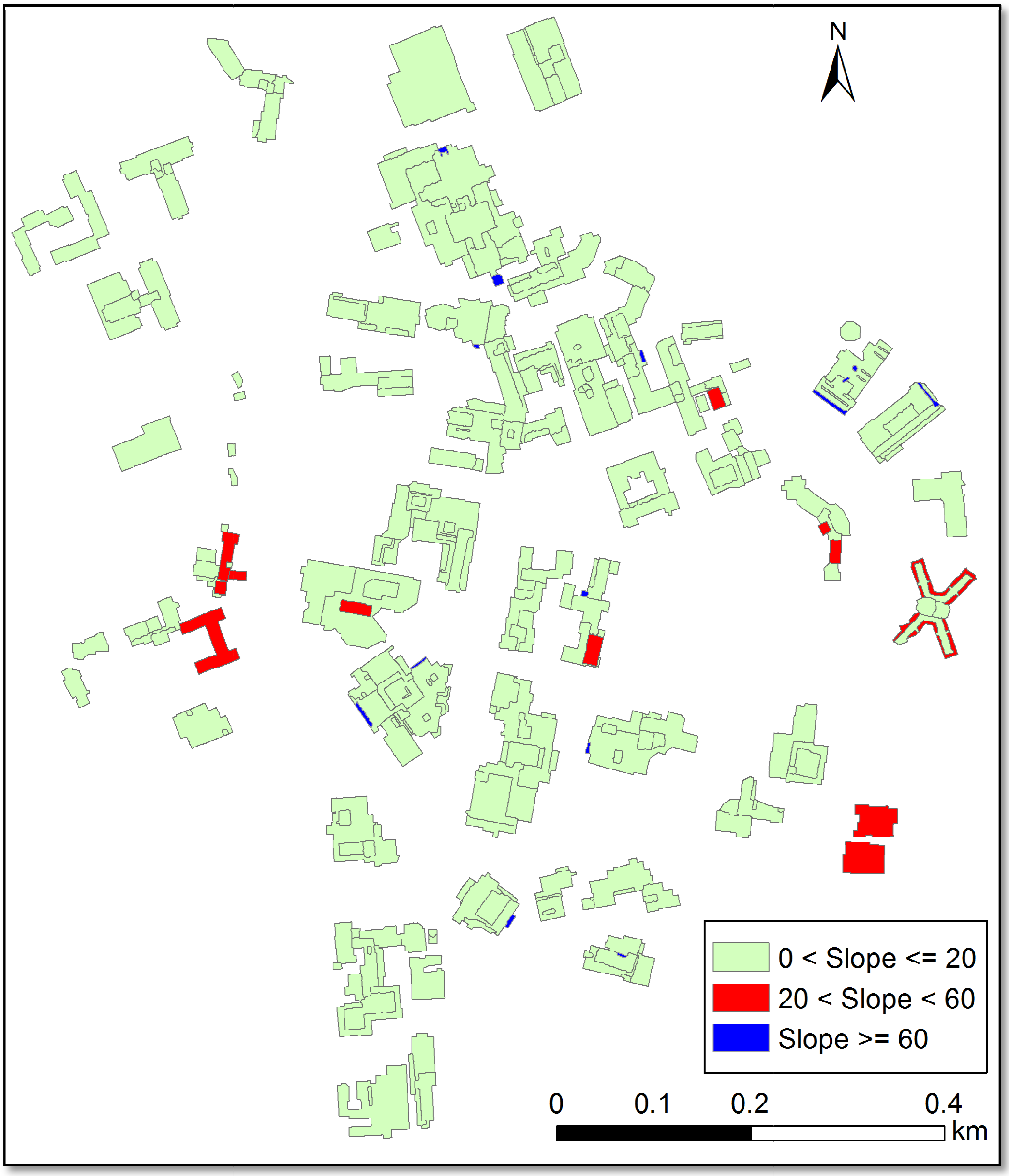


Figure 3.8 **Mean slope values of the detected roof planes**

Based on the author's knowledge of the study area, there are many pipe structures on top of the building roofs, which increase the average slope values for the roof segments, thus the slope threshold applied here is a little larger than the usual cases. In this study, slopes of less than 20 degrees mostly indicate flat roof segments, and pitched roof segments are associated with slopes between 20 and 60 degrees. Very few roof segments have slopes of greater than 60 degrees, most of which are located in the areas with high image noises or that with sudden change of elevation such as building boundaries and jump edges. Those roof segments with slopes greater than 60 degrees are actually flat roof segments thus should be assigned as flat. Therefore, roof segments with average slopes less than 20 degrees are classified as flat and those with average slopes in the range of 20 and 60 degrees as pitched. Those roof segments with slopes of more than 60 degrees are considered as flat in this study (Figure 3.8).

3.2.7 3D Building Reconstruction and Visualization

Up to this step, both flat and pitched roof segments have been identified. For the flat, each roof segment already represents one roof plane. Thus the flat-roof buildings can be reconstructed easily by extruding each roof segment within the flat building to the respective average height. Compared with flat roof segments, roof segments with pitched surface are more difficult to reconstruct since they may contain multiple roof planes and cannot just be extruded to a universal height. The following steps describe the reconstruction techniques for the pitched roof segments.

(1) Detect ridgelines within pitched roof segments

Even though the pitched roof segments are detected from the classification of roof shape types, the intra-structures of which are not well identified using the Canny detector because there are no obvious elevation difference at ridgelines in the nDSM data. Considering that the aspect image of the nDSM data can clearly distinguish different roof planes of the pitched roof segments without the need for defining the dominant building orientation, the aspect is used to classify sub-segments within the pitched roof segments. The ridgelines are identified by applying the Canny edge detector to the aspect image within the boundaries of the pitched roof segments (Figure 3.9 (a)). Similar edge processing is employed to create connected polylines (Figure 3.9 (b)). Up to this step, each region within the polylines represents a single pitched roof plane.

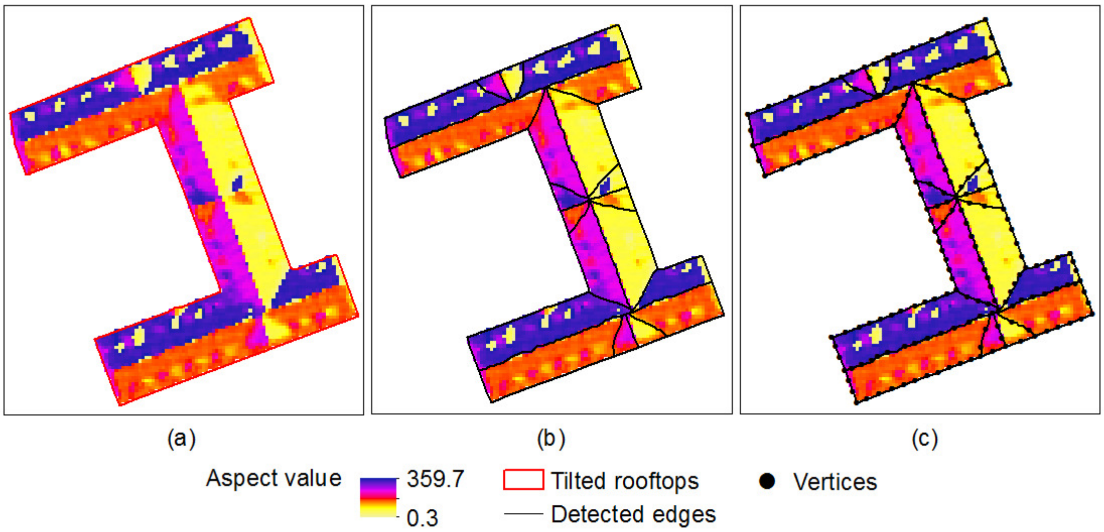


Figure 3.9 **Illustration of pitched roof segment reconstruction. (a) Aspect value within the pitched roof segment; (b) Detected edges within the pitched roof segment; (c) densified vertices along the edges of pitched roof segment.**

(2) Assign averaged height values to the densified vertices

Those polylines are then densified in ArcGIS 10 by inserting vertices along polylines so that there are enough and evenly distributed vertices for creating 3D structures in the visualization (Figure 3.9 (c)). The densified vertices of these planes are assigned with the corresponding height values of the nDSM data, thus every vertex has an elevation value representing the building height in that location. However, the assigned heights of those vertices cannot be all the same due to many reasons (e.g. the positional accuracy of roof segment outlines and ridgelines, noise along building edges caused by interpolation), which can affect the visualization performance. To compensate for this problem, the vertices located at the same edges (outlines or ridgelines) are selected respectively, and the mean height values of the vertices are then calculated and re-assigned to each vertex.

(3) Generate Triangulated Irregular Network (TIN) structures

In the following step, the vertices with re-assigned height values are used to construct Triangulated Irregular Network (TIN) structures. Since the vertices from one pitched roof segment can possibly form triangle nets with vertices from another pitched roof segment, the pitched roof polygons are used as masks on the vertices so that the derived TIN structures are completely within each pitched roof polygon. A visual comparison between the TIN structures with and without the aforementioned re-assignment step is shown in Figure 3.10. The TIN structures without height value averaging have many sags and crests, some roof planes even have overlaps with other planes, making the visualization less accurate and appealing. In contrast, the TIN structures generated from the processed points are much better visualized.

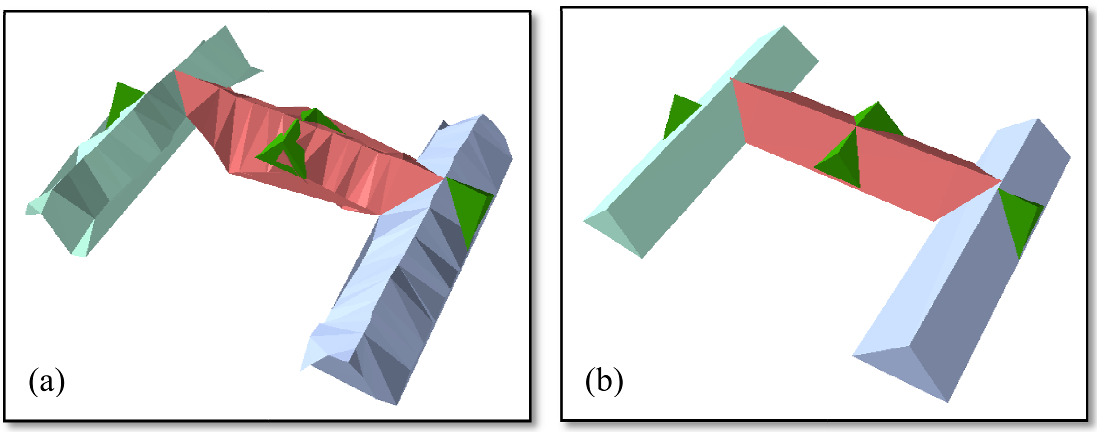


Figure 3.10 **Performance of the height value re-assignment. (a) TIN structure derived from vertices without height value re-assignment; (b) TIN structure derived from vertices with height value re-assignment.**

Different visualization strategies are applied to the flat and pitched roof planes. For the flat, the roof planes within the same buildings are assigned to a same ID number, and the average height of each roof segment is used to extrude the segment polygon by adding the average height to the base height of the buildings. In this study, the base height is the elevation in the DTM data. For the pitched, the base frames and the rooftops are dealt with separately. The average heights of the densified vertices located at the roof segment boundaries are assigned to the corresponding roof segments, which are extruded by adding the assigned values to the base height to represent the base frames. Finally the TIN structures of the pitched roof segments are piled upon the base frames (Figure 3.11).

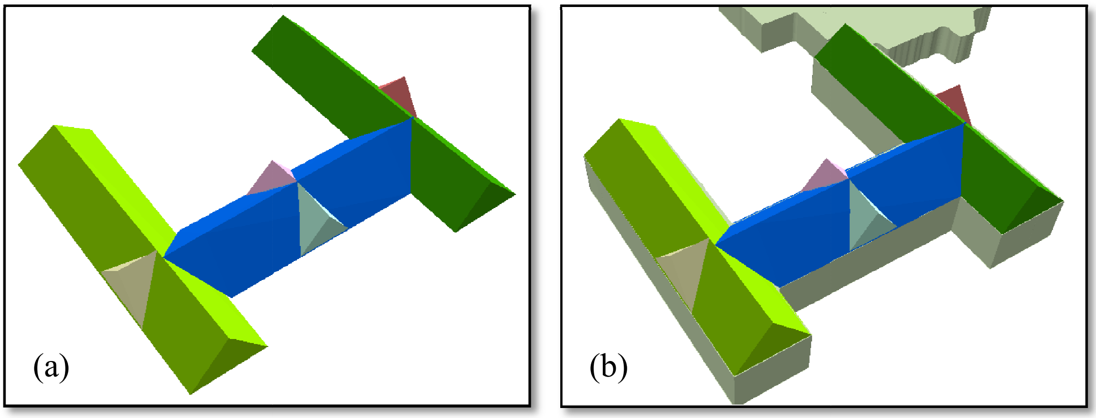


Figure 3.11 **Visualization of the pitched rooftop. (a) TIN structures of the pitched rooftop; (b) visualization of the whole building after adding the base frame.**

3.3 Results

The proposed approach is applied to the LiDAR data in the study area. The panoramic view of the reconstructed 3D building models is shown in Figure 3.12, and two enlarged views are displayed in Figure 3.13 and Figure 3.14. Moreover, the buildings within the red circle in Figure 3.12 are visualized with texture information (Figure 3.15).

The final visualization is able to show not only the structures of the flat roof segments (structures with colors from red to blue in Figure 3.12 to 3.14), but also that of the pitched roof segments (dark green structures in Figure 3.12 to 3.14) as well. Compared with the visualization of buildings with single heights, multiple-roof visualized buildings are more appealing and close to the real situation.

