UCGE Reports
Number 20282

Department of Geomatics Engineering

Integration of Photogrammetric and LiDAR Data for Accurate Reconstruction and Visualization of Urban Environments

(URL: http://www.geomatics.ucalgary.ca/research/publications)

by

Anna M. Y. Jarvis

December 2008

UNIVERSITY OF CALGARY

Integration of Photogrammetric and LiDAR Data for Accurate Reconstruction and Visualization of Urban Environments

by

Anna M. Y. Jarvis

A THESIS
SUBMITTED TO THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE
DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF GEOMATICS ENGINEERING
CALGARY, ALBERTA
DECEMBER, 2008

Abstract

Urban infrastructure and city layout is a topic of great interest in recent years. According to trends, the number of people living in urban areas is steadily increasing. Accurate and user-friendly methods of spatial information visualization and analysis are required to aid in the management and planning of urban area expansion. Three-dimensional (3D) city models, in particular, can be used for this purpose. Common sources of remote sensing data employed for 3D models of urban areas include Light Detection and Ranging (LiDAR) data and imagery. These data sources can be used to obtain the main features of a city model, namely the terrain and the buildings. The focus of this research is the generation of a digital building model (DBM) that is then added to a ground TIN (Triangulated Irregular Network) to produce accurate realistic visualization of 3D environments, implemented within a GIS platform. The production of the DBM is performed through the use of both data sources. The LiDAR data is used to obtain initial building-primitive boundaries as well as to obtain the height information for the created building model. The image data is used to refine the rough building-primitive boundaries derived from the irregular LiDAR data. Once the DBM is created, it is added to a ground TIN of the area under consideration, to produce a refined TIN surface model. The TIN is then converted to raster to obtain an enhanced DSM which is used in turn to generate improved true-orthophotos. The refined orthophotos are then draped over the refined digital surface model (DSM) for realistic visualization of the 3D urban environment. Quantitative and qualitative analysis of the final products were performed and analyzed using real data. Results from RMSE analyses found that the proposed method achieved results similar to the manual generation of a DBM, which were in the range of 33cm.

Acknowledgements

I would like to take a moment to thank all those to whom I owe a great deal of gratitude for their help and support throughout my research studies.

First and foremost I must express my sincere gratitude to my supervisor Dr. Ayman Habib, for his guidance and expertise over the past two years. Through working with him, I have gained priceless experience and knowledge during the various research projects and collaborations. In addition, his comments and guidance during the preparation and writing of this thesis are greatly appreciated. I am also very grateful to Dr. Quazi Hassan, Dr. Xin Wang, and Dr. Dan Jacobson for their time invested in the careful review as well as comments provided towards the write-up of this research work.

Acknowledgements must also be given to the University of Calgary IT Department (especially Tom McCaffrey) for providing the data that was used in testing the proposed research methodology, as well as the funding provided by NSERC, ETRI, and GEOIDE (SII-43 and IV-17). Many thanks are in order for my colleagues in the Digital Photogrammetry Research Group (DPRG) for both their scholastic and emotional support, in particular; Changjae Kim, David Chang, Ruifang Zhai, Juliano and Ana Kersting, Ki-In Bang, and Mohannad Al-Durgham, as well as former member Nand Jha. Furthermore, I would like to thank my friends from back home, as well as those I have made during my time in Calgary. In addition, I want to thank Mitch for his support and constant encouragement, as well as my parents and my brother Michael who have always supported and encouraged me to pursue my ambitions and dreams.

Dedicated To My Beloved Parents

Hélène and Gary

For their constant love and support.

"Life's a journey, not a destination"

Aerosmith

Table of Contents

List of Tables

List of Figures and Illustrations

List of Symbols, Abbreviations and Nomenclature

Symbol	Definition
2D	Two Dimensional
3D	Three Dimensional
BA	Bundle Adjustment
DBM	Digital Building Model
DSM	Digital Surface Model
DTM	Digital Terrain Model
EOPs	Exterior Orientation Parameters
EQC	External Quality Control
GCP	Ground Control Point
GIS	Geographic Information System
GPS	Global Positioning System
GSD	Ground Sampling Distance
INS	Inertial Navigation System
IOPs	Interior Orientation Parameters
IQC	Internal Quality Control
LiDAR	Light Detection And Ranging
QA	Quality Assurance
QC	Quality Control
RMSE	Root Mean Squared Error
TIN	Triangulated Irregular Network

CHAPTER ONE: INTRODUCTION

1.1 General Introduction

1.1.1 Overview

Urban planning and city layout is a topic of great interest in recent years. According to current trends, the number of people living in urban areas is steadily increasing (Nichol et al., 2007). To aid in the management and planning of urban area expansion, fast and accurate methods of spatial information visualization and analysis are required. Historically, the paper map was the main interface between map-maker and user. These maps contain a fixed array of attributes and an invariant scale, which limits their use (Longley et al., 2005). In addition to the paper map, Ranzinger (1997) summarizes some of the other traditional visualization techniques used to convey 3D information: elaborate wood or pasteboard models, perspective drawings, etc. These techniques, in particular the wood and the pasteboard models, can be quite expensive and the models are not easily modified without significant effort. Recently, digital representations of three dimensional city models have become a commonly employed tool for urban analysis and visualization, partly fuelled by the increasing computer capabilities which have allowed for the growth in the representation of three-dimensional (3D) environments. Moreover, 3D urban planning has become one of the most often quoted areas of human activity (Zlatanova et al., 2002). Planners, architects, and other decision makers can visualize possible construction projects and explore various changes, such as building location and size, in real-time. In addition, a computer based 3D urban model allows users to navigate through space and examine areas of interest from different view angles and scales. City models can also be utilized to simulate floods and other disasters, which are employed for planning and prevention purposes. Other applications for 3D city models include marketing, investing, tourism, and telecommunications. This vast range of users and applications of 3D city models has led to new research activities in an effort to meet the need for fast and accurate techniques for the creation of 3D urban models. The following section will investigate the various data sources that can be used for the generation of 3D models.

1.1.2 Common Data Sources

When building a 3D city model, it is important to consider which features to include in the final model. The base features for a city model, are undoubtedly the surface model of the area, as well as ortho-imagery which can be used to add texture information to the model. In addition to these crucial layers, most models contain roads, trees, parks, and street signs (Ranzinger, 1997). Common methods of obtaining three-dimensional positional information for the 3D model features are photogrammetric, LiDAR (Light Detection and Ranging), and surveying techniques. Photogrammetry is a common method used to obtain 3D information from 2D imagery, in which mathematical calculations are performed to obtain the object space coordinates of points based on: the camera internal characteristics; the location and orientation of the camera; image measurement; and ground control data.

In contrast to photogrammetry where a point must be measured several times (in overlapping images), LiDAR systems measure an object space point once. The LiDAR system consists of a laser scanner, a GPS receiver, and an inertial navigation system

(INS). The LiDAR system functions by emitting a laser pulse and measuring the return time. Based on the return time of the pulse, the distance between the scanner and the measured point can be obtained, and used in conjunction with the GPS and INS observations to compute the object's coordinates (Wolf, 2000).

A third method of obtaining three-dimensional attribute information is through field surveying. Surveying techniques are perhaps the most traditional form of obtaining positional information. There are several surveying techniques, which include the use of Total Stations and/or GPS surveying techniques. Surveying can obtain very accurate attribute coordinates (up to the millimetre level (Anderson and Mikhail, 1998)), but involves a significant amount of field work as well as post-processing. In addition to these outlined methods, data can be obtained through map digitizing as well as from data located within existing databases. The organization, integration, analysis, and display of the various data sources can be performed within a geographic information system (GIS). The integration of the data within a GIS is discussed in the next section.

1.1.3 Integration in a GIS platform

Once the data sources for a desired 3D city model are determined, the software environment in which the model is to be built and displayed must be selected. The integration of a 3D city model within a GIS platform can be very advantageous. Urban infrastructure information including city streets, building locations, and road signs, are often readily available in existing GIS databases. The existing GIS layer information can be incorporated into the 3D city models. In addition, the increasing availability and affordability of data acquisition systems coupled with the increasing computer capabilities allow for the creation of detailed and accurate 3D city models within a GIS. Furthermore, the use of pre-existing GIS functions limits some repetition in programming efforts. For instance, Shiode (2001) notes that 3D urban models located within a GIS platform benefit from pre-existing functions such as; view-shed analysis, spatial queries, and shadow analysis. In addition, these models inherit GIS storage, management, and editing capabilities (Köninger, 1998). Moreover, Shiode (2001) notes that, in comparison to 3D GIS city models, photo-realistic CAD-type models lack the attribute data that can be stored in a GIS model, which is often necessary for certain analysis purposes. Thus it is a natural extension to incorporate 3D models and capabilities into the traditional GIS platform.

1.2 Problem Statement and Research Objectives

To avoid social and environmental problems that can arise from rapid urbanization, accurate and current geospatial information is required in a form that is simple to understand and analyze. Although the reconstruction of urban infrastructure and mapping of city layout is not a new topic of interest, there is a need for practical and accurate tools for exploiting geospatial data acquired by modern remote sensing systems. Digital 3D city models can be utilized to address this need. There is a vast range of users (governments, planners, environmental agencies, telecommunications and utility companies, consultants, architects, etc.) who require increasingly sophisticated 3D city models to plan and monitor urban services and impacts (Shiode, 2001). 3D city models are powerful instruments in the visualization of urban and built-up environments as they provide important information that is easy to understand (Shiode, 2001). Nichol et al. (2007) outlines some required improvements for more effective urban models, in terms of data sources used and the 3D environment, to allow for the creation of accurate 3D models in a timely manner. Two of the outlined requirements include: 1) The use of high resolution imagery coupled with improvements in automated recognition of complex objects to avoid large quantities of manual digitizing; and 2) The development of automatic 3D segmentation algorithms to obtain accurate 3D building models from LiDAR data. The complimentary information provided by photogrammetric and LiDAR data is one of the main reasons these two sources of data are commonly used for urban planning (Nichol et al., 2007), often within a GIS framework. For instance, photogrammetry provides texture information, good horizontal accuracy, and offers detailed information regarding building boundaries. LiDAR in contrast offers dense information on homogenous surfaces, and provides good vertical accuracy. Table 1.1 outlines the respective advantages and disadvantages of these two data sources, and thus highlights how the complimentary nature of these data can be utilized to obtain optimal accuracy and a more complete description of the object space. In addition, the integration of GIS with remotely sensed data can aid in the automation of procedures involved in 3D city modeling. For these reasons, this research focuses on the integration of the photogrammetric and LiDAR data within a GIS platform.

Table 1.1: Complimentary nature of Photogrammetric and LiDAR data

Photogrammetry Pros	**LiDAR Cons**
Provides colour information	No colour information
Good horizontal accuracy	Less accurate horizontal information
Good information for building boundaries	Less accurate building boundaries (due to irregularity of point data)
LiDAR Pros	**Photogrammetry Cons**
Dense information on homogeneous surfaces	Almost no positional information on homogeneous surfaces (due to matching problems)
Day or night data collection possible	Only day time data collection possible
Directly acquires 3D information	Post-processing is needed to obtain 3D information
Good vertical accuracy	Less accurate vertical information

In this work, the LiDAR and photogrammetric data are used to produce the main components of a 3D model, namely orthophotos and a digital surface model (DSM) over the area of interest, and thus the quality of the produced 3D model is largely dependent upon the quality of these two important features. In some applications, the 3D urban model is created by simply draping ortho-imagery over a DSM of the area. This method may be sufficient for medium resolution imagery over relatively smooth terrain; however, more sophisticated procedures are required when dealing with high resolution imagery over urban environments. When the DSM of an urban area is derived from LiDAR, the irregular and sparse nature of LiDAR data leads to a degraded quality of building boundaries in the DSM. Furthermore, the quality of the orthophotos, which is directly affected by the DSM, will contain jagged building boundaries. Therefore, if the quality of the DSM can be enhanced, the surface model will be improved as well as the generated ortho-imagery.

This work outlines a framework for the integration of photogrammetric and LiDAR data within a GIS platform, for realistic 3D visualization of urban environments. To refine the DSM over an urban environment, a digital building model (DBM) for the area is obtained through the use of both LiDAR and photogrammetric data. When the DBM is added to a digital terrain model (DTM) of the urban area, a crisp surface model can be achieved. Improved true-orthophotos can then be generated using the refined DSM, as opposed to those generated from the original (unrefined) DSM. The improved true-orthophotos are then draped over the refined DSM, thus obtaining an accurate 3D realistic visualization of an urban environment. Figure 1.1 outlines the workflow for the creation of a 3D realistic visualization of an urban model starting from the input data to the final output product.

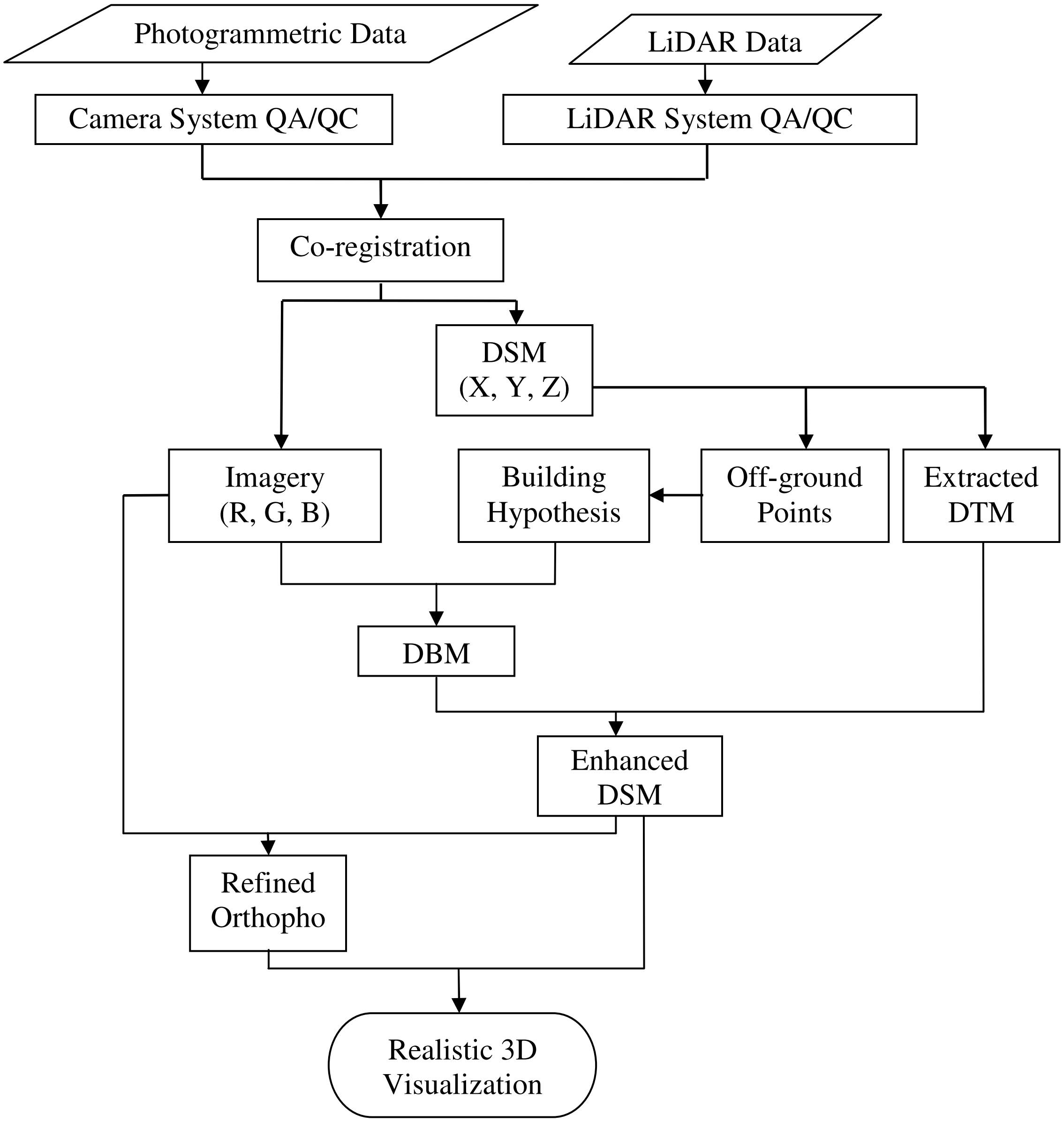


Figure 1.1: Workflow for 3D realistic visualization of an urban environment through the integration of LiDAR and photogrammetric data.

1.3 Thesis Outline

Chapter Two starts with a literature review of methods used for the production of a DBM, DSM, and ortho-imagery. An analysis follows outlining the inherent limitations in these methods. Chapter Three presents the pre-requisites that are required for the proposed DBM generation procedure. These pre-requisites include: camera and LiDAR quality assurance (QA) and quality control (QC); and image and LiDAR data co-registration. This last pre-requisite is the entry point of the proposed procedure for obtaining a DBM through a semi-automated procedure, involving LiDAR data and perspective imagery. The DBM generation procedure is presented in Chapter Four, and outlines the use of the DBM to enhance the DSM and the ortho-imagery. Experimental results and accuracy assessment of the outlined procedure are presented in Chapter Five, and final conclusions and recommendations for future work are then discussed in Chapter Six. Figure 1.2 depicts the final product of the proposed system, where a DBM is utilized to obtain a refined DSM and improved ortho-imagery, thus allowing for the creation of an accurate and realistic visualization of an urban environment.

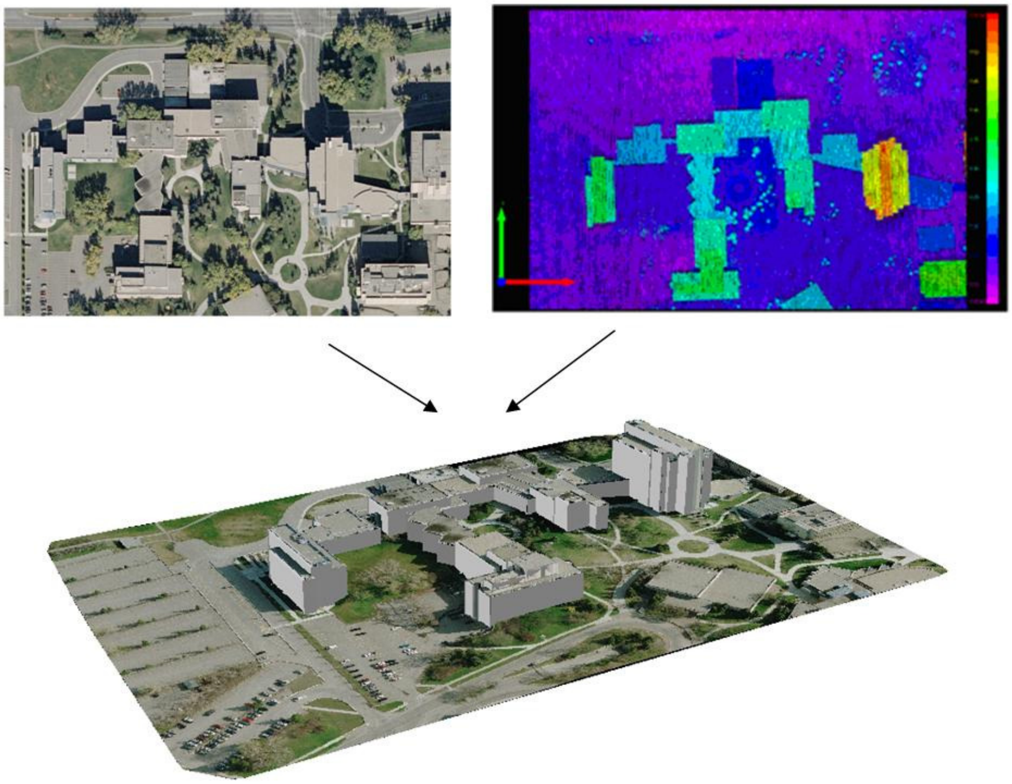


Figure 1.2: Integration of LiDAR and photogrammetric data to obtain an accurate 3D realistic model of an urban area.

CHAPTER TWO: BACKGROUND

Chapter one introduced the main features or layers of a 3D city model, namely the orthophotos and a DSM of the urban area. In addition, it was highlighted that refined orthophotos and DSM can be obtained through the use of a DBM. This chapter discusses the concepts of orthophotos, DSM, and DBM in terms of their purpose and the methodology for their creation through the review of previous research work conducted in these areas.

2.1 Orthophoto Generation

The creation of ortho-imagery is outlined in this section. Ortho-imagery is important for use in many fields of applications, and is very frequently used in a GIS, often as a background as well as to give texture information to a DSM. Note that the terms orthophoto and ortho-imagery will be used interchangeably in this work.

2.1.1 Introduction

The purpose of orthophoto production is to remove the effects of terrain relief and sensor tilt from the captured perspective imagery, thus producing an image with uniform scale that has the same characteristics as a map. The produced orthophotos often serve as one of the main layers in a 3D GIS model, and can also be used to directly measure distances, angles, and areas (Wolf, 2002). A common method used to produce orthophotos is differential rectification, where the grey values from imagery are assigned to each output cell of the orthophoto (Novak, 1992; Bang et al., 2007). The differential rectification procedure requires perspective imagery, a DSM, the exterior orientation parameters (EOP), and the interior orientation parameters (IOP). The EOP define the position and orientation of the camera at the moment of exposure relative to the object space coordinate system, while the IOP define the internal camera characteristics (coordinates of the principal point, camera constant, and distortion parameters). In this procedure, a uniform grid is defined over the output orthophoto plane, and the DSM is used to obtain the elevation, Z(X,Y), for each grid cell. Through the use of the collinearity equations, EOP, and IOP, the corresponding image point (x,y) is located. The grey value at the image location (x,y) is determined through use of a resampling technique, and this grey value is assigned to the orthophoto grid cell (X,Y), as depicted in Figure 2.1.

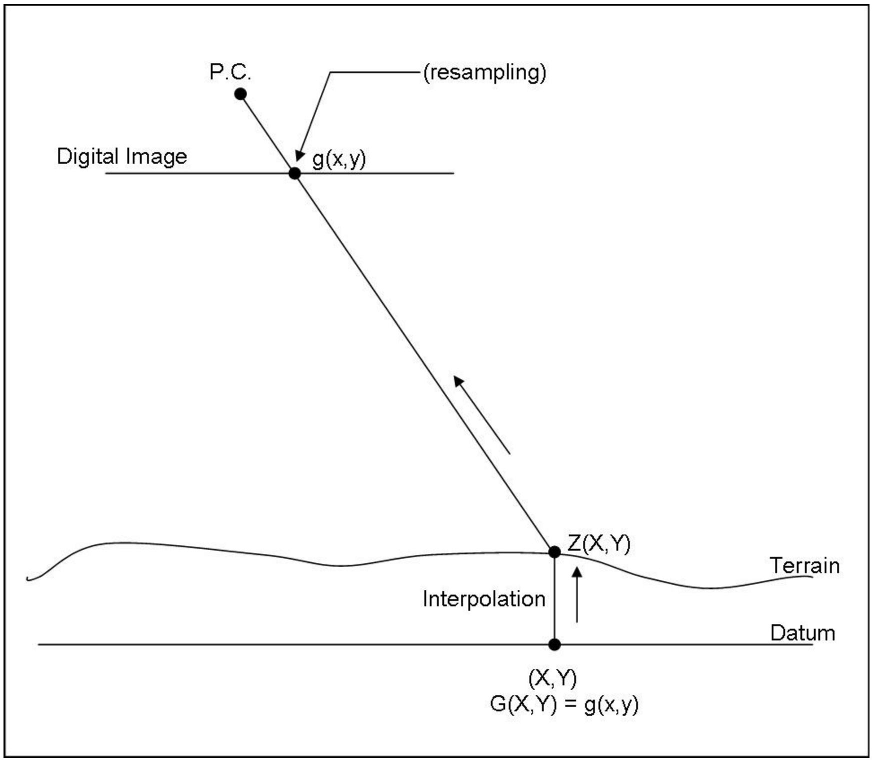


Figure 2.1: Concept of differential rectification.

One of the main limitations of the differential rectification procedure, however, is shown in Figure 2.2. This problem is known as the double mapping problem, which occurs when two object space points are competing for the same image location. Figure 2.2 depicts a scenario where the projected rays of a ground point and a building roof point yield the same location on the input image. The datum cells labelled *A* and *B* will thus be assigned the same grey value, shown as g(x,y) in the figure. As a result of tall buildings and other tall features (e.g., trees) occluding neighbouring areas in the perspective imagery, *ghost images* are produced in the orthophotos. Figure 2.3 illustrates the double mapping problem using a real dataset. A black outline is drawn over the double mapped areas to highlight the effect.

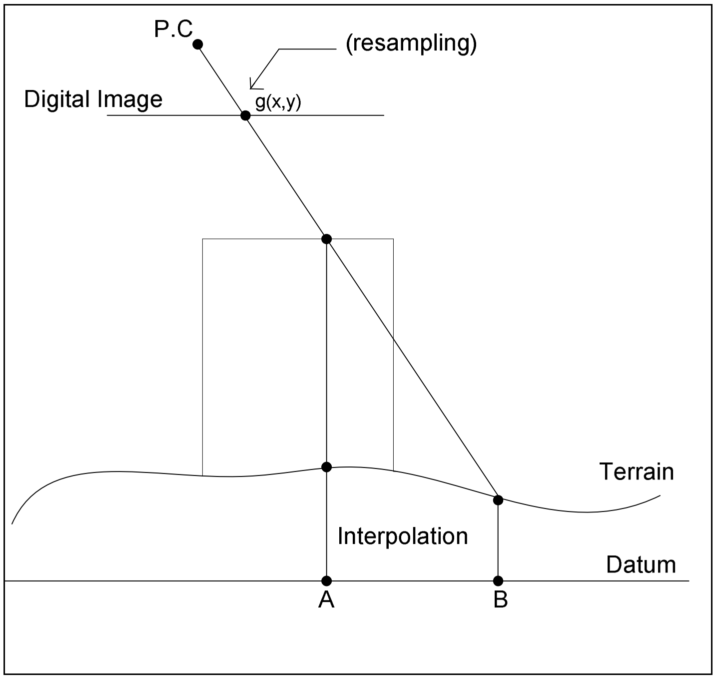


Figure 2.2: Illustration of the double mapping problem.

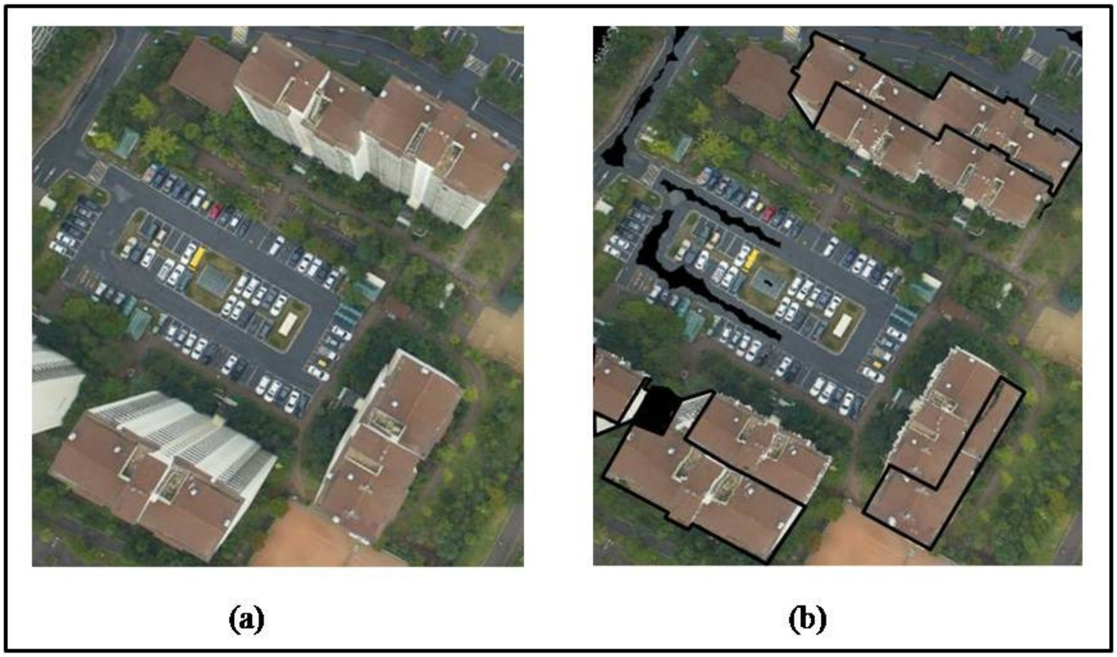


Figure 2.3: Perspective image (a) and corresponding orthophoto, with double mapped areas outlined in black (b).

Differential rectification is therefore not well suited for orthophoto generation with large scale imagery over urban areas (Skarlatos, 1999), and the following section outlines alternative methodologies more suited for urban environments.

2.1.2 True Orthophoto Generation

To avoid having ghost images in the output orthophotos, some techniques have been developed to produce true-orthophotos, where ghost images are removed through visibility analysis and occlusion detection. A majority of the true-orthophoto generation techniques are based on the Z-buffer method (Bang et al., 2007; Habib et al., 2007a; Rau, 2002). As discussed in the previous section, double-mapping occurs when two object space points are competing for the same image location. The Z-buffer method proposed in Amhar et al. (1998) addresses the situation of two (or more) object space points competing for the same image location, by considering the distances between the perspective center and each competing object space point. The point that is closest to the perspective center is considered to be a visible point, while any other competing points are deemed invisible in the image. The implementation of the Z-buffer method utilizes three 2D arrays of the same dimension as the input image (these will be referred to as Z-buffer image arrays), and one visibility map of the same dimension as the input DSM. The Z-buffer image arrays and the visibility map are shown in Figure 2.4. When a DSM cell is projected onto a perspective image, for example at an image location (x,y), the X-coordinate of the DSM cell is stored in the (x,y) location of one of the 2D image arrays (labelled “X” in Figure 2.4), while the Y-coordinate of the DSM cell is stored in the (x,y) location of a second 2D image array (labelled “Y” in Figure 2.4). The third image array, labelled “Distance”, is used to store the distance between the perspective center and the DSM cell. A visibility map of the same dimension as the input DSM is also established and used to store information about the visibility of each DSM cell. For example, looking at Figure 2.4, if the DSM cell A is first considered, the X and Y coordinates of this cell, in addition to the distance from the cell to the image perspective center (call this *dA*) are stored in the image arrays. Furthermore, the cell is marked *visible* in the visibility map. When the DSM cell B is visited, the distance from cell B to the perspective center will be computed (*dB*) and found to be smaller than the distance *dA*. In this case, the X and Y coordinates of cell B, as well as distance *dB*, will now be stored in the image arrays (replacing the previous values obtained from cell A). The cell corresponding to B in the visibility map will be marked *visible*, while the cell corresponding to A in the visibility map will be changed to *invisible*. After all DSM cells have been visited, the image arrays are used to project the image grey values onto the orthophoto plane. The visibility map is then used to locate occluded areas, which are represented by the cells labelled as *invisible* in the visibility map, and are marked as black cells in the output orthophoto. Figure 2.5 compares the orthophoto generation methods that have been outlined thus far, applied to real data. Figure 2.5a shows a building located in a perspective image, Figure 2.5b displays the resulting orthophoto after differential rectification is performed (notice the ghost image of the building), and Figure 2.5c displays the result from the Z-buffer method, where the majority of the ghost images have been replaced by black cells, as no colour information is available for the occluded areas.

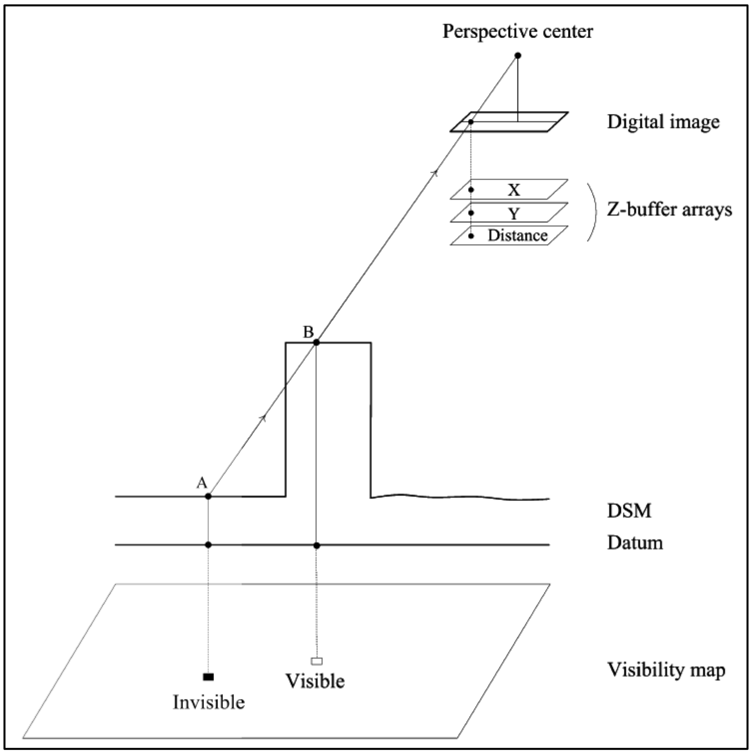


Figure 2.4: Z-buffer methodology, as depicted in Habib et al. (2007a).

Figure 2.5: A building in a perspective image (a), in an orthophoto produced using differential rectification (b), and in an orthophoto produced using the Z-buffer method (c).

Although the Z-buffer method is commonly used for the creation of true-orthophotos, the method has some limitations. One of the limitations is the sensitivity to the sampling interval of the DSM in relation to the ground sampling distance (GSD) of the image. For instance, if the DSM cell size is smaller than the image GSD, false occlusions in flat areas will be detected, as neighbouring DSM cells will be projected onto the same image pixel. An example of this case is depicted in Figure 2.6, for the neighbouring cells labelled A and B. In this case, the distance between the perspective center and cell B is smaller than A, and thus cell B will be labelled as visible (V) and cell A will be falsely labelled as invisible (I).

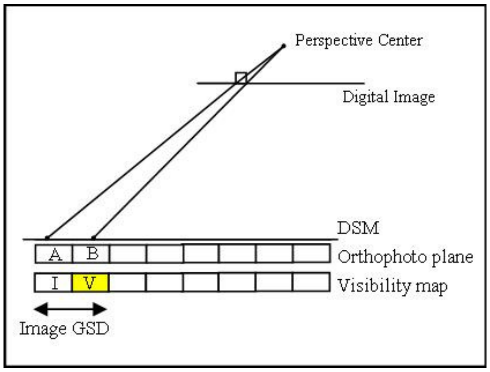


Figure 2.6: False Occlusions can occur when the DSM cell size is smaller than image GSD.

For the case shown in Figure 2.6, if the DSM cell size were made equivalent to the image GSD, these false occlusions would not occur. However, unlike Figure 2.6 where the terrain is flat, if tall buildings are present in the DSM, this selection of cell size will result in false visibilities. Figure 2.7 depicts such a situation, where the DSM cell size is chosen to be equivalent to the image GSD on the terrain. When cell A is visited, the projection of A onto the image will yield the pixel that is highlighted in red. As no other DSM cell has been projected onto the red image pixel, DSM cell A is recorded as *visible* in the visibility map. When DSM cell B is visited, the cell location is projected onto the blue pixel in the digital image. As no other DSM cell has yet been projected onto this image location, DSM cell B is also labelled as *visible.* Next, DSM cell C is projected onto the green pixel in the image. Once again, no other DSM cell has yet been projected onto this image location, thus DSM cell C is labelled as *visible*. DSM cell D is then projected onto the image, at the location of the purple image pixel. No other DSM cell has been projected to this image location, and thus cell D is also labelled as *visible*. When the next cell is visited, DSM cell E, this cell location is projected onto the blue image pixel. Notice, however, that DSM cell B was also projected onto this same image location. As the distance between the perspective center and DSM cell E is smaller then cell B, cell B is now labelled as *invisible,* while cell E is labelled *visible.* A similar situation is encountered when cell F is visited, and projected onto the same image location as cell D – the purple pixel. The distance between the perspective center and DSM cell F is smaller than cell D, and so cell D is labelled *invisible* and cell F is labelled *visible*. This process will continue until all DSM cells have been visited. Observe that through this process, DSM cell C will be labelled *visible*, even though it is in the building's occlusion. The reason this occurs will now be explained. The image GSD on top of the building in Figure 2.7 will be smaller than the image GSD on the terrain, since the building roof is closer to the image perspective center. This will result in the DSM cell size being greater than the image GSD on the roof top, and thus there will be a higher probability that some of the image pixels are not associated with all the DSM cells in the roof region. This is shown in Figure 2.7b, where cell E is projected onto the blue image pixel, and cell F is projected onto the purple image pixel. Note that no roof point is projected onto the green image pixel. Therefore, when the DSM cell C is projected onto the green pixel, it is *falsely* labelled as visible since it is the only cell that has been projected onto the green image pixel. Therefore this false visibility occurs as a result of having non-compatible GSD values between the terrain and building roofs (Habib et al., 2007a).

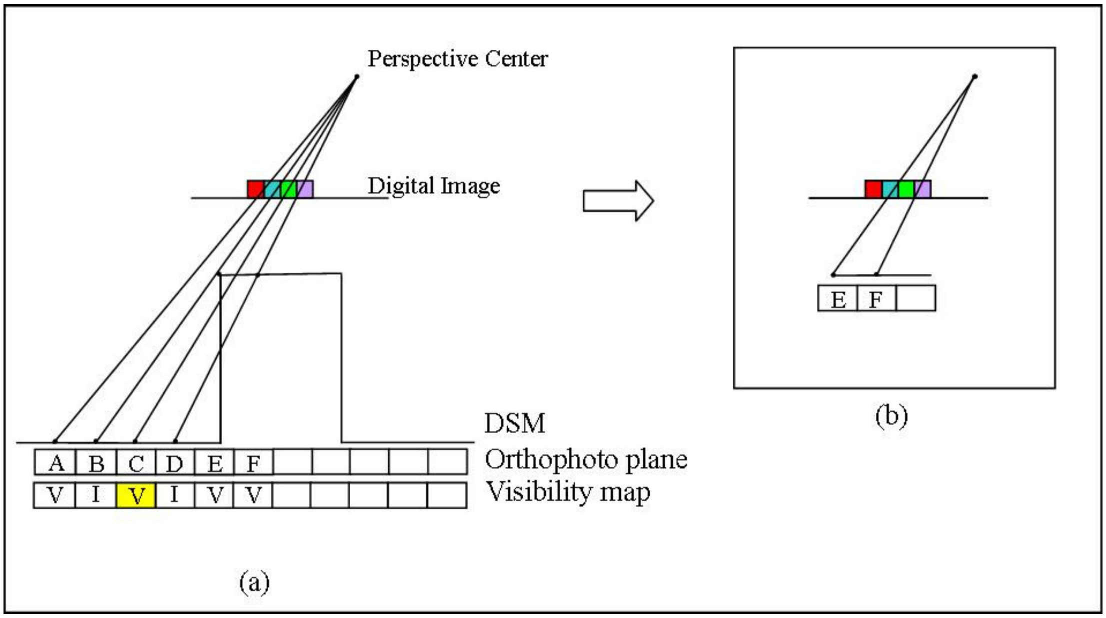


Figure 2.7: False visibilities in occluded areas due to incompatible GSD values at the terrain and rooftops.

Another limitation of the Z-Buffer method is commonly referred to as the M-portion problem which arises from narrow vertical structures. When narrow buildings are present in the DSM, the terrain points located in the occluded area directly beside the building can be falsely deemed visible (shows as a yellow cell in Figure 2.8). To overcome this limitation, a DBM is required such that pseudo groundels can be introduced along the building walls, thus avoiding the false visibilities caused by tall narrow structures. Figure 2.9 depicts the use of pseudo groundels (shown in red), and for detailed analysis on the pseudo groundels, refer to Bang et al. (2007).

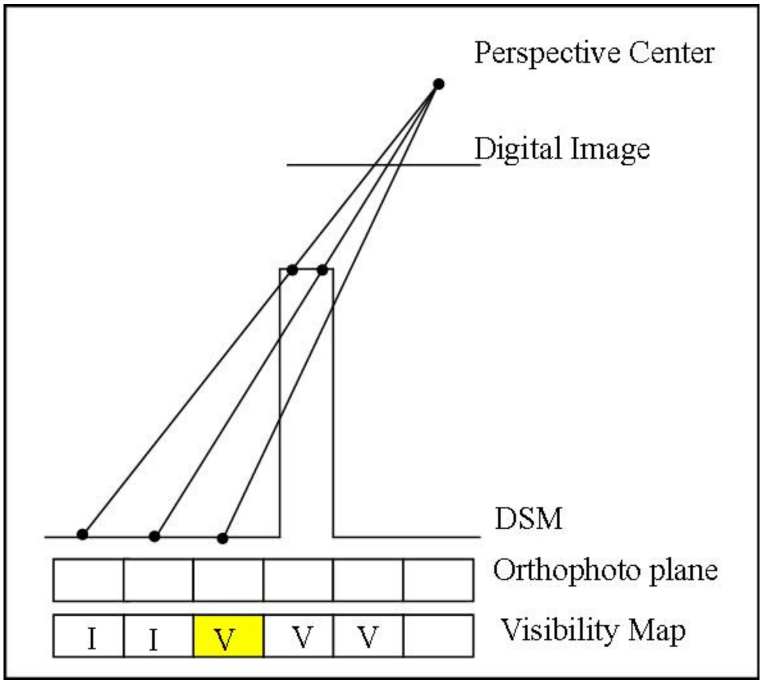


Figure 2.8: Depiction of false visibilities due to the presence of tall narrow structures.

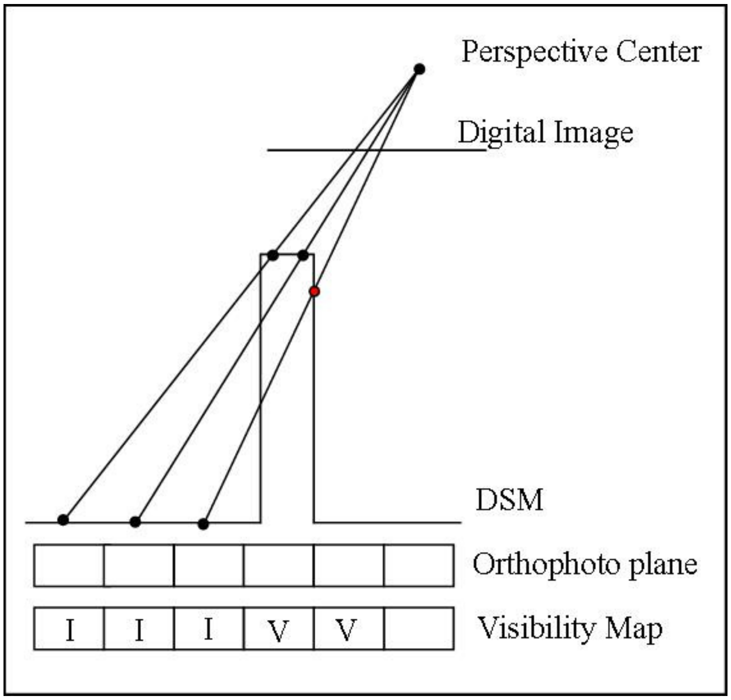


Figure 2.9: Use of pseudo groundels to solve the M-portion problem.

Kuzmin et al. (2004) commented on several limitations involved in image-based approaches for the creation of true-orthophotos: memory requirements; need for intermediate structures (DBM, groundels); processing speed; and dependency on resolution. Kuzmin et al. (2004) therefore proposed a new method that would be fast, accurate, and resolution independent, but that still requires the availability of a DBM. This resulted in the development of a polygon-based methodology for true-orthophoto generation. The first step in this approach is to generate an orthophoto through the conventional differential rectification procedure. The procedures following this are then aimed at locating the double mapped areas, and marking them black in the output orthophoto. A DBM is used to provide surface polygons. The polygons are then projected onto the image plane, and analysis of the overlap of the individual polygons is performed. The “top” polygons, i.e. the polygons closest to the perspective center, are kept in the orthophoto while all other polygons are deemed as hidden areas (see Kuzmin et al., 2004 for more details on this method).

Thus far, the outlined true-orthophoto generation procedures fall into the category of distance-based methods. Alternative methodologies for the production of true-orthophotos have been developed and tested in Habib et al. (2007a) where in contrast to distance-based methods, two angle-based methods for true-orthophoto generation are proposed. The methodologies are based on the effect of relief displacement in perspective imagery. Relief displacement, which is the extent of the projection of the top and bottom of a vertical structure onto an image, is the source of occlusions in imagery. This displacement takes place along radial directions from the image space nadir point (Mikhail, 2001). The angle-based methods address the detection of occlusions by sequentially checking the off-nadir angles (herein referred to as α) to the line of sight that connects the DSM points with the perspective center of the image. The methods proceed along a radial direction starting from the object space nadir point, where there is no relief displacement and thus this point is always visible in an image. As the procedure moves outwards from the nadir direction, the angle α will continue to increase in the absence of occluded areas. As the angle increases, each DSM point will be considered as visible in the image, for example, DSM cells *A, B, C* and *D* in Figure 2.10. An occlusion will be detected if the α angle decreases while proceeding away from the nadir point, along a radial direction. Therefore, according to Figure 2.10, the DSM point *E* will be determined as an occluded point, as the angle αE is smaller than the angle of the previous visible DSM point, angle αD. The next visible point will occur only when the α angle becomes greater than that of the last visible point, which is shown as DSM cell *F* for the situation depicted in Figure 2.10.

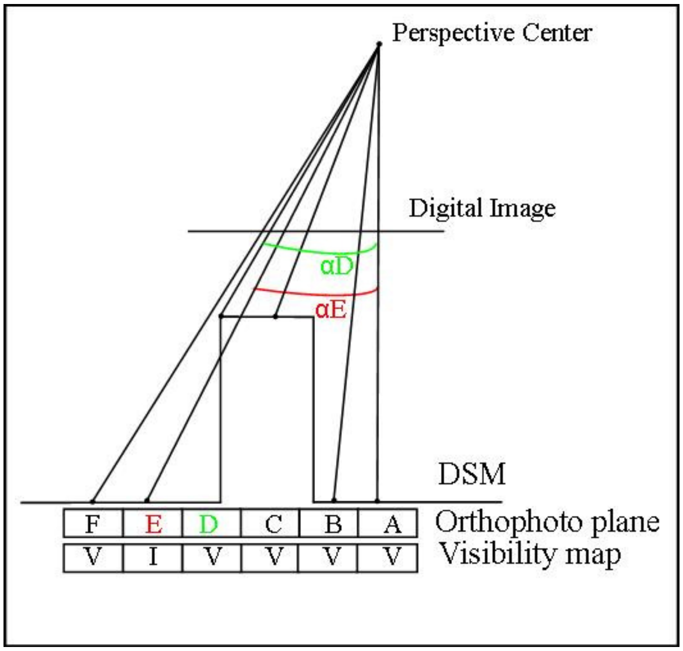


Figure 2.10: Concept of angle-based true-orthophoto generation.

The advantages of using an angle-based methodology for true-orthophoto generation are that the performance does not require a DBM of the area, and is independent of the DSM cell size and image GSD. For detailed descriptions of the two angle-based procedures, namely Spiral Sweep and Adaptive Radial Sweep, please refer to Habib et al. (2007a). Sample orthophotos produced using some of the true-orthophoto generation techniques discussed in this section are shown in Figure 2.11, as reported in Habib et al. (2008c). From this figure, it can be observed that, in comparison to the common Z-buffer method (Figure 2.11b), the two angle-based methods (Figures 2.11c and 2.11d) have detected occlusion areas more completely than the former.

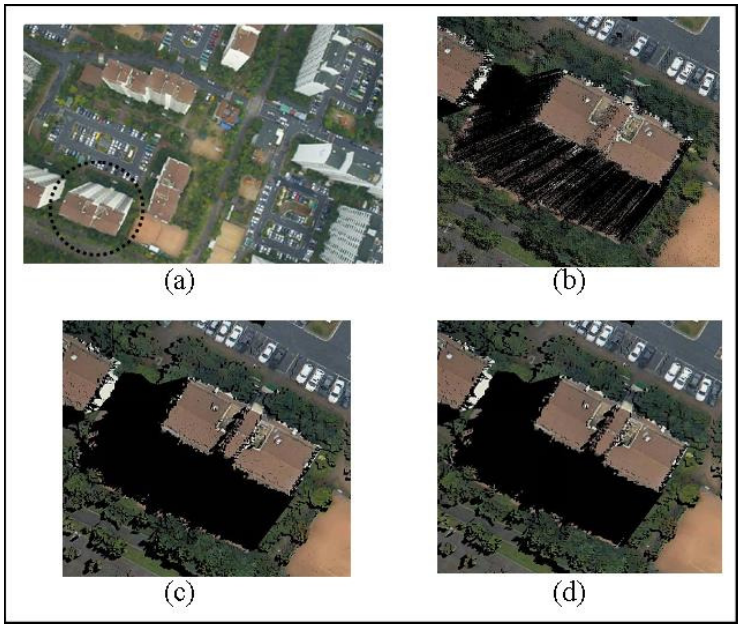


Figure 2.11: Original perspective image (a), true-orthophotos generated using Z-Buffer method (b), spiral sweep method (c), and adaptive radial sweep method (d).

The production of true-orthophotos using the above-mentioned angle-based techniques offers a significant improvement over the traditional orthophoto generation procedures. However, the quality of the true-orthophotos will always be depended upon the quality of the DSM. For instance, when LiDAR data is used, interpolation of the irregular data is performed to obtain a raster grid that can be used as the DSM in the production of true ortho-images. The nearest neighbour interpolation method is often used, in particular for urban areas, to avoid blurring any edge information contained within the LiDAR point cloud. Nonetheless, interpolation errors and the point spacing of the LiDAR footprints can result in jagged building boundaries in the produced ortho-imagery. Figure 2.12 illustrates the effect of these problems. In Figure 2.12a the small circles represent the LiDAR footprints, where the roof points are shown in black and the ground points in light green. When the data is interpolated to raster grid, each cell is assigned the elevation value of the nearest LiDAR point. From this figure it is seen that the LiDAR footprints do not fully match the true building boundary (shown as the brown outline), and the results from interpolation cause further deviation to the boundaries. Figure 2.12b shows an orthophoto produced using a LiDAR derived DSM. To avoid the jagged building boundaries shown in Figure 2.12b, it is therefore important to obtain an accurate and complete DSM of the area.

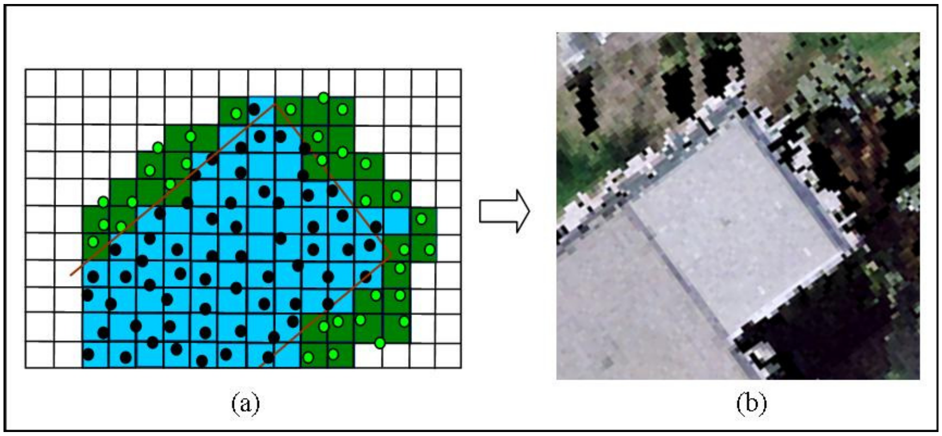


Figure 2.12: Effect of interpolation errors and LiDAR point spacing on the DSM (a) and true-orthophoto (b).

2.2 DSM Generation

A DSM of an area should ideally contain all object-space components, from the ground and vegetation to buildings of all sizes. Objects such as trees and vegetation, however, are often not included in a DSM, and thus it is common for a DSM to be composed of only the ground and buildings (Nielsen, 2004). Traditionally, a DSM of an area of interest has been obtained through photogrammetric procedures, where a given surface point must be identified in at least two images in order to obtain 3D coordinates of the given point. The determination of conjugate points can, however, involve complicated matching procedures (Toutin, 2004), and ground control points (GCPs) are often required which can involve a significant amount of field work. Furthermore, Mayunga et al. (2005) states that DSMs that are generated through the process of image matching are not sufficient due to poor ground sampling as well as matching errors, which can be caused by various factors such as poor image quality, occlusions, and shadows. This section will review previous methods utilized for DSM generation from imagery (satellite and aerial), LiDAR data, and IFSAR, while outlining the benefits and limitations of each approach.

Toutin (2004) analyzed the capabilities of Quickbird satellite imagery, which provide a resolution of about 0.61m. It was found that the accuracy of the DEM produced from Quickbird imagery was highly dependent upon the land types present in the area under investigation. For example, areas containing bare ground provided accuracies below 1.3m, while areas containing natural as well as man-made surfaces provided accuracies ranging from 3.4-6.7m. As urban areas comprise numerous man-made surfaces, it can be inferred that the use of photogrammetric procedures in satellite imagery will not yield an accurate and complete DSM of an urban environment. In other research, aerial imagery has been utilized to obtain elevation information. In Habib et al. (2008a), experiments were performed with frame analogue imagery, collected at a flying height of 770m. The photos were digitally scanned at a resolution of 12 microns to obtain a 6cm GSD. A root mean squared error (RMSE) analysis showed that manual measurements in the aerial imagery could achieve vertical accuracy in the range of 30cm, and planimetric accuracy of around 14cm. Aerial imagery was also utilized for DSM generation in Baltsavias et al. (1995). Large scale imagery scanned at a resolution of 15 microns was used in an automated DSM generation program, which utilizes stereo correlation to automatically obtain a DSM from perspective imagery. The image data provided four-way image overlap, and the final DSM was obtained by merging six separate DSMs that were obtained through the "pairwise matching of images." It was found that the obtained DSM was "unsatisfactory" in terms of the modeling of buildings. It was concluded that poor modelling of discontinuities in the DSM (for individual buildings, as well as several buildings close to one another) was the result of an insufficient density of measurements. In addition, numerous matching errors were found to occur near buildings due to the presence of shadows, occlusions, and trees. Thus, although the use of aerial imagery can allow for improved determination of elevation information in comparison to satellite imagery, the following paragraphs will outline alternatives that offer yet improved vertical accuracies.

A different option for DSM generation is through the use of LiDAR data. LiDAR directly determines 3D coordinates of each point it measures, and is more reliable in terms of obtaining surface data in comparison to the traditional stereo photogrammetric techniques (Weidner and Förstner, 1995). LiDAR provides a vertical accuracy between 5-20cm (Optech, 2008), and the LiDAR data delivery can be much faster, as the data is acquired in digital form and the system directly obtains range measurements (Baltsavias, 1999). Recent trends show that LiDAR systems are more frequently used to obtain a dense 3D point cloud representation of the object space surface. In Baltsavias (1999), a good comparison of LiDAR and photogrammetric data is presented. Each data source is described as possessing different advantages and disadvantages, depending on the desired product. However, in terms of DSM generation for urban environments, it was concluded that airborne laser scanning was the best option. The factors contributing to this conclusion were the density and accuracy of the data collection; this allows for an easier detection and reconstruction of 3D objects (in particular buildings) from a LiDAR-derived DSM than from DSMs that are obtained through an image matching procedure. For many applications, the use of LiDAR data in its irregular format can be time-consuming, and thus in many cases the irregular data is converted into a grid format. When using LiDAR data to obtain a DSM, the DSM cell size should be chosen such that it is small enough to avoid data loss during the resampling process. On the other hand, the cell size should be large enough to avoid redundancy in the derived DSM. It can thus be a good option to select a cell size equivalent to the average point spacing of the LiDAR data. In terms of the resampling technique, the nearest neighbour interpolation method is a good option, as explained in section 2.1.2. The irregular nature of the LiDAR point cloud leads to a degraded quality of building boundary representation in the DSM, and furthermore the irregular LiDAR point cloud can produce scissoring effects around building boundaries in the produced orthophotos (Figure 2.13). In Gamaba and Houshmand (2000), the use of interferometric radar (IFSAR) is discussed in terms of DSM generation capabilities. This data source was found, however, to be unsuitable for use in urban environments due to the "the insufficient spatial resolution, multiple scattering due to the building geometries, and layover effects, in addition to the intrinsic IFSAR system level noise."

Figure 2.13: Scissoring effects in orthophoto due to original LiDAR-derived DSM.

Regardless of the chosen data source for DSM generation, abrupt changes in height can have an adverse effect on the quality of the DSM. Therefore, numerous research activities have been aimed at devising procedures for the refinement of a DSM, in particular through the addition of a digital building model (DBM). A review of research on DBM generation is thus investigated in the next section.

2.3 DBM Generation

To address the issues outlined in the previous section, a DBM can be created and added to a DTM to produce a crisp DSM that represents the buildings and terrain of an urban environment. Although the surface model of an urban area contains features such as trees, cars, lampposts, etc., it would be difficult to refine these objects in the DSM. For this reason, when a DBM is created it is added to a DTM, as opposed to the original DSM. Figure 2.14a shows a perspective image of a parking lot containing trees and lampposts. A true-orthophoto generated using the original DSM for this area is shown in Figure 2.14b. It is seen in the image that the effect of the irregular LiDAR point cloud causes the trees and lampposts in the produced orthophotos to contain occlusions around these features. Although these occlusions can be filled by generating a mosaic of several orthophotos (Figure 2.14c), the trees will still appear degraded in quality. Therefore, when the trees are not considered in the surface model through the use of the DTM, the trees will simply be projected onto the ground, and will be more visually appealing in the 3D model (Figure 2.14d). Therefore, the refined DSM is obtained by adding the DBM to a DTM. Once the improved DSM is obtained, it can be used to produce true-orthophotos with improved building boundaries. Figure 2.15a shows an initial DSM, which can be compared to the refined DSM in Figure 2.15b. Figures 2.15c and d show the produced orthophoto using both the initial and refined DSM, respectively. The improved orthophoto obtained in Figure 2.15d can then be draped over the modified DSM to obtain a realistic 3D city model.

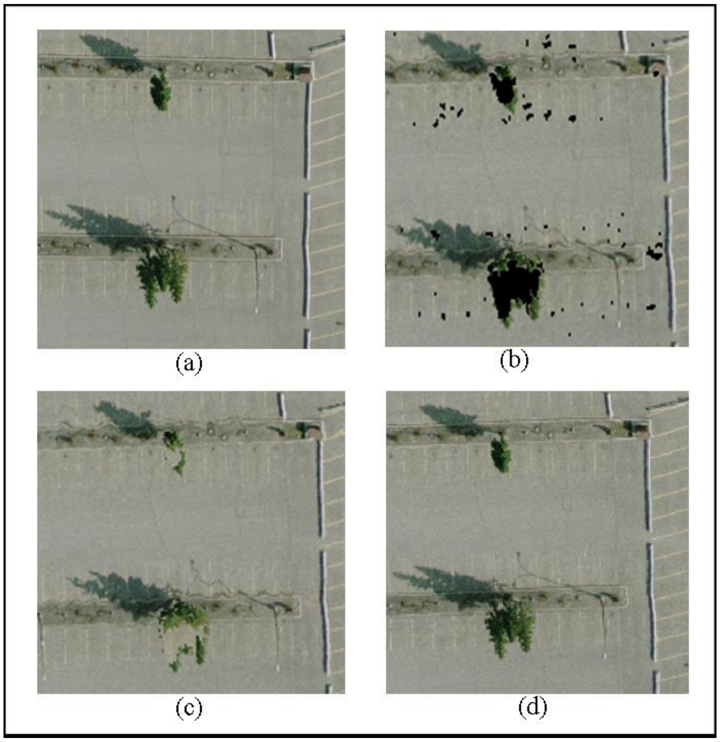


Figure 2.14: A perspective image of a parking lot containing trees, shrubs, and lampposts (a), a true-orthophoto of the same area using a DSM (b), the true-orthophoto after occlusion filling (c), and a true-orthophoto generated using a DTM (d).

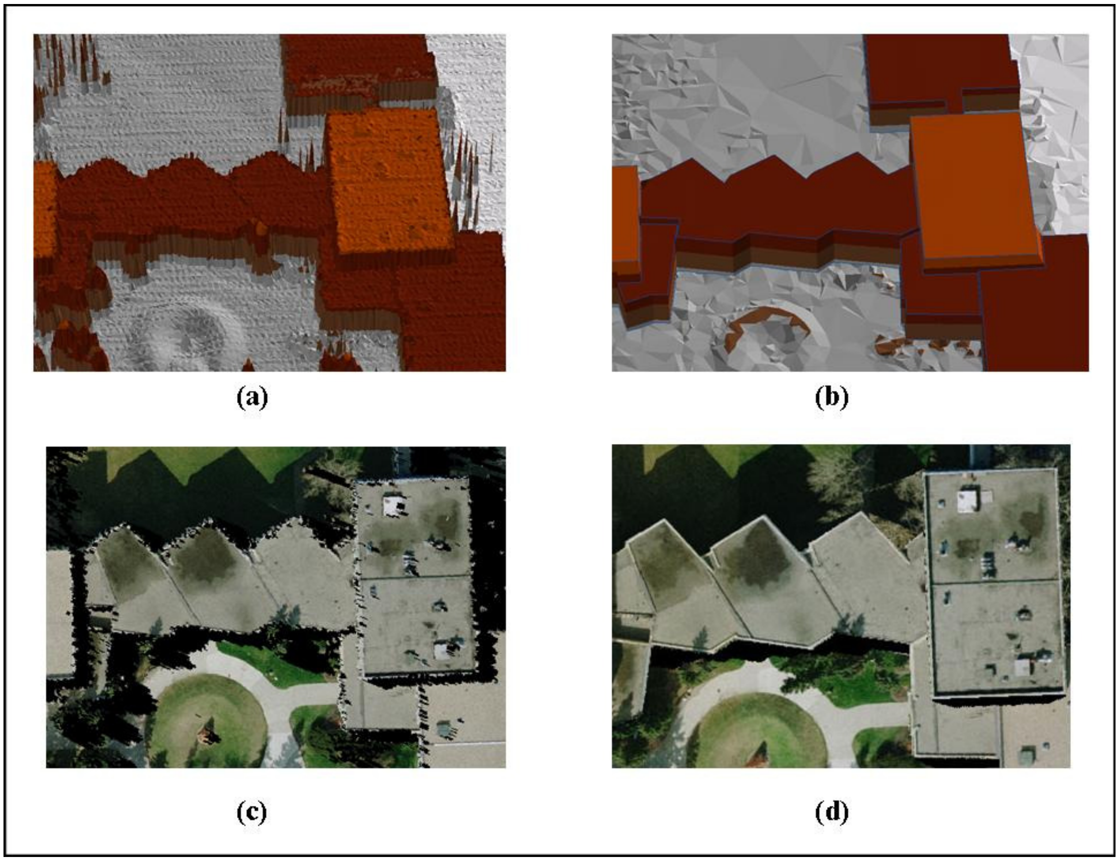


Figure 2.15: A comparison of an initial DSM (a) and a refined DSM (b), as well as initial true-orthophoto (c) and improved true-orthophoto (d).

The DBM generation process can be separated into two steps; building detection and building reconstruction (Ma, 2004). Building detection is a process in which buildings are differentiated from all other objects within the data, to obtain building hypotheses. The term building hypothesis is used to describe regions in the data that are hypothesized as building locations. Building reconstruction utilizes the building hypothesis to derive building model parameters (such as height, width, etc.) that are required to obtain a 3D building model. DBM generation strategies are categorized into two groups in Chen et al. (2008). The first category is commonly called the model-driven approach, and the second category of DBM generation strategies is the data-driven approach. The model-driven approach is referred to as a *top-down* method since the process begins with a set of pre-defined types for possible building-primitives. Information derived from the data is then investigated to determine which 3D model primitives best fit the different areas of the dataset. Sample building primitives utilized in the work of Halaa et al. (1998) are displayed in Figure 2.16. The model-driven approach is also referred to in some work as constructive solid geometry (Halaa et al., 1998). Since the approach uses a finite set of possible building primitives, it is easy to implement. On the other hand, the approach cannot model all building types. The building types that can be successfully modeled using the model-driven approach tend to be simple buildings with flat or gable rooftops (Ma, 2004). The data-driven approach for DBM generation deals with point features (such as building corners), linear features (building boundaries), and planar features (building rooftops) which are extracted from the input data. This approach is referred to as a *bottom-up* method since features are initially extracted from the point cloud, after which the features are used to construct the DBM. This approach has the advantage that it can handle generic models without making assumptions regarding the shape of the buildings contained within the data. As such, this approach can theoretically handle any building type, although on the down side, this approach involves complex implementation (Chen, 2008; Ma, 2004).

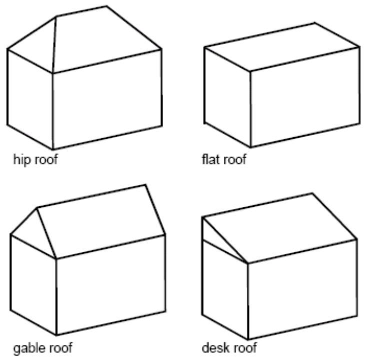


Figure 2.16: Building primitives used in the model-driven approach in Halaa et al. (1998).