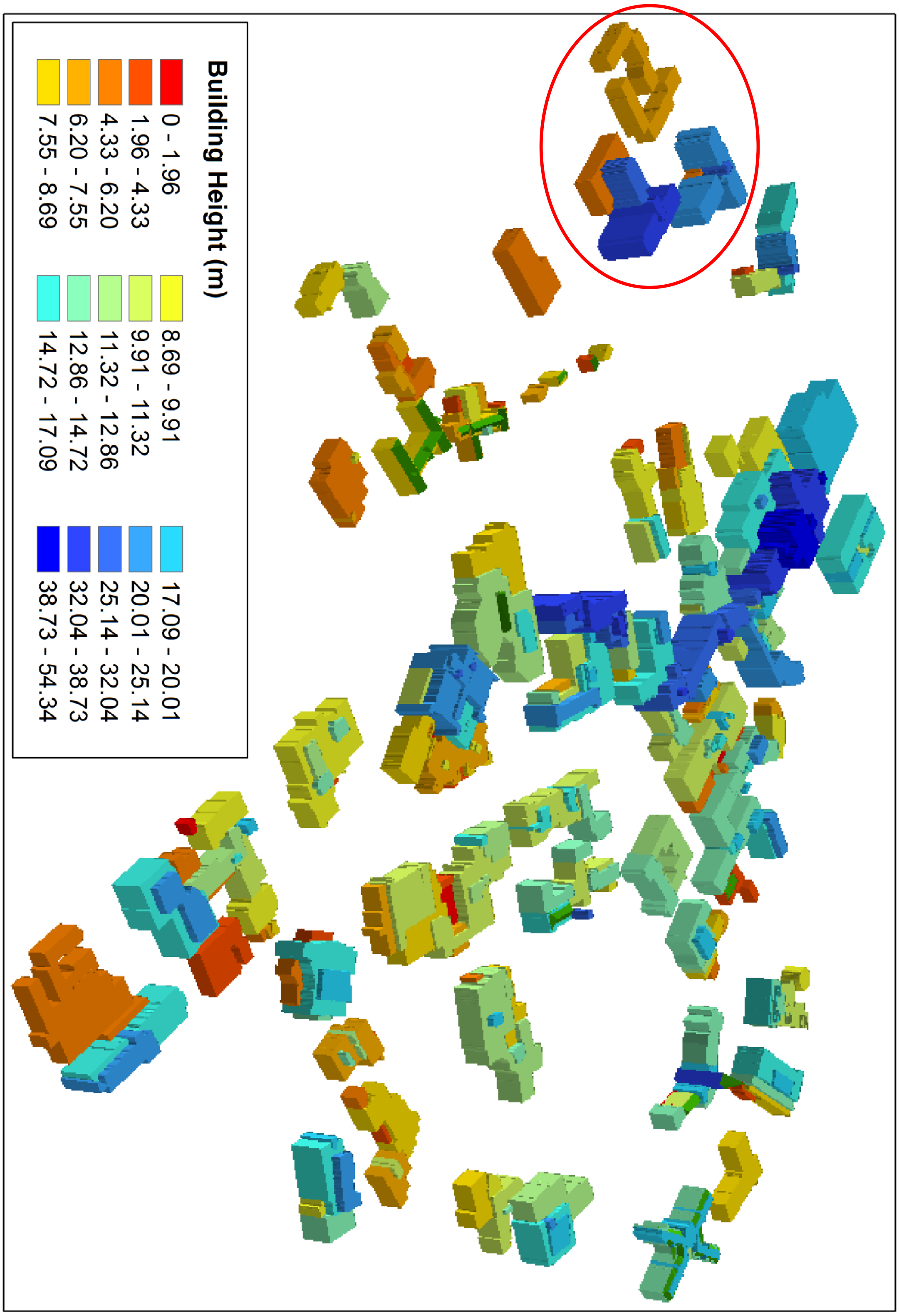


Figure 3.12 **Panoramic view of 3D building reconstruction**

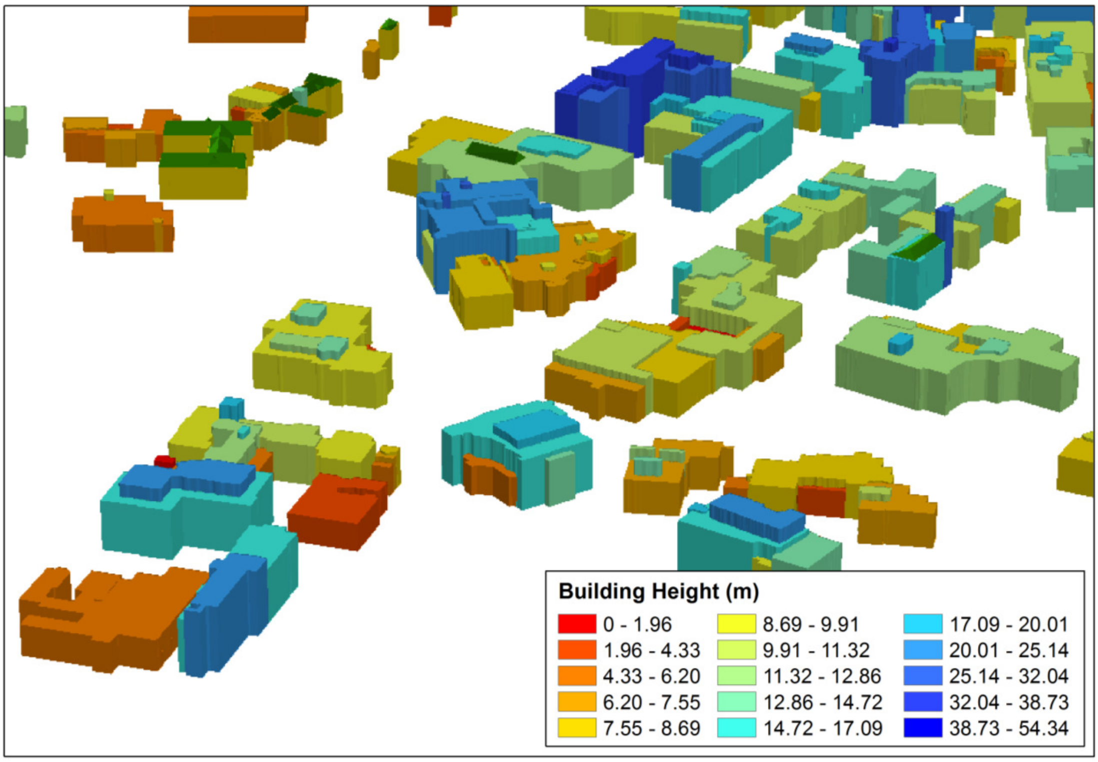


Figure 3.13 **Partial view 1 of 3D building reconstruction**

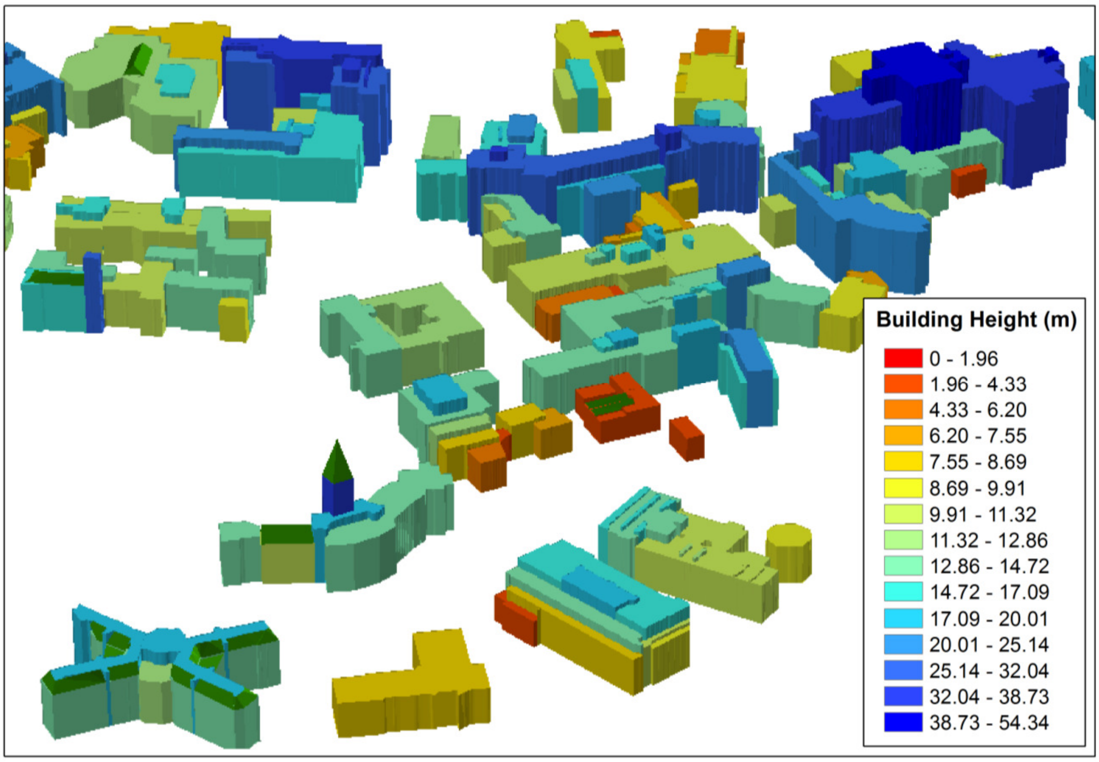


Figure 3.14 **Partial view 2 of 3D building reconstruction**

Figure 3.15 **Visualization of 3D buildings with texture information**

3.3.1 Accuracy Evaluation of LiDAR Elevation Data

In this study 50 building heights are measured and used as reference to assess the accuracy of LiDAR elevation data. The heights are acquired using a laser rangefinder (Model: TruPulse™ 300). The TruPulse™ 300 equipment is capable of measuring horizontal distance, vertical distance, slope distance and inclination measurements with ±0.3 m accuracy. Since the x, y coordinates of building corners measured by Garmin Etrex – Vista GPS can have considerable errors due to its ±3-5m accuracy and the poor satellite signals blocked by buildings, the validation points shown in Figure 3.16 are manually digitized according to the photos took during height surveys. Originally there were 50 building heights collected in the survey, but the time difference between LiDAR data acquisition (year 2006) and the survey (year 2012) may cause some height inconsistencies since there may be building construction (e.g. tear down buildings, construct more levels) during this interval. After examination, 38 points are chosen for assessing the accuracy of LiDAR elevation data. Those validation points are assigned with heights measured by TruPulse™ 300. The corresponding height values in the LiDAR nDSM data are extracted as well to compare with the reference heights. The differences between reference heights and heights extracted from LiDAR for the 38 points are calculated in Table 3.1. The average height difference is 0.25 meters, and the standard deviation is 0.29 meters. The scatter (xy) plot is drawn in Figure 3.17 based on the values in Table 3.1. The slope rate of the trendline is 1.004, which is very close to 1. Considering the accuracy of height measured by TruPulse™ 300 is averaged 0.3 - 0.5 meters, the LiDAR derived nDSM data are regarded very accurate and suitable for 3D building modeling.

Figure 3.16 **Locations of the validation points**

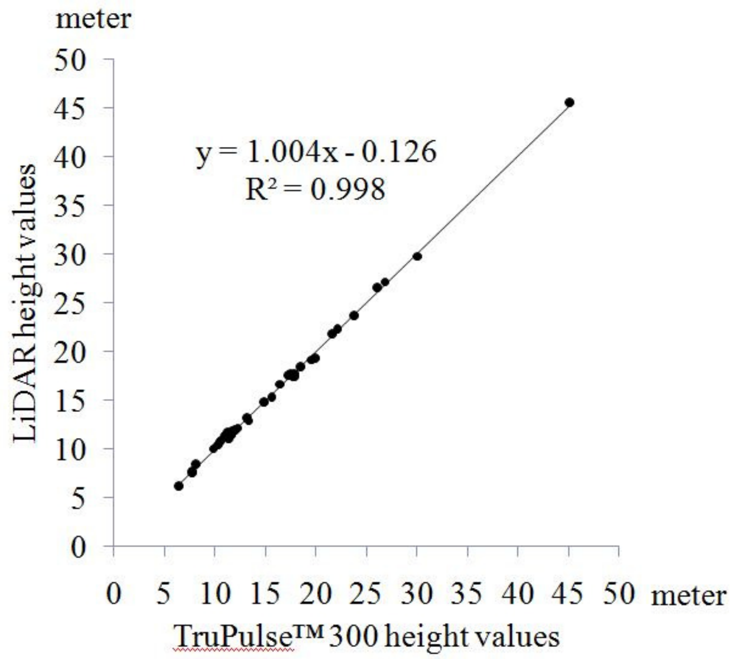


Figure 3.17 **Scatter (xy) plot for height values of LiDAR data and TruPulse™ 300**

Table 3.1 **Differences between reference heights and extracted heights from LiDAR**

Id	TruPulse™ 300	LiDAR values	Absolute Difference	Id	TruPulse™ 300	LiDAR values	Absolute Difference
1	17.30	17.52	0.22	20	8.20	8.37	0.17
2	10.40	10.65	0.25	21	15.60	15.27	0.33
3	12.00	11.95	0.05	22	23.80	23.65	0.15
4	11.60	11.61	0.01	23	11.60	11.51	0.09
5	6.50	6.10	0.40	24	19.60	19.20	0.40
6	10.50	10.71	0.21	25	12.20	12.17	0.03
7	22.10	22.26	0.16	26	13.30	12.95	0.35
8	26.80	27.20	0.40	27	17.80	17.35	0.45
9	7.80	7.53	0.27	28	17.90	17.35	0.55
10	10.90	11.23	0.33	29	11.40	11.03	0.37
11	7.80	7.58	0.22	30	26.10	26.47	0.37
12	11.70	11.40	0.30	31	12.00	11.88	0.12
13	13.20	13.10	0.10	32	14.90	14.76	0.14
14	21.70	21.71	0.01	33	11.30	11.64	0.34
15	11.80	11.93	0.13	34	18.50	18.41	0.09
16	9.80	10.07	0.27	35	30.10	29.79	0.31
17	16.50	16.69	0.19	36	10.30	10.45	0.15
18	17.60	17.80	0.20	37	20.00	19.28	0.72
19	17.90	17.63	0.27	38	45.20	45.52	0.32
							Average 0.25
							Standard Deviation 0.29

3.3.2 Evaluation of the detected roof planes

Since several techniques are used to detect roof segments, which are further classified as flat and pitched roof planes, it is important to evaluate the performance of the two-stage strategies, considering the accuracy will determine the later reconstruction method. In this study, the number of the detected roof planes are calculated and compared with that of the reference roof planes. The confusion matrix of the classification of flat and pitched roof planes is calculated as well. Since it is difficult to obtain ready-made reference data of building roof planes, the reference roof planes are manually digitalized based on the hillshade image of the LiDAR nDSM data and used for the completeness evaluation. Those reference planes are further classified as the flat and pitched roof planes and used for the confusion matrix calculation.

(1) Completeness of the detected roof planes

A comparison between the reference roof planes and the planes derived from the LiDAR data is made to examine the completeness of detected results. The detailed comparisons of results are shown in Figure 3.18. The completeness percentage is calculated by dividing the number of extracted roof planes by the actual amount of roof planes identified from reference data. Very small roof parts (example 1 in Figure 3.18) are not regarded as planes in this study. All 50 buildings in the study area are used for evaluation of the completeness and the results are shown in Table 3.2.

The overall completeness of the roof plane identification is very high (91.4%) considering the complex roof structures of buildings in the study area. The roof planes are completely identified in the 60% of buildings. About 26% of buildings are partially identified with completeness of larger than 80%. The rest of the buildings have a completeness of less than 80%. The completeness may be influenced by the number of roof planes each building has. Buildings that have fewer roof planes tend to obtain lower completeness if the same numbers of roof planes are unidentified (building #2 and #27).

Table 3.2 **Completeness of detected roof planes**

*B#	D#	R#	C%	B#	D#	R#	C%
1	7	7	100	21	1	1	100
2	1	2	50	22	3	3	100
3	5	5	100	23	2	2	100
4	2	2	100	24	1	1	100
5	4	4	100	25	17	20	85.0
6	5	5	100	26	3	4	75.0
7	1	1	100	27	8	9	88.9
8	19	21	90.5	28	5	6	83.3
9	6	7	85.7	29	1	1	100
10	6	6	100	30	14	18	77.8
11	4	4	100	31	15	28	53.6
12	22	24	91.7	32	4	6	83.3
13	2	2	100	33	6	8	75.0
14	32	35	91.4	34	11	11	100
15	3	3	100	35	2	2	100
16	1	1	100	36	22	23	95.7
17	1	1	100	37	3	3	100
18	12	12	100	38	25	28	89.3
19	7	7	100	39	13	14	92.9
20	1	1	100	40	8	8	100
41	4	4	100	46	5	7	71.4
42	6	7	85.7	47	5	5	100
43	2	2	100	48	8	9	88.9
44	5	10	50.0	49	15	15	100
45	6	6	100	50	6	9	66.7
Average completeness = 91.4%							

B#: Building number R#: Number of the reference roof planes
D#: Number of the detected roof planes C%: Completeness percentage

(2) Confusion matrix of the flat and pitched roof planes classification

Although the completeness can indicate the efficiency of the roof plane identification, the classification accuracy of the flat and pitched roof planes is still unclear. Evaluation of the roof shape classification should be examined as well since the classification is the foundation based on which reconstruction strategies are applied differently for the flat and pitched roofs. Therefore, another evaluation method is conducted by comparing the number of the flat, pitched and unidentified roof planes from the LiDAR nDSM data with the reference data, and creating the confusion matrix as shown in Table 3.3.

Table 3.3 **The confusion matrix for the classification of roof shape**

	Reference data			
Extracted data	Flat	Pitched	Unclassified	User's accuracy
Flat	304	7	0	97.8%
Pitched	0	57	0	100%
Unclassified	40	12	0	0%
Producer's accuracy	88.4%	75.0%	-	Overall accuracy = 88.8%

In the result, 368 (311 + 57) out of the 420 roof planes are extracted. The overall accuracy of the classification is found to be 88.8%. The user's accuracy of the pitched roof planes is 100%, which means there is no commission error for the pitched plane extraction. The user's accuracy of the flat roof planes is 97.8%, indicating 7 pitched roof planes in the reference data are mistakenly classified as the flat, which can be later improved by applying different cut-off thresholds on the slope of the nDSM data. For the producer's accuracy, 88.4% of flat roof planes are detected among the total 344, while 25% of pitched roof planes in the reference data are classified as others (since those pitched roof planes are not detected in the first place), a large portion of which result from the incapability of the identification of the protruding windows upon the pitched roof planes.