This section will highlight some of the previous methods employed for DBM generation, through the use of various data sources. In some of the previous literature on DBM generation, a single data source has been utilized. This section will review the use of imagery alone for DBM generation, as well as the single use of LiDAR data. The use of image data for DBM generation has the advantage that building boundaries are accurately defined in imagery. In addition, automated edge detection can be performed on the involved imagery to aid in the automatic detection of building boundaries. In some cases, however, difficulties may be encountered when information is not available due to occlusions and shadows (Baillard, 1999). Furthermore, it may be difficult to clearly locate building boundaries when there is weak contrast between the building roofs and the surrounding area (Chen et al., 2008). LiDAR data has also been used as a single data source for DBM generation. LiDAR provides important shape information regarding the building roofs (flat, slanting, etc.) as well as accurate height information. In addition, Priestnal et al. (2000) comments on several other benefits obtained from the use of LiDAR for the creation of a DSM: offers an inexpensive method for the production of accurate DSMs; can be obtained under various weather conditions (day and night); and offers dense point data over surface features. Wang et al. (2008) note that the accuracy of buildings obtained from LiDAR alone suffer due to point spacing, scanning angle, etc. To overcome some of the limitations inherent with the use of a single data source, many authors have proposed DBM generation methodologies that utilize the integration of multiple data sources, including the integration of: LiDAR and vector maps; imagery and vector maps; and LiDAR and photogrammetric data. When LiDAR and vector maps are integrated, the vector maps can be used to provide building boundary locations, thus reducing any effort involved in building boundary detection. Vector maps have also been integrated with aerial imagery to improve on the performance obtained using image data alone. The vector maps can be used to aid in determining the possible locations of building boundaries within the imagery, and therefore aids in the detection of building boundaries. The use of vector maps with either LiDAR or imagery improves the automation involved in generating the DBM, however these method assumes that the vector maps are error-free (i.e., provide accurate boundaries) and up to date (Chen et al., 2008). Furthermore, it is assumed that the roof borders match with the ground plans and thus that there are no roof overhangs (Halaa et al., 1998). In other research, photogrammetric and LiDAR data are integrated. There is strong evidence that these data sources compliment each other (Baltsavias, 1999), and their integration can produce accurate and reliable 3D building models. When these sources are integrated, a common approach is to extract information from the LiDAR data to obtain a building hypothesis, after which the image data is utilized to refine the building boundaries (Wang et al., 2008). Figure 2.17, modified from Habib et al. (2008c), gives a flow chart of the different data sources that will be reviewed in this section, in the context of DBM generation.

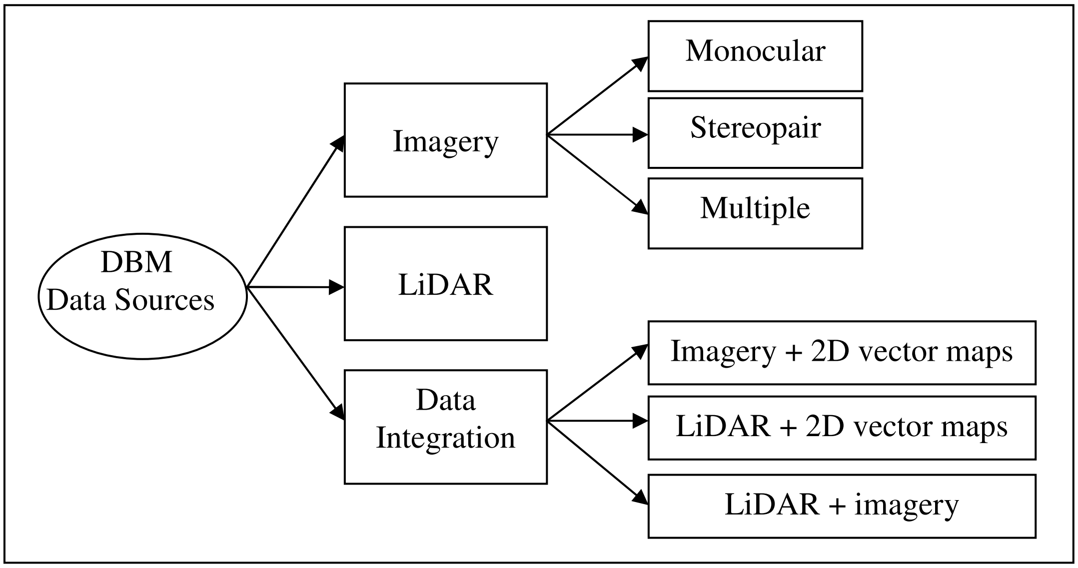


Figure 2.17: Potential data sources for DBM generation (modified after Habib et al., 2008c)

2.3.1 DBM from Imagery

The use of imagery for DBM generation has been utilized by numerous authors in previous research, and can be categorized as: use of monocular imagery (single image); stereopairs; and multiple images. Various forms of imagery, such as aerial and satellite, have been employed for DBM generation purposes over the past several years, and this section gives a brief review of some previous methods that utilize image data. The extraction of buildings from monocular imagery has been a topic of research for the last twenty years. Common techniques employed in DBM generation from monocular imagery include region growing and shadow analysis. Shufelt (1999) performed a comparative analysis of several building extraction systems that utilize monocular imagery. The reviewed systems include: 1) BUILD+SHAVE; 2) VHBUILD; and 3) PIVOT. The BUILD+SHAVE building extraction system is a hybrid system that combines two separate programs, namely BUILD and SHAVE. When BUILD+SHAVE is used for DBM generation, several assumptions are made: a vertical image is used; perspective effects are negligible and thus ignored; all buildings are roughly rectangular in shape; and the solar azimuth angle is available. The BUILD program is used first, to obtain the building boundaries in image space. The program detects linear features in the input vertical image. Any lines that meet at right angles are intersected to obtain corners, after which the corners are used to obtain rectangular buildings. If some corners of a building are missing, the program uses symmetry to complete the building. The solar azimuth angle is then used to determine the sides of each box that are expected to cast shadows. If shadows are found in the expected areas, the building is included in the final 2D building hypothesis. The SHAVE program then computes the average length of the shadow for a given building, as well as the GSD at the center of the building. The GSD is used to obtain the length of the shadow in the object space, which is used in conjunction with the solar azimuth angle to obtain the height of a building in the object space. The second building extraction system reviewed by Shufelt (1999) is VHBUILD, which improves on the previous system as it no longer assumes nadir acquisition geometry. Furthermore, buildings with flat rooftops as well as gable rooftops can be modelled by this system. The system utilizes the same shadow analysis as BUILD+SHAVE to obtain a building hypothesis, after which the height information is obtained by using a “vertical edge finder” on the buildings in the perspective imagery. Experimental results for the BUILD+SHAVE and the VHBUILD building extraction systems have shown a high sensitivity to the image type. For instance, they have shown poor performance in images that contain numerous and complex buildings. The PIVOT building extraction system uses a vanishing point detection algorithm to detect building corners, which are then used to form rectangular and triangular 3D primitives that are subsequently connected to form 3D buildings. The height attribute in this method is computed using a combination of shadow analysis and vertical line detection. In comparison to the previous methods, the PIVOT system produced the best results, largely in part due to the nature of the height computation of this system (i.e., the use of both shadows and vertical line detection of building walls visible in the perspective imagery). The three building extraction systems BUILD+SHAVE, VHBUILD, and PIVOT are classified as model-driven methods, as information obtained from the imagery is used to form a fixed model type; rectangular buildings with either flat or gable roofs. That is, the process uses a set of pre-defined types for possible building primitives.

Single imagery was also utilized in Nevatia et al. (1997), in which a system was developed for the extraction of buildings. In their work, it is noted that the use of single imagery for DBM generation can be more difficult in contrast to other input data types, such as stereo images and range images, as 3D information cannot be directly obtained from single imagery. The system developed in Nevatia et al. (1997) is restricted to parallelogram and rectangular shaped buildings with flat roofs, and thus is not suited for more complex building models. Since this method assumes these pre-defined types of building primitives, it is model-driven. The method begins with the detection of linear edges in imagery, after which parallelogram hypotheses are created. The validity of the hypotheses is then assessed by investigating evidence of nearby walls and shadows, which are also used to obtain elevation information, in a similar fashion as outlined in Shufelt (1999). It is shown by experimental results that in some cases buildings that are partly occluded by trees or have low heights are not detected through the use of this system.

Although single imagery can be used to detect certain types of buildings under specified conditions, the use of multiple images can produce a more robust system with more complete results. Multiple images can be used in several different ways. They can be analyzed as a stereo-pair, or the imagery can be analyzed individually after which the individual results are combined. The use of multiple images can be very advantageous, as some buildings not visible in a first image may be visible in a second (e.g., due to different view angle). When stereo imagery is used for building extraction, image matching techniques are commonly used. The matching techniques, however, can have several limitations such as failing to match in the presence of occlusions and shadows (Wang et al., 2008). In addition to investigating the use of single imagery for DBM generation, Nevatia et al. (1997) also investigated the use of multiple images, where hypotheses for buildings obtained from one image are projected into another image. The proposed method is restricted to rectilinear shaped buildings with flat roofs. Since the method assumes a pre-defined type of building-primitive, it is categorized as a model-driven approach. The use of multiple images was also employed in the work of Baillard (1999). In his work, line-matching is performed where epipolar geometry is used to asses endpoint correspondence between tentatively matched lines, after which cross-correlation is used to check the similarity of the line’s neighbourhoods. Once lines from two images are matched, they are then projected into a third view to confirm the matched lines. Subsequent steps use the detected lines to obtain roof planes. As this method deals with the extraction of features (in this case lines) from the input data, it is a data-driven approach. Moreover, it is a polyhedral data-driven approach, which is a sub model of the data-driven model. Polyhedral models assume that the buildings are composed of planar surfaces, bounded by straight lines. The roof planes and the final building reconstruction obtained using this approach is contingent on the accuracy and number of detected lines. Elaksher et al. (2003) also believe that to obtain accurate and complete building descriptions, multiple images are required. Moreover, they state that the use of multiple images will compensate for cases when a building is partially obscured in more than one image. In their work, four images are used for building wire-frame extraction. Their approach begins with a segmentation procedure, in which regions in the imagery are deemed as roof and non-roof areas, based on size, shape, and intensity values. Since no assumptions are made regarding the building shape, this is a data-driven method, and the procedure requires both EOP and IOP of the involved imagery and utilized camera, respectively. Corresponding roof regions in overlapping imagery are located, and lines from these areas are matched (i.e. this research targets polyhedral models). In this work, it was found that the average coordinate accuracy that could be achieved was around 0.9m in horizontal accuracy and 0.6m in vertical accuracy, using aerial imagery with a scale of 1:4000, scanned at 30 μm with a 12cm GSD. For six tested buildings, the planimetric root mean squared error (RMSE) was computed for each of the derived buildings. It was found that the planimetric RMSE of the six buildings ranged from 0.4m-

1.5m. The vertical RMSE was also computed for each individual building, and the results ranged between 0.2m-1.0m. In Fraser et al. (2002), a data-driven method using stereo IKONOS satellite imagery was used to extract building models. 3D building information was obtained by matching conjugate points in overlapping imagery. GPS surveyed roof corners were used to evaluate the accuracy of the DBM extraction, and it was found that a planimetric accuracy of around 0.7m and vertical accuracy of about 0.9m could be achieved. It was further concluded that about 15% of buildings were not detected, including both small and large buildings. In addition, some buildings were generalised into simpler forms. Some reasons provided for these limitations were the presence of shadows and occlusions. Due to these limitations, as well as complex matching algorithms involved when using imagery alone for DBM generation, other authors have sought out alternative methods utilizing other data sources to detect and extract buildings.

2.3.2 DBM from LiDAR

In other research work, LiDAR data is used for the creation of a DBM. Some of the advantages that can be gained through the use of laser scanning data as opposed to image data are that factors such as variations in surface reflectance, as well as costly matching techniques which can be error prone, do not have to be considered. A common approach to DBM generation from LiDAR data is to segment planar roof patches. Although this method can be quite successful in the determination of roof planes, the roof boundaries can often be difficult to accurately define using the LiDAR data alone. In Maas and Vosselman (1999), the irregularly distributed LiDAR point cloud is used to segment the roof planes by grouping the laser scanner footprints in a 3D cluster (or parameter) space, based on two slope parameters and one distance parameter. Each LiDAR footprint defines a plane in the cluster space. The algorithm identifies the roof planes based on the number of planes that intersect in a bin in the cluster space. That is, bins containing a large number of points are identified as the planes of the roof. All points located on the same plane will be clustered and segmented as a planar roof patch. To find the crest lines of gable roofs, all pairs of segmented roof planes are intersected. In addition, straight line roof boundaries are obtained using the Douglas-Peucker algorithm, followed by an approximation algorithm that forces the lines to be parallel or perpendicular to the main orientation of the building. To obtain the building walls, the height of the walls are obtained by intersecting vertical planes with the roof edges, while the ground level of the walls is obtained from the height of the lowest LiDAR footprint located near the building. At intersections of four or more roof planes, a common node is computed by averaging the individual corner points. The results of this method were promising, and complex roof structures were reconstructed, however, no ground truth was available to asses the accuracy of this polyhedral data-driven model (Figure 2.18).

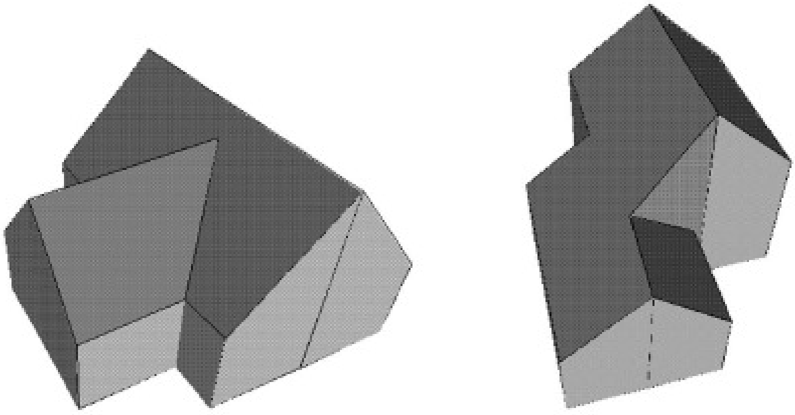


Figure 2.18: Complex buildings modeled in Maas and Vosselman (1999).

In Maas (1999) building detection and modelling was performed through the use of texture measures and height information, in a data-driven approach. In the first step, the LiDAR data is interpolated into a raster grid. In this work, the term texture measure is used to refer to local variations in height. It was hypothesized that roads and building rooftops would yield low height variations, while trees would produce large local variations. Some of the utilized texture measures include: height of data points; differences between the maximum and minimum height values within a given window; the results of applying a Laplacian and Sobel filter; and local slope information. A selection or all of the texture measures can then be used as the input to a supervised classification. Results showed that the use of texture information can be quite valuable for DBM detection and reconstruction, although some post-processing will be required. The reason for this is that some misclassified points appear as noise within objects, as well as some of the building edges are misclassified as vegetation. The DBM generation procedures reviewed up to this point involve the use of a single data source. The remainder of this chapter will review methods that utilize multiple data sources, in the efforts to increase the automation and accuracy of DBM generation.

2.3.3 DBM from Data Integration

The literature reviewed up to this point involves the single use of a data source. Although in some cases this can produce sufficient results, the single use of either LiDAR or imagery have several limitations. The single use of LiDAR data suffers from inaccurate building boundary definition due to the irregular nature of a LiDAR point cloud. When image data is used on its own, complex matching problems are encountered.

Many researchers have found that more complete and accurate building models can be obtained through the integration of different data: LiDAR, aerial imagery, satellite imagery, vector maps, etc. This section will investigate the use of data source integration for DBM generation.

LiDAR and 2D map data are integrated in the work performed by Halaa et al. (1998). As ground plans are often readily available within a GIS, these data sources were chosen in their work. A model-driven (or CSG) approach is utilized in their work, where buildings are created by a combination of one or more building primitives. In particular, the ground plan is split into small rectangular areas, after which the DSM data is used to determine the best fitting building primitive for each area, which is accomplished by segmenting the DSM into planar surfaces. As this is a model-driven approach, a limitation of the method is that not all building types can be accurately modelled. Therefore this method is more suited to environments containing simple building structures. Furthermore, the authors comment that their method will produce inaccurate building models if the roof surface contains any building part (such as a bay), that is not included in the ground plan. Chen et al. (2008) investigated DBM generation through the integration of LiDAR point clouds with large scale 2D vector maps. The reasoning given for the integration of these two data sources is that the vector maps can be used to provide accurate boundaries for the buildings, while the LiDAR data will be used to obtain dense information regarding the roof shape. An advantage gained through the use of the vector map is that it reduces any work aimed at building detection, and thus also increases the automation level of the procedure. The data-driven method proposed consists of three main steps. The first step is data pre-processing. In this step, the vector map is used to obtain the building outline, after which a point-in-polygon procedure is employed to locate the LiDAR points that fall within a given building. The second step is roof analysis, in which the extraction of linear and planar roof features is performed. In this step, co-planar points are grouped in order to extract the different roof faces. In the third step, the final model is produced by regularizing the lines. The Split-Merge-Shape method is then applied to the generalized lines to form enclosed areas. The accuracy of the derived building model is assessed in several ways. The height consistency between the LiDAR data and the reconstructed building roofs is investigated. In addition, the planimetric accuracy of the building corners are compared to coordinates obtained from stereoscopic measurements, where an estimated horizontal accuracy of 0.3m was obtained from the manual measurements in overlapping images. It was found that the vertical RMSE is around 0.2m (where the accuracy of the LiDAR data for the system used is quoted as 0.15m, with an average point spacing of 0.8m). The computed horizontal RMSE was found to be no greater than 0.6m. Furthermore, it was found that around 80% of buildings in the dataset were correctly reconstructed, and the authors comment that the missing buildings could be reconstructed if a higher density of LiDAR data were available. This work notes that the reconstruction results are thus dependent on the point cloud density, and the vector maps were assumed error free. In addition, when integrating multiple data sources, registration problems can be an issue. For instance, when vector maps are obtained from existing GIS databases, information regarding the data acquisition and accuracy could be missing or inaccurately documented.

Section 2.3.1 reviewed the use of imagery for DBM generation. Suveg and Vossleman (2000), however, stress that the sole use of image data for automated building reconstruction will not be sufficient, in large part due to lost information caused by occlusions, areas of low contrast, poor perspective, etc. And thus, to overcome these drawbacks, image data can be combined with other sources of data. In their work, Suveg and Vossleman (2000) utilize a GIS map as a secondary source of data. The ground plans are projected onto the imagery to aid in determining the possible locations of building corners and edges. The Foerstner operator, which is designed to locate the intersection of lines (i.e. corners), edge endpoints, as well as the center of circular features, is employed in this work. In particular, the authors use the operator to detect building lines and corners, after which matching procedures are performed on detected lines and corner points in overlapping imagery to obtain elevation information. The ground plan of a building is then manually partitioned, and the line and corner primitives (box, wedge, or rectangular primitive) that fit best for a given partitioned area are produced. As building primitives are used in this work, this method is considered model-driven. Finally, the building primitives are merged to obtain a 3D building model. This section has thus far shown that the inclusion of ground plans derived from an existing GIS database, is employed to aid in the detection of building boundaries in either imagery or LiDAR data. There are, however, some limitations inherent in this approach: the methods assume the ground plans are accurate and up to date; in some cases it is assumed that the roof borders match exactly with the ground plans, and thus there is no roof over-hang (Halaa et al., 1998). Furthermore, registration problems can be an issue when integrating multiple data sources, as the data acquisition is often performed at different epochs, and the data is commonly processed by different operators.

Numerous previous research efforts have utilized LiDAR and photogrammetric data for DBM generation. Several authors have acknowledged that these two data sources supplement each other, and thus through their integration accurate and reliable buildings can be extracted (Wang et al., 2008). When photogrammetric data is used on its own, many systems require vertical images as the input data. In addition, the methods encounter difficulties due to the low contrast between roofs and wall, while contrast between roofs and the ground is often greater. This causes significant problems when walls are used to obtain the height information. In addition windows and doors can interfere in the boundary detection procedure, and sometimes the necessary shadows are not visible, in particular in urban areas due to occlusions from neighbouring structures. In addition, many methods assume buildings are rectilinear with flat roofs (Lin, 1998). Furthermore, the number of lines detected in an image can be immense. When using imagery alone, complex analysis must be performed to decide which detected lines should be considered in the process. This can require a grouping process, for example, where groups of lines are analyzed in terms of their proximity, etc. (Lin, 1998). Wang et al. (2008) comments that buildings obtained from LiDAR should be refined through the use of imagery. In particular, the LiDAR building boundaries can be refined by projecting the LiDAR boundaries onto vertical imagery, and investigating the difference between the projected boundaries and the extracted image building boundaries. Figures 2.19a and 2.19b show results from Wang et al. (2008). Figure 2.19a shows the building footprints as extracted from the LiDAR data, while Figure 2.19b shows the projected LiDAR footprint for a given building onto a vertical image. The authors comment that the differences between the boundaries from the projected LiDAR data and the boundaries in the imagery will differ as the buildings extracted from LiDAR data may not be very accurate due to the LiDAR point spacing, scanning angle, etc. The differences between the projected LiDAR lines and image extracted lines are found to be within a few pixels, and thus an affine transformation is used to refine the building boundaries obtained from LiDAR: the transformation parameters are estimated through the use of the computed distance between projected and extracted roof edges. Although an affine transformation is commonly used to solve registration problems, in this work the LiDAR buildings serve as initial building hypotheses, after which the LiDAR building boundaries are matched with the image boundaries to obtain buildings with improved accuracy. That is, the building boundaries obtained from LiDAR are aligned with the image building lines by estimating the transformation parameters (i.e. the scale in the x and y directions, the correction for non-orthogonality, rotation angle, and translation parameters). After the building boundaries are refined, a *best* oblique image for each building is selected and used to add texture to the building walls.

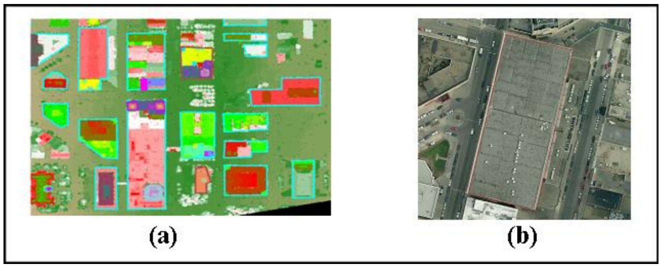

(a) (b)

Figure 2.19: LiDAR derived buildings (a), and projection of LiDAR boundary onto vertical imagery (b), according to methods proposed in Wang et al. (2008).

LiDAR and photogrammetric data were integrated in a GIS platform in Palmer and Shan (2002). The integration within a GIS allowed the use of existing powerful GIS functions and tools that are used to process the LiDAR data and visualize a 3D urban model through semi-automated procedures. The LiDAR is converted to a grid format, after which the grid is classified based on height only, to obtain building polygons. The classification results are vecotrized and regularized within the GIS, after which orthophotos are draped onto the 3D model. Although this method is straight-forward to implement and visualize due to the assistance of the GIS functions, the method does not yield an accurate 3D model that could be used for engineering purposes. Rottensteiner et al. (2004) also opted to integrate LiDAR and imagery for the extraction of buildings in urban areas. In their work, the LiDAR-derived DSM is first classified into ground and non-ground points, based on various criteria. Homogeneous segments are then located in overlapping imagery, through the use of image feature extraction techniques. The image planes are then matched with the corresponding DSM roof planes, to refine the initial segmentation of the DSM. Finally, building boundaries are then obtained by projecting linear features from imagery, which are located in the vicinity of the roof patches, onto the roof planes. Edge merging is performed, and finally closed building boundaries are produced. To evaluate the results of this data-driven method, building polygons were manually digitized in orthophotos, and used to determine the detected building completeness and correctness. The authors compute the building completeness as the ratio of the number of boundary segments that are correctly established to the total number of boundary segments in the area of interest. The term building correctness on the other hand is computed as the ratio of the number of boundary segments that are correctly established to the total number of boundary segments that are established. It was found that the algorithm was not successful in separating trees from buildings in some cases, and buildings smaller than 30m² were not detected. Chen et al. (2005) also utilize LiDAR and photogrammetric data in a data-driven approach, and their results aim at obtaining sub-meter accuracy for the buildings. The procedure first performs building detection, through region-based segmentation, followed by object-based classification. In the region-based segmentation, the LiDAR data is resampled into a raster grid and the pixels with similar attributes are merged. The attributes used for merging include the elevation information from LiDAR and the spectral information from imagery. After the pixels are grouped into regions, an object-based classification is performed to locate building regions (according to elevation, spectral information, texture roughness, and shape). In the building reconstruction stage, planar patches are extracted from the LiDAR data in the detected building regions. Initial building edges are then obtained from the 3D planes. These initial edges are used to predict the locations of the building boundaries in imagery. The Hough Transform is used to obtain image lines in the predicted regions, and the image lines are projected into object space using the height information of the 3D planes and the available exterior orientation parameters. Finally a split-merge-shape (SMS) method is applied, to yield final building boundaries. A 1/1000 scale topographic map with an image GSD of 0.1m was used as the source of ground control in their work, and the RMSE were computed. The values obtained from the RMSE analysis were 0.45m, 0.56m, and 0.70m in X, Y, and Z respectively. Through experimental results it was noted that buildings smaller than 35m² were not detected.

2.4 Summary

The literature reviewed in this chapter demonstrates the vast range of methodologies and data sources utilized by various researchers for orthophoto, DSM, and DBM generation. In Section 2.1, it was established that angle-based true-orthophoto procedures are required to obtain accurate ortho-imagery over urban environments. It was noted that the quality of the ortho-imagery is dependent on the quality of the DSM. Therefore, Section 2.2 outlined several methods used in previous research to obtain a DSM. It was found that a DSM obtained from either imagery or LiDAR did not accurately represent the surface of an urban environment, and thus a refinement of the DSM is required. It was concluded that this can be accomplished by adding a DBM to a DTM, and thus Section 2.3 investigated various sources of data that could be used for DBM generation. It was found that the sole use of LiDAR data was limited due to the irregularity of the LiDAR point cloud, and that the use of image data alone encountered difficulties due to complex image matching techniques, further hampered by occlusions and shadows contained within the imagery. The integration of LiDAR with 2D vector maps, as well as aerial imagery with 2D vector maps, was also reviewed. The addition of the 2D vector map aided in the automation of the DBM generation procedure, however problems occurred when the vector maps did not accurately depict the buildings. The advantages of integrating LiDAR and photogrammetric data were then addressed, and previous research in this topic was reviewed. Each method reviewed has different levels of automation and model accuracy. As mentioned in Chapter 1, the methodology proposed in this work utilizes the complimentary nature of LiDAR and photogrammetric data for semi-automated production of an urban 3D model of high accuracy, which can be employed for engineering purposes.

CHAPTER THREE: PRE-REQUISITES FOR DBM GENERATION

There are several pre-requisites that are necessary for accurate data integration and final 3D model production. The pre-requisites that are outlined in this chapter include: quality assurance (QA) of photogrammetric and LiDAR mapping; quality control (QC) of photogrammetric and LiDAR products, followed by the co-registration of the two data sources.

3.1 Photogrammetric and LiDAR QA and QC

QA procedures are necessary to obtain the best possible accuracies from the utilized systems, and deal predominantly with planning and data acquisition activities. QC activities investigate the performance of the data collection systems. During QC procedures, checks are performed to insure data integrity and correctness. This section briefly discusses the involved QA/QC procedures.

For a camera system, the key QA procedure is camera calibration, however, other factors to consider include the source of control that will be used to georeference the imagery, as well as planning the data collection in terms of the base-height ratio, flying height, the number and distribution of tie points, the ground sampling distance (GSD), image overlap, and camera stability analysis. Since camera calibration is a crucial component for the QA of photogrammetric mapping, this will be addressed first. Deriving accurate 3D measurements from imagery is contingent on precise knowledge of the internal camera characteristics. These characteristics, which are usually known as the interior orientation parameters (IOP), are derived through the process of camera calibration, in which the coordinates of the principal point, camera constant, and distortion parameters are determined. The calibration procedure can be performed in a laboratory, indoor, or in-situ. For indoor and in-situ calibration procedures, control information is required such that the IOP may be estimated through a bundle adjustment procedure. The type of control commonly used are surveyed point targets, however Habib and Morgan (2003) have outlined a low-cost easy to establish calibration technique where the use of linear features in camera calibration was proposed as a promising alternative. In recent years, the use of digital cameras for photogrammetric purposes has become more prevalent, with a wide spectrum of designs, in particular, for medium-format digital cameras. Furthermore, new users have entered the mapping market. Several authors in the field have thus devised an automated methodology for an in-door camera calibration (Habib and Morgan, 2003; Habib et al., 2007b). In addition to camera calibration, stability analysis should be conducted on the imaging system. If a camera is stable, the IOP obtained from multiple camera calibration sessions should not vary over time. Stability can be assessed through a comparison of two sets of IOP derived from two separate calibrations of the same camera (Habib et al., 2006). More specifically, if the object space reconstructed by the two sets of IOP is equivalent, then the camera is deemed stable. Flight planning is an important QA activity that should be conducted before data collection is performed. Habib et al. (2007c) performed an in-depth analysis on the affect of the number and distribution of ground control points (GCP), the number of tie points, the georeferencing method, and the percentage of image overlap on the accuracy of the derived object space for both a large-format analogue camera and a medium-format digital camera. The achievable accuracies under different flight configurations were investigated for indirect georeferencing, direct georeferencing using onboard GPS/INS, and GPS-controlled areal triangulation. The research findings for the medium-format digital camera are now summarized. It was found that when there is a limited availability of tie points, direct georeferencing using GPS/INS performed significantly better in comparison to GPS-controlled georeferencing. In terms of indirect georeferencing, it was found that the number and distribution of GCP has a significant impact on ground point accuracy, in particular in terms of the z-direction. When there is a high number of GCP however, the accuracy is comparable to that achieved from direct georeferencing. When the number of available tie points is increased, direct georeferencing yields slightly better accuracy, although the results from GPS-controlled and indirect georeferencing (with a high number of GCP) are also sufficiently accurate. The final analysis performed in this research investigated the effect of sidelap on the accuracy of the derived object space. When sidelap was increased from 20% to 60%, it was found that all methods yield similar and accurate results. In addition, the vertical accuracy obtained through indirect georeferencing is not as strongly affected by the number of GCP. That is, when sufficient side lap exists in the images and there are a good number of available tie points, the number of GCP that are required is reduced. Such an analysis can allow users to select the appropriate equipment and ground control information, based on desired product accuracy.

In terms of QC procedures for photogrammetric products, photogrammetric triangulation is based on redundant measurements. This provides quantitative measures for the precision of the data. The precision can be measured by the variance component, as well as the variance-covariance matrix of the computed object space coordinates. As these two measures asses the integrity and internal/relative quality of the data, this type of quality control is often referred to as internal quality control (IQC). The accuracy of the photogrammetric data is an external or absolute quality control measure, and is often referred to as external quality control (EQC). The EQC of photogrammetric products is measured through the use of check point analysis, where some check points visible in the imagery have been independently measured and their coordinates are compared with the outcome from the photogrammetric reconstruction.

Similar to QA for photogrammetric mapping, the main QA activity for LiDAR mapping is the calibration procedure. The aim of LiDAR system calibration is the estimation of systematic errors. The quality of the derived point cloud is dependent upon the systematic and random errors in system measurements and parameters (Habib, 2008c). Part of the calibration process involves careful planning of the flight and control surface configurations (Habib et al., 2007d). A calibration flight is performed over a control test field. The discrepancies between the control surface and the LiDAR data are used to determine the systematic errors in the LiDAR system parameters. These errors can include the bore-sighting spatial and rotational offsets, and biases in the system measurements which include the range measurements and mirror angles (Habib et al., 2007d). Mission planning is another important aspect of QA for LiDAR mapping. Aspects of mission planning include: understanding of the area to be mapped to limit the number of occluded areas in the obtained data, the flying height, the pulse repetition rate, scan angle, amount of overlap between flight lines, GPS baseline, and mission time.

To asses the quality of the derived LiDAR point cloud, LiDAR QC procedures can be conducted. In contrast to photogrammetric data, which provides redundancy in the observations that can be used to assess photogrammetric IQC, LiDAR data lacks redundancy in the process for deriving the point cloud coordinates. That is, it is impossible to identify conjugate points in overlapping LiDAR strips, due to the irregular nature of LiDAR data. To evaluate the degree of consistency between overlapping LiDAR strips, LiDAR IQC procedures can utilize conjugate surface elements. The main concept of this approach is that conjugate surface elements in overlapping strips should match as well as possible, in the absence of biases in the system parameters and measurements (Habib, 2008c). To evaluate the external or absolute quality of the LiDAR data, EQC procedures can be employed. A common approach to LiDAR EQC involves the use of signalized targets that are placed along the LiDAR strip and which are used in a check point analysis. This check point analysis is similar to that employed for photogrammetric EQC, although a segmentation procedure must be performed on the LiDAR data to obtain the LiDAR derived coordinates for each signalized target (Csanyi and Toth, 2007).

3.2 Co-Registration

When integrating data from different sources, the datasets must be registered to a common reference frame. Photogrammetric georeferencing, the process of relating the image and ground coordinate systems, is performed by defining the position and orientation information (exterior orientation parameters - EOP) of the camera at the moment of exposure relative to the object space coordinate system. The EOP can be derived either through indirect georeferencing using ground control information or through direct georeferencing when GPS/INS is available onboard the imaging platform. The quality of the reconstructed surface is affected by the accuracy of the EOP, and thus it is essential that these parameters be determined to a high degree of accuracy. In direct georeferencing, the IMU body frame attitude and the GPS phase centre position are directly measured using onboard GPS/INS systems. In addition, the IMU boresighting angles and the GPS antenna offsets relative to the camera perspective centre are computed to determine the position (X_0, Y_0, Z_0) and attitude (ω, φ, κ) of the camera at the time of exposure with respect to object space coordinates frame. Direct georeferencing is computationally efficient as well as economical in the long run since no ground control is required. The quality of the geo-referencing is highly sensitive to the accuracy of the system calibration as well as the IOP of the implemented camera.

Indirect georeferencing has traditionally been the preferred georeferencing procedure due to its accuracy and robustness against IOP biases (Cramer et al., 2000). In indirect georeferencing, the EOPs are determined using ground control, where the most common type of control involves the use of ground control points obtained through field surveying procedures. This form of ground control can be costly in terms of the required time and effort but has proven to be an accurate source of control for the georeferencing procedure. With the recent developments in spatial acquisition systems, however, some reliable alternative forms of control are becoming available. For instance, in recent research, LiDAR-derived features are used as the source of control for image georeferencing. Some of the procedures for use of LiDAR derived control are outlined in the rest of this chapter.

LiDAR georeferencing is directly established through the GPS/INS components of the LiDAR system. Therefore, LiDAR-derived control features can be utilized for the georeferencing of the photogrammetric data relative to the LiDAR reference frame. The use of LiDAR derived control has two main advantages. First, it allows for a straight-forward/direct co-registration of LiDAR and photogrammetric data, which has numerous benefits such as effective production of 3D city modeling and orthoimage generation. In addition, the use of LiDAR control features can eliminate the need for ground control points which simplifies the indirect georeferencing procedure and makes it more affordable. Since the LiDAR footprints are irregularly distributed, however, no point-to-point correspondence can be assumed between the photogrammetric and LiDAR data. As such, it is almost impossible to identify distinct conjugate points in overlapping photogrammetric and LiDAR data. Consequently, LiDAR-derived patches and linear features can be used as control information for the georeferencing of the photogrammetric data. The planar patches are obtained through a segmentation technique (Kim et al., 2007), and a sample result for an area of interest is shown in Figure 3.1a and Figure 3.1b. Linear features are obtained through the intersection of neighbouring planar patches (Figure 3.1c). Details on the methods for the incorporation of patches (or areal features) and linear features in photogrammetric georeferencing are explained in detail in: Habib et al. (2004a); Habib et al. (2007e); Aldeglawy et al. (2008); Habib et al. (2008c); and Habib et al. (2008a). These authors present several alternative approaches, and this section will now discuss in some detail one of the proposed methods for each of the LiDAR-derived features, namely patches and lines.

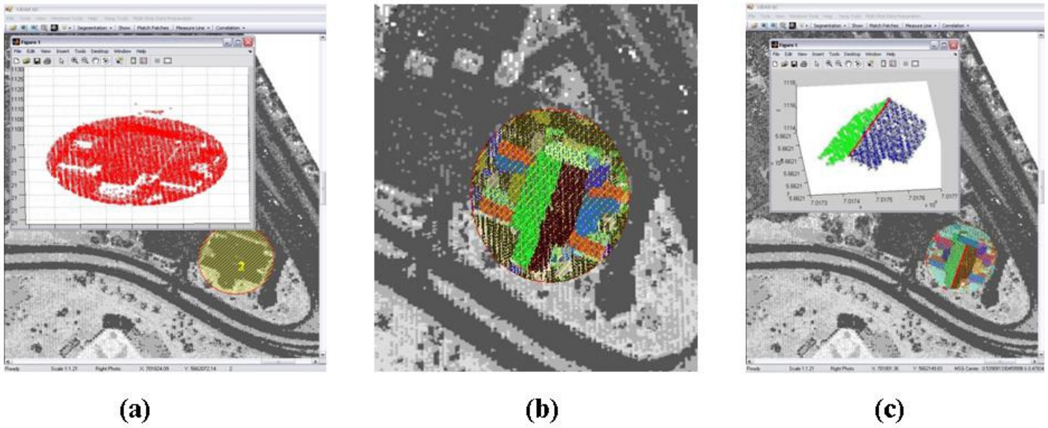


Figure 3.1: LiDAR point cloud extraction (a), Segmented planar patches (b), and extracted linear features through planar patch intersection (c).

The approach for the use of LiDAR derived patches that will be reviewed is referred to as the coplanarity-based incorporation of planar patches. The steps for this procedure include the extraction of the LiDAR patches from the LiDAR point cloud, followed by the incorporation of LiDAR-derived areal features in the photogrammetric triangulation procedure. In this approach (shown in Figure 3.2), the planar patch in the imagery is defined by a minimum of three points, for example points *a*, *b*, and *c*, which are located in the image space, while the LiDAR patch is defined as a set of LiDAR points in object space (Habib et al., 2007e). The points, *a*, *b*, and *c* should be visible in at least two overlapping images. The collinearity equations are used to relate the image space coordinates of the points *a*, *b*, and *c* to their object space coordinates (*A*, *B*, *C*). The LiDAR points belonging to a certain planar-surface patch should coincide with the photogrammetric patch representing the same object space surface. The coplanarity of the

LiDAR and photogrammetric points can be mathematically expressed through Equation 3.1 which outlines how to incorporate the LiDAR points into the photogrammetric triangulation. In physical terms, this constraint means that the normal distance between any LiDAR point P and the corresponding photogrammetric surface consisting of the three points A, B, and C should be zero. In other words, the volume of the tetrahedron comprised of the four points (A, B, C and P) should be zero, as these points belong to the same plane. This constraint is applied for all LiDAR points located within this surface patch. It is important to mention that in order to de-correlate the estimated parameters in the bundle adjustment procedure, one should make sure to use planar patches with varying slope and orientation (Habib et al., 2007d).

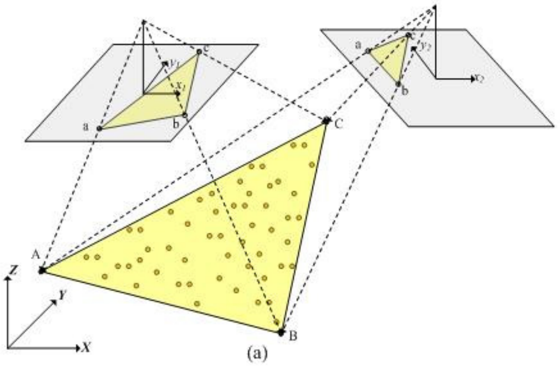


Figure 3.2: Coplanarity-based incorporation of planar patches in photogrammetric triangulation (Habib et al. 2007e).

$$V = \begin{vmatrix} X_P & Y_P & Z_P & 1 \\ X_A & X_A & X_A & 1 \\ X_B & X_B & X_B & 1 \\ X_C & X_C & X_C & 1 \end{vmatrix} = \begin{vmatrix} (X_P - X_A) & (Y_P - Y_A) & (Z_P - Z_A) \\ (X_B - X_A) & (Y_B - Y_A) & (Z_B - Z_A) \\ (X_C - X_A) & (Y_C - Y_A) & (Z_C - Z_A) \end{vmatrix} = 0 \tag{3.1}$$

Where: $(X,Y,Z)_{A,B,C}$ are the object space coordinates of image points a,b,c and (X_P,Y_P,Z_P) are the object space coordinates of any LiDAR footprint, $P=1$ to n, where n is the number of extracted LiDAR points in the areal feature.

Similar to LiDAR-derived areal features, LiDAR-derived linear features can be used as control information for the georeferencing of the photogrammetric data. One of the methods proposed in previous research (Habib et al., 2004a) for the integration of LiDAR linear control features in a photogrammetric triangulation procedure will now be reviewed. This procedure is referred to as the coplanarity-based incorporation of linear features. Neighbouring planar patches are identified and intersected to obtain LiDAR-derived linear features. This technique defines a line in object space by its two end points. These two points in the object space are extracted from the LiDAR data. In the image space, the line is defined by a group of intermediate points (shown as k'' in Figure 3.3). Each of the intermediate points satisfies the coplanarity constraint shown in Equation 3.2. In it, vector $\vec{V}_1$ is the vector from the perspective centre to the first LiDAR end point of the line, vector $\vec{V}_2$ is the vector from the perspective centre to the second LiDAR end point of the line, and vector $\vec{V}_3$ is the vector from the perspective centre to any intermediate image point on the line. The constraint in Equation 3.2 indicates that these three vectors are coplanar, and can be introduced for all the intermediate points along image space linear features. For this procedure, a minimum of two non-coplanar line segments is needed (Habib et al., 2004b).

$$\left(\vec{V}_1 \times \vec{V}_2\right) \bullet \vec{V}_3 = 0 \qquad (3.2)$$

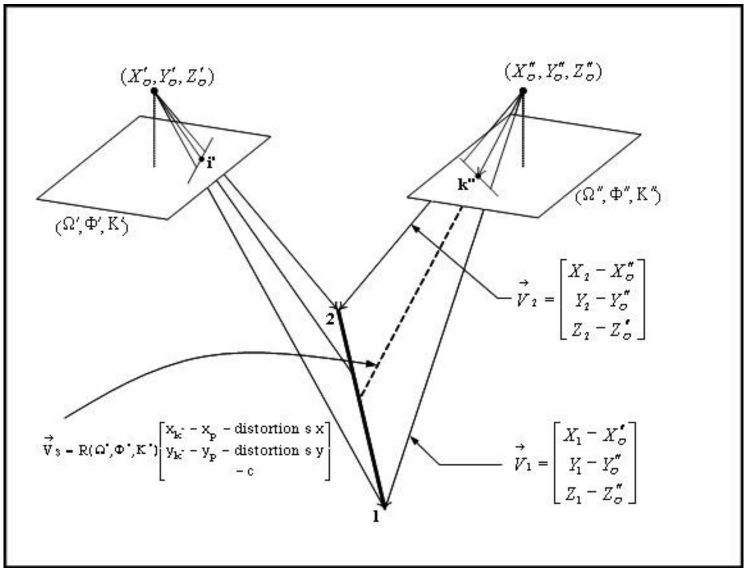


Figure 3.3. Perspective transformation between image and LiDAR straight lines and the coplanarity constraint for intermediate points along the line (Habib et al., 2004a).

3.3 Summary

This concludes the review of the pre-requisites for the proposed methodology for DBM generation. In this chapter, photogrammetric and LiDAR mapping quality assurance was described. The purpose for these pre-mission procedures is to ensure that the data meets the user requirements. Photogrammetric and LiDAR data quality control was then discussed. The purpose of the quality control is to ensure the desired data quality has been obtained, and is thus an important pre-requisite for any work. Then, the need for the co-registration of photogrammetric and LiDAR data was described, in which it was concluded that several advantages can be gained through the use of LiDAR-derived control features for image georeferencing. The following chapter makes use of the co-registered datasets to obtain accurate digital building models over an urban environment.

CHAPTER FOUR: DBM GENERATION METHODOLOGY

4.1 Introduction

The co-registered LiDAR and photogrammetric data are now available for use in the proposed DBM generation procedure, and the workflow of the involved steps is outlined in Figure 4.1. The raw LiDAR point cloud is used to obtain a building hypothesis. This is accomplished by first classifying the LiDAR data into ground and non-ground points. Next, the non-ground points are classified into those belonging to planar surfaces and those belonging to rough surfaces. The reason planar surfaces are desired is that this work focuses on buildings with roofs composed of planar surfaces. Furthermore, these planar roof surfaces are referred to as building-primitives in this work. Neighbouring points that belong to planar surfaces are then grouped, according to their proximity in 3D. The output from this process is a building hypothesis. The next step is to generate initial building-primitives, by performing LiDAR segmentation on the individual building hypotheses. Once the initial building-primitives are obtained, a building-primitive of interest is selected and the best perspective image that contains the chosen building-primitive is located. The best image will be the one in which the chosen building-primitive is located as close as possible to the center of the image, and thus where many if not all building-primitive boundaries will be visible. Canny Edge detection (Canny, 1986) is then performed on the perspective image, and detected edges that meet certain criteria are extracted and displayed to the user. In this research, these outlined steps have been linked sequentially. That is, the user simply inputs the raw LiDAR point cloud and image dataset, then selects a building-primitive of interest, after which the extracted lines from imagery are displayed for their analysis. At this point, the user can visually inspect the output image lines, and if any visible building-primitive boundary lines have not been located automatically, manual editing can be performed to digitize any missing lines. Once the user is satisfied, the lines that are to be automatically projected into 3D must be selected. The projected 3D lines are then displayed, and if there are any portions of the building-primitive boundary that have not been located using the best image, the user can choose to extract lines for the same building-primitive in a second image. This process continues until the user is satisfied with the located 3D lines, or until no more perspective images are available. Automated snapping of neighbouring edges in the object space can then be chosen, followed by manual digitizing if any building-primitive boundaries are still missing. After the boundaries of all building-primitives are obtained through this process, an automated process converts the extracted rooftops into a digital building model. Two output options are available at this time: the DBM can be added to a ground TIN to obtain a refined TIN surface model that is displayed within ArcGIS, or the DBM can be exported to KML format which can be loaded and viewed in Google Earth. As a final step, the refined TIN surface model is exported to raster and used as the DSM for the creation of true-orthophotos. This chapter will outline each of these described steps in sequential order, showing intermediary results.

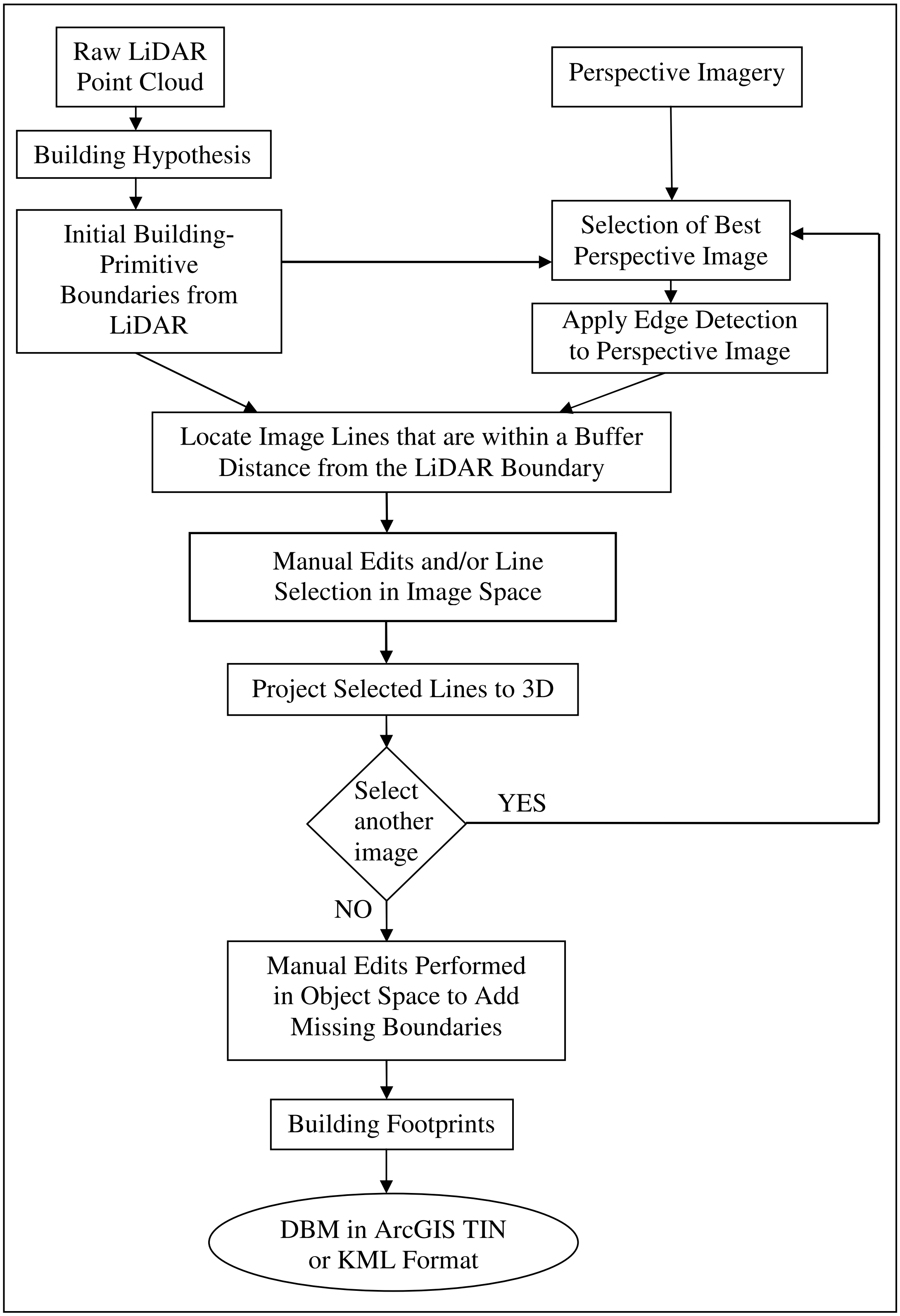


Figure 4.1: DBM Generation Workflow.

4.2 Building Detection and Primitive Generation

This process utilizes the LiDAR data to obtain a building hypothesis, followed by building-primitive generation: planar patches constituting the rooftops. The main steps in generating the initial building-primitive boundaries include: 1) Point cloud classification into ground and non-ground points; 2) Building hypothesis generation; and 3) Rough building-primitive boundary generation. Detailed explanations of these three steps are explained in the following sub-sections.

4.2.1 Ground/Non-Ground Classification of LiDAR data

In order to obtain initial building-primitive boundaries, the LiDAR footprints must be classified into ground and non-ground points. Although there exist several different approaches for LiDAR point classification, the method that will be explained in more detail is a novel ground/non-ground point classification technique developed recently by Chang et al. (2008). In this classification procedure, a perspective center is simulated. The LiDAR point cloud is first converted to a regular grid by using the nearest neighbour resampling technique. The classification process is fully automated and is founded on the knowledge that in a perspective projection, non-ground objects cause occlusions. The top and bottom of above-ground objects are projected as two different points in a perspective projection, and this displacement, which occurs along radial directions from the image space nadir point, causes occlusions in the imagery. In a similar procedure to the angle-based true-orthophoto generation techniques outlined in Section 2.1.2, occlusions are detected by sequentially checking the off-nadir angles (recall that this is referred to as α) to the line of sight that connects the DSM points with the perspective center of the image. The detection of occlusions will be used to locate the points causing the occlusions, i.e. non-ground points. As we proceed away from the nadir point, it is expected that α will increase (Chang et al., 2007; Habib et al., 2007a). As α increases, the checked DSM points will be deemed visible. In Figure 4.2 for instance, the point B will be deemed visible, since α_B is greater than α_A. Whenever there is an apparent decrease in the angle α while proceeding away from the nadir point, an occluded point is located. This will occur at point C in Figure 4.2, since α_C is less than α_B. A backward tracking procedure is then conducted in order to locate the occluding points, which will be presumed to be non-ground points. The tracking begins from the last visible cell and starts tracing back towards the nadir point until the α angle is less than the off-nadir angle of the first occluded point. In the case depicted in Figure 4.2, this backward tracking will proceed until the α angle is less than α_C. As α_A and α_B are both greater than α_C, the points A and B will be classified as non-ground points.

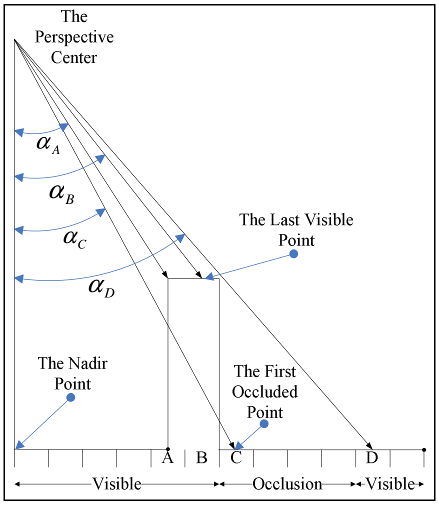
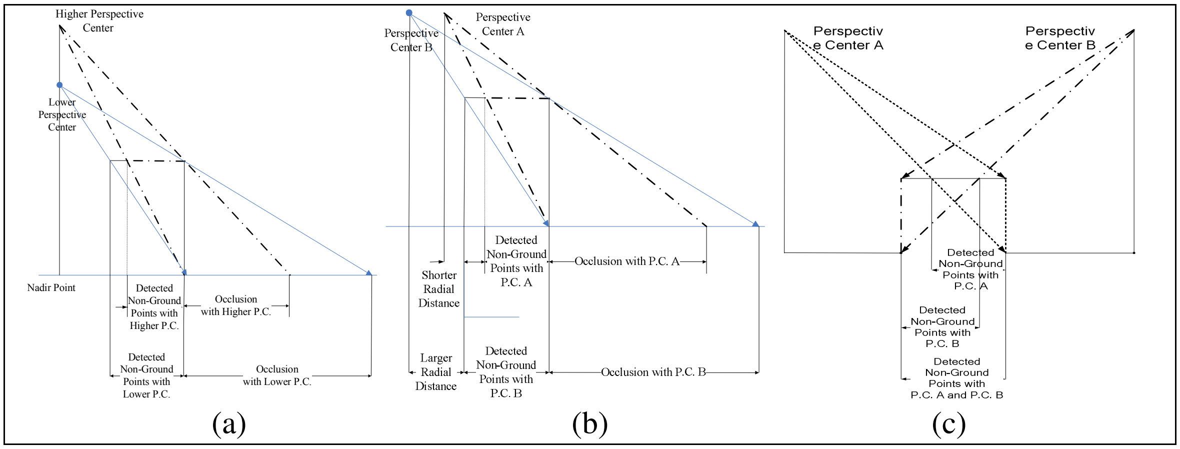


Figure 4.2: Detection of occlusions and occluding points in a perspective view.

The best criteria for the simulated perspective center location will now be explained. As depicted in Figures 4.3, the best location is to position the perspective center at an elevation that is as close as possible to the heights of the buildings and located at a significant distance from the buildings. This will enhance the capability of the procedure in detecting occluding points, as depicted in Figure 4.3a and 4.3b. The reason for this is that under these criteria, the buildings in the imagery cause a greater number of occluded points (shown in Figure 4.3) and thus more of the occluding building roof points can be located. Furthermore, a more complete classification can be obtained through the use of multiple perspective centers. In Figure 4.3c, it is seen that some non-ground points that were not detected using perspective center A have been detected using perspective center B. By merging the results from these two perspective centers, all non-ground points in this figure have been detected.



(a) (b) (c)

Figure 4.3: The effect of using different PC heights (a), different locations (b), and the use of multiple PCs (c), (figures modified from Chang et al., 2007).

Noise in the data as well as surface roughness and high-frequency components (for example cliffs) can lead to falsely detected non-ground points. A statistical filter is thus applied to reclassify any falsely classified ground points. Experimental results showed that this robust classification procedure is successful and performs well in urban environments. Therefore, the classified ground and non-ground points can be used with confidence in the following steps for the creation of a building hypothesis for an urban area. Figure 4.4a shows an aerial image of an area of interest, Figure 4.4b shows the original LiDAR data, and Figure 4.4c shows the resulting classified ground and non-ground points.

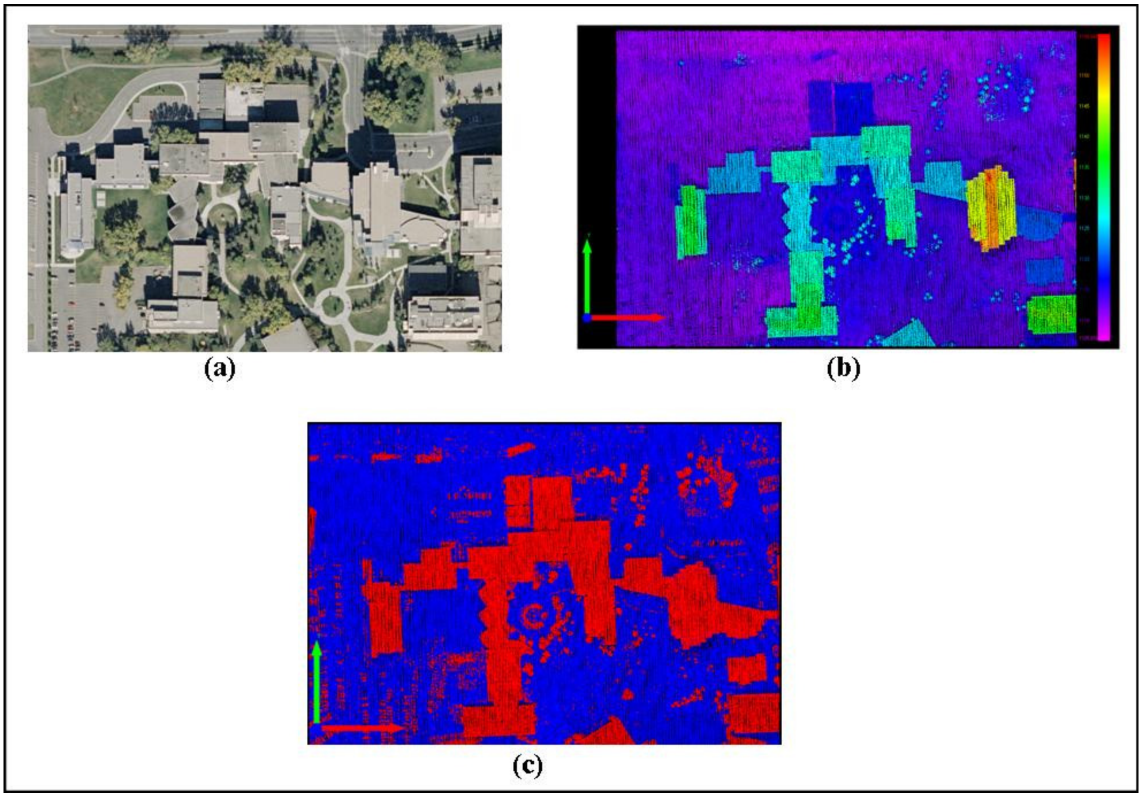

(a)
(b)
(c)

Figure 4.4: An aerial image of an area of interest (a), the original LiDAR data for the area of interest (b), and the classified ground (blue) and non-ground (red) points (c).

4.2.2 Building Hypothesis Generation

The classified LiDAR point cloud can be used to develop hypotheses for the individual buildings that are located in the mapped area. This section outlines the steps involved in obtaining the building hypothesis for an area of interest. First, the non-ground points are further classified into those belonging to planar surfaces and those belonging to rough surfaces. This is accomplished by first defining a spherical neighbourhood centered at each non-ground point. The non-ground points in the spherical neighbourhood are used to compute an initial plane through the use of a least-squares adjustment procedure. The inverse normal distances from the fitted plane to each point in the neighbourhood are used as weights in a second plane fitting procedure, where points that are further from the plane are assigned smaller weights than points closer to the plane. This process is repeated until the plane parameters do not change between iterations, or until a maximum number of iterations are performed. At the end of this iteration process, a final plane is obtained. A buffer is then created above and below the final fitted plane, where the size of the buffer is chosen based on the expected noise in the LiDAR point cloud. This buffer creates what is referred to as a cylindrical neighbourhood (Figure 4.5). A non-ground point is then classified as belonging to a planer or rough surface based on its spherical and cylindrical neighbourhoods. The ratio of the number of points in the cylindrical neighbourhood to the number of point in the spherical neighbourhood will be around 1 for relatively flat surfaces, and significantly less than 1 for points belonging to a rough surface. In addition, if the plane fitting process does not converge, the point under investigation will be classified as belonging to a rough surface. Figure 4.6 shows a sample scenario when the non-ground point on a building roof will be classified as belonging to a planar surface (Figure 4.6a), and a scenario when a non-ground point located on a tree will be classified as belonging to a rough surface (Figure 4.6b).

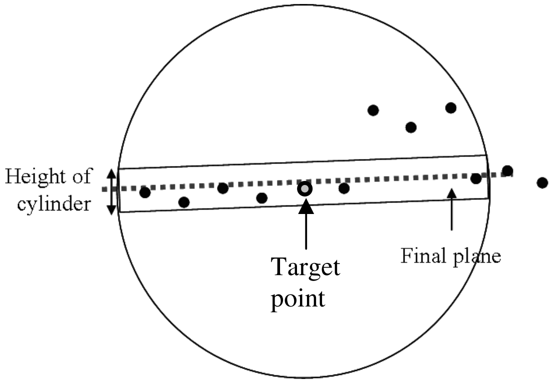


Figure 4.5: Spherical and cylindrical neighbourhood definitions of a non-ground point.

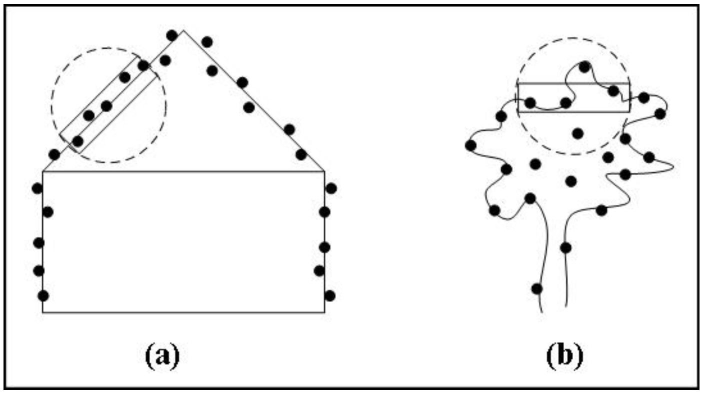


Figure 4.6: A sample case for when the ratio will be computed as 1 for a roof point (a), and a case where the ratio for a rough surface, such as a tree, will be computed as significantly less than 1 (b), (Kim et al., 2007).

After all non-ground points are classified as belonging to a planar or rough surface, the neighbouring points belonging to planar surfaces are then grouped, based on their proximity in 3D. The groups are then analyzed to form building hypothesis by considering the size and height characteristics of each group. This filters some of the groups out, which do not match a given criteria for possible building size and height. The process as outlined in this section is proposed in Kim et al. (2007). Figure 4.7 shows the results of the generated building hypotheses using the classified points shown in Figure 4.4b. The different colours in Figure 4.7 represent the different building hypotheses.

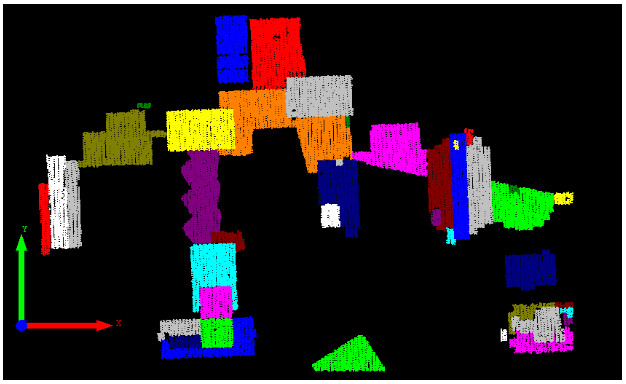


Figure 4.7: Generated building hypotheses for the area of interest, shown in Figure 4.4a.

4.2.3 Building-Primitive Generation

LiDAR segmentation techniques can be used to generate rough building-primitive boundaries from the building hypotheses. The groups obtained from the building hypothesis generation may contain points from more than one roof surface. This can occur, for instance, when a roof structure consists of a series of connected planes with different slopes and aspects. . LiDAR segmentation, in general, is used to relate the LiDAR footprints to physical objects. When dealing with urban environments in particular, the goal is to relate footprints to planar roof patches. The segmentation process is thus applied on each individual building hypothesis to segment points that belong to physically different planes. Detailed descriptions of this procedure are available in Kim et al. (2007) and Habib et al. (2008c), although a brief summary of the three main steps is presented here: 1) First the neighbourhood of a point is defined; 2) A set of attributes is computed for each point, based on its neighbourhood; and 3) A clustering procedure is then conducted based on the computed attributes to group points belonging to the same planar roof surface.

The definition of a point's neighbourhood significantly affects the computed attributes for the given point. For the purpose of planar patch segmentation, it has been found that an adaptive cylinder neighbourhood definition, first introduced in Filin and Pfeifer (2006) and as explained in Section 4.2.2, is optimal. Through this definition, points located on building walls are not considered in the neighbourhood of rooftop points. Figure 4.8 depicts the adaptive cylinder neighbourhood definition.

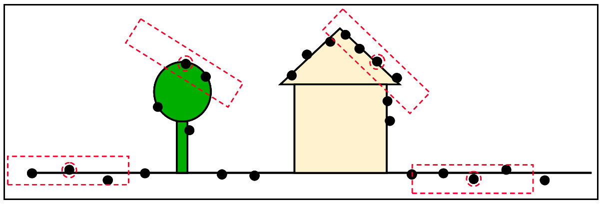


Figure 4.8: Adaptive cylinder neighbourhood definition (Filin and Pfeifer, 2006).

The neighbours of a point are then utilized to compute attributes for each point. For each point and its neighbourhood, a plane of best fit is computed. The normal distances from two predefined origins to the plane are computed, and stored in a 2D accumulator array (Figure 4.9). The reason two origins are used is to insure that points belonging to different planes which are located at the same distance from one origin (for example the three planes shown in Figure 4.9a, which are all at the same normal distance from origin 1) are not stored in the same cluster within the 2D accumulator array. Through this outlined process, all points on the same plane will form a cluster in the parameter space. Figure 4.9b shows the clusters in the 2D accumulator array obtained for the three planes shown in Figure 4.9a.