3.3.3 Visual Assessment of the 3D Reconstruction Results

The visual assessment of the reconstructed 3D building models indicates that the method successfully reconstructs most of the 3D building models by extruding the flat roof planes and creating TIN structures for the pitched roof planes. Several reconstructed 3D building models are shown in Figure 3.18. Example 1, 2, 3 and 4 are buildings with flat roof planes, example 5 is the building with pitched roof planes, and example 6 is the building combined with both flat and pitched roof planes. Examples of reconstructed 3D buildings that contain errors in roof configuration are shown in Figure 3.19.

For the flat, most of the jump edges are successfully detected using the Canny algorithm, which is very effective in identifying edges with less height variations than the typical edges (example 3 and 4 in Figure 3.18). However, there are still some flat planes that fail to be identified (example 1 and 2 in Figure 3.19), since the height difference is too small to be detected. Due to the processing criteria, the detected edges shorter than 5 meters are removed. Besides eliminating noise edges, the criteria also unfavorably eliminate small roof segments. Considering that the point density of the LiDAR data in this study is only 1 point / m^2, which is not sufficient for the detection of detailed roof structures (Alexander et al., 2009), and the shapes of the extracted small roof segments are irregular, the small planes are considered as the same planes as their larger surrounding planes (example 1 in Figure 3.18 and example 3 in Figure 3.19).

The method is very efficient in detecting buildings which have mixed roof shapes with a pitched roof segment piling upon a larger flat roof segment. Even though the ridgelines of the pitched roof segments are not identified at the first level of the edge detection due to the non-obvious height variations, the method is able to detect the intra-structures of the pitched roof segments in the later steps. Roof segments with slopes of less than 20 degrees are classified to be flat using the classification threshold. Some pitched roof segments with slopes of less than 20 degrees thus are mistakenly categorized as flat ones. A major drawback of the performance is that chimneys and protruding roof structures on the pitched roof segments are not well visualized, the main reason for which, as mentioned above, is the low point density of the LiDAR data, making it difficult to trace corresponding edges and reconstruct properly. The detection and reconstruction performance would be better if LiDAR data with higher point densities are used.

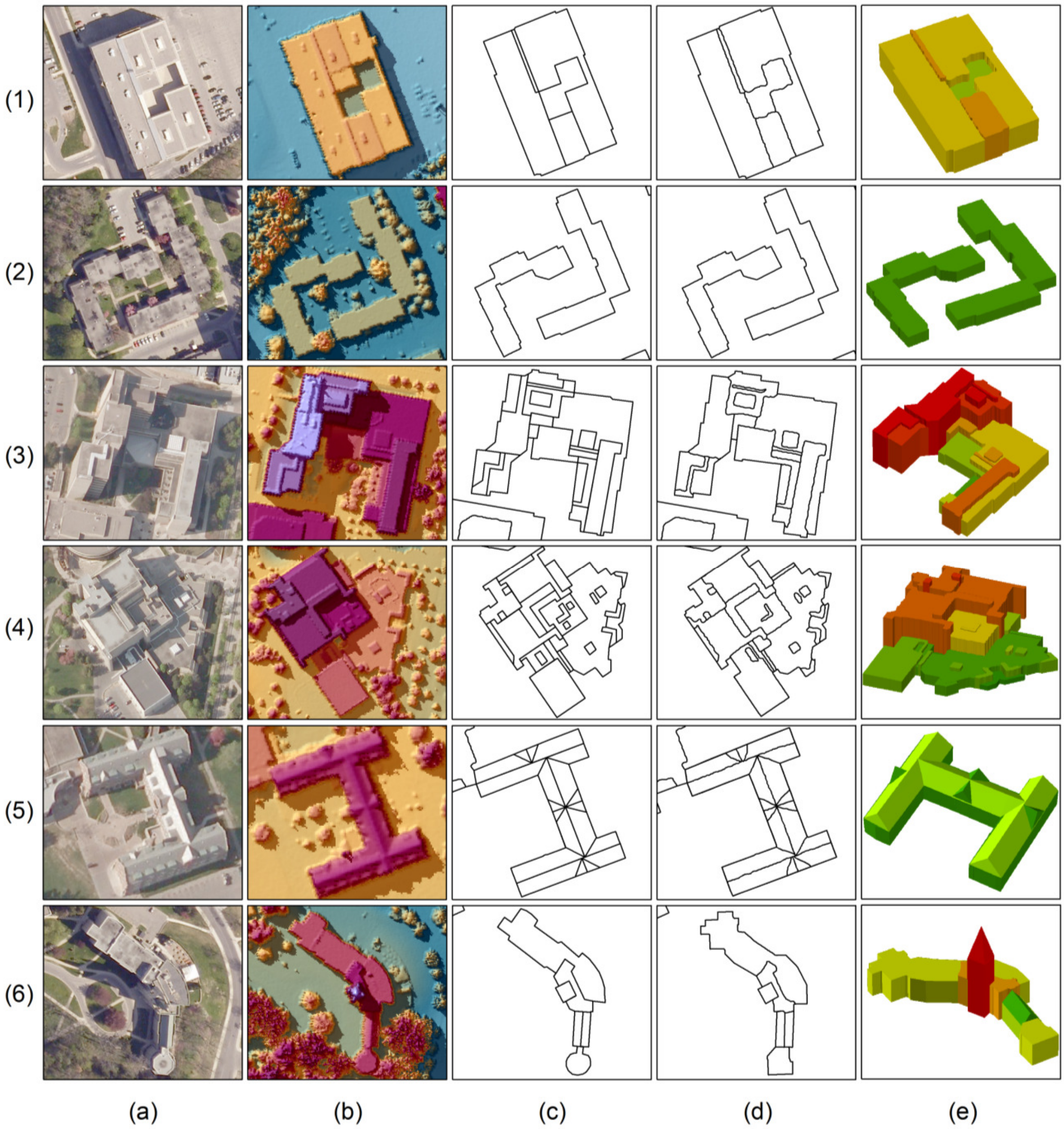


Figure 3.18 **Six samples of edge detection results and reconstruction models (from row (1) to row (6)). From left to right, column (a): aerial photos; column (b): hill-shaded images generated from the nDSM data; column (c) reference planar facets derived from the LiDAR nDSM data; column (d): planar facets detection results and; column (e): the final 3D models.**

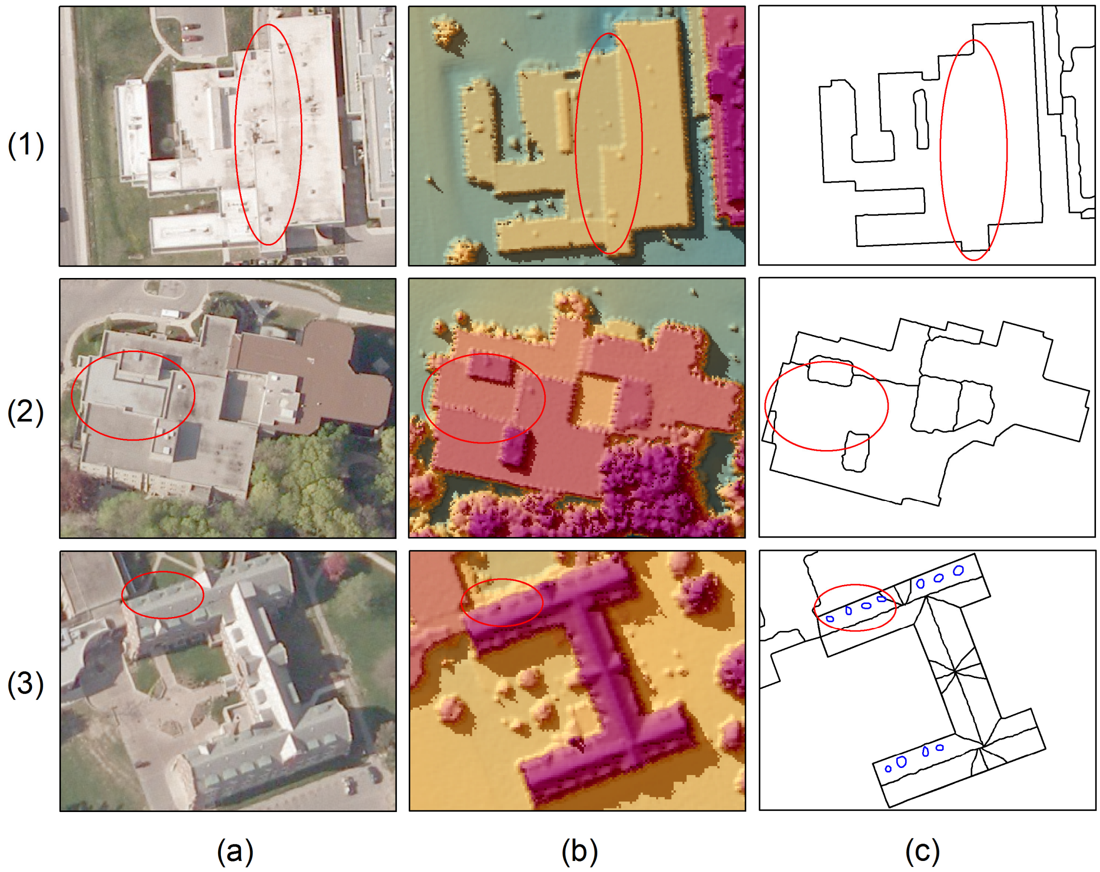


Figure 3.19 **Missed or misinterpreted roof planes. (1) (2) and (3) are examples of the poor performances. (a) aerial photos; (b) hillshade images of the nDSM data; (c) extracted building roof structures**

3.4 Conclusions and Discussions

A GIS-based technique is developed in this paper to reconstruct 3D building models from the airborne LiDAR data. Despite the aforementioned problems, the method presented in this study has successfully modeled 3D buildings using the LiDAR data and the 2D building footprints extracted in Chapter 2. Most previous studies use either LiDAR point clouds or interpolated grid data to segment roof planes and reconstruct building models. The major difficulties of the previous studies in reconstructing 3D building models include the identification of height variations (jump edges) between flat roof segments and the detection of ridgelines and planes in the pitched roof segments. Instead, the method presented in this study identifies roof planes by detecting edges within buildings, based on which GIS techniques are used to process and reconstruct building models. The main differences between this study and the previous ones are: (1) the direct edge detection from the LiDAR nDSM data, compensating for the problem of the identification of jump edges and ridgelines; (2) classification of roof shape types, and (3) the separate modeling strategies for the flat and pitched roofs. The method has some applicability and reproducibility, the generated 3D building models can be further utilized as fundamental data for various applications such as urban design, urban heat island studies and flood risk analysis. The main software used in this study is ArcGIS, which is the industrial standard software for GIS and can be widely accessed in educational institutions, government organizations and private industries.

The GIS-based reconstruction method presented here is proven to be effective for creating 3D building models from LiDAR data in urban environments. The proposed method in this study is able to detect most of the jump edges and identify 90% of the flat roof planes. The ridgelines in the pitched roof segments are extracted sufficiently by applying Canny edge-detection algorithm to the aspect image of the nDSM data. 91.4% overall completeness of the roof-plane identification is achieved, which is successful considering the complicated building structures in the study area. The roof planes in 60% of buildings are completely detected, while there are only less than 25% buildings, of which roof planes are partially identified with less than 80% completeness. The classification accuracy of the roof shape types (flat and pitched) are examined as well. The confusion matrix is calculated based on the flat, pitched and unidentified roof planes to evaluate the classification performance. The overall accuracy of the classification is 88.8%, with the user's accuracy of flat planes 97.8% and that of pitched planes 100%. In contrast, the producer's accuracies for the two roof shape types are 88.4% and 75% respectively. The low producer's accuracy of pitched planes indicates the incapability of identification of chimneys and protruding windows upon the pitched roofs.

Despite the merits of the proposed method, some aspects still need to be improved in future work. For example, the point density of the LiDAR data used in this study is 1 point / m^2, which is insufficient in detecting small roof planes in the flat roof segments and the protruding structures in the pitched roofs as well. LiDAR data with higher point densities should be used if detailed roof structures need to be identified. The cut-off angle threshold in the classification of pitched roof segments can be further modified by smoothing the slope image of the nDSM data, since there are some lines structures on top of the roof planes causing the high slope values. Generalization methods also can be used to regularize the detected building edges in the future work.

3.5 References

Alexander, C., Smith-Voysey, S., Jarvis, C., Tansey, K., 2009, Integrating building footprints and LiDAR elevation data to classify roof structures and visualize buildings. *Computers, Environment and Urban Systems*, Vol. 33, pp. 285-292.

Arefi, H., Hahn, M., 2005. A morphological reconstruction algorithm for separating off-terrain points from terrain points in laser scanning data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*36 (Part 3/W19), pp. 120–125.

Beraldin, J. A., Blais, F. and Lohr, U. 2010. Laser scanning technology. Chapter 1 in *Airborne and Terrestrial Laser Scanning* (Eds. G. Vosselman and H. G. Maas).Whittles, Caithness, Scotland.320 pp. 1-42.

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

Brennan, R. and Webster, T.L. 2006. Object-oriented land cover classification of LiDAR-derived surfaces. *Canadian Journal of Remote Sensing*, Vol. 32, No. 1, pp. 162-172.

Canny, J., 1986, A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.Vol.8, No. 6, pp. 679-698.

Elaksher, A.F., Bethel, J.S., 2002, Reconstructing 3D buildings from LiDAR data. *ISPRS Symposium 2002 Photogrammetric Computer Vision*, Graz, Austria, pp.102-107.

Elberink, S., and Vosselman, G., 2009.Building reconstruction by target based graph matching on incomplete laser data: analysis and limitations. *Sensors*, Vol. 9, No. 8, pp. 6101–6118.

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

Forlani, G., Nardinocchi, C., Scaioni, M., Zingaretti, P., 2006, Complete classification of raw LiDAR data and 3D reconstruction of buildings. *Pattern Analysis Application*, Vol. 8, pp. 357-374.

Gong, P., Sheng, Y., Biging, G.S., 2002, 3D model-based tree measurement from high-resolution aerial imagery. *Photogrammetric Engineering and Remote Sensing*, Vol. 8, No. 11, pp. 1203-1212.

Kabolizade, M., Ebadi, H., Mohammadzadeh, A., 2012, Design and implementation of an algorithm for automatic 3D reconstruction of building models using genetic algorithm. *International Journal of Applied Earth Observation and Geoinformation*, Vol. 19, 104-114.

Kim, K.H., Shan, J., 2011, Building roof modeling from airborne laser scanning data based on level set approach. *Journal of Photogrammetry & Remote Sensing*, Vol. 66, pp. 484-497.

Lehrbass, B. and Wang, J. 2012. Urban tree cover mapping with relief-corrected aerial imagery and LIDAR. *Photogrammetric Engineering and Remote Sensing*.Vol. 78, No. 5, pp. 473-484.

Noronha, S. and Nevatia, R., 2001, Detection and modeling of building from multiple aerial images. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 23, No. 5, pp. 501-518.

Rottensteiner, F., 2003, Automatic generation of high-quality building models from LiDAR data. *IEEE Computer Graphics and Applications*, Vol. 23, No.6, pp. 42-50.

Sampath, A., Shan, J., 2010. Segmentation and reconstruction of polyhedral building roofs from aerial LiDAR point clouds. *IEEE Transactions on Geoscience and Remote Sensing*, Vol. 48, No.3, pp. 1554-1567.

Satari, M., Samadzadegan, F., Azizi, A., Maas, H., 2012, A multi-resolution hybrid approach for building model reconstruction from LiDAR data. *The Photogrammetric Record,* Vol. 27, No. 139, pp. 330-359.

Shan, J. and Toth, C.K., 2009, Topographic laser ranging and scanning: principles and processing. *CRC Press*, Taylor & Francis, London, pp. 307-334.

Sithole, G., Vosselman, G., 2004. Experimental comparison of filter algorithms for bare-Earth extraction from airborne laser scanning point clouds. *ISPRS Journalof Photogrammetry and Remote Sensing*, Vol. 59, No. 1-2, pp. 85–101.

Tarsha-Kurdi, F., Landes, T., Grussenmeyer, P., 2007, Hough-transform and extended RANSAC algorithms for automatic detection of 3D building roof planes from LiDAR data. *Proceedings of the ISPRS Workshop on Laser Scanning 2007 and SilviLaser*, Espoo, Finland, pp. 407-412.

Verma, V., Kumar, R. and Hsu, S., 2006. 3D building detection and modeling from aerial LIDAR data. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Washington, DC, USA. Vol. 2515, pp. 2213–2220.