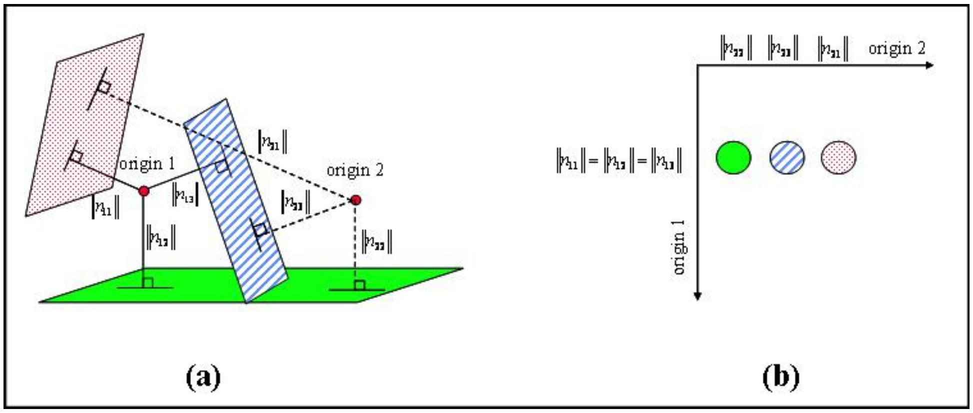


Figure 4.9: The use of two origins from which to compute the normal distance attributes (a), and 2D accumulator array (b), (Kim et al., 2007).

Figure 4.10 shows an accumulator array that contains two planar surfaces. The points contributing to the peaks are identified, and the segmentation results for the two planar surfaces are shown in Figure 4.10b. Therefore, through this outlined process, the points belonging to the same planar surface can be obtained (Figure 4.11a).

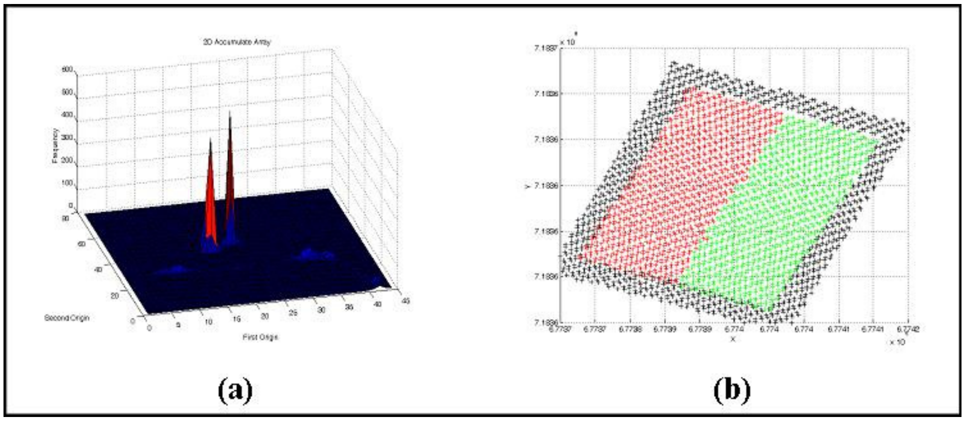


Figure 4.10: Accumulator array (a) and segmented patches for a site with two planar roof surfaces (b), (Kim et al. 2007).

The final step in obtaining the initial rough building-primitive boundaries from LiDAR employs the modified convex hull approach (Jarvis, 1977) to obtain boundary outlines of the segmented clusters (Figure 4.11b). In addition, the plane parameters for each segmented cluster is computed and stored for use in later processing steps. These initial boundaries are now ready for use in the DBM generation procedure presented in the following section.

Figure 4.11: The results from segmentation of the building hypotheses (a), and the initial building-primitive boundaries (b), for the area of interest shown in Figure 4.4.

4.3 LiDAR and perspective imagery for DBM generation

The advantages of integrating LiDAR and photogrammetric data have been known for several years. Weidner and Förstner (1995) note that LiDAR data is a promising option for obtaining 3D information that could be useful during image analysis. Baltsavias (1999) notes that the two forms of data are fairly complementary and through their integration numerous benefits can be obtained including more accurate and complete products, in addition to creating new application areas. Baltsavias notes that when dealing with photogrammetric data, the planimetric accuracy is much greater than the vertical accuracy whereas the opposite is true for airborne laser scanning, where the planimetric accuracy is significantly lower than the vertical accuracy. For these reasons, as well as those outlined in the preceding chapters, the methodology introduced in the following sections is a semi-automated procedure for DBM generation based on the integrated use of LiDAR and photogrammetric data, while requiring minimal user interaction. The refined DBM that is reconstructed in the following sections can be used to improve the surface model of an urban environment, particularly in areas where sharp surface discontinuities, such as building roof boundaries, are present.

4.3.1 Edge Detection within an Image Buffer

In this work, the building-primitives are assumed to be bounded by straight lines. The use of perspective imagery can be a practical option for DBM generation, due to the high visibility of building boundaries. This allows the use of linear feature detection to automate the location of the many building edges, and can also provide more accurate locations for the building-primitive boundaries, in comparison to those derived from LiDAR data (Figure 4.12). Note that the boundary lines displayed in Figure 4.12a represent the detected lines that are within a buffer distance from the LiDAR boundary that is shown in Figure 4.12b. The addition of LiDAR in the building-primitive boundary location process can be very beneficial. Although the initial LiDAR-derived boundaries do not provide accurate building-primitive boundaries, they can be used to assist in the automatic building-primitive boundary extraction from perspective imagery. In addition, the segmented LiDAR data can be used to obtain the plane equations of the rooftops, and thus provides important and accurate height information for the building-primitive boundary points. Furthermore, the integration of these two data sources allow for the determination of 3D information from a single image (i.e., monoploting), and thus does not require any matching of conjugate points in overlapping imagery. The developed procedure begins from the initial building-primitive boundaries. A building-primitive of interest is then manually selected, and the best image from which to extract the selected building-primitive is investigated.

Figure 4.12: Automated linear feature extraction from perspective imagery (a), and segmented roof boundaries from the raw LiDAR data (b).

Image Selection

When creating a DBM from perspective imagery, the first step will be to investigate the selection of the best image from which to extract the boundaries of a specific building-primitive. The best image for a given building-primitive is the image with the shortest distance between its perspective centre and the chosen building-primitive's centroid (Figure 4.13), and thus where none (or as few as possible) of the building-primitive edges will be occluded. When a building-primitive of interest is selected, the distance between the building-primitive centroid and each image perspective centre is computed to locate which image provides the shortest distance. By selecting the image through this criterion we are trying to avoid the relief displacement from neighbouring higher structures for a given building boundary. Figure 4.14 displays the concept of choosing the best image. Once the best perspective image for a certain area has been located, automated linear feature extraction techniques can be used.

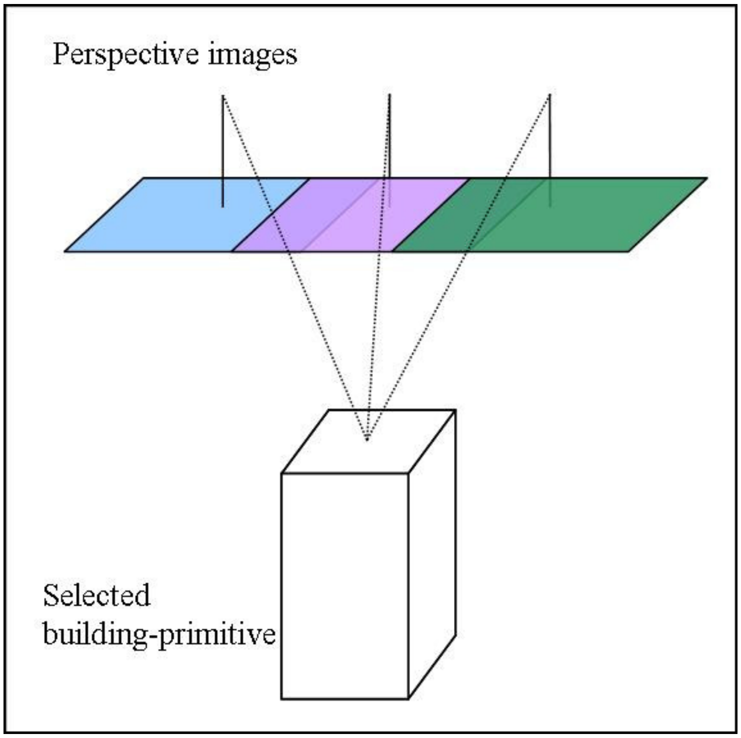


Figure 4.13: Selection of the best image for a given building-primitive.

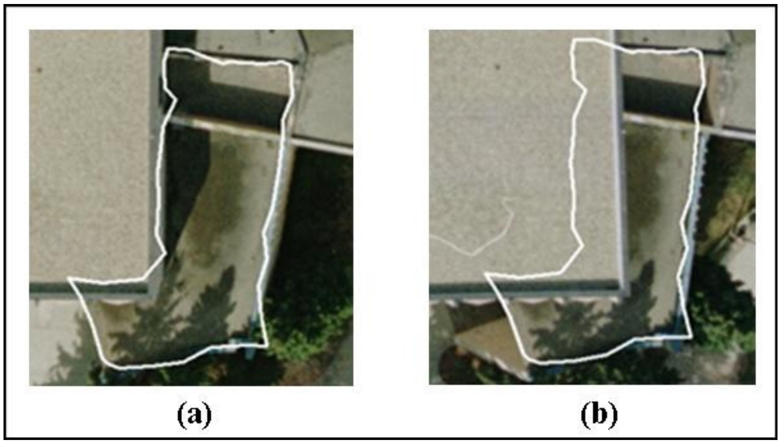

(a) (b)

Figure 4.14: Location of the best image (a) to avoid having occluded building boundaries (b). Note that the white lines represent the projected initial building-primitive boundaries from LiDAR.

Generation of Image Buffer from LiDAR

Once the best image from which to extract a given building-primitive is located, automated edge feature extraction techniques are used. When edge detection procedures are performed on imagery, a large number of linear features are usually detected. This can make the identification of the building-primitive lines of interest quite time consuming and less reliable (Figure 4.15).

Figure 4.15: Sample output from edge detection algorithm, showing the large quantity of edges that are detected for the given building-primitive.

The LiDAR initial boundaries can play an important role in simplifying this process. The boundary vertices of the initial building-primitives, from the LiDAR raw data, can be used to create a buffer around the current building-primitive boundary of interest in the imagery, after which only the area of the image contained within this buffer is considered in further processing steps. Before any buffer operations are performed, however, the LiDAR boundary points must be projected onto the perspective image. This is accomplished through the use of the collinearity equations, to obtain the 2D image coordinates of the 3D boundary points. A black and white binary image of the area surrounding the initial building-primitive is then created. This black and white image is used to represent areas of the image that are within a buffer distance from the LiDAR boundary, where the pixel value of 1 is assigned for pixels in the buffer zone, and 0 for pixels outside the buffer zone (and thus not of interest). The buffer is created by sequentially moving through the list of the projected LiDAR boundary points. For each neighbouring boundary vertices, for example call these *point(i)* and *point(i+1)*, the distance between these points is computed (in pixels), and a moving window of a specified dimension is moved along the line connecting these two points (Figure 4.16). In Figure 4.16, the red circles denote the projected LiDAR boundary points, and the orange square represents the moving window that assigns values of 1 to each pixel that is located along its path.

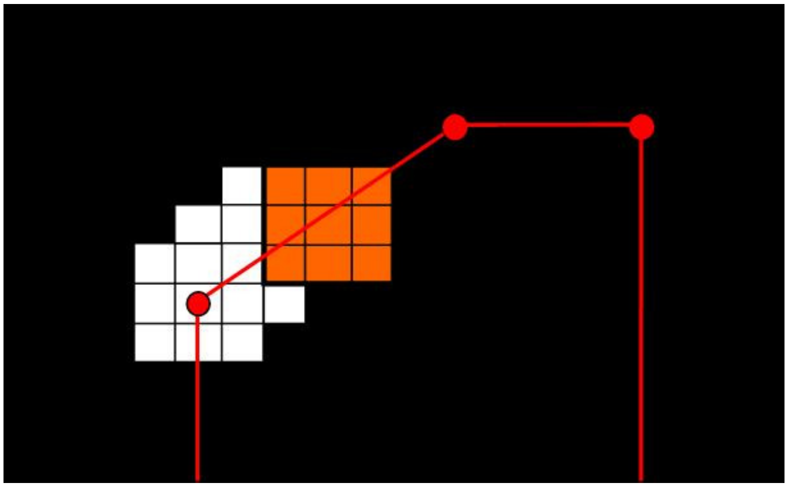


Figure 4.16: Moving window used to create a buffer around a building-primitive.

An appropriate window size must be chosen to guarantee that the true building boundary is located within the buffer area. The selection of the buffer size should thus be made while considering the average point spacing of the LiDAR data. It can be inferred that the true building boundary must be within a distance equal to the average point spacing from the initial boundary of a building-primitive. An assumption is made here that the quality of the segmentation results is very good. Therefore, the chosen buffer size should be twice the average point spacing. Figure 4.17 depicts the concept behind the creation of the buffer image. This buffer image is to be used at a later stage in this process, to reduce the number of lines to consider after Canny Edge Detection has been applied to the perspective image.

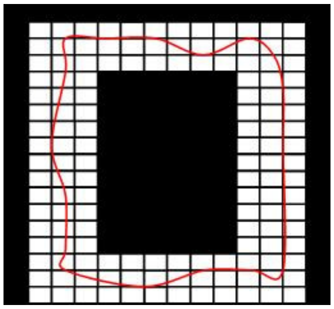


Figure 4.17: Depiction of the concept of the buffer creation (white), based on the irregular LiDAR building-primitive boundary (red line).

Canny Edge Detection

Building-primitive boundaries are often represented by visible edges in imagery. Existent edge detection algorithms can thus be employed to detect edges from imagery. Canny Edge Detection is an algorithm commonly used in various edge detection applications, including this research work. Although the Canny Edge detector algorithm is more complex than other edge detectors, it is often viewed as a superior method (Canny, 1986). Canny aimed at improving on existing edge detection methods, by stating three objectives that a good edge detection algorithm should achieve. These objectives are; 1) Low error rate, 2) Edge points are well localized, and 3) Single edge point response. The low error rate aims at finding all edges, and ensuring non-edges are not falsely detected as edges. The well localized edge response objective states that the detected edges should be as close as possible to true edges. For the third objective, the single edge response, the detector should return one single point for each true edge that is detected. It was found that the best way to accomplish these objectives was to first eliminate noise in the image through the use of a Gaussian filter. The gradient of the smoothed image is obtained using the Sobel operator, after which the magnitude and direction of the gradient are computed in order to estimate the strength of the edge. Nonmaxima suppression is performed next, to insure thin edges are produced. The final step in the Canny Edge detection procedure is a thresholding operation. The purpose of this step is to remove false edge points, and avoid gaps between the detected lines. Experiments outlined in Gonzalez and Woods (2008) have shown that through the use of the Canny Edge detector improvements in detail of the edges are obtained, and irrelevant features are removed. Furthermore, improved continuity and thinness is obtained, which makes the Canny Edge detector the preferred algorithm in many applications (Baillard, 1999; Gonzalez and Woods, 2008).

Once Canny Edge detection has been performed, all edges detected in the image will be saved in an output image, including non-building-primitive boundary edges, which are not of interest in this research. The buffer image produced by the initial LiDAR building boundary, as previously described, can now be used to filter out any detected edges that are not within the area of interest, namely the building-primitive boundary area. Figure 4.18 shows a sample of this procedure. Figure 4.18a displays the output from the Canny Edge detection over a given building-primitive, where in addition to the boundary edge pixels of the building-primitive, many other edges have been detected. Figure 4.18b shows the projected LiDAR building boundary, and the result of retaining only the detected edges located within a buffer around the LiDAR building boundary is shown in Figure 4.18c.

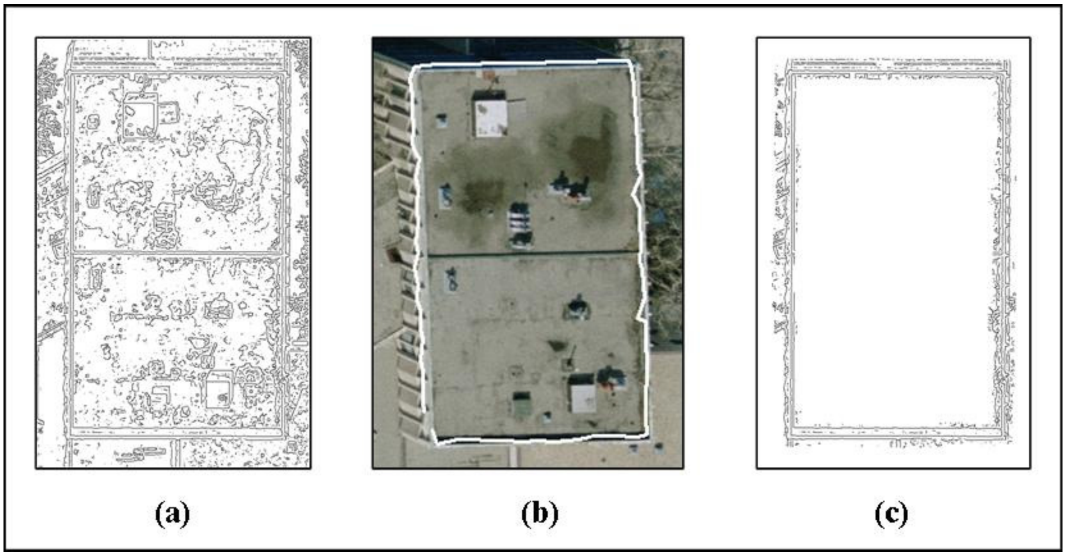


Figure 4.18: Detected edge pixels for a given building-primitive (a), the projected LiDAR building-primitive boundary (b), and the resulting edge pixels located within a LiDAR buffer area (c).

4.3.2 Extracting Image Lines and Obtaining Precise Boundary Segments

After the image buffer is used to locate the detected edge pixels that are near the LiDAR building-primitive boundary, the next step is to search through these edge pixels and connect neighbouring edge pixels to form chains of connected edge elements. This is accomplished by locating the chain endpoints and tracking along the connected edge pixels to form the chains (referred to as edge tracking). Once all chains of connected edge elements are located, edge-splitting is performed followed by straight line fitting, to obtain straight line segments. Adjacent line segments are then merged, as well as collinear line segments, based on several constraints. The resulting line segments are then displayed to the user, who must select the precise boundary segments that are to be projected into 3D. If any building-primitive boundary segments are visible in the imagery but have not been automatically detected, the user can manually digitize these missing line segments. The user-selected line segments are then projected into 3D space using a monoploting procedure. More details will now be provided for each of these steps.

Edge Tracking

The resulting binary image produced from the Canny Edge detection on the perspective image, within the LiDAR buffer zone, is now traced to connect neighbouring edge pixels to form chains of connected edge elements. The chaining process starts by first locating the image coordinates of a chain endpoint. The concept used for endpoint detection is that an edge endpoint will only have one neighbour that is also an edge point. Therefore, edge endpoint detection is performed by sequentially checking every pixel in the image, and determining at which locations two edge pixels are found in a moving 3x3 window, centered on the *checked* pixel. Figure 4.19 depicts the different types of cases where an edge endpoint is located.

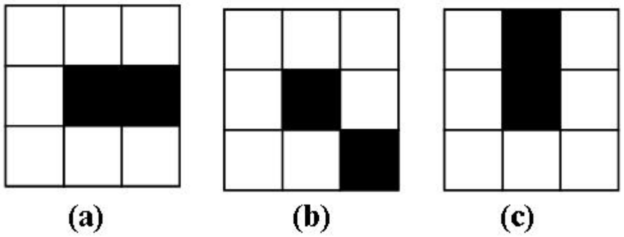


Figure 4.19: Three different types of edge endpoints; horizontal (a), diagonal (b), and vertical (c).

When an edge endpoint is located, the pixel coordinates are stored and a moving window is then used to trace along the connected edge elements (storing each pixel location) until another endpoint is reached. The coordinates of the traced edge are stored, and the traced chain elements are then deleted from the image, so that it is not accidentally traced again. After searching through the entire edge image, all edge chains will be obtained. During this process, any chains that are shorter than a given threshold value will be rejected and not used in further processing steps.

Edge Splitting

The results from the edge tracing may contain curved line segments. This research assumes building boundaries to be straight lines, and thus the next step is to split the traced edges into straight line segments. This is accomplished using a method similar to the Douglas-Peucker algorithm (Douglas and Peucker, 1973). A line is connected between the two endpoints of a tracked edge (shown as a red dashed line in Figure 4.20). The perpendicular distance between every pixel in the tracked (or traced) edge and the dashed line is computed. If the maximum perpendicular distance, shown as "d" in Figure 4.20, is greater than a threshold, the tracked edge is split into two edge segments. This step differs from the Douglas-Peucker method, as the tracked edge is actually split into two under this criterion. The Douglas-Peuker method is used to simplify lines, not to split them. In their method, when the distance d is smaller than a threshold, the line is simplified by joining the endpoints of the red dashed line in Figure 4.20. In this work, the same splitting process is performed on the split edge segments, and if necessary, the segments are further split into smaller edge segments. This process is applied to all tracked edges, and the resulting split edge segments are stored and used in the following section.

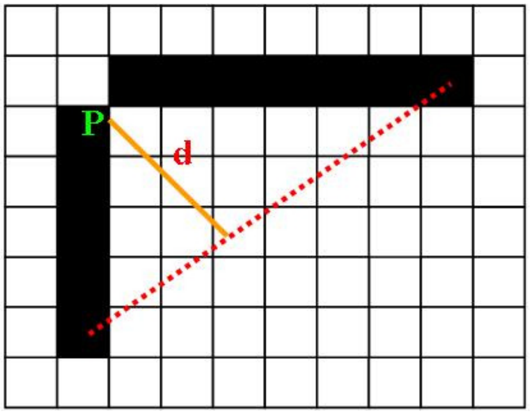


Figure 4.20: A tracked edge that will be split into two lines at the point P if the distance d is greater than a threshold.

Before moving to the next step, however, line fitting is performed. Figure 4.21 shows a case where a split edge is not straight after the edge-splitting is performed. Line fitting is thus applied to all split edges. This is done using all the edge points along the split edge in a least-squares line-fitting procedure. The straight lines are then used in following section.

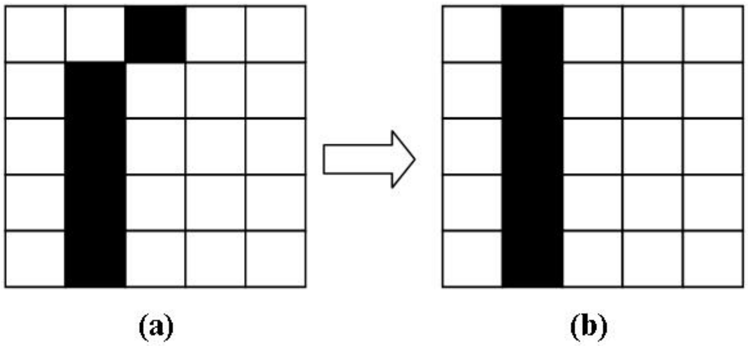


Figure 4.21: Possible output line from line-splitting process (a) where line-fitting is performed to obtain straight-line segments (b).

Line Merging

There are two types of line merging steps that are performed on the image space lines. The first involves the merging of adjacent lines, which are lines that overlap with each other to some degree. The second line merging is performed on neighbouring lines that are roughly collinear. Figure 4.22 show a sample of adjacent lines versus roughly collinear lines.

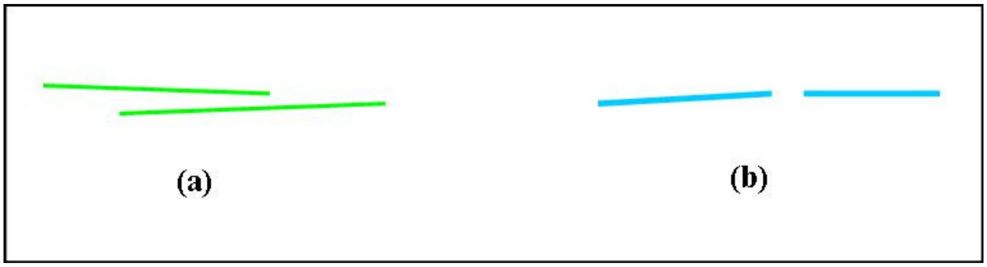


Figure 4.22: Sample adjacent lines (a), and roughly collinear lines (b).

The first step in line merging is to investigate the line directions. If the angle between two adjacent lines is larger than a threshold, then these two lines are not considered further as possible lines to be merged (Figure 4.23). If, however, the angle is smaller than a given threshold, then these two lines are still considered as possible lines to be merged. A reasonable angular threshold to use is 10º, as any lines that have significantly different line directions should not be merged.

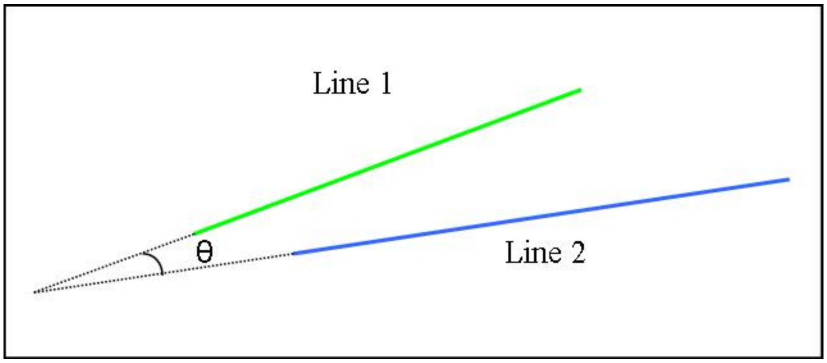


Figure 4.23: Angle constraint for line merging.

Another constraint that is considered, deals with the computation of a line that connects the two line midpoints, which is used as a measure of the normal distance between two adjacent lines. This is best explained by consulting Figure 4.24. In this figure, M1 represents the midpoint of Line 1, while M2 represents the midpoint of Line 2. A line is then connected from M1 to M2, and let us call the length of this new line "*l*". The midpoint of this connecting line is labelled Mm. Therefore, the distance from M1 to Mm is *l/2*. This is also true for the distance from Mm to M2.

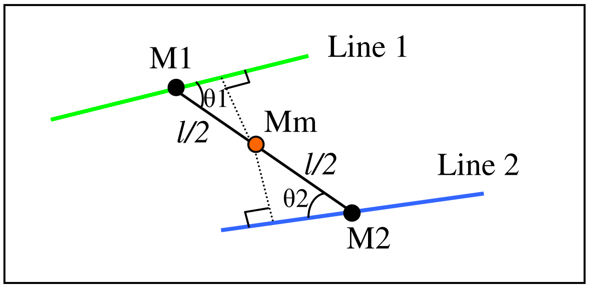


Figure 4.24: Computation of perpendicular distance from the midpoint Mm to Line 1 and Line 2.

The perpendicular distance from point Mm to Line 1 can be computed as $l/2 \cdot \sin\theta_1$. Similarly, the perpendicular distance from point Mm to Line 2 can be computed as $l/2 \cdot \sin\theta_2$. These values are used in the constraint in Equation 4.1, as proposed in Zuxun et al. (2004) and give an indication of the normal distance between the lines. If the sum of these two distances is below a given threshold, these lines are kept. In this work, the threshold was chosen as 2 pixels.

$$\frac{l}{2}\sin\theta_1 + \frac{l}{2}\sin\theta_2 < threshold \tag{4.1}$$

A third constraint is introduced to assess the separation between the endpoints of two line segments. If the line endpoints are separated by a significant distance they should not be merged. Figure 4.25 depicts this new constraint, which is expressed by Equation 4.2. Figure 4.25a depicts this constraint when applied to two nearly collinear lines, while Figure 4.25b depicts the constraint applied to adjacent lines. From Figure 4.25b it can be seen that the left hand side of Equation 4.2 will be negative, and thus all adjacent lines that have passed the previous constraints (i.e., angle and normal distance constraints) will also pass the line separation constraint.

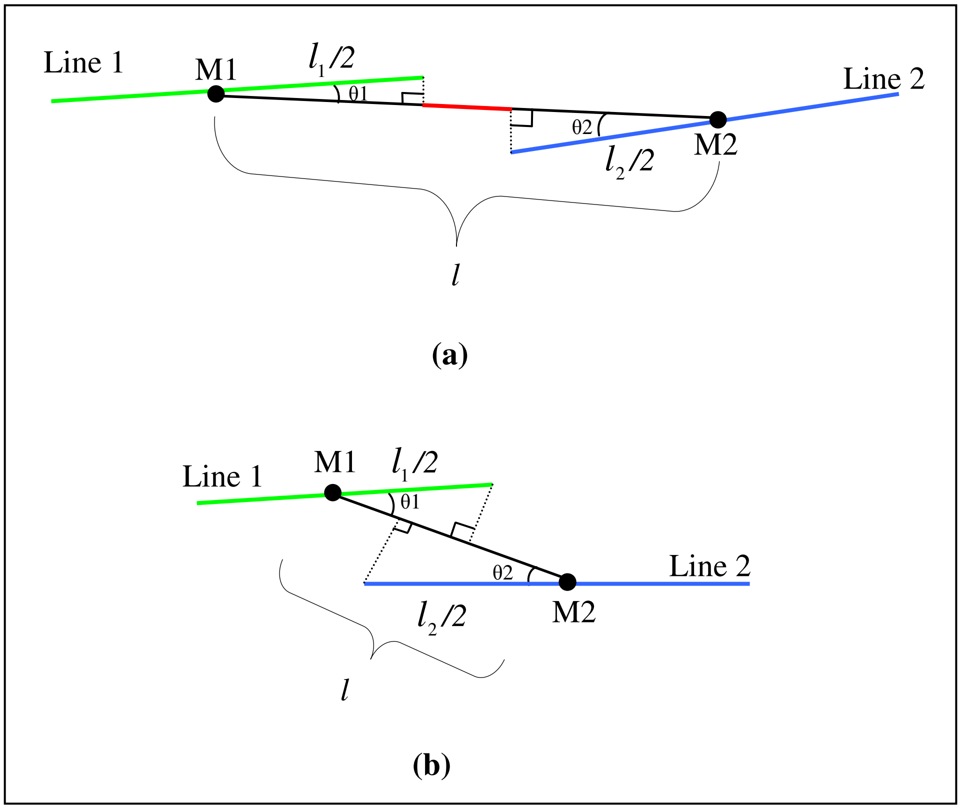


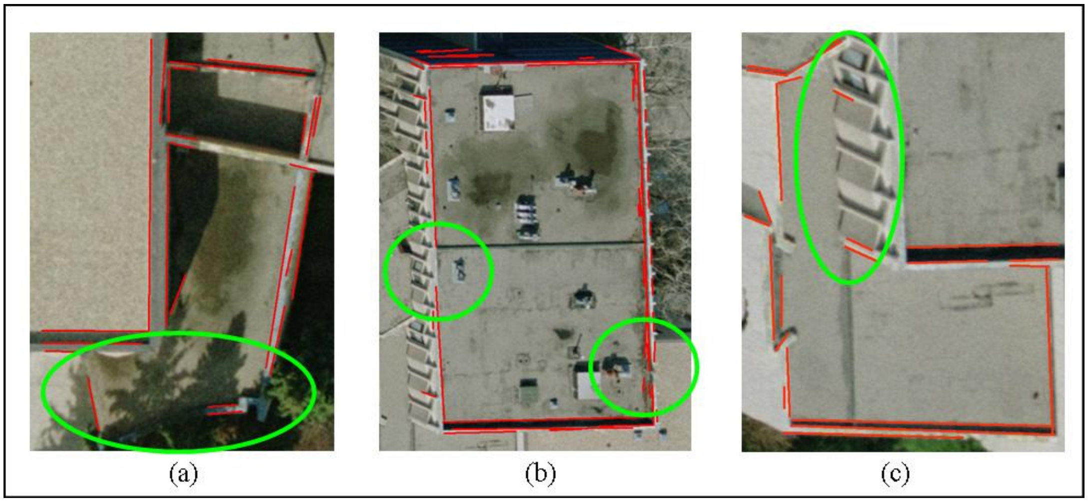
Figure 4.25: Line separation constraint for collinear lines (a), and the effect of applying this constraint to adjacent lines (b).

$$l-(\frac{l_1}{2}\cos\theta_1+\frac{l_2}{2}\cos\theta_2)<threshold \qquad (4.2)$$

Note that the left-hand side of Equation 4 expresses the distance highlighted in red in Figure 4.25. If this distance is smaller than a threshold (chosen as 10 pixels in this work), and the previous two constraints have been met, the line endpoints will be used in a least-squares line fitting procedure, to obtain a merged line. The three outlined constraints deal with both neighbouring/overlapping and almost collinear line segments. This concludes the line extraction from imagery procedures. The lines produced at the end of these outlined steps are now ready for use in the following sections.

2D/Image Space Manual Editing

Following the boundary line extraction from imagery, the user can manually add any missing boundary lines that are visible in the perspective image. Missing lines can occur for several reasons. For instance, when a boundary is in shadow, the edge detection will likely not be able to detect the building-primitive edge, and will more likely detect the edge of the shadow or simply not locate any edges in the boundary region. This situation is shown circled in green in Figure 4.26a, where the shadows cast by the trees prevent automatic line detection for part of the building-primitive boundary. In other cases, if there is not enough contrast between the building roof and wall or ground, the edge detection may fail. This situation is depicted in the green circled area in Figure 4.26b. In addition, if the boundary is composed of small protruding parts, the line tracking process may disregard these edges due to length constraints employed during the edge tracking process. A scenario of this case is shown in Figure 4.26c. In all three figures the red lines shown are the detected lines obtained from the above outlined process. At this point in the process, the user has the option to manually digitize any missing boundaries that have not been located through the automated procedure.



(a) (b) (c)

Figure 4.26: Detected lines and building image clip, demonstrating problems in the straight-line detection process.

Once all visible boundary lines are obtained, either through the edge detection process or from manual digitizing, the user manually selects the lines, which represent the precise boundary segments for the building primitive in question, which are to be projected into 3D. The line selection performed by the user is a vital step in this process. If the selection of 2D lines was implemented automatically, there would be some cases in which incorrect lines would be selected. The lines that are chosen manually, on the other hand, can then be used with confidence. This is important as the accuracy of the reconstructed building-primtives is dependent upon the accuracy (and number) of detected lines. Lines chosen through an automated process will always have certain cases in which wrong selections are made. Therefore, the manual digitizing and line selection options that are offered to the user is a key contribution to obtaining accurate and more complete building-primitive lines.

The selected lines are then automatically projected into 3D, and this projection is performed through a monoploting procedure, where monoploting is the process of obtaining 3D information from measurements in a single image (Makarovic, 1973). The regular monoploting procedure is an iterative process, and in some cases the solution will diverge. However, in this research we do not perform iterations, as the plane of the building-primitive in question is already defined by the LiDAR segmentation. In addition to not requiring the implementation of an iterative procedure, there is no risk of the monoploting solution diverging. The plane equations for the building-primitives are used in conjunction with the collinearity equations, to directly obtain 3D information from a single 2D image. The collinearity equations are shown in Equations 4.3 and 4.4, with the plane equation shown in Equation 4.5.

$$x_a = x_p - c\frac{r_{11}(X_A - X_o) + r_{21}(Y_A - Y_o) + r_{31}(Z_A - Z_o)}{r_{13}(X_A - X_o) + r_{23}(Y_A - Y_o) + r_{33}(Z_A - Z_o)} \tag{4.3}$$

$$y_a = y_p - c\frac{r_{12}(X_A - X_o) + r_{22}(Y_A - Y_o) + r_{32}(Z_A - Z_o)}{r_{13}(X_A - X_o) + r_{23}(Y_A - Y_o) + r_{33}(Z_A - Z_o)} \tag{4.4}$$

$$aX_A + bY_A + cZ_A + \mathrm{d} = 0 \tag{4.5}$$

The object space coordinates of the endpoints of the established precise boundary segments are obtained through this monoploting process, and are displayed in 3D for the user. The following section outlines the options available once object space 3D lines are obtained.

4.3.3 Building modelling in Object Space

The previous section outlined the detection, digitization, selection, and projection of building-primitive boundary lines from the best image for a given building-primitive, into 3D object space. In some cases, there may be some missing building-primitive boundaries, which were not visible in the chosen perspective imagery. There is thus an option available to the user to explore other images to complete the boundary of the building-primitive in question. When the user decides to explore another image for a given building-primitive, an option can be chosen to select the next best image from which to extract lines. This option repeats the previously outlined steps, and in addition, the object space lines obtained from the *best image* are projected into the image space of the *next best image*. Figure 4.27 shows an instance when using the *next best image* option is needed. The best selected image for a given building-primitive has one boundary that is not visible (and is circled in red in Figure 4.27a), whereas the next best image for this building-primitive has a clear view of this boundary (which is circled in green in Figure 4.27b). Manually selected/digitized lines from previous images plus the detected straight line segments from the current image are displayed together in the same image space, as shown in Figure 4.28.

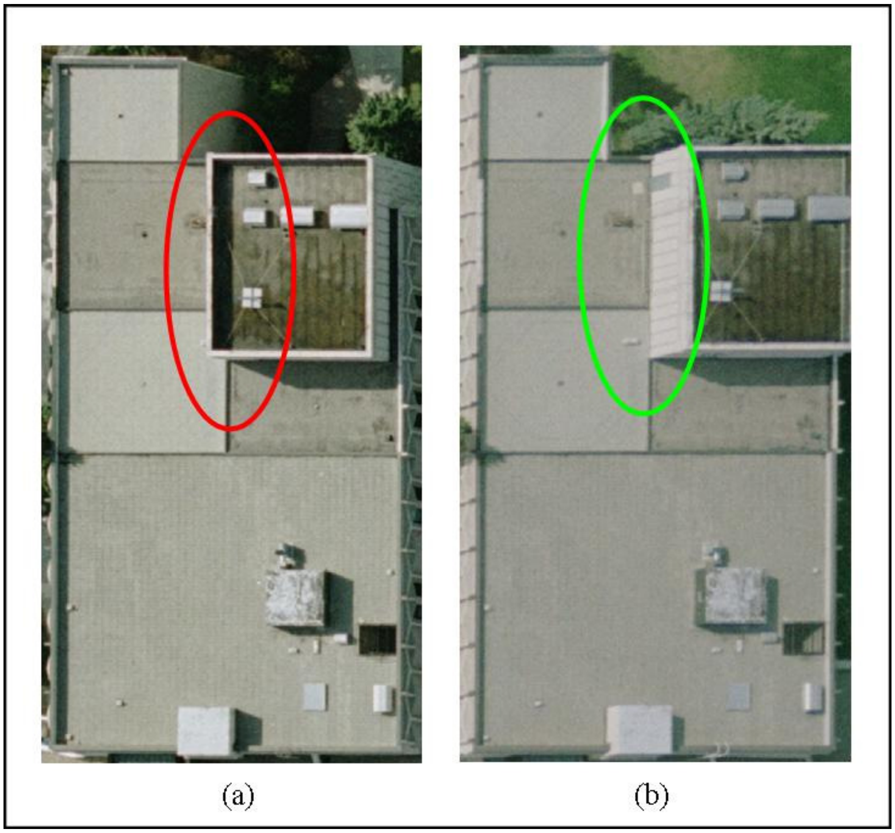


Figure 4.27: A building-primitive in which a boundary line within the red ellipse is not visible in the best image (a), but is clearly visible within the green ellipse in the next best image (b).

Figure 4.28: The display of detected lines from multiple images - red lines from previous image, green from current.

The final step in obtaining the complete building-primitive boundary is performed in object space. In some cases, some boundary segments for a given building-primitive may not be visible in any of the images contained in the dataset, or small gaps may be present between neighbouring building-primitive boundaries. When one of these situations occurs, these lines must be created/edited in 3D to obtain a closed building-primitive roof boundary. There are two options available to the user. The first option, called *3D Prunning*, has been implemented to snap neighbouring lines together if they are separated by small gaps. A threshold for the snapping process has been set, by the author, to 2m. Therefore, any neighbouring lines that are separated by 2m or less will be snapped together automatically. Figure 4.29 shows a sample case when this option can be very useful. In Figure 4.29a, two boundary segments of the building-primitive have not been automatically located, due to low contrast near the building edges. These two lines are then manually digitized on the image (in image space), as shown in Figure 4.29b. Note however, that in Figure 4.29b there are small gaps between many neighbouring lines. It would be quite time consuming to manually add line segments for all these gaps. Figure 4.29c shows the lines for this building-primitive after being projected into 3D. The snapping option has then been used to obtain the closed 3D building-primitive outline shown in Figure 4.29d.

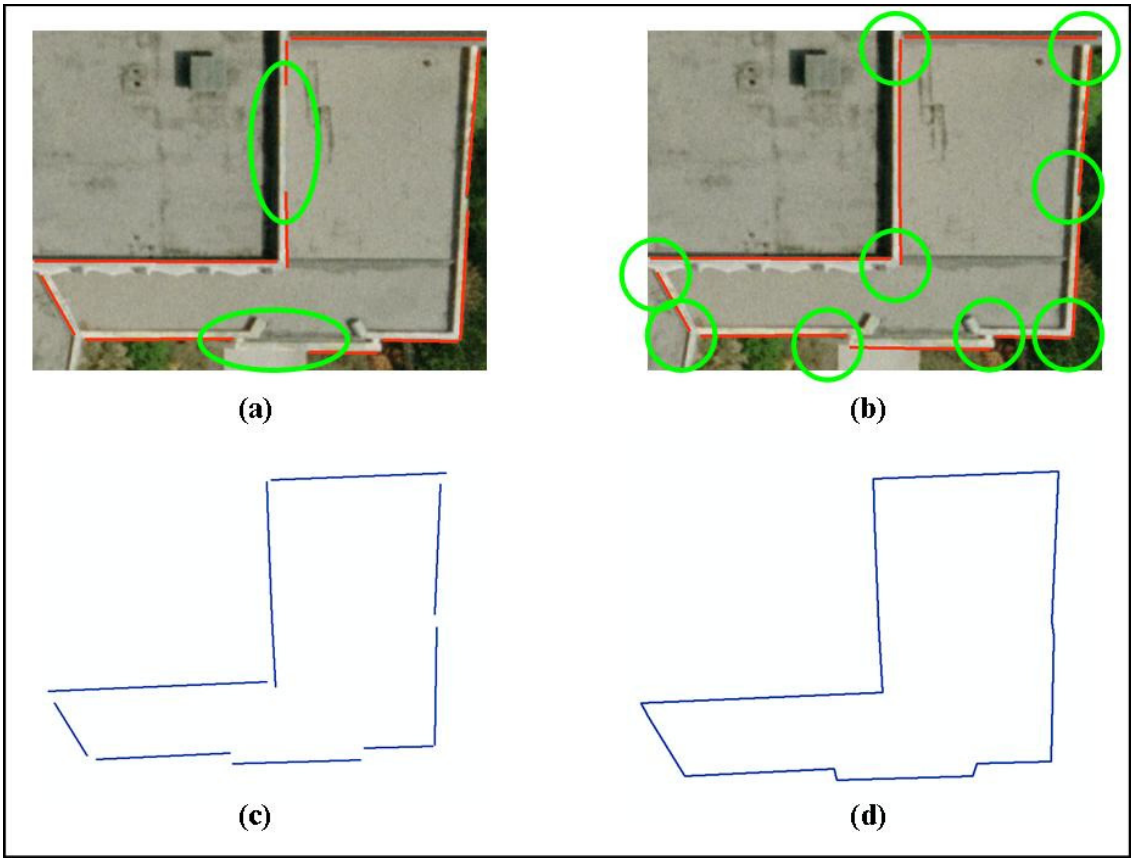


Figure 4.29: A building-primitive with some boundary lines not detected automatically, due to low constrast (a), the boundary lines after manual editing, with gaps between all lines (b), the projected lines of the given building-primitive into 3D (c), and the results from using the snapping option (d).

The logic behind the snapping function will now be explained. Figure 4.30 shows three sample cases where gaps are present in the boundary of a given building-primitive. For the first case, circled in blue in Figure 4.30, two neighbouring lines that form corners are to be snapped together. The program will snap these lines by computing the intersection point of the two lines, and then changing the coordinates of the line endpoints to match the coordinates of the intersection point. The Z-coordinate is then obtained from the roof plane equation. In some cases the extracted neighbouring lines have an orientation that causes their computed intersection point to lie outside the extent of the building-primitive, that is, not between the neighbouring line endpoints (shown in Figure 4.30a and 4.30b, circled in red). In these cases, a new line is simply placed between the neighbouring lines. Although this produces a slightly jagged boundary (Figure 4.30c), it is temporary as it will be compensated for in later processing steps when the Douglas-Peucker algorithm is applied to all building boundary lines. A third case is displayed in Figure 4.30, circled in green, where two neighbouring lines are collinear. Under this constraint their intersection point cannot be computed. When two neighbouring lines have a similar slope, the snapping function will simply connect the endpoints of the collinear or almost collinear lines.

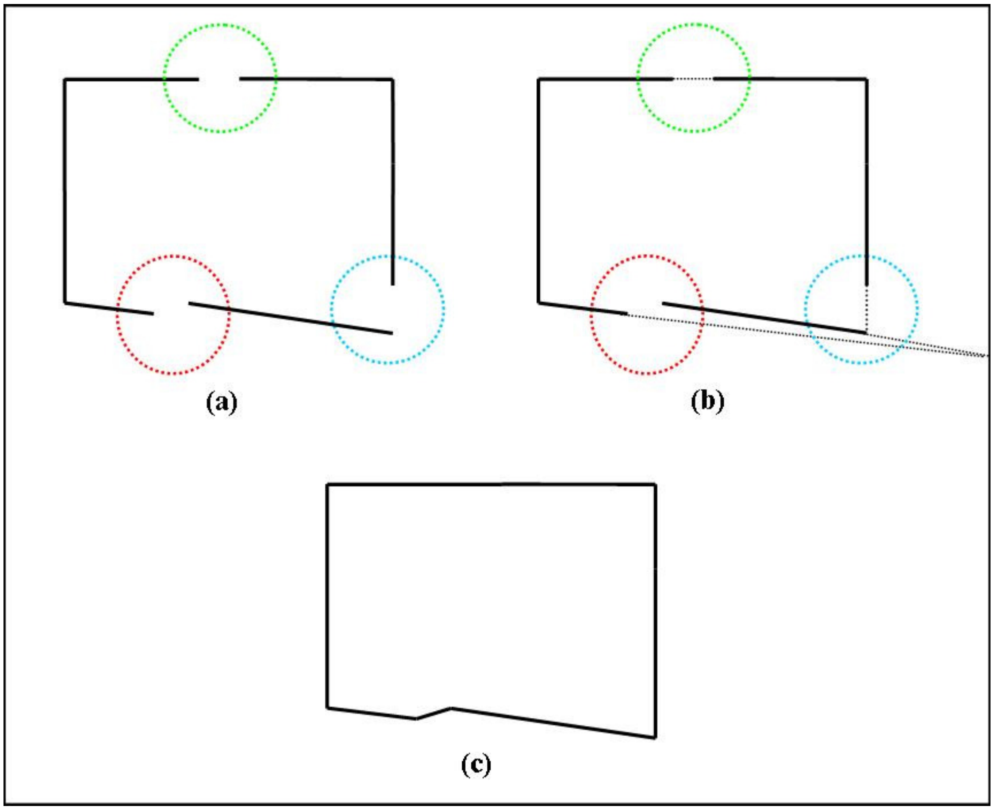
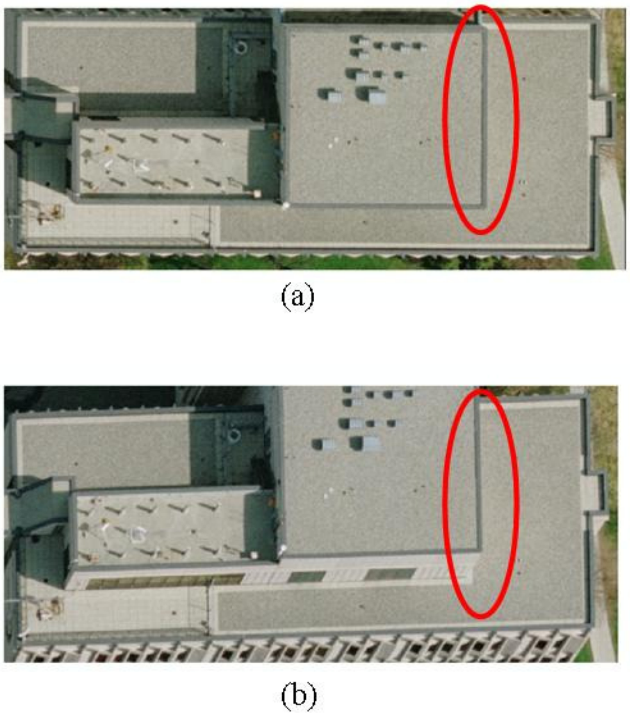


Figure 4.30: Three cases for gaps between neighbouring lines (a), the connection of neighbouring lines to the intersection point or the joining of line endpoints (b), the result from snapping the lines together, and the resulting building-primitive boundaries(c).

As previously mentioned, there are two options for closing the building-primitive boundaries in object space. If the first option, snapping the lines together, does not close all building-primitive boundaries (recall there is a threshold for the snapping process), then manual editing is required. Figure 4.31 shows a building-primitive that led to this situation, where one of the building-primitive boundaries was not visible in any image, as it is occluded by a neighbouring building-primitive. Figure 4.31a shows the best image for the given building-primitive and Figure 4.31b shows the next best image for the same building-primitive. Figure 4.32 shows the building-primitive boundary lines before and after the snapping option is used.



(a)

(b)

Figure 4.31: Images of a building-primitive in the best image (a) and next best image (b), where a one boundary is not visible, thus requiring manual digitizing in 3D.

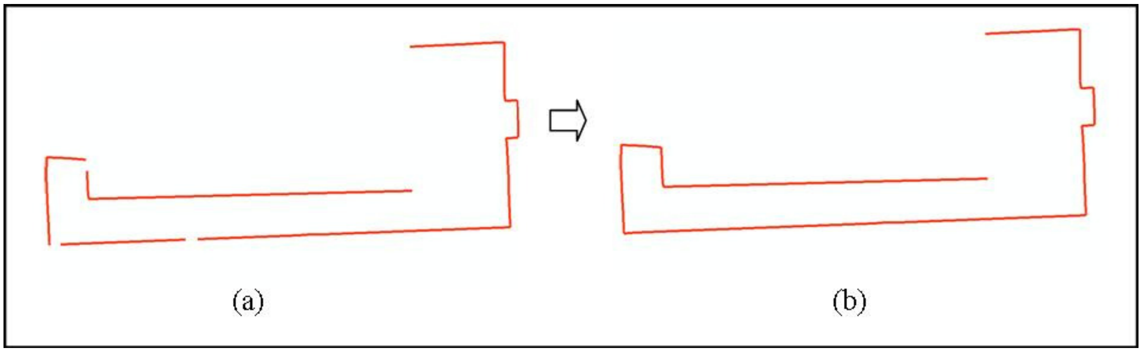


Figure 4.32: The boundary lines for a building-primitive, before (a) and after (b) using the snapping option.

Note that for this case there is still a boundary segment for this building-primitive that is missing. A tool is available to the user for manual digitizing in 3D, to obtain any missing boundary lines. For this tool, the LiDAR initial boundaries can be used as a guide for digitizing, if desired (Figure 4.33). In Figure 4.33a, the initial LiDAR boundaries are displayed as polygons, and the established boundary segments are displayed in red. By using the LiDAR boundary as a guide, the user can be confident that the two lines beside the gap should be joined (Figure 4.33b). Note that during the manual editing in 3D, the ArcGIS snapping function that snaps newly added lines to existing lines is enabled.

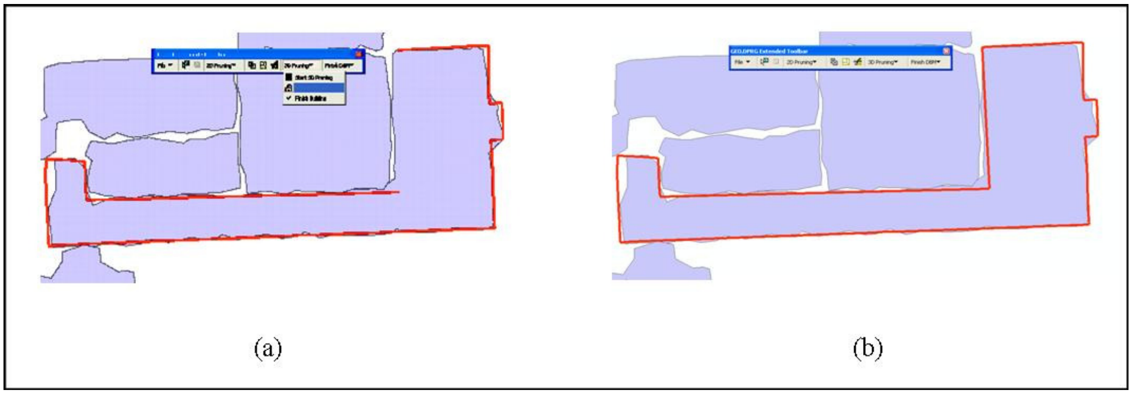


Figure 4.33: Use of the LiDAR boundary for manual editing in 3D, where a) and b) represent the boundary outline before and after editing.

Once all the boundaries of a building-primitive have been obtained, the boundary line segments are used to obtain a building-primitive polygon. This step is necessary, as the next step requires the input building-primitives to be in polygon format. When the polygon is created, the Douglas-Peucker algorithm is applied to the boundary lines. This algorithm aims at preserving directional trends in linear features based on a tolerance that is used to specify the amount of simplification that is desired (Douglas and Peucker, 1973). The generalized building-primitive polygons are then used in the next section to create a refined DSM, by adding the building-primitive polygons to a DTM of the area. Figure 4.34 shows a sample of a building-primitive before and after the generalization is performed. In Figure 4.34a, there is a jagged line segment circled in red (produced by the 3D snapping function). The Douglas-Peucker algorithm generalizes the building-primitive boundary to obtain the result shown in Figure 4.34b.

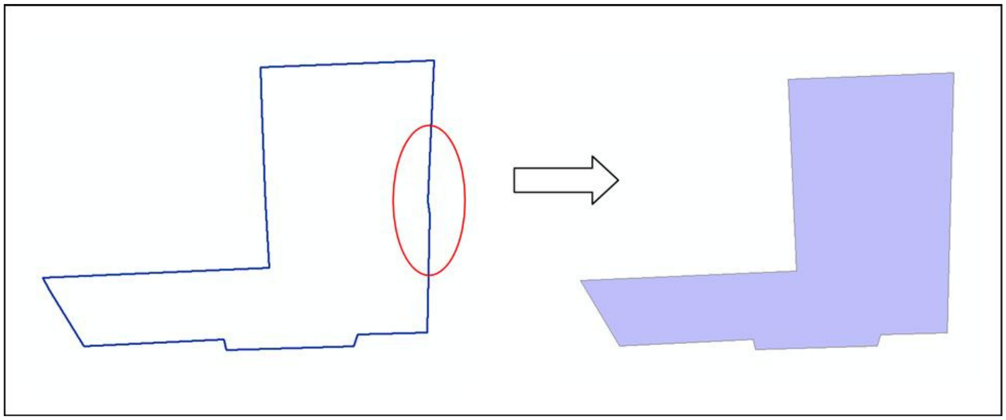


Figure 4.34 Generalization of building boundaries to remove the small bend in the building outline circled in red (a), through the use of the Douglas-Peucker algorithm to obtain straight boundary segments (b).

4.4 Generating a Refined DSM

The outlined process discussed so far is applied sequentially to the building-primitives one by one. The discussed procedure in this section is applied while simultaneously considering all the building-primitives. This section outlines the process in which the rooftop polygons are added to a ground TIN to obtain a refined DSM. First, to obtain a DBM both the rooftops and the building footprints are required. The rooftops acquisition for the individual building-primitives is performed as outlined in Section 4.3. Several steps are required to obtain the building footprints. First the roof patches are aggregated based on their proximity into new polygons which form the external building boundaries (Figure 4.35). The elevation information for the footprints are obtained by projecting the aggregated polygons onto the ground TIN, and the Z-coordinates for the footprint vertices are obtained by linear interpolation of the ground TIN. Before the rooftop and footprint polygons can be incorporated into the ground TIN, one final modification must be made, due to the 2.5D limitation of TIN models. That is, each X-Y location on the TIN can only have one elevation point. In reality however, the rooftop and footprint will have two different elevations for the same X-Y pair. To address this limitation of the TIN structure, the building footprints are slightly expanded outwards thus avoiding truly vertical walls. This expansion however is very slight and thus does not have a significant effect on the final output DBM. In addition, any ground nodes located within the building boundaries or within a specified buffer distance (10cm) from the building footprints are first selected and deleted from the ground TIN. The final step is the addition of the building rooftops and footprints to the ground TIN (as hard lines).

Figure 4.36a shows the building rooftops and footprints, while Figure 4.36b shows the DBM that has been added to the ground TIN.

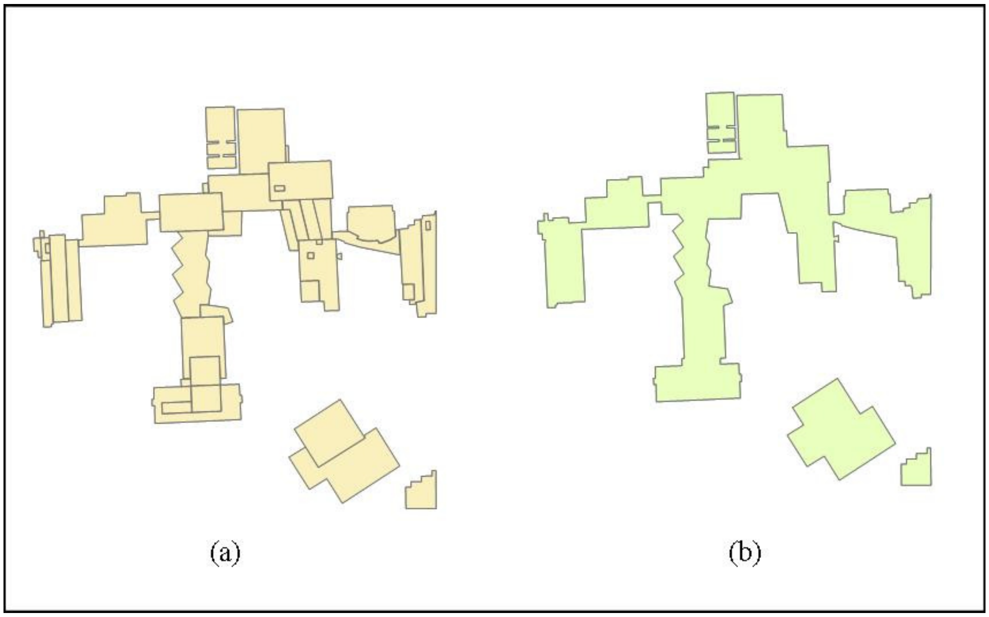


Figure 4.35: The building rooftop polygons (a) and the aggregated building footprints (b).

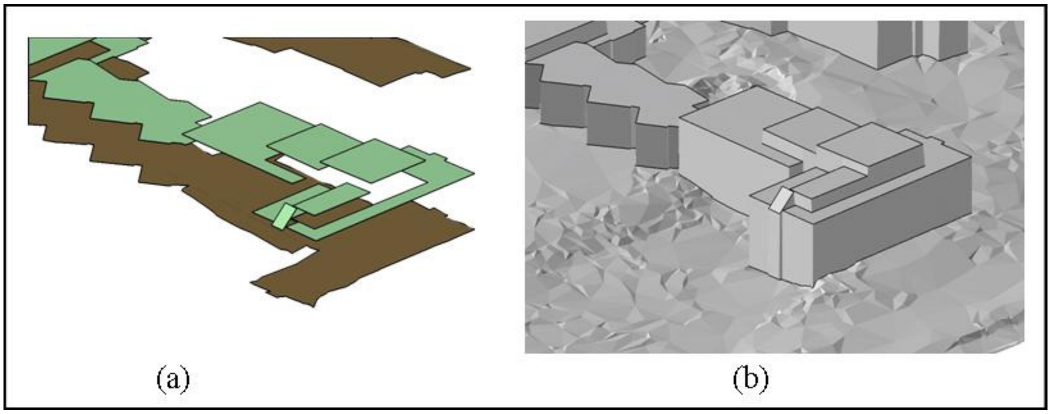


Figure 4.36: Building rooftops and derived footprints (a), and the TIN surface model from adding the buildings to TIN (b).

The resulting TIN surface model would not look as visually appealing if it was obtained by adding the DBM to the original TIN surface model obtain from the LiDAR footprints (as opposed to the ground TIN). Reasons for this are that the representation of other non-ground features are not accurately modelled in the LiDAR derived TIN, and thus are not visually appealing in either the refined surface model or true ortho-imagery. Figure 4.37 shows a sample "refined" TIN surface model that is obtained by adding the reconstructed DBM to the original TIN surface model, in comparison to a refined TIN surface model obtained by adding the DBM to a ground TIN. To obtain a refined DSM that can be used to produce improved true-orthophotos, the TIN shown in Figure 4.36b is converted to raster using the nearest neighbour resampling technique. This method is chosen to maintain the established boundaries.

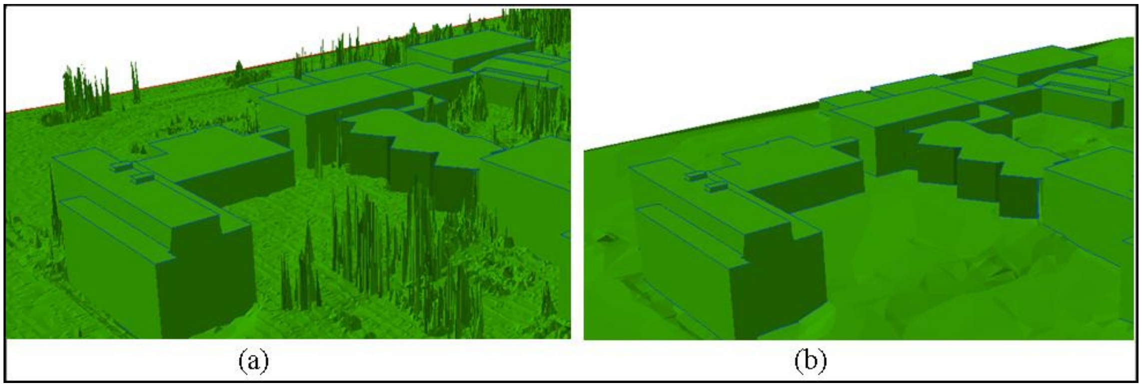


Figure 4.37: Refined TIN surface model that is obtained when the DBM is added to the original TIN surface model (a), in comparison to when the DBM is added to a ground TIN (b).

4.5 Refined True-Orthophoto Generation

The enhanced DSM obtained in the previous section can be used to provide a more complete surface model for a 3D realistic visualization of an urban area. Chapter 1 outlined that in addition to a surface model, the most important feature of a city model is the ortho-imagery that is draped over the surface model. The use of the refined DSM for true-orthophoto generation is thus the last objective to discuss in this research. The first step is to export the refined TIN surface model to a DSM, which can be used for true-orthophoto generation. Experimental results for the improved true-orthophotos are presented in Chapter 5. In addition, a comparison between the resulting orthophotos from two variations of the refined DSM is performed: the first DSM is obtained by adding the DBM to the ground TIN, whereas the second DSM is obtained by adding the DBM to the original TIN surface model. Figure 4.38a shows the true-orthophoto obtained using the original DSM, and Figure 4.38b shows the true-orthophoto generated using a refined DSM that was obtained by adding the DBM to a ground TIN. It can be observed that the building boundaries are clearly defined in Figure 4.38b, as opposed to Figure 4.38a where the building boundaries are jagged. Figure 4.38c shows the 3D visualization of the dataset using the original orthophotos and surface model, while Figure 4.38d shows the refined 3D visualization obtained using the refined surface model and improved true-orthophotos.

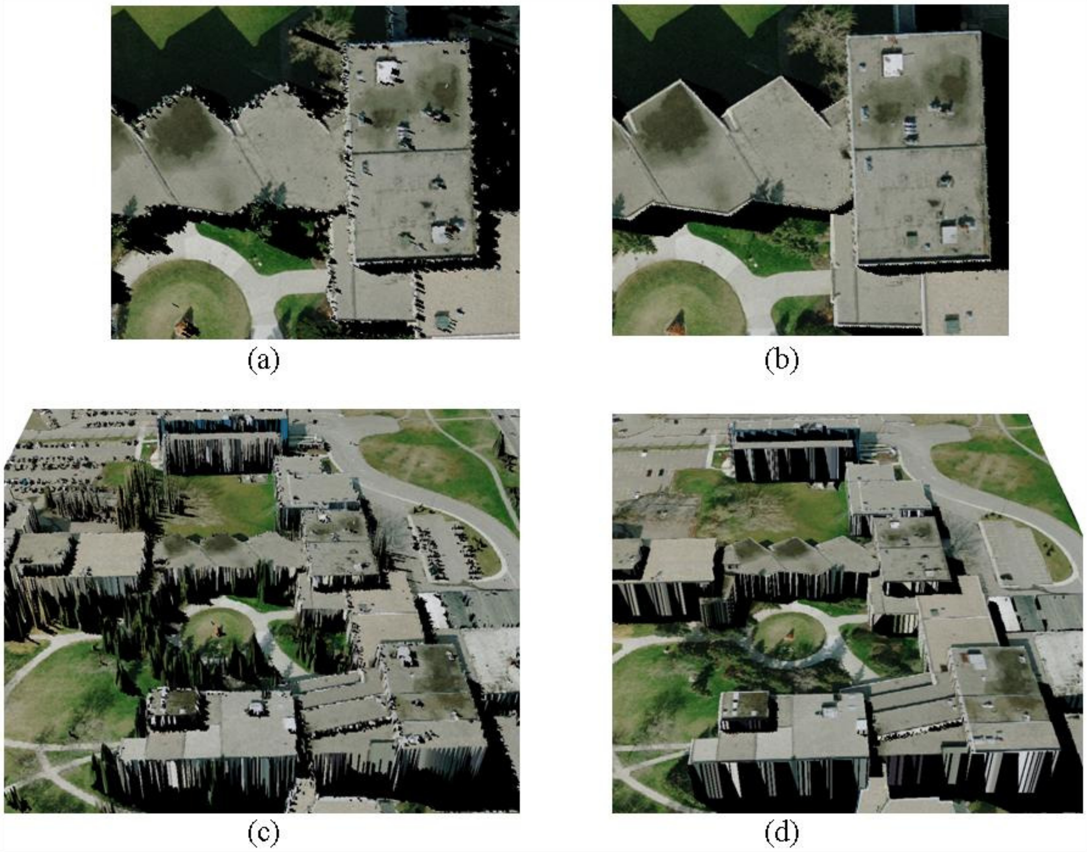


Figure 4.38: True-orthophoto generated using the initial DSM (a), true-orthophoto generated using the refined DSM (b), the 3D visualization using the initial surface model and true-orthophoto (c), and the 3D visualization using the refined surface model and improved true-orthophoto (d).