Vosselman, G., Dijkman, S., 2001, 3D building model reconstruction from point clouds and ground plans. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. 34,pp. 37-43.

Wang, L. and Chu, C.-H.H., 2009.3D building reconstruction from LiDAR data. *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*, San Antonio, Texas, USA.pp. 3054–3059.

Yong, L., 2011, Automatic edge extraction by LiDAR-optical data fusion adaptive for complex building shapes. *International Symposium on Lidar and Radar Mapping 2011: Technologies and Applications*, Nanjing, China.

Zhang, K., Yan, J., Chen, S.C., 2009, Automatic 3D building reconstruction from airborne LiDAR measurements. *IEEE Urban Remote Sensing Joint Event*, Shanghai, China.

Zhou, G., Albertz, J., Gwinner, K., 1999. Extracting 3Dinformation using temporal–spatial analysis of aerial image sequences, *Photogrammetry Engineering &Remote Sensing*, Vol. 65, No.7, pp. 823–832.

Zhou, G., Song, C., Simmers, J., Cheng, P., 2004. Urban 3D GIS from LiDAR and digital aerial images. *Computers & Geosciences*, Vol. 30, pp. 345-353.

Chapter 4

Conclusion

4.1 Summary

It is important to map building footprints and reconstruct 3D building models because the information can be used as fundamental GIS data, which can be applied in multiple scientific and engineering applications such as urban planning and design, urban climate and environmental studies, and emergency response applications. Remote sensing data have been regarded as valuable sources for the extraction of building information for decades. LiDAR technology has shown great potential in extracting building information. LiDAR data not only possess high resolution quality, but also can provide accurate vertical information along with multiple returns and intensity data. The research presented here includes a method for urban 2D building footprint extraction using the rule-based object-oriented classification approach and a method for 3D building model reconstruction in a GIS environment from airborne LiDAR data.

Chapter 2 proposes a rule-based object-oriented approach, including some new techniques, to extract building footprints using the airborne LiDAR data. The semi-automatic method is applied to the LiDAR data for campus of the University of Western Ontario, London, ON. Detailed procedures are presented for the pre-processing, segmentation and the rule-based classification of buildings. Finally the reference building footprints are used to examine the performance of proposed method.

Chapter 3 develops a GIS-based reconstruction method for 3D building models from the airborne LiDAR data. The same study area is applied and the 2D building footprints derived from Chapter 2 are used in this research as well. The method presented here identifies roof planes by directly detecting edges on the LiDAR nDSM data within building footprints, based on which GIS techniques are used to reconstruct 3D building models. The digitized and classified roof planes from the hillshade of the nDSM data are used as reference data for the evaluation of the roof-plane identification and roof shape classification. The completeness of roof plane identification and the confusion matrix for the roof shape classification are calculated respectively. Visual assessment of the reconstructed 3D building models is made as well.

4.2 Conclusions

The study has successfully answered the research questions from Chapter 1. The specific answers are presented as follows:

1. The results of the proposed rule-based object-oriented extraction of building footprints prove to be satisfactory. The overall accuracy of 94.0% and commission error of 6.3% with overall Kappa of 0.84 achieved.

2. The results of the three contrastive methods indicate that NDTI and intensity play very important roles in object-based building extraction. The object-based method with only LiDAR nDSM data has the highest commission error, while adding NDTI remarkably reduces the error by 29.0%. The intensity data contribute to the final result by improving 1.5% of overall accuracy and decreasing 1.8% of commission error making the joint effort with NDTI in the proposed method.

3. Roof structures can be identified and the 3D building models can be reconstructed using GIS-based strategies from the LiDAR data. The performance is satisfactory considering the complexity of the study area.

4. The GIS-based reconstruction method is simple and effective for creating 3D building models from LiDAR data. 91.4% completeness of the roof plane identification is achieved, among which the roof planes of 60% of buildings are completely detected. The overall accuracy of roof shape classification is 88.8% with the user's accuracy of flat planes 97.8% and that of pitched planes 100%.

4.3 Contributions

The main contribution of the study in Chapter 2 is to develop an effective object-based method for building footprint extraction. The method can successfully separate buildings from trees and generate accurate building footprints. The NDTI and intensity data are used as the main parameters in rule-based object-oriented approach, and the roles of the two parameters played in the proposed method are examined, which indicate that both parameters should be used in the object-based classification to achieve the best extraction result. The conclusions can support the related research on LiDAR-based building footprint extraction in the future.

Chapter 3 is one of the few studies that use GIS-based techniques to reconstruct 3D building models from the LiDAR data. The proposed reconstruction strategy is able to greatly reduce the problems from previous studies. The GIS-based approach presented here can be applied in the reconstruction of 3D building models from LiDAR data since the implementation is less complicated yet effective enough to achieve satisfactory results.

The implementation of the two methods in other urban areas needs to be examined in the future, the application of this study in systematically generating 2D and 3D building information appears promising.

4.4 Possible Future Research

4.4.1 Object-based classification

Future research can potentially improve the result of building footprints in several different aspects. For the proposed rule-based object-oriented method, building footprints would be extracted more accurately by adding other parameters for the rule-based classification. Many parameters describing the spatial and spectral information of the objects can be examined in the future. Additionally, since the rule-based classification is used in the object-based method, the "Fuzzy membership" can be used in the classification of building and non-building classes. The Fuzzy rule-base allows image objects to have membership in more than one class, and uses membership functions to represent the imprecise nature of the properties and relationships that can characterize building and non-building classes (Aldred and Wang, 2011). Although the proposed object-based technique is able to extract some small buildings, the rest are left out due to their spatial and spectral disparity from typical buildings. The fuzzy membership may improve the efficiency of detecting small building objects by comparing their fuzzy values returned by the contributions of the object features (e.g. shape, texture), and the class membership of the objects are assigned as the possibility of the objects belonging to the class. The fuzzy functions would benefit classifying more small buildings.

4.4.2 GIS-based Reconstruction method

Despite the merits of the proposed method, some aspects still need to be improved for the reconstruction of 3D building models. For example, the point density of the LiDAR data used in this study is 1 point / m^2, which is insufficient to detect very small roof planes in the flat roof segments and protruding structures in the pitched roof segments as well. LiDAR data with higher point densities should be used if detailed roof structures need to be identified. The cut-off angle threshold in the classification of the pitched roof segments can be modified and applied in multiple levels in order to increase the classification accuracy. Another aspect to be improved is the post-processing of the detected edges. No generalization method is used to regularize the detected building edges. Future work can include the improvement of smoothing and squaring the detected edges, thus better visualization of 3D building models can be achieved.

4.5 References

Aldred, D. and Wang, J. 2011. A method for obtaining and applying classification parameters in object-based urban rooftop extraction from VHR multispectral images. *International Journal of Remote Sensing*, Vol. 32, No.10, pp. 2811-2823.

Appendix 1

The pseudo-code for the rule-based classification

 # Initialize the classes, input layers and pre-defined parameters

Initialize:

Input[3]=[nDSM_FR, nDSM_LR, Intensity, NDTI].

Class[1] =[Building_Candidate, Unclassified]

SpectralDifference = 1

 # PerformMultiresolution segmentation and generate the first level of image objects

MultiresolutionSegmentation(Input, W_{nDSM_FR} = 1, W_{nDSM_LR} = 1, W_{NDTI} = 1, ScaleParameter = 10)

    return ImageObject1

 # Iterate Spectral Difference segmentation until the spectral difference is no larger than the user-defined threshold of 1, and generate the second level of image objects

WhileSpectralDifference> 1

    SpectralDifferenceSegmentation(ImageObject1, W_{nDSM_FR} = 1, W_{nDSM_LR} = 1, $W_{Intensity}$ = 1, W_{NDTI} = 2, SpectralDifference = 1)

        return ImageObject2

 # In the classification of buildings, the first step is to assign the basic qualified image objects to building candidates.

Assign ImageObject2 to Unclassified

If the mean value of NDTI$_{\text{Unclassified}}$<= 0.001 **AND** the mean value of nDSM_LR$_{\text{Unclassified}}$>= 2.5 **OR** the mean value of nDSM_FR$_{\text{Unclassified}}$>= 30

    Assign Unclassified to Building_Candidate

If the value of Roundness$_{\text{Building_Candidate}}$<= 0.41 **AND** the value of Area$_{\text{Building_Candidate}}$<= 190

    Assign Building_Candidate to Unclassified

 # The second step is to assign adjacent building objects to building candidates.

While (the value of Roundness$_{\text{Unclassified}}$>= 3 **OR** the mean value of nDSM_LR$_{\text{Unclassified}}$>= 2.5) **AND** the value of Rel. border to$_{\text{Building_Candidate}}$>= 0.6

    Assign Unclassified to Building_Candidate

    Merge all the objects of Building_Candidate

 # The third step is to assign building edge objects to building candidates.

If (the value of Roundness$_{\text{Unclassified}}$>= 2.8 **OR** the mean value of nDSM_LR$_{\text{Unclassified}}$>= 6) **AND** the value of Rel. border to$_{\text{Building_Candidate}}$>= 0.4

    Assign Unclassified to Building_Candidate

    Merge all the objects of Building_Candidate

While the mean value of nDSM_LR$_{\text{Unclassified}}$>= 3.5 **AND** the mean value of nDSM_LR$_{\text{Unclassified}}$<=5.3 **AND** the value of Rel. border to$_{\text{Building_Candidate}}$>= 0.5

Assign Unclassified to Building_Candidate

Merge all the objects of Building_Candidate

 # After classification, the next step is to post-process the building candidates.

If the value of Area$_{\text{Building_Candidate}}$<= 200

Assign Building_Candidate to Unclassified

If (the value of Area$_{\text{Building_Candidate}}$> 200 **AND** the value of Area$_{\text{Building_Candidate}}$< 1000) **AND** (the value of Rectangular Fit$_{\text{Building_Candidate}}$< 0.9 **OR** the value of Rectangular Fit$_{\text{Building_Candidate}}$>= 0.997)

Assign Building_Candidate to Unclassified

Merge all the objects of Building_Candidate

Merge all the objects of Unclassified

 # Export the building candidates

ExportVectorLayer (Building_Candidate, Polygon)

Appendix 2

Comparisons of edge detection operators

Edge detection is the process of identifying and locating sharp brightness changes or discontinuities presented in an image. Those brightness changes are typically organized into a set of boundaries of objects or line segments termed edges. Classical edge detectors usually convolve the image with an operator, which is constructed to detect the gradients while be insensitive to homogeneous areas (Jähne et al., 1999). There are many edge detection algorithms available, such as Sobel operator, Robert's cross operator, Prewitt's operator, Laplacia of Gaussian and Canny detector, each of which is designed to detect certain types of edges yet has its own limitations. All the aforementioned operators are available in ENVI software.

A2.1 Sobel Operator

Sobel operator has a pair of 3 by 3 convolution kernels as shown in Figure A2.1. d_y is simply the d_x rotated by 90 degrees. The kernels are constructed to detect edges running vertically and horizontally relative to the pixel grid, d_x is to detect vertical edges and d_y responds to horizontal edges. The two kernels can be used separately for the image to generate separate gradients, which can be then combined to calculate the absolute magnitude (g(i,j) in Figure A2.1) of the gradient at each pixel.

$$g(i,j) = \{d_x^2(i,j) + d_y^2(i,j)\}^{\frac{1}{2}}$$

$$d_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad d_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Figure A2.1 **Equations of Sobel operator**

A2.2 Robert's cross operator

Robert's cross operator has a pair of 2 by 2 convolution kernels as shown in Figure A2.2. D_2 is simply D_1 rotated by 90 degrees, which is very similar to the Sobel operator. The kernels are designed to detect edges running at 45 degrees to the pixel grid, and the detection directions of the two kernels are perpendicular with each other. Similar to the Sobel operator, the two kernels can be used separately for the image to produce separate gradient results, which are then combined to calculate the absolute magnitude of the gradient at each pixel. G(i,j) in Figure A2.2 is just another expression as the g(i,j) in Figure A2.1

$$G(i,j) = |f(i+1,j+1) - f(i,j)| + |f(i+1,j) - f(i,j+1)|$$

$$D_1 = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \qquad D_2 = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

Figure A2.2 **Equations of Robert's cross operator**

A2.3 Prewitt's Operator

Prewitt's operator is also very similar to the Sobel operator to detect vertical and horizontal edges in the image. The operator has a pair of 3 by 3 convolution kernels as shown in Figure A2.3. The only difference between the Prewitt's operator and the Sobel operator is the numbers in the kernel.

$$g(i,j) = \{d_x^2(i,j) + d_y^2(i,j)\}^{\frac{1}{2}}$$

$$d_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \qquad d_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Figure A2.3 **Equations of Prewitt's operator**

A2.4 Laplacian of Gaussian

The Laplacian operator is an isotropic detector of the second spatial derivative of an image. Since the second derivative of an image refers to the change rate of brightness presented in the image (the change rate is the maximum when the second derivative of an image reaches zero), the Laplacian operator can be used for edge detection. Based on the definition of the Laplacian, some commonly used 3 by 3 kernels are shown in Figure A2.4.

$$L(x, y) = \frac{\partial^2 I(x, y)}{\partial x^2} + \frac{\partial^2 I(x, y)}{\partial y^2}$$

$$H_1 = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad H_2 = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad H_3 = \begin{bmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{bmatrix} \quad H_4 = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Figure A2.4 **Equations of Laplacian operator**

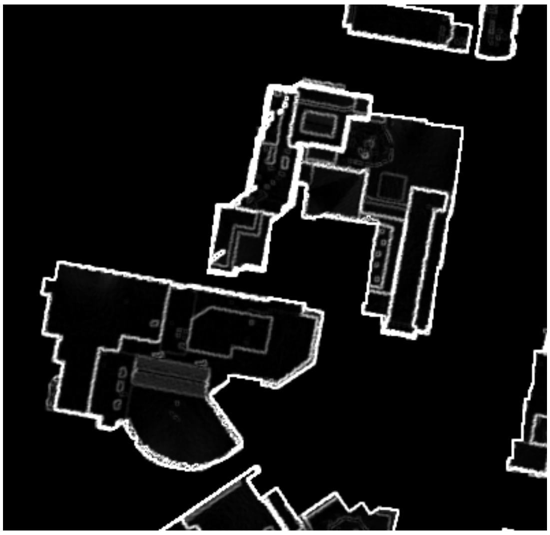
A2.5 Comparisons of the operators

Since the aforementioned operators are available in ENVI software, all the mentioned operators are applied to the nDSM of the last return image. The results are shown in Figure A2.5

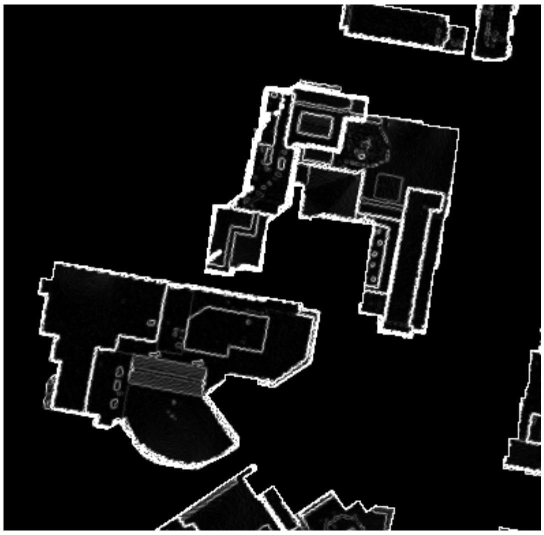
The results of Sobel's operator, Robert's cross operator and Prewitt's operator are quite similar. The detected edges are thick and blur. There is also detection of false edges with low values in the image. As mentioned before, Sobel's operator, Robert's cross operator and Prewitt's operator are more capable of detecting edges running certain directions, which, in this study, are not the best candidates for edge detection within buildings because of the random building orientations presented in the study area.

As for Laplacian operator, two results are generated using different kernel sizes. However, there is no obvious difference in the two images. Besides the thick edge problem, the results generated by Laplacian operator are influenced by the noise. The kernels are approximately second derivative measurements on the image, and both the noises and the edges contain high frequency content, thus Laplacian operator is very sensitive to the noise, which results in less accurate identification of the detected edges.