4.6 Summary

This chapter has outlined the proposed DBM generation methodology, performed through the integration of photogrammetric and LiDAR data. Building detection and primitive generation using the LiDAR footprints was first reviewed. In particular, the method proposed in Chang et al. (2007) for ground/non-ground classification of the LiDAR data was outlined, followed by a discussion on building hypothesis and building-primitive generation procedures as proposed in Kim et al. (2007). The building-primitives and perspective imagery were utilized in a semi-automated procedure to obtain a DBM of an urban area. The DBM was then added to a ground TIN and converted to raster to obtain a refined DSM, which was then used to obtain improved true-orthophotos. Together the refined raster surface model and improved true-orthophotos can be used to obtain an accurate and realistic 3D visualization of an urban environment.

CHAPTER FIVE: EXPERIMENTAL RESULTS

5.1 Introduction

To assess the performance of the proposed methodology, experiments were carried out using a real dataset. Both LiDAR and image data were utilized, and qualitative and quantitative analyses were performed on the generated output. The qualitative analyses involve a visual comparison of the DSM, orthophotos, and 3D models that are produced using: 1) the refined DSM obtained by adding the DBM to the original TIN surface model, and 2) the refined DSM obtained by adding the DBM to a ground TIN. The quantitative analysis was conducted through root mean squared error (RMSE) computations between building corner coordinates obtained manually (using photogrammetric techniques) and the building corner coordinates obtained from the proposed methodology. Repeatability of the implemented DBM generation process was also investigated. The developed software program will be described in this chapter, and intermediate results are shown and discussed.

5.2 Dataset Description

The data obtained for the conducted experiments is of the University of Calgary and surrounding area. This data is ideal for testing the proposed methodology, as the university campus contains numerous buildings of various sizes and shapes. The photogrammetric data includes nine photos in three strips. The photos were captured by an RC30 frame analogue camera, with an average flying height of 770m, and a focal length of 153.33mm. The image scale is 1:5000, and the photos were digitally scanned at 12 microns resolution to obtain a 6cm GSD. With a pixel size of 12 microns and an image measurement accuracy of 1 pixel, the expected horizontal accuracy is around 0.06m, while the expected vertical accuracy is around 0.30m. Ten LiDAR strips with an average point spacing of around 0.8m were captured in two flight missions over the study area (six strips in the first mission and four strips in the second mission), with an Optech 3100 sensor. The data was captured with a flying height of 1000m for the first flight mission, and 1400m for the second. According to manufacturer specifications for the given flying height, the expected accuracy for the LiDAR data is a vertical accuracy of 15cm for both flight missions, while the horizontal accuracy for the first flight mission is 50cm, and 70cm for the second. The overlap between the LiDAR data and imagery can be seen in Figure 5.1. In accordance with the pre-requisites outlined in Chapter 3, the data were co-registered as described in the following section.

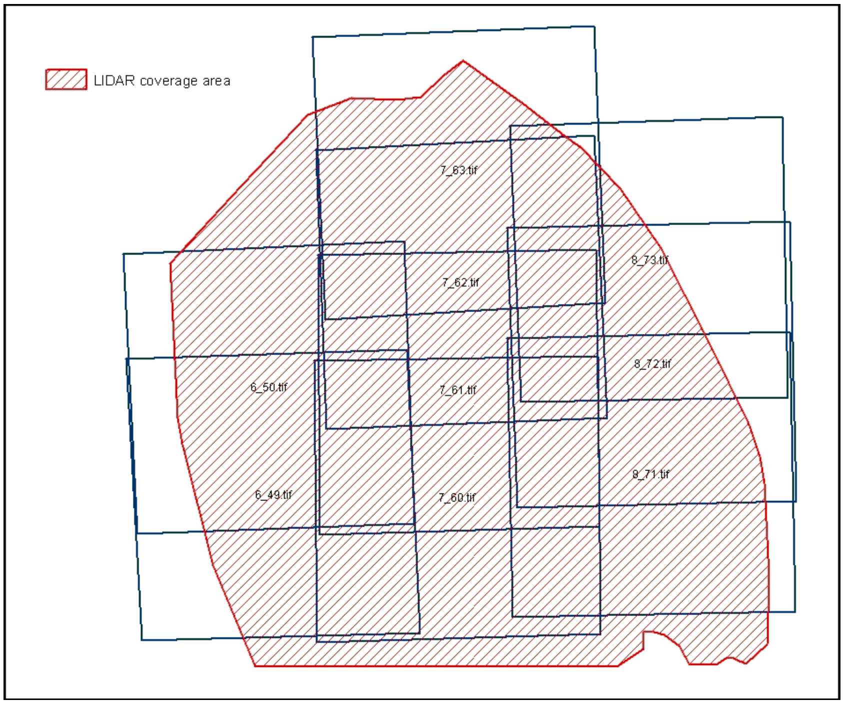


Figure 5.1: Overlap between LiDAR and photogrammetric data.

5.3 Co-Registration

The co-registration of the utilized data had been performed prior to this research using LiDAR-derived linear features, to georeference the involved imagery. It was found that using LiDAR-derived features to obtain the image EOP was a better option in comparison to the use of ground surveyed control points, for several reasons that will be discussed. A comparative analysis was performed on the use of ground control points (GCP) versus LiDAR-derived features for image georeferencing, in an independent study. A static GPS survey of 24 GCP was conducted around the University of Calgary campus. Out of the 24 GCP, 8 points were used to georeference the imagery using a conventional photogrammetric bundle adjustment (BA) procedure. The remaining 16 surveyed ground points were used as check points. A root mean squared error analysis (RMSE) was performed between the 16 check points and their values as computed from the photogrammetric BA. The results are summarized in the second column of Table 5.1. With a pixel size of 12 microns and an image measurement accuracy of 1 pixel, the expected horizontal accuracy is around 0.06m, while the expected vertical accuracy is around 0.30m. From Table 5.1, it can be seen that the expected accuracies match closely with the results computed in this experiment ($\boldsymbol{RMSE_X}$, $\boldsymbol{RMSE_Y}$, $\boldsymbol{RMSE_Z}$,). The results from the georeferencing of the imagery using LiDAR-derived linear features are presented in column 3 of Table 5.1. A relatively large amount of bias is present in the results ($\boldsymbol{Mean_{\Delta X}}$, $\boldsymbol{Mean_{\Delta Y}}$, $\boldsymbol{Mean_{\Delta Z}}$), which is not present in the results from georeferencing experiments using GCP. The reason for this is that a bias has been identified between the LiDAR reference frame and the reference in which the ground control points were surveyed. Moreover, a bias in the LiDAR system parameters was suspected as well. The error amount (**σ_X, σ_Y, σ_Z**), however, is reasonable. In particular, the horizontal standard deviation is similar to the results from using GCP, while a noted vertical standard deviation improvement is obtained. A possible explanation for this is that many more LiDAR-derived control features (50) were used in comparison to the number of GCP used (8). That is, the improved vertical accuracy may be due to the higher redundancy. In addition, the linear features are derived from a plane fitting and intersection procedure, which reduces the noise in the LiDAR data. Therefore the vertical accuracy of the linear features is significantly better than 15cm. The effect of the bias value is seen in the final RMSE values (**$RMSE_X$, $RMSE_Y$, $RMSE_Z$, $RMSE_{Total}$**), which are larger than those presented in the second column of Table 5.1.

Table 5.1: Mean, standard deviation, and *RMSE* analysis of the Check Points using surveyed GCP and LiDAR linear control features

METHOD	**Surveyed GCP** **- 8 Control Points** **- 16 Check Points**	**Coplanarity Method, Linear Features** **- 50 Control Lines** **- 24 Check Points**
$Mean_{\Delta X}$ (m)	0.05	1.00
$Mean_{\Delta Y}$ (m)	-0.01	-0.76
$Mean_{\Delta Z}$ (m)	0.14	0.65
σ_X (m)	0.09	0.10
σ_Y (m)	0.11	0.10
σ_Z (m)	0.25	0.11
$RMSE_X$ (m)	0.10	1.00
$RMSE_Y$ (m)	0.10	0.77
$RMSE_Z$ (m)	0.28	0.66
$RMSE_{Total}$ (m)	0.32	1.43

A qualitative comparison between these two types of georeferencing procedures was also performed. Two orthoimages were generated using the angle-based true orthoimage generation methodology outlined in Section 2.1.2. They were generated using the perspective image shown in Figure 5.2, a digital surface model, and the two sets of EOPs resulting from using GCPs as well as from the use of LiDAR derived linear features as the source of control for image georeferencing. The generated orthoimages are illustrated in Figures 5.3a and 5.3b. From these images it can be seen that the orthoimages generated using the LiDAR data as a source of control appears to be more accurate than the orthoimage generated using GCPs. This can be observed in the orthophotos in Figure 5.3, where there are more traces of building boundaries projected onto the ground in the latter orthoimage (Figure 5.3a). Therefore, the EOP derived from the LiDAR control ensures better co-registration between the photogrammetric and LiDAR data.

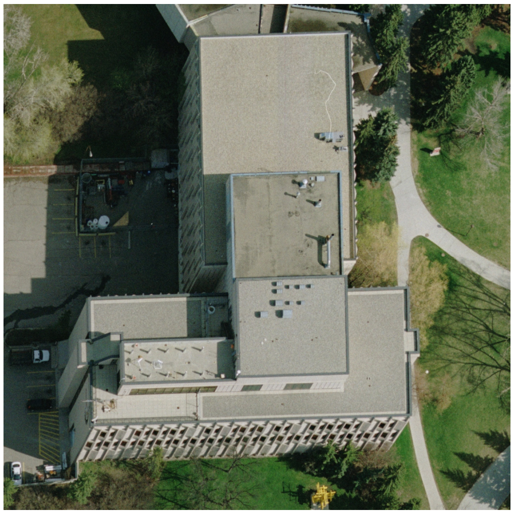


Figure 5.2: Perspective image of a building on the University of Calgary campus.

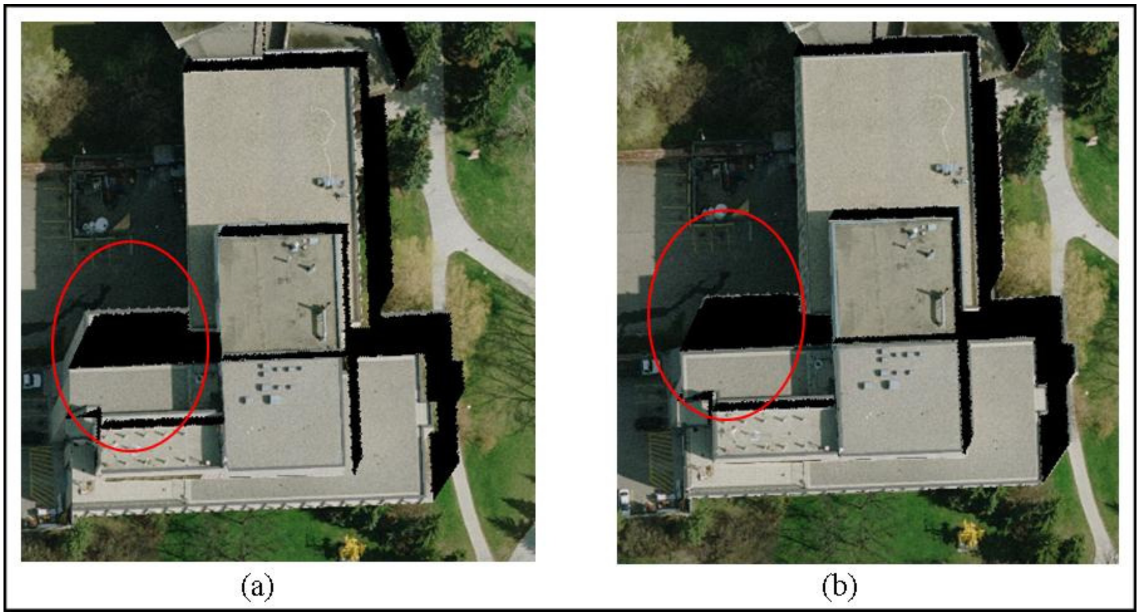

(a) (b)

Figure 5.3: Orthoimage produced using GCP as the source of control (a), and orthoimage produced using LiDAR derived linear features (b).

The reason for this is that any bias that exists in the LiDAR data will not be visible in the final orthoimages, when the source of control data and the digital surface model have both been obtained from the same (although biased) data source. To summarize the conclusions of this study, the use of LiDAR features and GCPs for georeferencing appeared to give compatible horizontal precision. On the other hand, LiDAR features seemed to give better vertical precision. The inferior quality of the orthophotos generated using GCP is due to co-registration problems between the reference frame in which the ground control was collected and the LiDAR reference frame. That is, the derived EOP are in the GPS reference frame, while the DSM used to produce the orthophoto is in the LiDAR reference frame. The bias that was identified between these reference frames contributes to the degraded quality of the produced orthophoto using the GCP as the source of control for image georeferencing.

5.4 Software Implementation

The implementation of the proposed methodology was performed within the ArcGIS platform, through the use of existing functions as well as from the addition of a custom toolbar which was added. The coding language used for the customized modifications of the ArcGIS functionality was performed using the native ArcGIS programming language ArcObjects, as well as in C#. This multi-platform integration was very advantageous, as the powerful functionality available within ArcGIS can be modified to suite the desired application, without having to program many processes from scratch. The developed program is called *GEO DPRG TOOLS EXTENDED*, and the purpose of this program is to reconstruct 3D building models using LiDAR and aerial image data. The program developed in this work is a semi-automated process. The use of a semi-automated DBM generation procedure offers several advantages: some steps are automated which can improve speed, while incorporating some manual work for times when automated methods fail or provide incomplete results. Table 5.2 lists the functions available within the ArcObjects library that were used in the developed software. When this research began, the algorithms for several procedures (LiDAR data classification, LiDAR segmentation, etc.) required for the proposed methodology were available within numerous different programs and programming languages. To allow a user to perform the full process of creating a refined DSM from the two input data (co-registered LiDAR and photogrammetric data), the existing programs were merged within an option on a drop-down menu of the implemented toolbar (Figure 5.4). The first step that is required, as outlined in Chapter 4, is the classification of the LiDAR point cloud into ground and non-ground points, after which the classified non-ground points are used to generate building hypotheses and initial boundaries for the building-primitives. This process is invoked through the *Compute initial boundary* option shown in Figure 5.4.

Table 5.2 List of ArcGIS functions used for various purposes in the developed software.

Purpose	ArcGIS function used
Generalization of the building boundaries	*Douglas-Peucker Algorithm*
Perform manual editing operations	*Editor Toolbar*
Selection of a building-primitive of interest, and selection of the image space lines	*SelectFeaturesTool*
Convert the refined TIN to Raster	*TinRaster*
Create the building footprints, and add the roof patches and footprints to the ground TIN	*AggregatePolygons*, *TinEdit*, *Buffer*, *InterpolateShape*

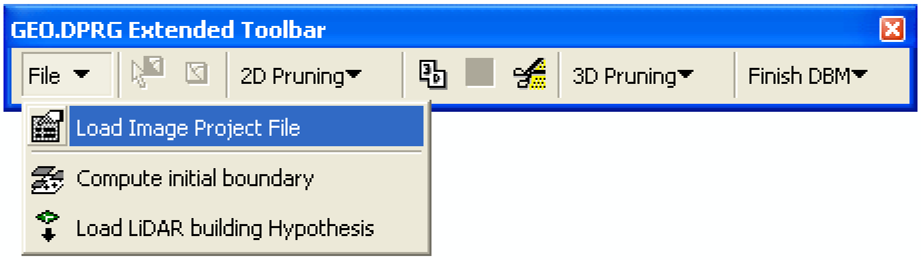


Figure 5.4 Toolbar containing links to all required functions in the proposed methodology.

For the area of interest (Figure 5.5a), the menu option takes the input LiDAR point cloud (Figure 5.5b) and classifies the points into ground and non-ground points (Figure 5.5c), generates a building hypothesis (Figure 5.5d), performs segmentation on the building hypothesis to obtain clustered building-primitives (Figure 5.5e), and finally obtains initial boundaries of the building-primitives (Figure 5.5f). The initial building-primitives are then loaded into the ArcGIS interface, in a data frame that is automatically labelled *Object space*. Figure 5.6 displays the polygons of the initial building-primitives for the hypothesized buildings in the dataset.

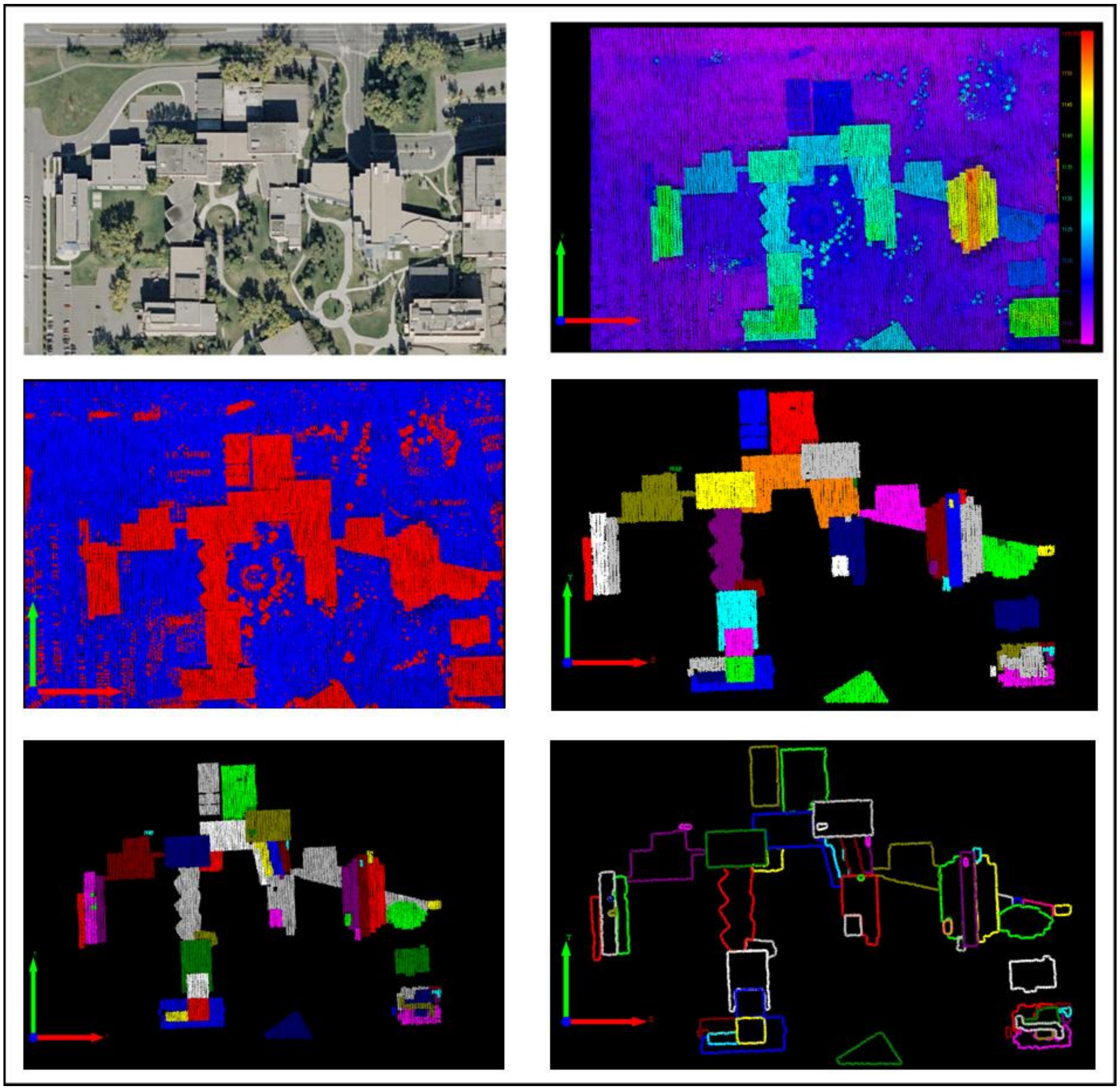


Figure 5.5: Areal image of the area of interest (a), the LiDAR data over the area of interest (b), the classified ground (blue) and non-ground points (red) over the same area (c), the generated building hypotheses (d), the clustered building-primitives (e), and the initial boundaries of the building-primitives (f).

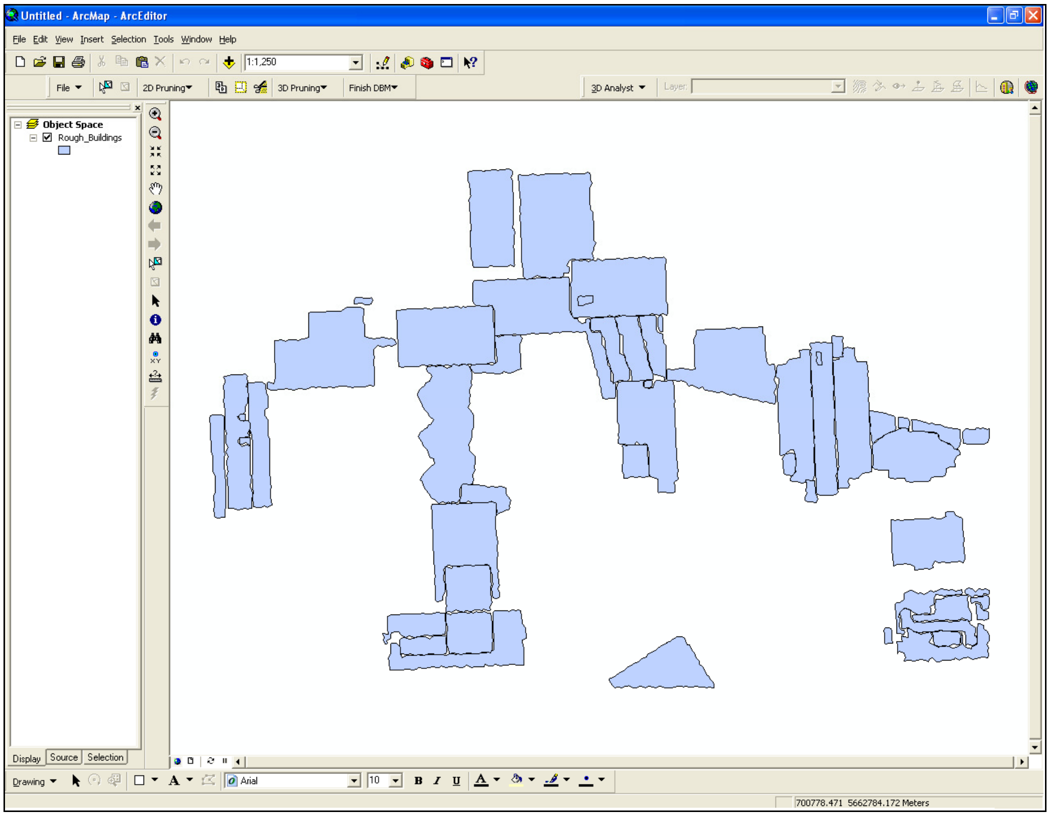


Figure 5.6: Display of polygons of the initial building-primitives.

As outlined in the explanations in Chapter 4, the user will start the DBM generation process by selecting a polygon from the interface display. Figure 5.7 shows the selection of a building-primitive in the software interface. The threshold values that were utilized in the experiments presented in this chapter are summarized in Table 5.3. Once a building-primitive is selected by the user, an option on the drop-down menu is chosen to extract lines from the best perspective image for this building-primitive. When this option is selected, Canny Edge detection is performed to locate the edge pixels in the perspective image. An image buffer is then created and used to locate the edge pixels that are located near the building-primitive boundary. Edge tracking (or chaining), edge splitting, line fitting, and line merging is performed automatically on the edge pixels located in the buffer area, and the lines obtained from this process are displayed in a new map, that will be called *Image space*. Figure 5.8a show the results from the Canny Edge detection on the chosen perspective image, and Figure 5.8b shows the extracted lines from the established buffer surrounding the initial LiDAR boundary after its projection into the image space, displayed on a clip of the building-primitive in the perspective image. In Figure 5.9a, the user has manually added lines that were not detected automatically. In particular, two added lines are circled in green in Figure 5.9a. The reason these lines were not detected automatically is that these segments of the boundary are composed of small sub-parts, as shown in a close-up view in Figure 5.9b. In Figure 5.9c, precise boundary segments are selected (highlighted in blue) and are ready to be projected into the object space map (through monoploting) for further processing. Please note that throughout this section, in some cases the images will be rotated to improve the display within this document.

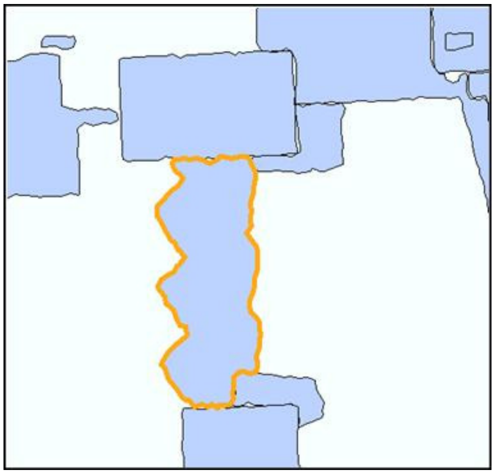


Figure 5.7: Selection of a building-primitive of interest, shown outlined in orange in the interface.

Table 5.3: Descriptions and values for the thresholds used for the experimental results contained in this chapter.

Threshold	Description	Value
Image Buffer	Size of the buffer on each side of the initial boundary, used for edge filtering.	14 pixels (0.8m in object space)
Douglas-Peucker Threshold for Edge Splitting	Threshold value used to determine if tracked edges should be split.	15 pixels
Line Merging Angle Constraint	Acceptable angle between neighbouring/ overlapping line segments in the line merging process.	10 degrees
Line Merging Normal Distance Constraint	Acceptable normal distance between neighbouring/overlapping line segments in the line merging process in the image space.	2 pixels
Line Merging Endpoint Separation Constraint	Acceptable endpoint separation of neighbouring line segments in the line merging process in the image space.	10 pixels
3D Snapping Threshold	Acceptable endpoint separation for intersection 3D lines.	2m
Douglas-Peucker Generalization Threshold	Threshold value for generalizing a line segment.	0.3m

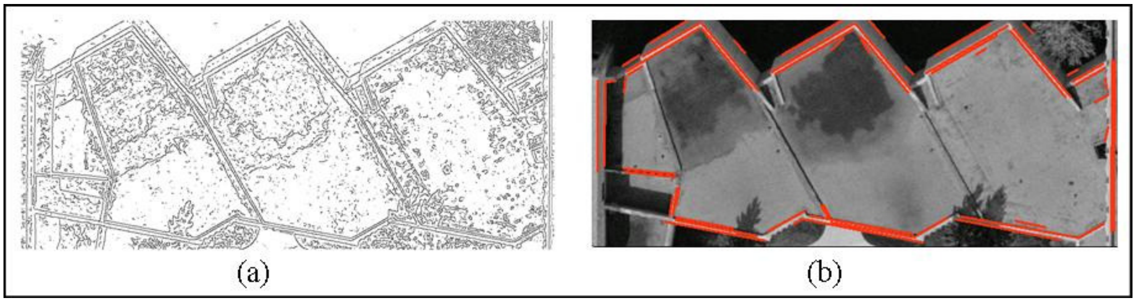


Figure 5.8: The results from the Canny Edge detection on the perspective image (a), and the line segments obtained from line extraction in the image buffer that was applied to the Canny output image (b).

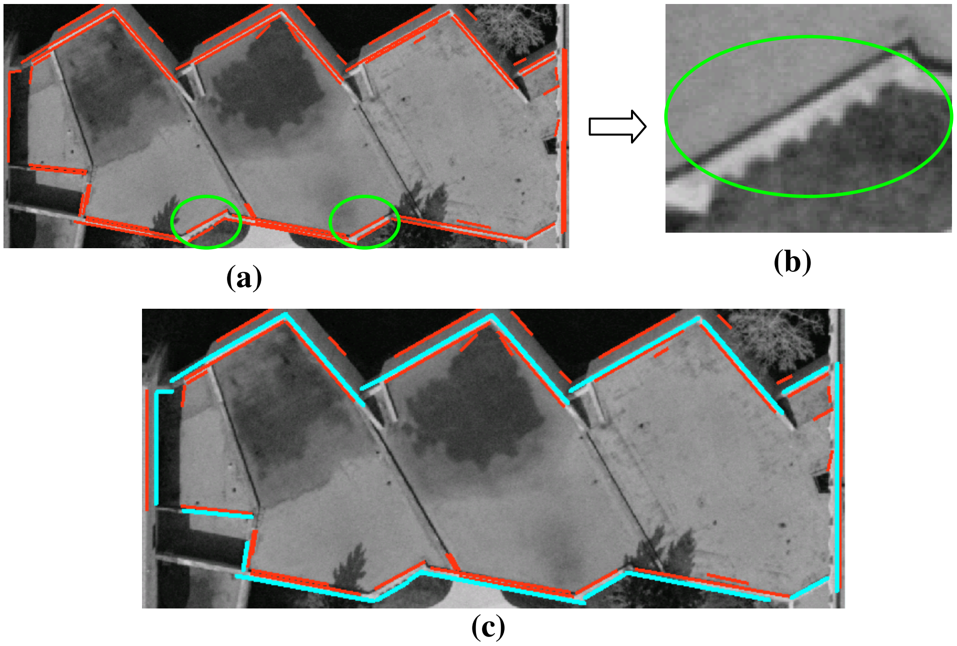


Figure 5.9: Line editing in the image space, where two manually added lines are circled in green (a), a close-up view of one of the boundary portions that was not detected automatically (b), followed by manual line selection of the precise boundary segments (c).

The selected image lines have been projected into 3D and are displayed on top of the LiDAR-derived initial building-primitive polygon in Figure 5.10. The gaps between the building-primitive lines must be closed, and thus the 3D pruning/snapping option is used for this building-primitive. Figure 5.11 shows a sequence of steps: Figure 5.11a shows the 3D building-primitive boundary lines after the lines have been project from image space to object space; Figure 5.11b shows the resulting lines after the 3D snapping option is selected; and Figure 5.11c shows the final building-primitive polygon that is obtained. Through this process, all the precise boundary segments for this building-primitive are extracted from a single image.

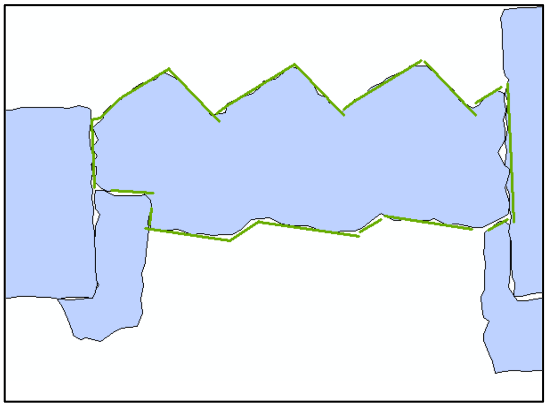


Figure 5.10: Projected image lines into 3D (green lines) on top of the initial building-primitive polygons (blue).