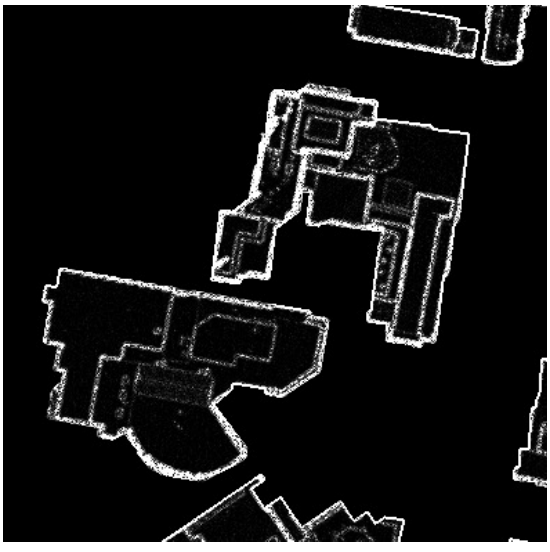
Compared with other operators, Canny operator outperforms with its multi-stage algorithm. The algorithm is designed to iterate the detection process until there are as many real edges as possible detected. The Canny operator is also capable of thinning the detected edges down to generate a better result. By comparison, Canny operator is finally chosen for the edge detection of buildings in this study.



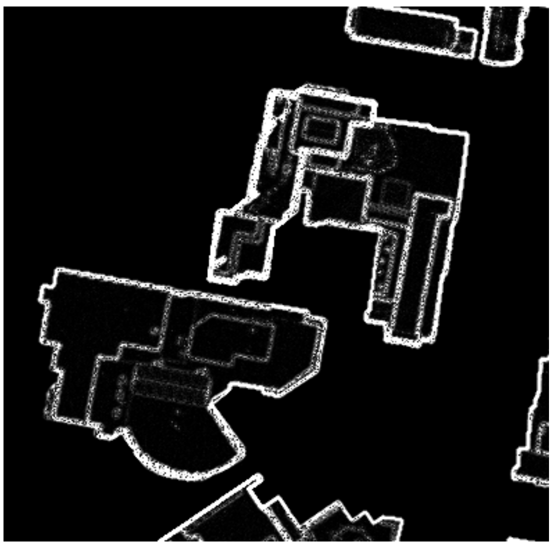
(a)



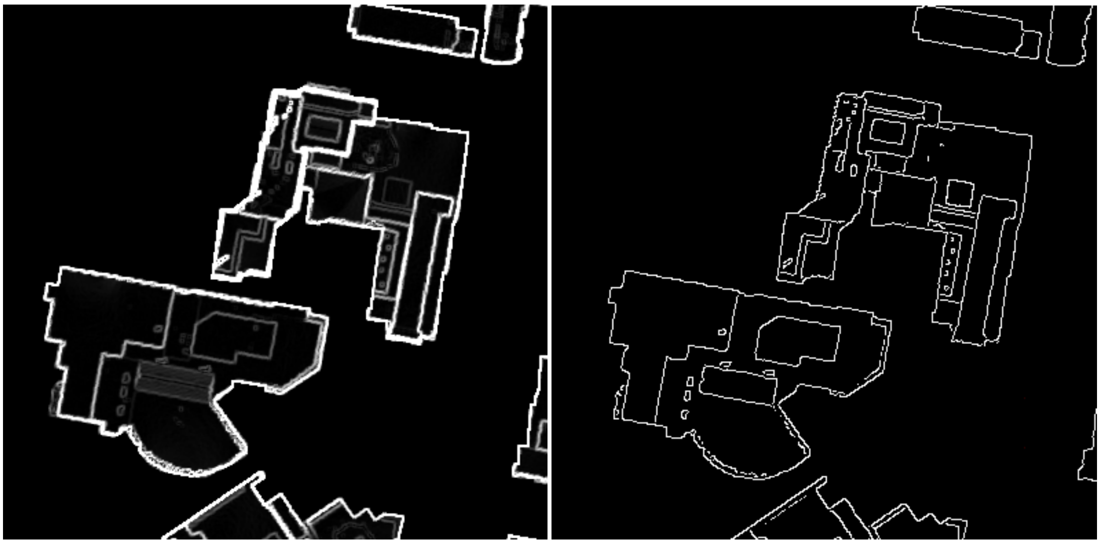
(b)



(c)



(d)



(e) (f)

Figure A2.5 **Results of edge detection using different operators. (a) Sobel operator; (b) Robert's cross operator; (c) Laplacian operator with 3 by 3 kernel; (d) Laplacian operator with 5 by 5 kernel; (e) Prewitt's operator and (f) Canny operator**

A2.6 References

Jähne, B.,Scharr,H., and Körkel,S. 1999. Principles of filter design. *In Handbook of Computer Vision and Applications*.Academic Press.

Appendix 3

Implementation of GIS-based 3D building reconstruction

In this appendix, a detailed description of the implementation of GIS-based 3D building reconstruction method is presented, which can be categorized as five main steps: edge detection, edge processing, boundary processing, classification of rooftops and building reconstruction. The main theories of the method have been elaborated in Chapter 4, thus only implementations are shown in the following sections.

A3.1 Edge detection

Canny operator is used for edge detection within building footprints. There are three parameters need to be specified for Canny operator: sigma, low threshold and high threshold. Sigma is the standard deviation of the Gaussian filter, of which typical values are from 0.6 to 2.5. Low threshold and high threshold are the hysteresis threshold values. The typical value range of low threshold is between 0.2 and 0.5 and that of high threshold is within 0.6 an 0.9. Thus a three-level loop is set to iterate the Canny algorithm with different combination of the three parameters. The code is written in IDL environment embedded in ENVI software. Details for the iteration can be seen as follows:

```
;----------------------canny edge detection------------------------------
;----------------------Writen By Ting Zhao-------------------------------
;---------------------Department of Geography----------------------------
;-----------------University of Western Ontario-------------------------

;Syntax: canny(A,highthreshold,lowthreshold,sigma)
;
;A is a 2D image array of numbers, typically of bytes
;
;Sigma is the standard deviation of a guassian smoothing filter.
;      Typical values are in the range 0.6 to 2.5. The choice
;      depends on the noisiness of the image.
;lowthreshold and highthreshold are the hysteresis threshold values.
;      They must be in the range 0<lowthreshold<highthreshold<1.
;      Typical value ranges are lowthreshold in 0.2-0.5 and
highthreshold in 0.6-0.9.
;-----------------University of Western Ontario-------------------------

; theGetData program below mainly initialize the parameters and set
; up the processing environment
```

```
ProGetData,ImgData = ImgData,ns = ns,nl = nl,nb = nb,Data_Type =
Data_Type,$
FileName = FileName,Map_info = map_Info
Envi_Open_File,FileName,R_Fid = R_Fid,/no_realize
ENVI_FILE_QUERY,R_fid,ns = ns,nl = nl,nb = nb,Data_Type = Data_Type
map_info = envi_get_map_info(fid=R_fid)
dims = [-1,0,ns - 1 ,0,nl - 1]
caseData_TypeOf
1:ImgData = BytArr(ns,nl,nb)     ;  BYTE  Byte
2:ImgData = IntArr(ns,nl,nb)     ;  INT  Integer
3:ImgData = LonArr(ns,nl,nb)     ;  LONG  Longword integer
4:ImgData = FltArr(ns,nl,nb)     ;  FLOAT  Floating point
5:ImgData = DblArr(ns,nl,nb)     ;  DOUBLE  Double-precision floating
EndCase
Fori = 0,nb-1DoBegin
Dt = Envi_Get_Data(Fid = R_Fid,dims = dims,pos=i)
ImgData[*,*,i] = Dt[*,*]
EndFor
End

;---------------------------------------------------------------------

procanny_test
t0=systime(1)

Filename='E:\Master_Research\LiDAR\BuildingReconstruction\LiDAR_UWO_cro
p_DEM_LR_NDSM_Extracted_0data.tif'
GetData,ImgData = ImgData,ns = ns,nl = nl,nb = nb,Data_Type =
Data_Type,FileName = FileName,Map_info = map_Info

for a = 0, 3dobegin;from 0.6 to 0.9, with 0.1 interval
for b = 0, 3dobegin;from 0.2 to 0.5, with 0.1 interval
for c = 0, 19dobegin;from 0.6 to 2.5
img_canny = canny(ImgData,HIGH=0.1*a+0.6,LOW=0.1*b+0.2,
SIGMA=0.1*c+0.6)
          x=0.1*a+0.6
          y=0.1*b+0.2
          z=0.1*c+0.6
Envi_Write_Envi_File,img_canny,Out_Name =
'E:\Master_Research\LiDAR\BuildingReconstruction\canny_test\LiDAR_NDSM_
canny_H'+strmid(strtrim(string(x),1),0,4)+'_L'+strmid(strtrim(string(y)
,1),0,4)+'_S'+strmid(strtrim(string(z),1),0,4)+'.tif',Map_info =
map_Info
endfor
endfor
endfor
print, 'The spent time is', systime(1)-t0, 'seconds'

end
;---------------------------------------------------------------------
```

More than 300 results are generated by this code. Visual examination is conducted by starting with the default values for the three parameters. Based on empirical observation, the performance of the edge detection is regarded as the best when the sigma was set to 0.7, with the high threshold 0.6 and the low threshold 0.4 (Figure 3.4).

A3.2 Edge processing

After edge detection, the building edges need to be vectorized for further processing. This step is automatically performed on the binary image of building edges using the "Vectorization" function from the ArcScan module of ArcGIS 10.

After clicking "Start editing" for the binary image of building edges, "Vectorization Setting" should be opened to set up the parameters for vectorization (Figure A3.1). Attempts of different combinations are conducted until the optimal vectorization is performed. The vectorized edges are stored as Polyline Shapefile in ArcGIS. The generated edges need initial processing, which is to delete polylines with length of less than 5 meters (Figure A3.2).

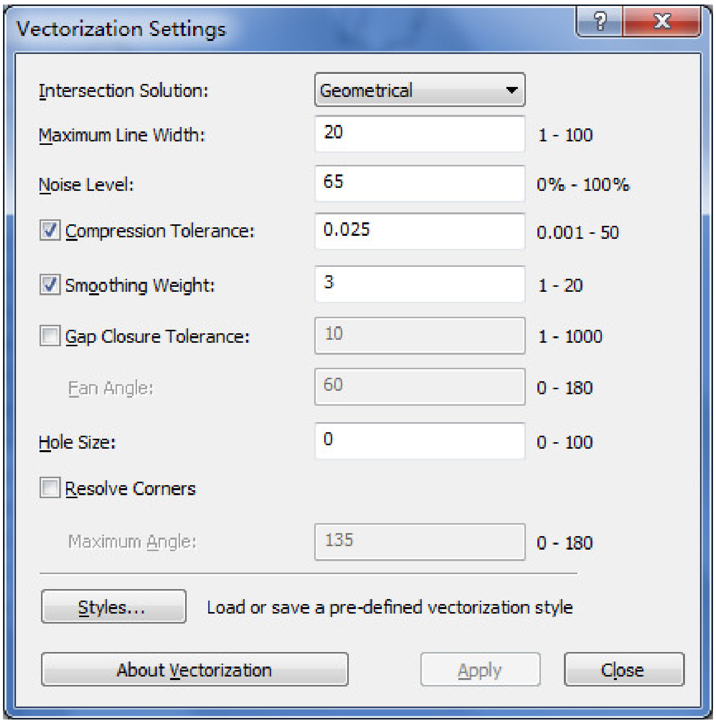


Figure A3.1 **Vectorization settings**

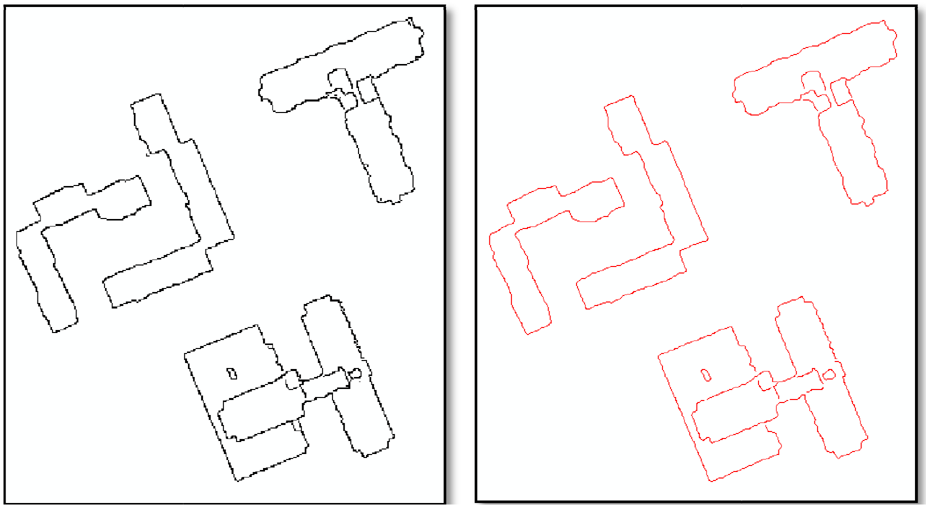


Figure A3.2 **Vectorization result. Left: binary image of building edges; Right: vectorized and initially processed building polylines**

Since many polylines are not closed up in the vectorization process, the next step is to close up the polylines, which is the precondition of transforming polylines to polygons. The topology rules of polylines are very important in this step. In ArcGIS, topology is the arrangement that defines how point, polyline and polygon features share coincident geometry (ESRI, 2011). Topology defines and enforces data integrity rules (e.g. there should be no gaps between polygons, there should be no intersection between polylines).

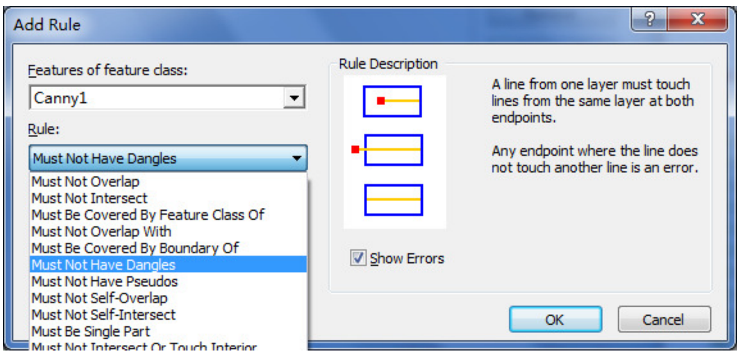


Figure ***A3.3*** **Topology rules for vector data in ArcGIS**

In this case, specifically, the endpoints of polylines that dangle in the air are detected by validating the topology rule "Must Not Have Dangles (Line)" for the vectorized polylines. Detailed steps are shown in Figure A3.4.

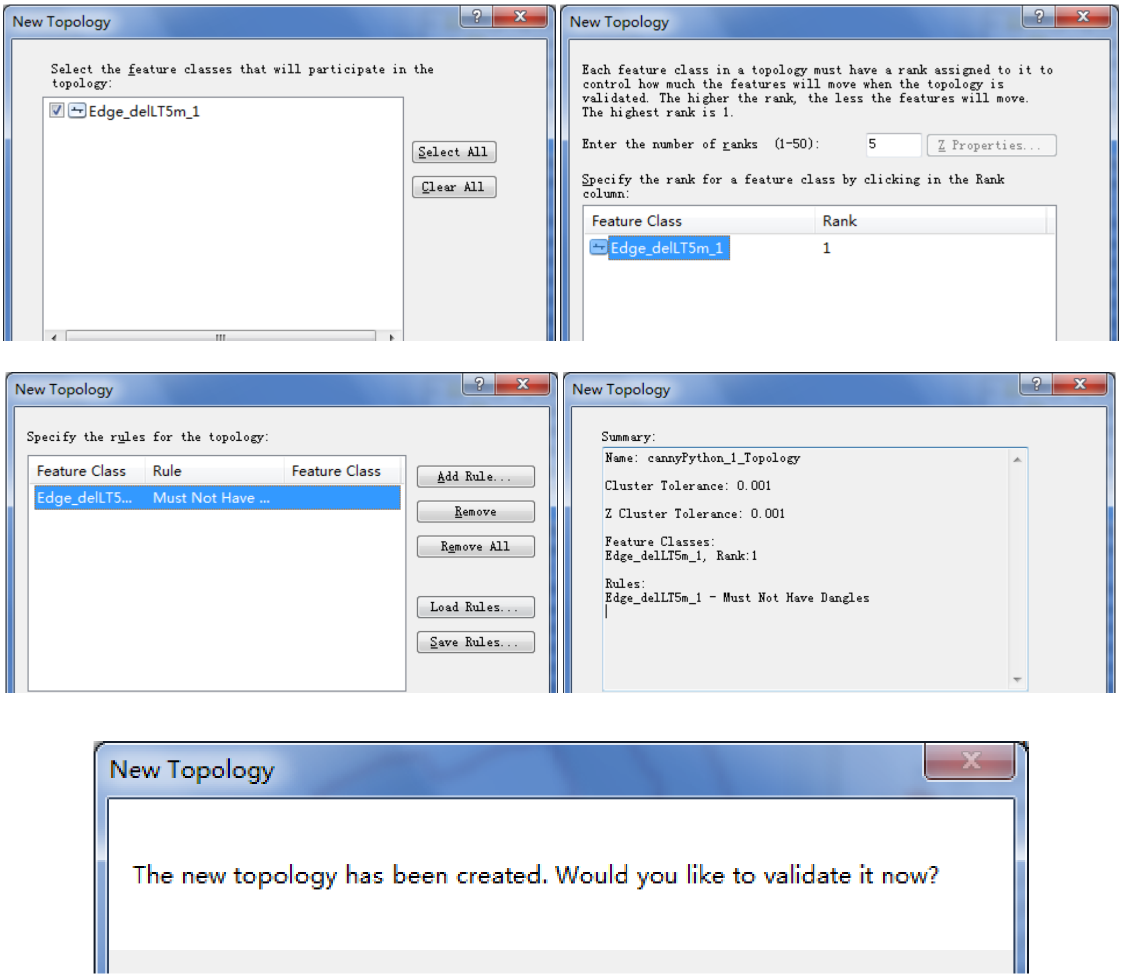


Figure A3.4 **Steps for creating topology rules for vectorized edges**

After validating the pre-defined topology rule, the topology errors can be checked. The "Fix topology error tool" (arrow with a red cross in the Topology toolbar) is used to select those dangled points (black squared points after selecting, pink squared points before selecting) to see the detailed information (Figure A3.5 Left). The error points need to be fixed by right clicking the error list to pop up fixing tools such as snap, extend and trim (Figure A3.5 Right). Considering most of the polylines with dangling endpoints are nearly perpendicular to their encountering polylines, an iteration with different levels of cluster tolerance thresholds (2 meters, 4 meters, 6 meters etc.) based on empirical observation is set for the "Extend" function to extend the dangled endpoints until most of them are joined to the polylines encountered on the way of extending direction.

Afterwards, the “Snap” function is then used to attach the rest of dangled endpoints to the nearest polylines with attempted snap tolerance values.

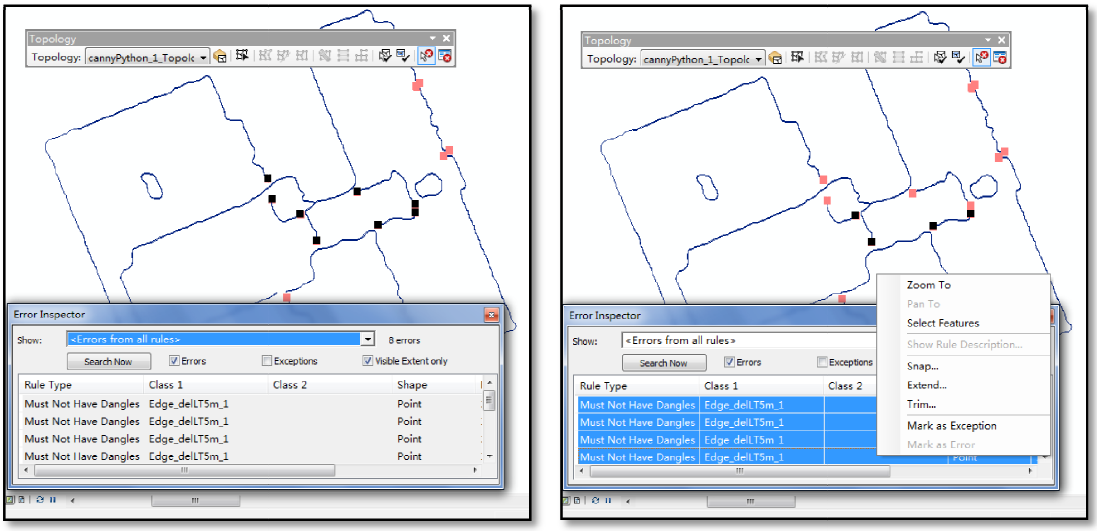


Figure A3.5 **Edge processing according to topology errors. Left: selection of error points; Right: edge processing for selected error points**

After edge processing, the closed edge polylines ((a) in Figure A3.6) are used to generate polygons that represent single roof planes, which is implemented by using the tool “Feature to Polygon” in ArcGIS (Figure A3.7). The generated polygons is shown in (b) of Figure A3.6. Noticing that there are many small polygons that can be merged with their surrounding larger polygons, small polygons with areas less than 20 m^2 are merged with neighboring polygons by dropping the shared border. This step is realized by selecting the polygons using “Select Layer by Attribute” and using “Eliminate” tool to join those small polygons to their surrounding larger polygons via dropping the shared border (Figure A3.8). The neighboring polygons refer to the ones that have the longest shared border with those small polygons. The revised polygons are shown in (c) of FigureA3.6.

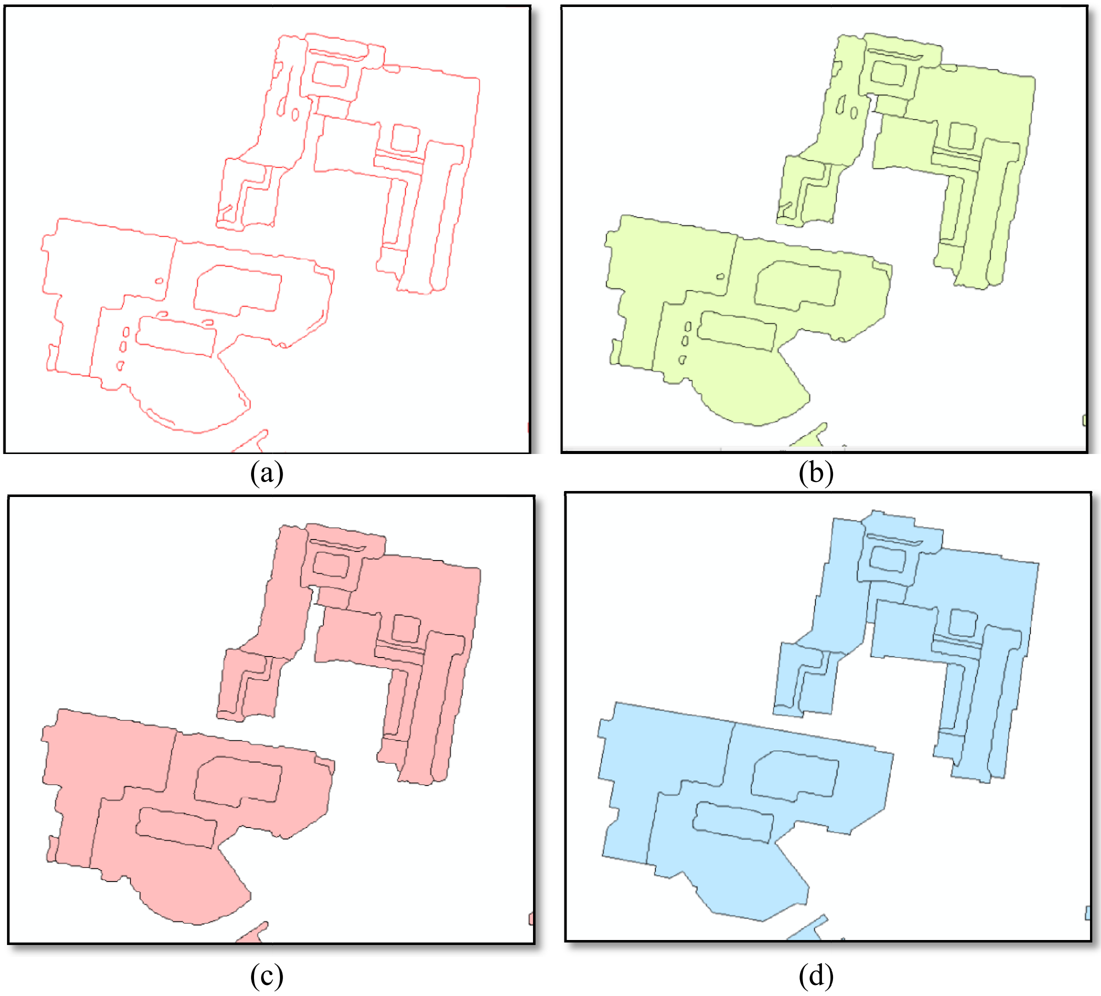


Figure A3.6 **Results after different stages of edge processing. (a) closed edge polylines; (b) polygons derived from closed edge polylines; (c) Revised polygons with small polygons removed; (d) polygons after boundary processing.**

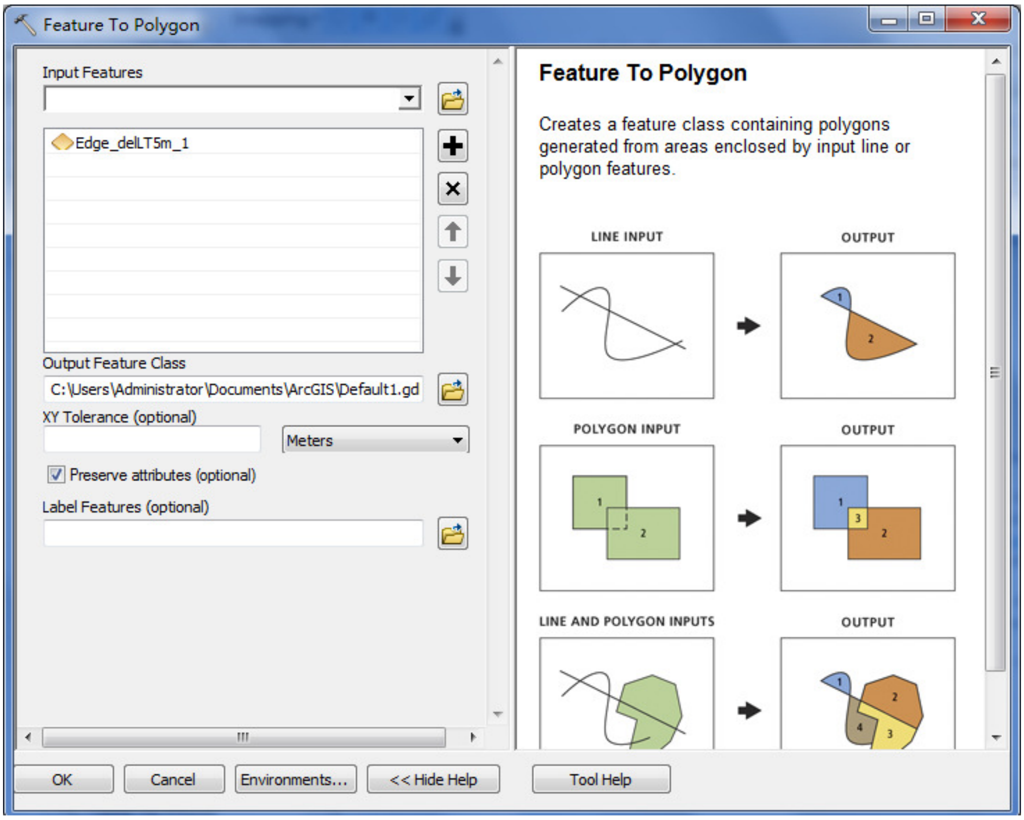


Figure A3.7 **"Feature to polygon" tool in ArcGIS**

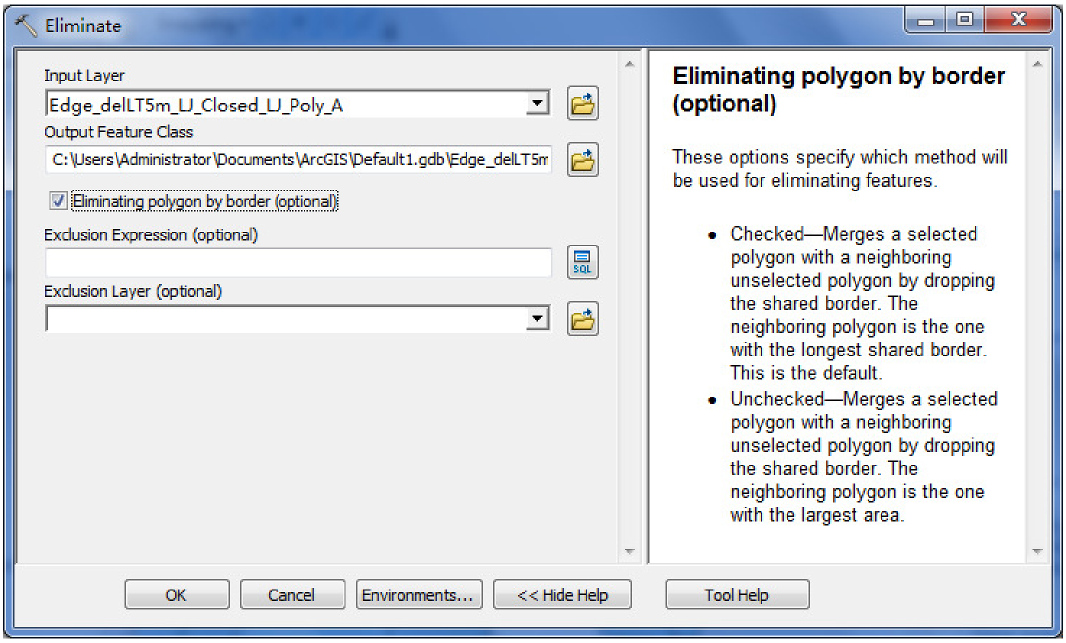


Figure A3.8 **"Eliminate" tool in ArcGIS**

A3.3 Boundary processing

Since the building edges are detected within the extent of the cropped nDSM data, the outlines of the polygons derived from the building edges do not perfectly overlap with the extracted building boundary. The final processing step is to overlay the edge polygons and the extracted building footprints to regularize building edges. This part is elaborated in Chapter 3 and stressed again in this session. The differential areas (pink areas shown in Figure 3.7 (c)) are identified using "Union" tool to overlap the polygons and the extracted building boundaries (Figure A3.9). Only the differential areas within the extracted boundaries (red boundary) are merged to the neighboring polygons with the longest sharing border using "Eliminate" tool again. The revised polygons are shown in (d) of Figure A3.6.

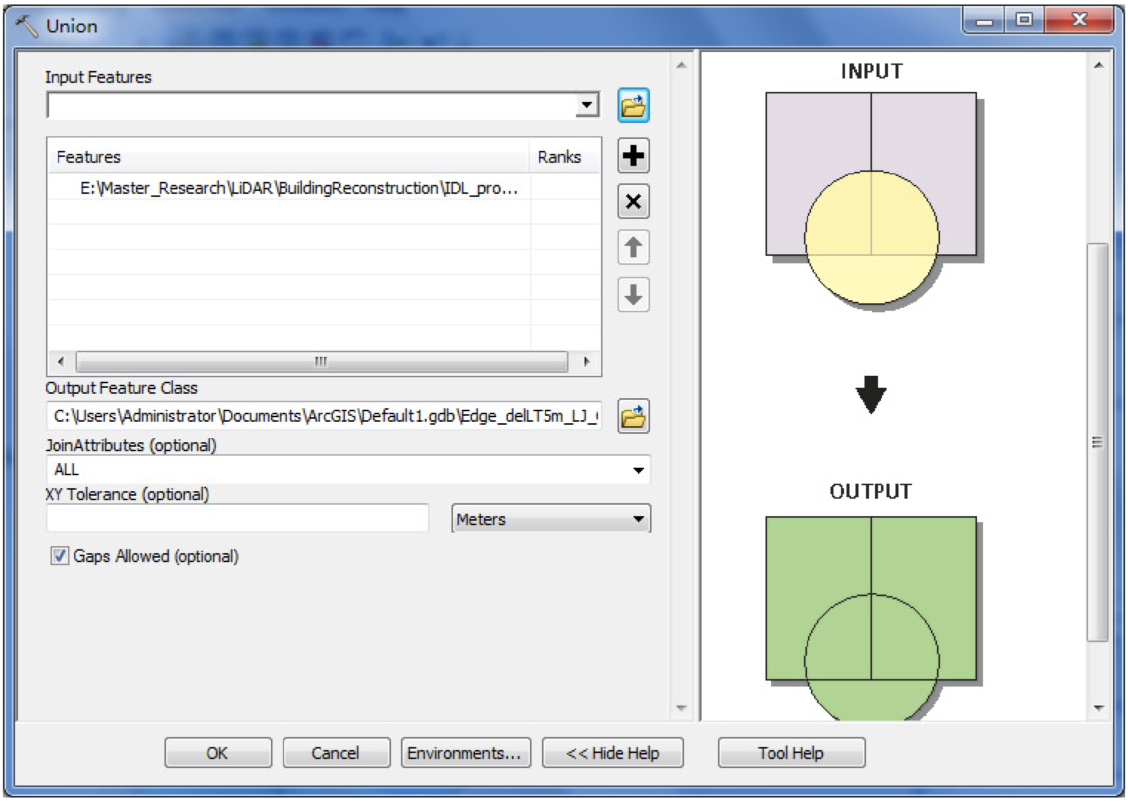


Figure A3.9 **"Union" tool to join the two polygon features together**

A3.4 Classification of roof shape

The average heights and slopes of the roof segments are assigned those roof segments using Zonal Statistics in ArcGIS 10 (Figure A3.10). The slope parameter of the nDSM data is used to classify the types of roof segments within the same building, for which two types of roof segments (flat and pitched) are considered. Based on author's knowledge of the study area, roof segments with average slopes less than 20 degrees are classified as flat roof segments and those with average slopes in the range of 20 and 60 degrees as pitched roofs. Those roof segments with slopes of more than 60 degrees are assigned to flat roof segments as well. This step is easily realized by classifying the polygons into three classes according to slope values (Figure A3.11). Polygons representing flat roof segments and pitched roof segments are exported separately as polygon shapefiles in ArcGIS

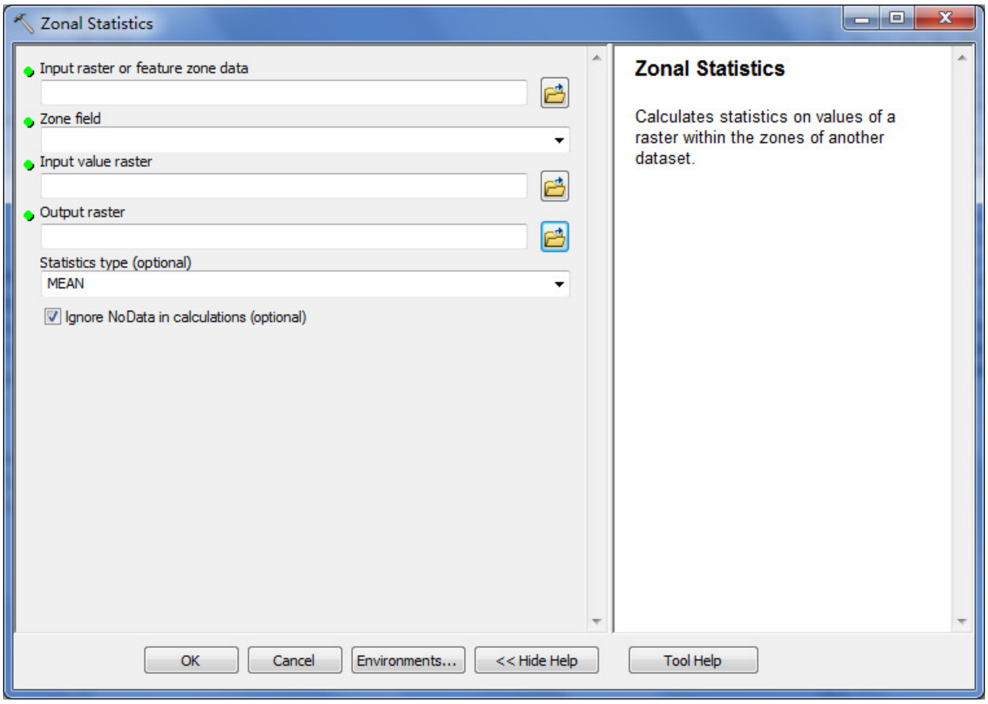


Figure A3.10 **The "Zonal Statistics" tool in ArcGIS**

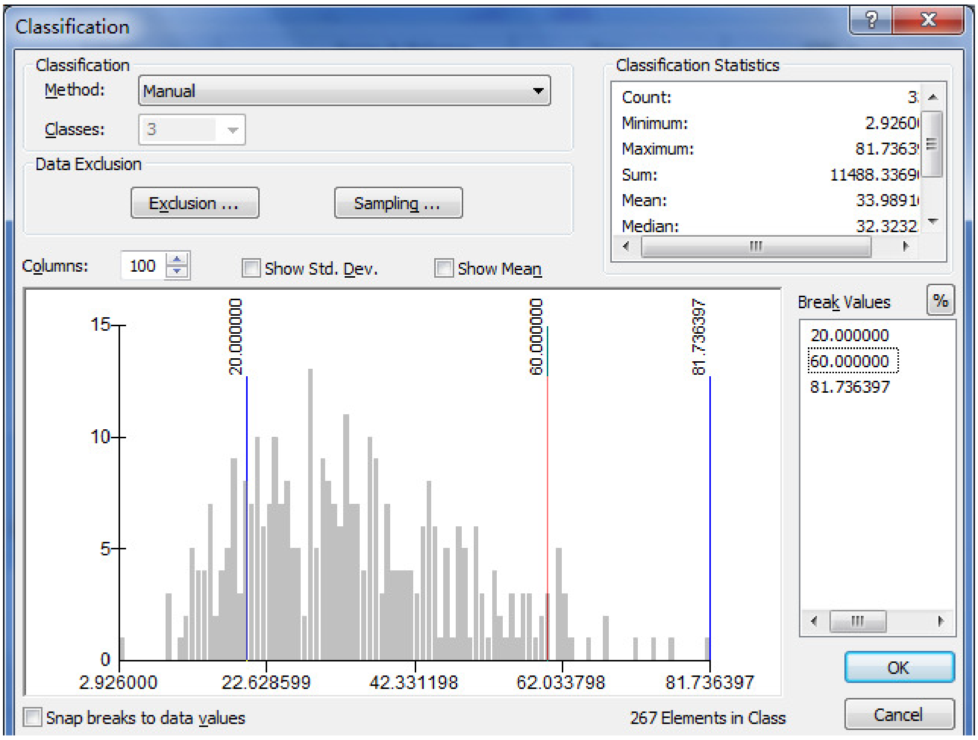


Figure A3.11 **Classify the slopes into three categories.**

A3.5 Building reconstruction

For the polygons representing flat roof segments, visualization can be easily made through applying the mean height values to polygons in the "Extrusion" function (Figure A3.12) of ArcScene, which is a platform for 3D modeling and visualization. By choosing the height values and adding them to each polygon's base height (in this study, the elevation in the DTM data), 3D models of flat roof segment are visualized in Figure A3.13.

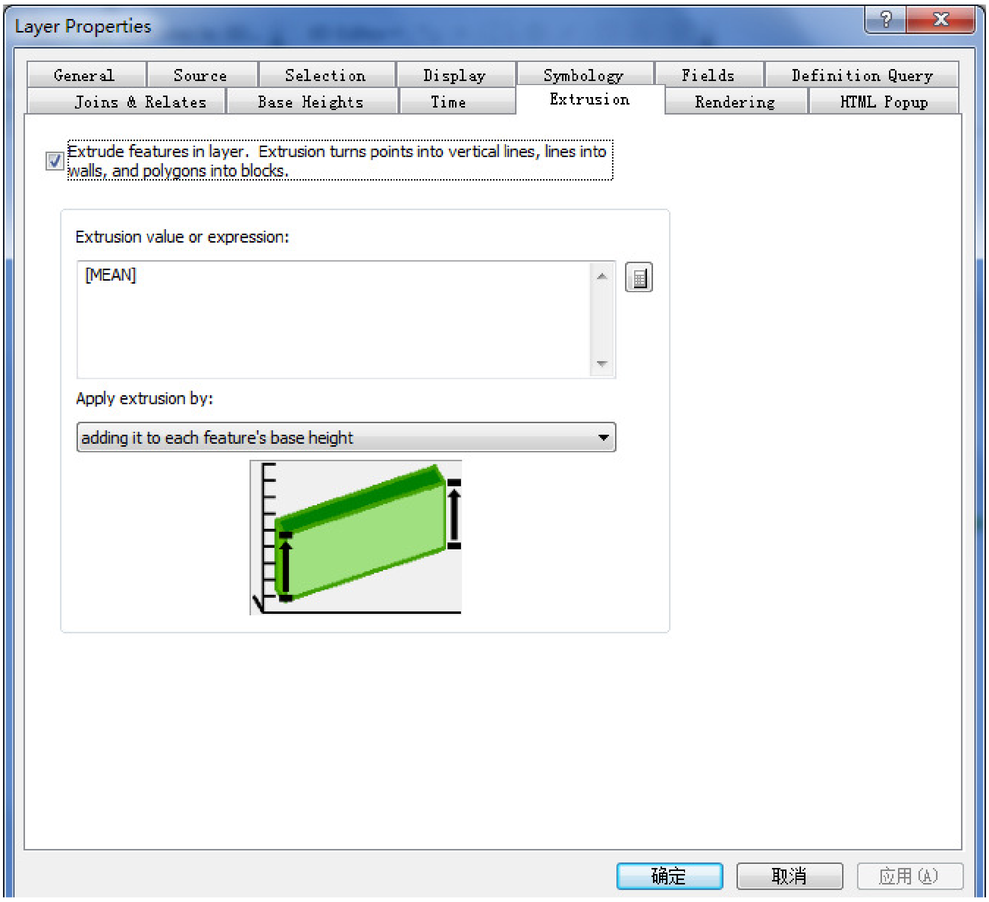


Figure A3.12 **"Extrusion" function in ArcScene**

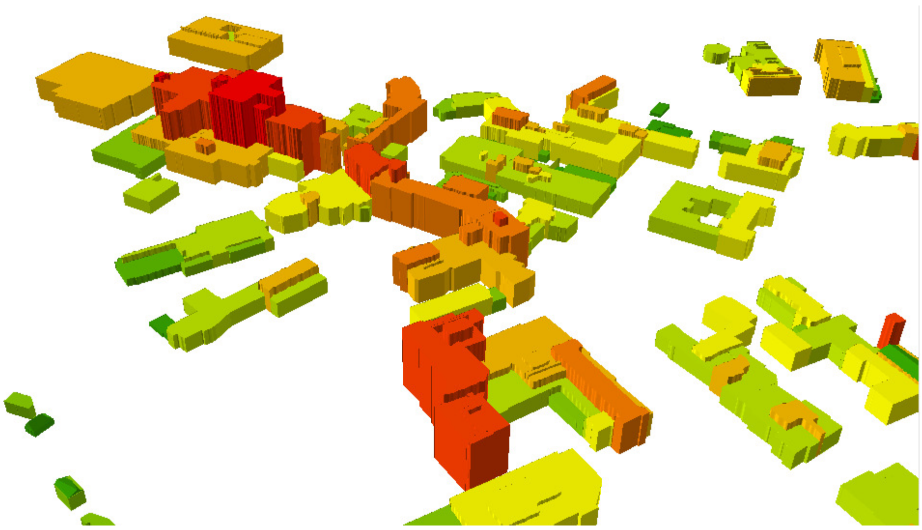


Figure A3.13 **3D visualization of flat rooftops**

Compared with flat roofs, roof segments with pitched surface are more difficult to visualize since they cannot just be extruded to a single height. Even though the pitched roof segments are detected from the classification of roof shape, the intra-structures of which are not well identified using Canny detector because there are no obvious elevation difference in ridgelines. The aspect parameter of the nDSM data can clearly distinguish different surfaces of pitched roof polygons without the need of defining the dominant building orientation. Therefore the aspect is used to classify sub-polygons within the pitched rooftops. "Aspect" tool in ArcGIS is applied to create the aspect image derived from the nDSM of the first return data (Figure A3.14).

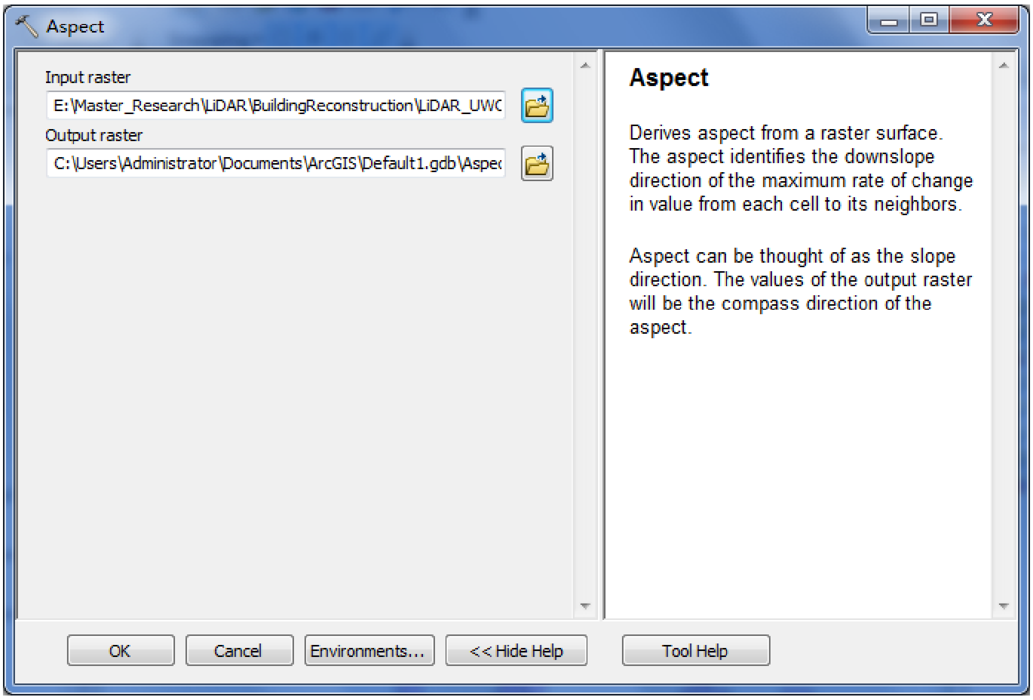


Figure A3.14 **"Aspect" tool in ArcGIS**

The ridgelines are identified by again applying the Canny edge detector to the aspect image within the boundaries of the pitched roof polygons. The sudden changes in aspect values can be easily detected by Canny algorithm. Afterwards, similar edge processing (see section A3.2 and A3.3) is employed to generate closed polylines. The main part of the Canny code for ridgeline detection can be seen as follows:

```
;-----------------------------------------------------------------------
procanny_test2_pitchedroof
t0=systime(1)

Filename='E:\Master_Research\LiDAR\BuildingReconstruction\IDL_programmi
ng\pitched_aspect3.tif'
GetData,ImgData = ImgData,ns = ns,nl = nl,nb = nb,Data_Type =
Data_Type,FileName = FileName,Map_info = map_Info

for a = 0, 3dobegin;from 0.6 to 0.9, with 0.1 interval
for b = 0, 3dobegin;from 0.2 to 0.5, with 0.1 interval
for c = 0, 19dobegin;from 0.6 to 2.5
img_canny = canny(ImgData,HIGH=0.1*a+0.6,LOW=0.1*b+0.2,SIGMA=0.1*c+0.6)
          x=0.1*a+0.6
          y=0.1*b+0.2
          z=0.1*c+0.6
Envi_Write_Envi_File,img_canny,Out_Name =
'E:\Master_Research\LiDAR\BuildingReconstruction\IDL_programming\canny_
test2\Pitched_canny_H'+strmid(strtrim(string(x),1),0,4)+'_L'+strmid(str
trim(string(y),1),0,4)+'_S'+strmid(strtrim(string(z),1),0,4)+'.tif',Map
_info = map_Info
endfor
endfor
endfor
print, 'The spent time is', systime(1)-t0, 'seconds'

end
;-----------------------------------------------------------------------
```

Those processed polylines (Figure A3.15 Left) are then densified in ArcGIS 10 by inserting vertices along polylines so that there are enough and evenly distributed vertices for creating TIN structures in the 3D visualization (Figure A3.15 Right).

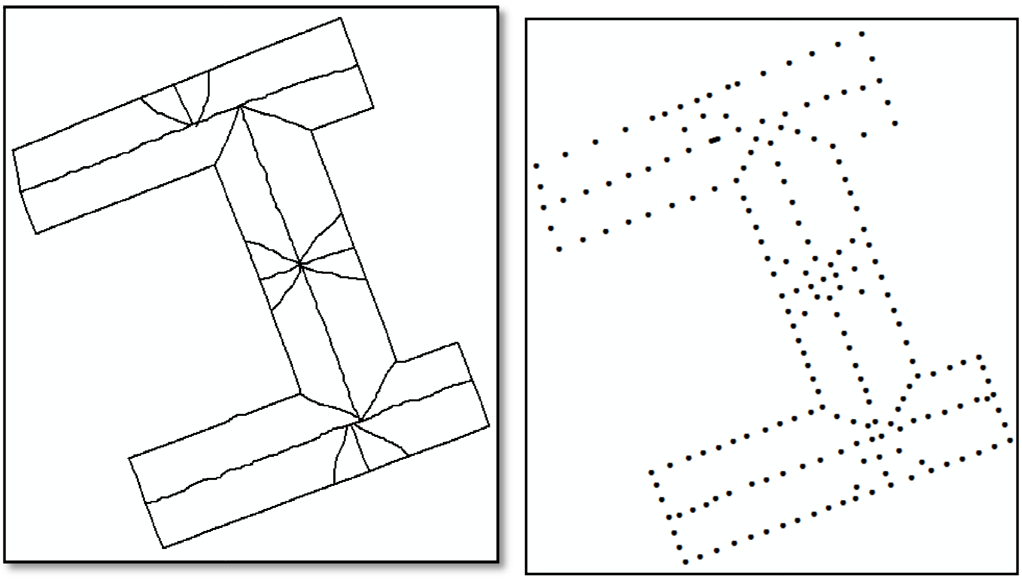


Figure A3.15 **Before and after performing "Densify" tool in ArcGIS**

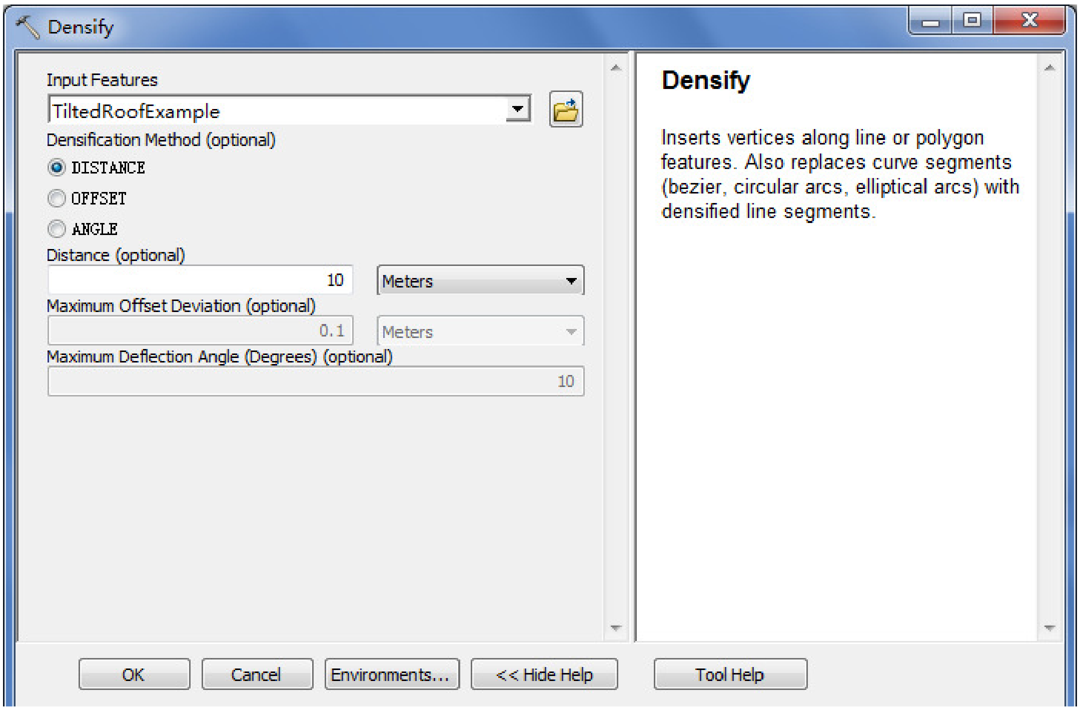


Figure A3.16 **"Densify" tool in ArcGIS**

In the "Densify" tool, the densification method and distance need to be specified. Usually the "Distance" method is chosen for the function. The distance should be adequate but not too concentrated to influence the performance of TIN structures. In this case, the distance of 10 meters is chosen based on empirical observation.

The densified points are then assigned with the corresponding height values of the nDSM data, thus every vertex has an elevation value representing the building height in that location. This step can be done using the "Extract Values to Points" tool in ArcGIS (Figure A3.17). However, the assigned heights of those vertices cannot be all the same due to many reasons (e.g. the positional accuracy of roof segment outlines and ridgelines, noises along building edges caused by interpolation), which can affect the visualization performance. To compensate the problem, the points located at the outlines of the pitched roof segments are selected by intersecting with the extracted building footprints (Figure A3.18), the mean height values of those points are then calculated and re-assigned to each point.

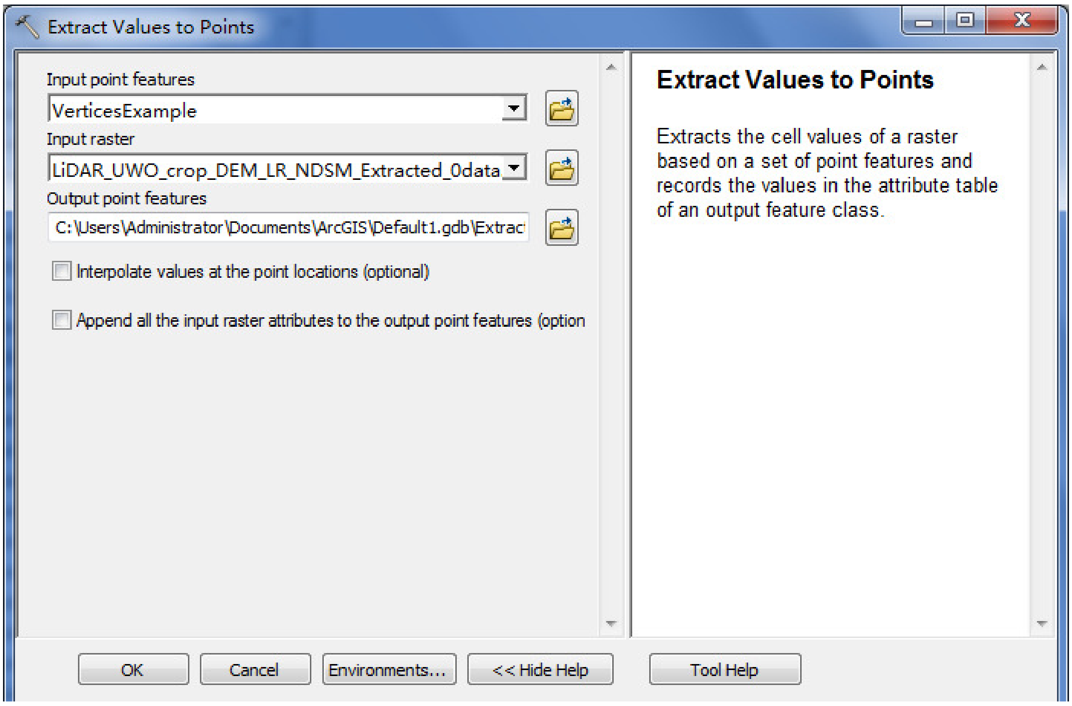


Figure A3.17 **"Extract Values to Points" tool in ArcGIS**

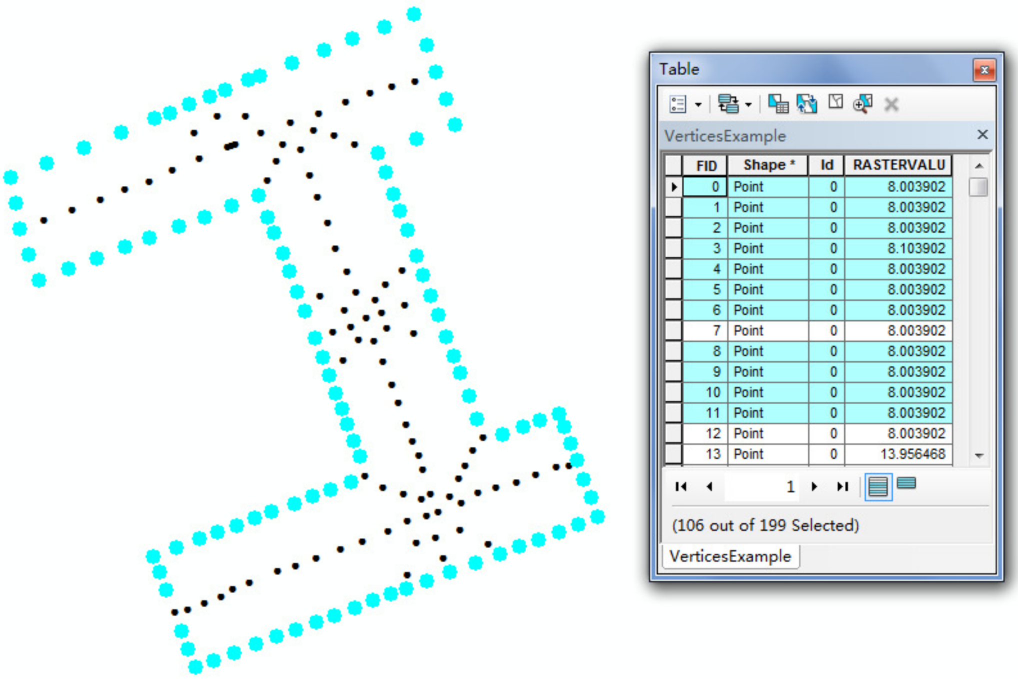


Figure A3.18 **Selected points in building edge areas (points highlighted in cyan)**

Triangulated Irregular Network (TIN) structures are created based on those vertices using the "Create TIN from Features" tool in ArcScene (Figure A3.19). However, the points from one of roof structures can possibly form triangle nets with points from another. Therefore, the boundaries of the pitched segments are used as masks on the vertices so that each TIN structure created within would represent a single pitched roof plane.

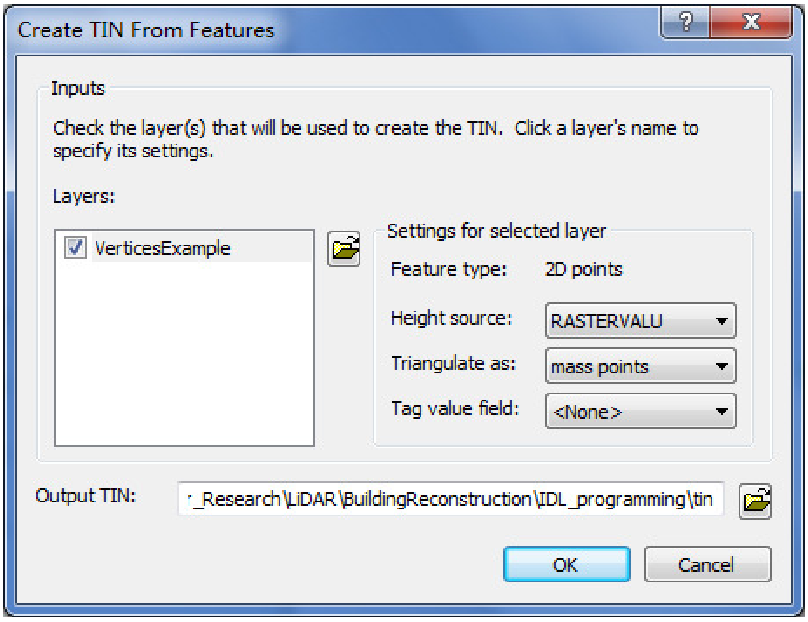


Figure A3.19 **"Create TIN From Features" tool in ArcScene**

For the final visualization of pitched roof planes, the base frames and the rooftops are dealt with separately. The average height values of the densified points located at the roof segment outlines are assigned the respective roof segments, which are extruded by adding the average heights to the base height to represent the base frames. Finally the TIN structures of the pitched roof planes are piled upon the base frames (Figure 3.11).

A3.6 References

ESRI, 2011. ArcGIS Desktop: Release 10. Redlands, *CA: Environmental Systems Research Institute.*

Appendix 4

Glossary of Terms and Abbreviations

DEM (Digital Elevation Model) –is often used as a generic term for DSMs and DTMs, only representing height information without any further definition about the surface.

DSM (Digital Surface Model) –represents the elevation of the earth's surface and includes all objects on it.

DTM (Digital Terrain Model) –represents the elevation of bare ground surface with all objects like trees and buildings removed.

GIS (Geographic Information Systems) –is a combination of the computer hardware, software, data, methods and people, and it aids in the collection, maintenance, storage, analysis, output and distribution of spatial data and information.

GPS (Global Positioning Systems) –is a Global Navigation Satellite System (GNSS) that gives precise x, y, z positional information, day or night, in most weather and terrain conditions, anywhere on or near the Earth surface.

IMU (Inertial Measurement Unit) –is an electronic device that measures and reports on a craft's velocity, orientation, and gravitational forces, using a combination of accelerometers and gyroscopes, sometimes also magnetometers.

LiDAR (Light Detection And Ranging) –is an active remote sensing technique that can measure the elevation and the reflected energy of targets by illuminating the target with laserlight. LiDAR is able to collect an accurately geo-referenced set of dense point clouds by recording reflected laser beams emitted by aircraft.

NAD (North American Datum) –is the official datum used for the primary geodetic network in North America.

nDSM(Normalized Digital Surface Model) –The difference model of a DSM and a DTM is called nDSM, which is a representation of the height of objects on a plane surface.

NDTI (Normalized Difference Tree Index) – can be calculated using the nDSM of the first return and last return to represent elevation difference between the two returns:

$$\mathrm{NDTI} = \frac{\mathrm{nDSM_{FR}} - \mathrm{nDSM_{LR}}}{\mathrm{nDSM_{FR}} + \mathrm{nDSM_{LR}}}$$

Generally, elevation difference between first return and last return data is caused by penetration through trees. Therefore NDTI can be used to remove trees from buildings in the non-ground objects.

RS (Remote Sensing) –is the science and art of obtaining information about an object area object, area, or phenomenon through the analysis of data acquired by a device that is not in contact with the object, area or phenomenon under investigation.

TIN (Triangular Irregular Networks) – TIN structures are a digital means to represent surface morphology. TINs are a form of vector-based digital geographic data and are constructed by triangulating a set of vertices (points).

UTM (Universal Transverse Mercator) – is a projected coordinate system that uses a 2-dimensional Cartesian coordinate system to give locations on the surface of the Earth. It is a variation of Mercator projection with horizontal position representation.

Curriculum Vitae

Name:	Ting Zhao
Post-secondary Education and Degrees:	Beijing Normal University Beijing, China 2007 - 2010 B.Sc. Geography (Resources Science and Engineering) The University of Western Ontario London, Ontario, Canada 2011 - 2013 M.Sc. Geography (GIS and Remote Sensing)
Honours and Awards:	Western Graduate Research Scholarship, UWO 2011 – 2013 ESRI GIS Graduate Scholarship, Faculty of Social Science 2012 – 2013 The Prof. Elaine Bjorklund Scholarship, Department of Geography 2012 – 2013
Related Work Experience	Research Assistant, Teaching Assistant The University of Western Ontario, Department of Geography 2011 - 2013

Conferences:

Zhao, T. and Wang, J., 2012."Object-based Extraction of Building Footprints from Airborne LiDAR Data", presented at the 33rd Canadian Symposium on Remote Sensing. June 11 – 14, 2012, Ottawa, Canada.

Zhao, T., He, C. and Zhao, Y., 2010."Monitoring Vegetation Dynamics in Haihe River Basin, China by Coupling Linear Trend Analysis and Change Vector Analysis", presented at the Academic Conference of physical geography & environmental change, July 31 – August 1, 2010, Shijiazhuang, Hebei Province, China.