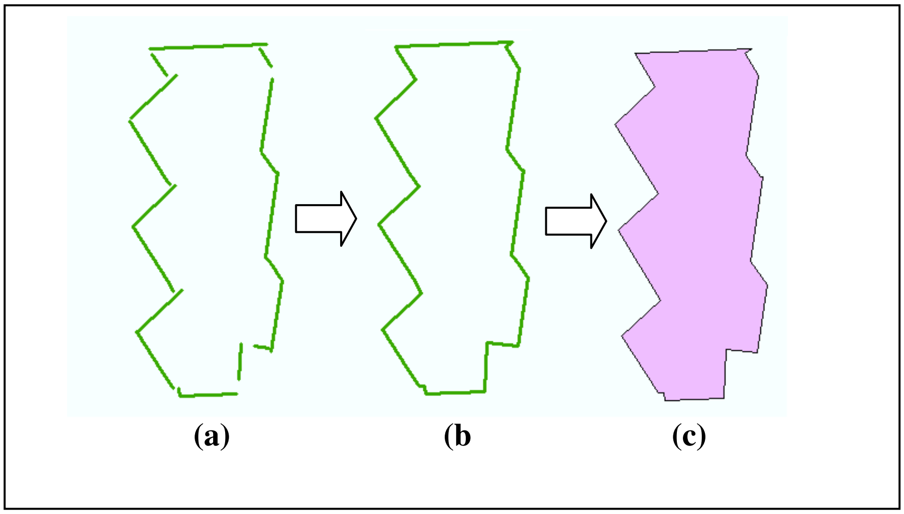


Figure 5.11: The boundary lines after projection into 3D (a), after using the 3D snapping option (b), and finally the closed building-primitive polygon (c).

The same process is repeated for a second building-primitive, shown in Figure 5.12. Figure 5.12a shows the building-primitive selected in the ArcGIS interface, and Figure 5.12b is the output from running the Canny Edge detection on the best chosen perspective image. The extracted lines from the established buffer surrounding the initial LiDAR boundary after its projection into the image space are displayed in Figure 5.12c. A portion of a boundary segment has not been detected automatically, due to low contrast near the building-primitive boundary. This area is circled in green in Figure 5.12c and 5.12d. The user has chosen to not manually digitize this gap in the image, and has selected the lines that are to be projected into 3D (highlighted in blue in Figure 5.12d).

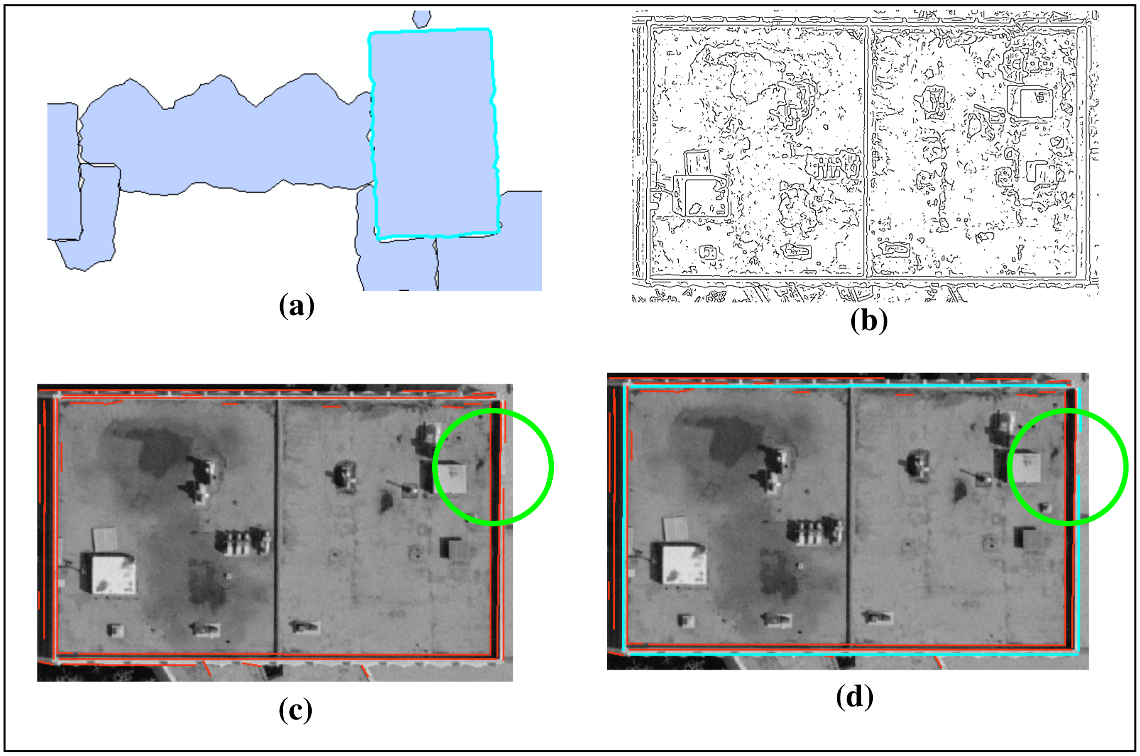

(a) (b) (c) (d)

Figure 5.12: Selected building-primitive in the object space (a), the Canny output image for the building-primitive (b), the extracted lines located within the image buffer, with a boundary portion that was not automatically detected circled in green (c), and the selection of precise building-primitive boundary segments (d).

Figure 5.13 shows the projected lines from image space into 3D. Figure 5.13a and 5.13b show the 3D lines before and after the snapping option is used. There is still a gap in the boundary of the building-primitive in Figure 5.13b. Although the option to select another image from which to extract the boundary lines is available at this point, from analysis of the LiDAR-derived building-primitive that is viewed in Figure 5.13c, it is clear that this gap should be closed by simply adding a straight line segment in this area. Therefore, manual editing in 3D is performed by the user, using the LiDAR-derived initial boundary as a guide to manually digitize the missing line segment. Figure 5.13d shows the boundary lines after manual editing is performed to obtain a closed building-primitive outline. The polygon obtained for this second selected building-primitive is shown alongside the first building-primitive in Figure 5.13e.

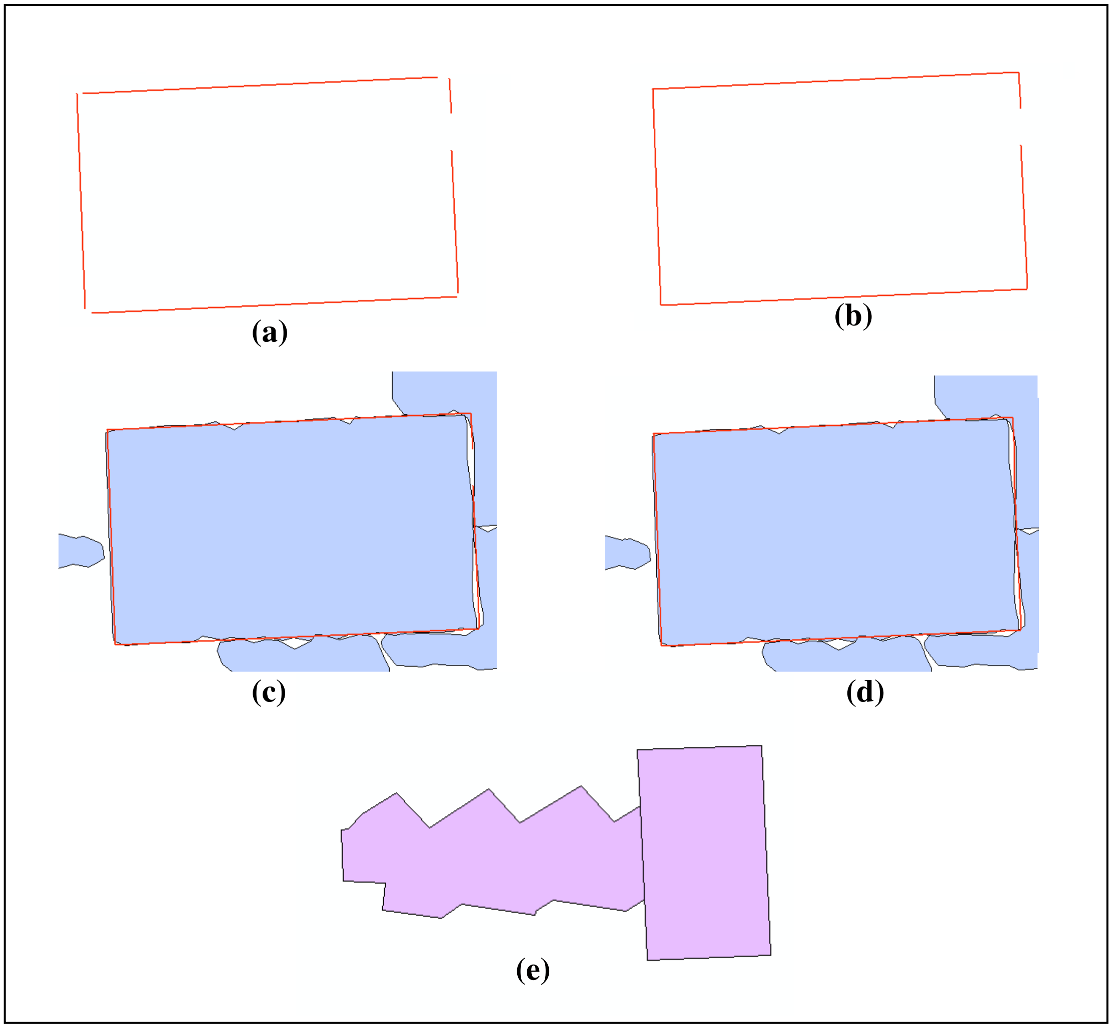
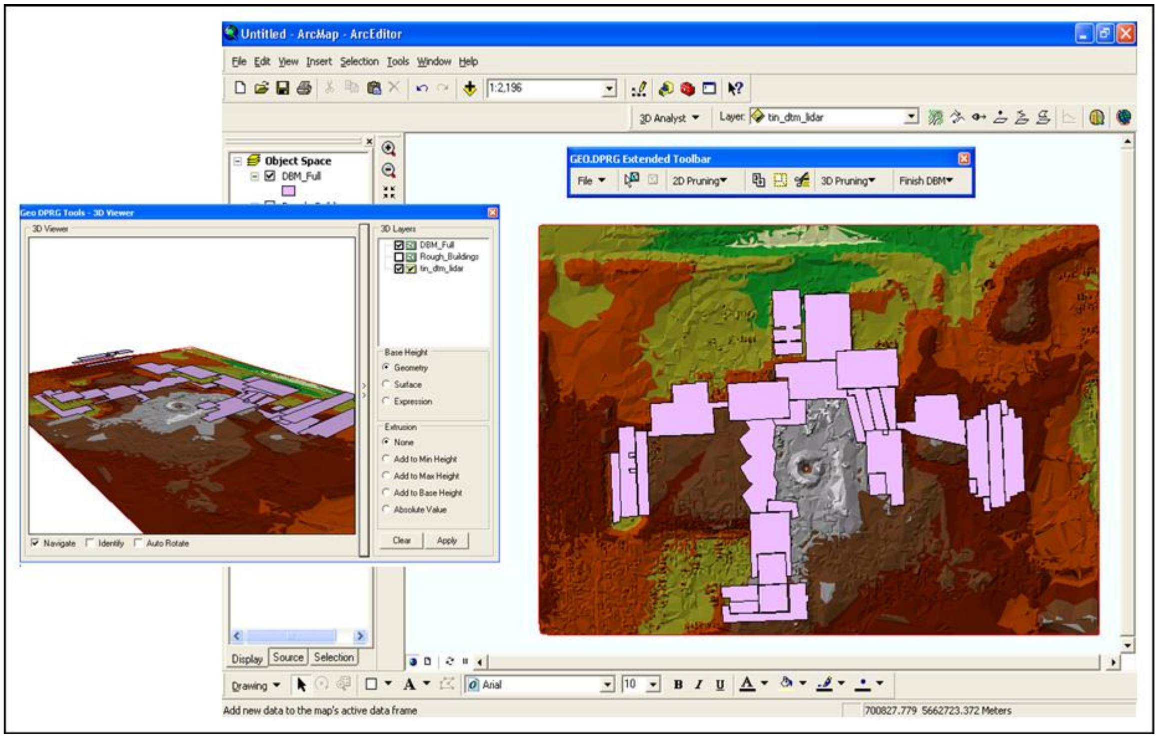


Figure 5.13: The projection of the image lines into the object space (a), the results after using the 3D snapping option (b), LiDAR-derived building-primitive used as a guide for manual digitizing in 3D (c), the results from manually digitizing the missing line segment (d), and the two polygons obtained from the selected building-primitives (e), which are shown in Figure 5.8 and Figure 5.12.

Once all building roof polygons are obtained, the user can select an option in the custom menu bar to add the roof polygons to a ground TIN, which can be used to export a refined DSM. The function that performs this operation utilizes several ArcGIS geoprocessing techniques and analysis tools (aggregation, buffer, etc.). In particular, an ArcGIS script available online in Visual Basic was converted into C#, with some slight modifications, and employed in Habib et al. (2008c) for DBM generation. Figure 5.14 shows the ArcGIS interface with the building roof polygons and TIN displayed, both in the main ArcGIS window and a custom 3D Viewer dockable window. The 3D Viewer window can be used at any time to view the GIS data layers in 3D. The resulting TIN surface model obtained by adding the buildings to the ground TIN is displayed in Figure 5.15a, and Figure 5.15b displays the created DBM in Google Earth.



☑ DBM_Full

☑ DBM_Full
☐ Rough_Buildings
☑ tin_dtm_lidar

Base Height
● Geometry
○ Surface
○ Expression

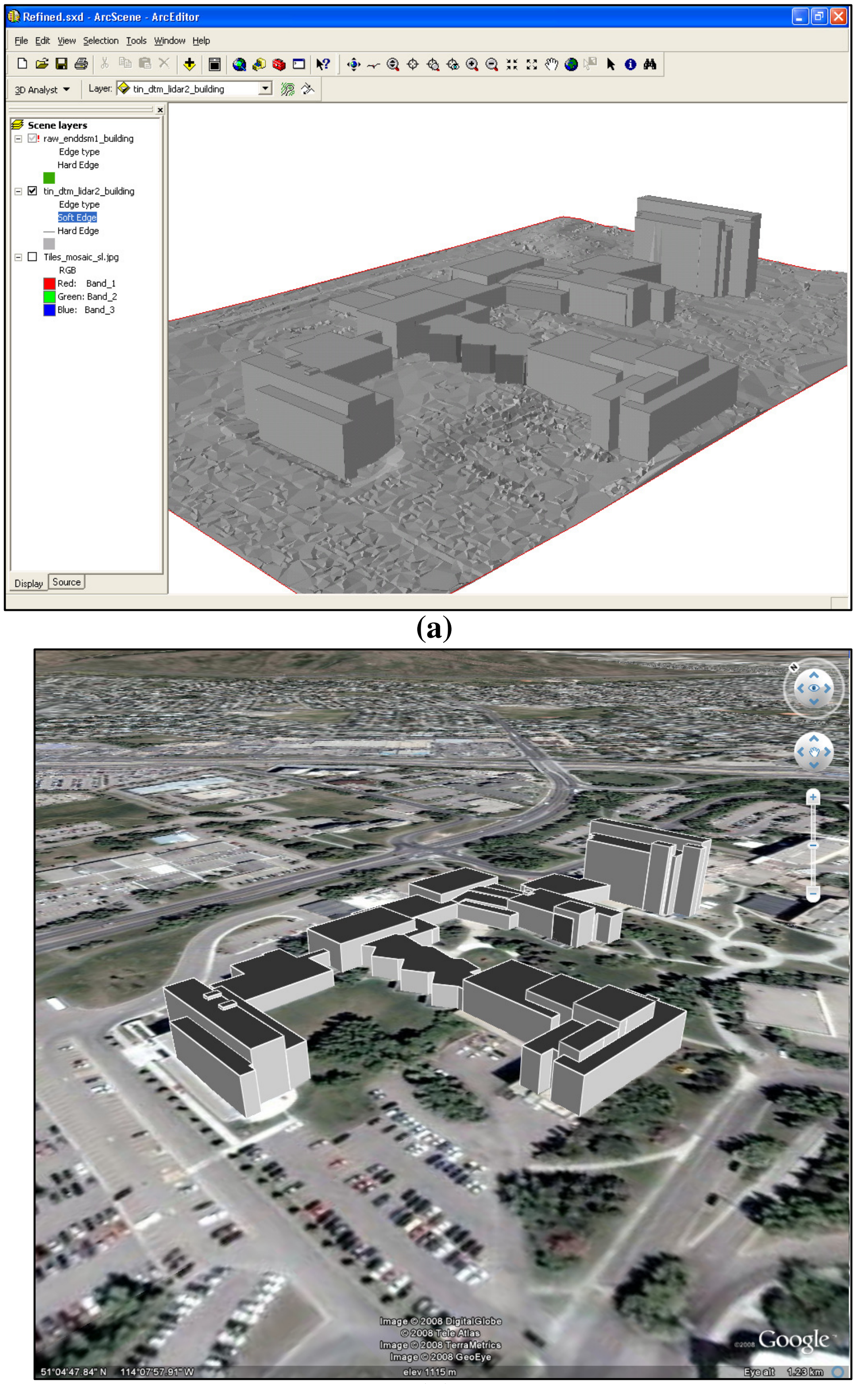
Extrusion
● None
○ Add to Min Height
○ Add to Max Height
○ Add to Base Height
○ Absolute Value

☑ Navigate ☐ Identify ☐ Auto Rotate

Add new data to the map's active data frame

700827.779 5662723.372 Meters

Figure 5.14: Building rooftop polygons obtained for the area of interest.



(a)

(b)

Figure 5.15: The resulting DBM added to a ground TIN in ArcGIS (a), and the created DBM displayed in Google Earth (b).

The TIN surface model that is obtained by adding the DBM to the ground TIN is then converted to raster. The raster is then exported to text file to obtain a refined DSM.

Improved true-orthophotos can be generated using the refined DSM, and draped on the refined DSM to obtain an improved 3D urban model. The following section conducts a qualitative analysis of the output DSM, DBM, and refined true-orthophotos. Note that the area of interest that is investigated in this work is ideal as it contains various shapes and sizes of building-primitives, and thus provides a robust test for the proposed methodology.

5.4.1 Qualitative Evaluation

This section will perform qualitative analysis and comparison between the true-orthophotos, the TIN surface model, and the 3D realistic visualization obtained using; 1) the original LiDAR-derived TIN surface model, 2) the TIN surface model derived by adding the DBM to the original LiDAR-derived TIN surface model, and 3) the refined TIN surface model obtained by adding the DBM to a ground TIN. To obtain the DSM that is required to generate the true-orthophotos, each of these TIN surface models were converted to raster data. Since the quality of the produced orthophotos is dependent upon the DSM quality, we will first investigate the three surface models shown in Figure 5.16. The TIN surface model obtained from the original LiDAR point cloud is shown in Figure 5.16a. In this figure the building boundaries are jagged, and the roof surfaces are not smooth due to various reasons (the presence of small features on the roof such as pillars, pipes, etc., as well as the vertical accuracy of the LiDAR footprints). Furthermore, the trees are represented by tall narrow cones in the surface model. It is evident from this figure that the surface model requires refinement. Figure 5.16b shows the resulting TIN surface model that is obtained by adding the derived DBM to the TIN representation of the original LiDAR point cloud. The building rooftops are now smooth, since the roof surfaces in the DBM are obtained using a plane-fitting equation, and thus look more natural. In addition, the building boundaries are crisp. The representation of the trees in Figure 5.16b is the same as in Figure 5.16a, since only the representation of the buildings in the surface model has been refined. Figure 5.16c shows the refined TIN surface model that is obtained by adding the DBM to a ground TIN. From Figure 5.16c, it is clear that the buildings look very realistic, and as the trees are no longer present in the surface model, the 3D visualization is more appealing then the 3D models shown in Figures 5.16a and 5.16b.

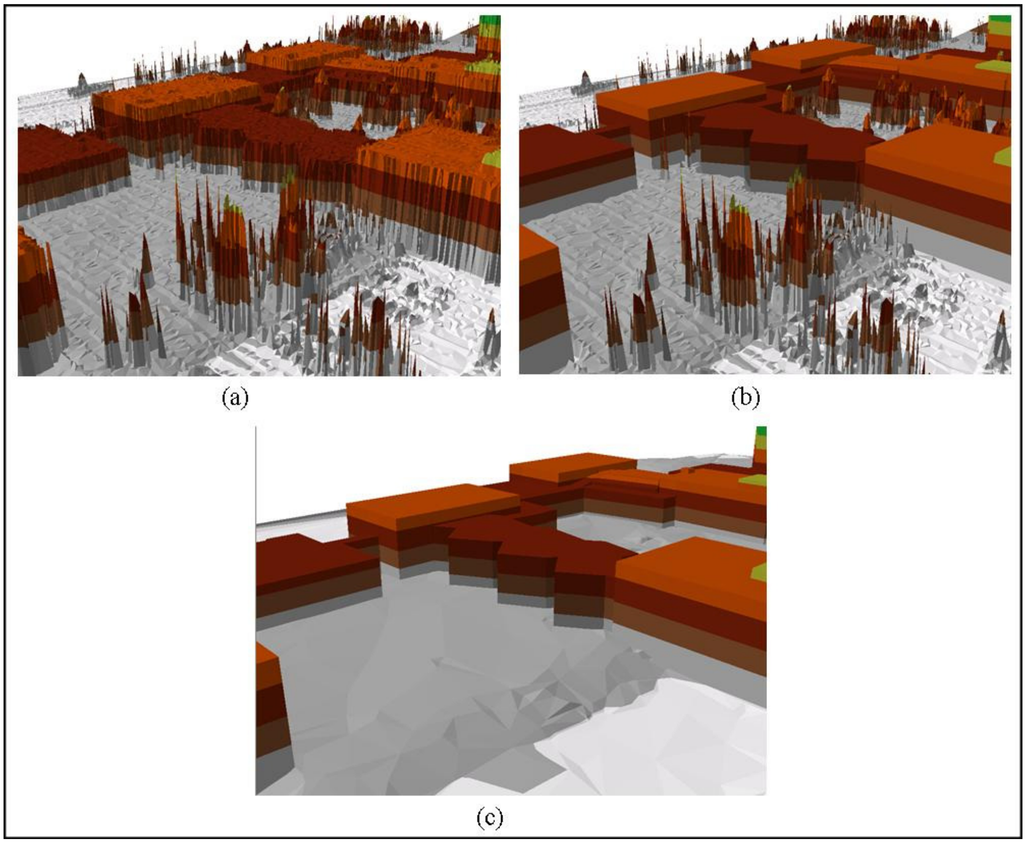


Figure 5.16: Three TIN surface models are shown: the original surface model coming from the LiDAR footprints (a), the surface model obtained by adding DBM to the original surface model (b), and the refined surface model obtained by adding DBM to a ground TIN (c).

These three TIN surface models were then each converted to raster to obtain a DSM that can be used to produce true-orthophotos over the same area. A comparison of the results is shown in Figures 5.17, 5.18, and 5.19. The original surface model obtained from the LiDAR footprints is used to produce the true-orthophoto mosaic shown in Figure 5.17a. Looking at the close-up view in 5.17b, several things are apparent. Part of the building outline has been projected onto the ground. This can be seen in the areas contained within the green ellipses in Figure 5.17b. In addition, the building boundaries are not crisp, and there are many occlusions (shown in black) in and around the trees. Figure 5.18a shows the mosaic of the true-orthophotos produced using the DSM obtained by adding the DBM to the original surface model. In the close-up view of the area bounded by the red box (Figure 5.18b), it can be seen that the building boundaries look quite crisp and there are no traces of the building boundaries on the ground, however the quality of the trees is still degraded and thus the image does not look fully realistic. Figure 5.19 shows a mosaic of the true-orthophotos created using the refined DSM, obtained by adding the DBM to a ground TIN. In this figure, the building boundaries are clearly defined, and the trees appear more natural, as they are simply projected onto the ground.

Figure 5.17: A mosaic of true-orthophotos obtained using the original DSM (a), with a close-up view of the area bounded by the red box (b).

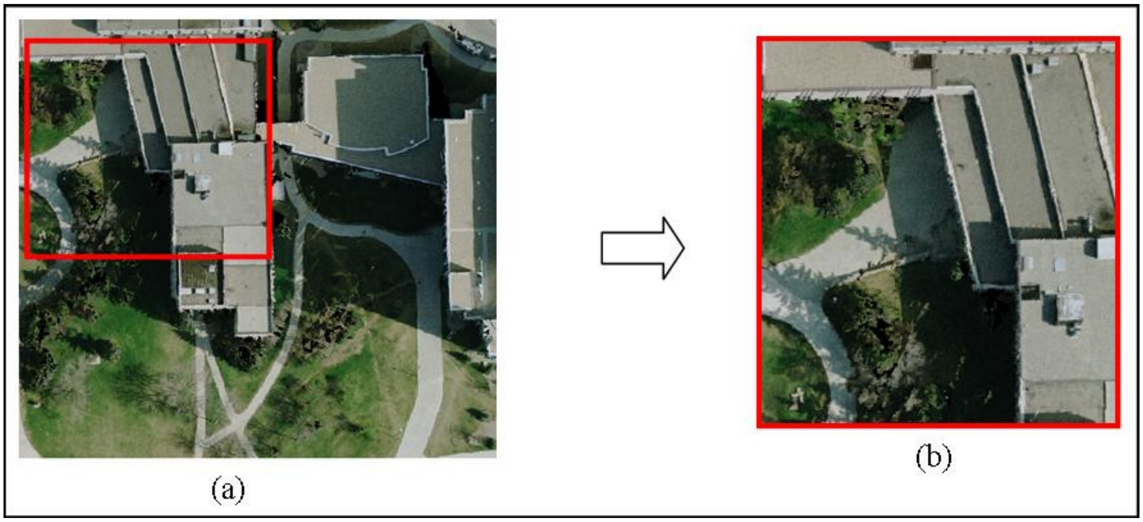


Figure 5.18: A mosaic of true-orthophotos created using a DSM obtained by adding the DBM to the original surface model (a), and a close-up view of the area bounded by the red box (b).

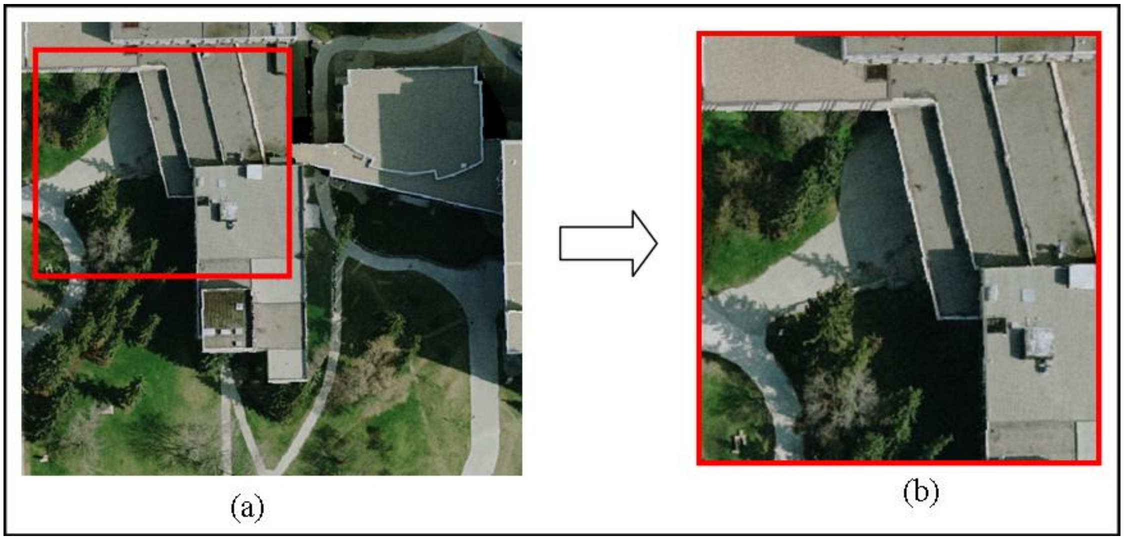
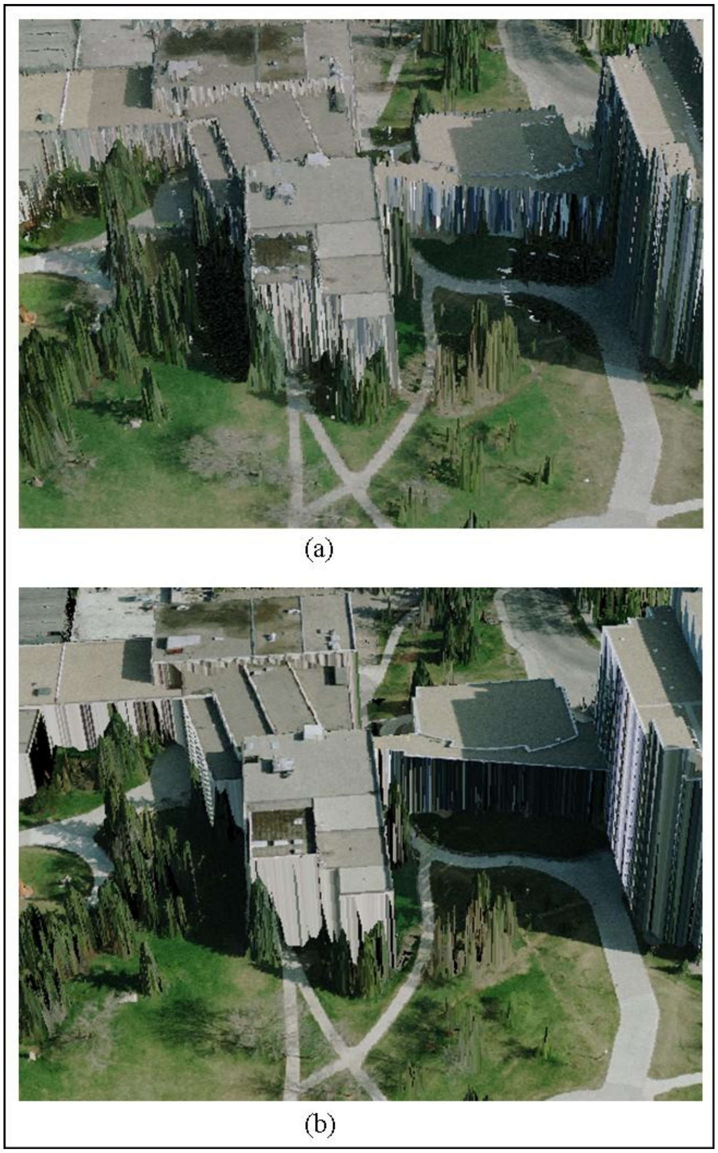


Figure 5.19: A mosaic of true-orthophotos created using the refined DSM that is obtained by adding the DBM to the ground TIN (a), and a close-up view of the area bounded by the red box (b).

Once the refined DSM and the true-orthophotos over the area of interest have been generated, the next step is to use these output products to create a 3D realistic visualization of an urban environment. This was accomplished by adding these two data layers into ArcScene, and the results are displayed in Figure 5.20 and 5.21. The 3D model obtained using the original DSM and true-orthophotos is shown in Figure 5.20a. In this image, it is evident that the building boundaries are jagged, and the trees appear as green spikes beside the buildings. Figure 5.20b displays the results obtained using the DSM derived by adding the DBM to the original TIN surface model. From this image, the building boundaries are improved, however the trees are still displayed as green spikes which cause the 3D model to be less visually attractive. Figure 5.21 shows the final 3D city model obtained using the refined DSM obtained by adding the DBM to a ground TIN. An extra step has been included to obtain the 3D model shown in Figure 5.21. A buffer “wall” around the buildings has been added, to enhance the visualization of the building walls. This was has been created by simply generating a copy of the building rooftop polygons, expanding these polygons slightly, and extruding the polygons to the ground. From a comparison of these three figures it can be concluded that the proposed procedure has successfully improved both the surface description and the texture information of an urban environment. Furthermore, the choice of adding the DBM to the ground TIN of the area has been validated through these results.


(a)

(b)

Figure 5.20: The 3D models created using the original LiDAR derived DSM (a), the DSM obtained by adding the DBM to the original DSM (b).

Figure 5.21: The refined DSM obtained by adding the DBM to the ground TIN.

To further confirm the choice of adding the DBM to the ground TIN as opposed to the TIN obtained from the original LiDAR footprints, some comparisons are shown in Figure 5.22. An area of the true-orthophotos generated using the DSM obtained by adding the DBM to the original LiDAR-derived TIN is shown in Figure 5.22a, and the refined DSM obtained by adding the DBM to the ground TIN is shown in Figure 5.22b. From these images, it can be seen that the trees in Figure 5.22a contain occlusions in and around them, whereas the trees in Figure 5.22b look more visually appealing. A comparison of trees in a second area shown in 3D is found in Figure 5.22c and Figure 5.22d. In figure 5.22c, several large trees are modeled as green spikes. The LiDAR footprints that hit the tree canopy produce the spikes, while in some cases the LiDAR footprints penetrate the canopy and return a ground point. For this reason, trees are often poorly modeled in a surface model obtained from LiDAR data. Figure 5.23 displays samples of other structures (such as cars, metal structures, and the campus entrance) on the surface that are poorly modeled by the LiDAR data, and thus appear degraded in the derived orthophotos. Figures 5.23a, 5.23c, and 5.23e are true-orthophotos produced using a DSM that is obtained by adding the DBM to the original LiDAR-derived TIN model, while Figures 5.23b, 5.23d, and 5.23f are true-orthophotos from a DSM that is obtained by adding the DBM to a ground TIN.

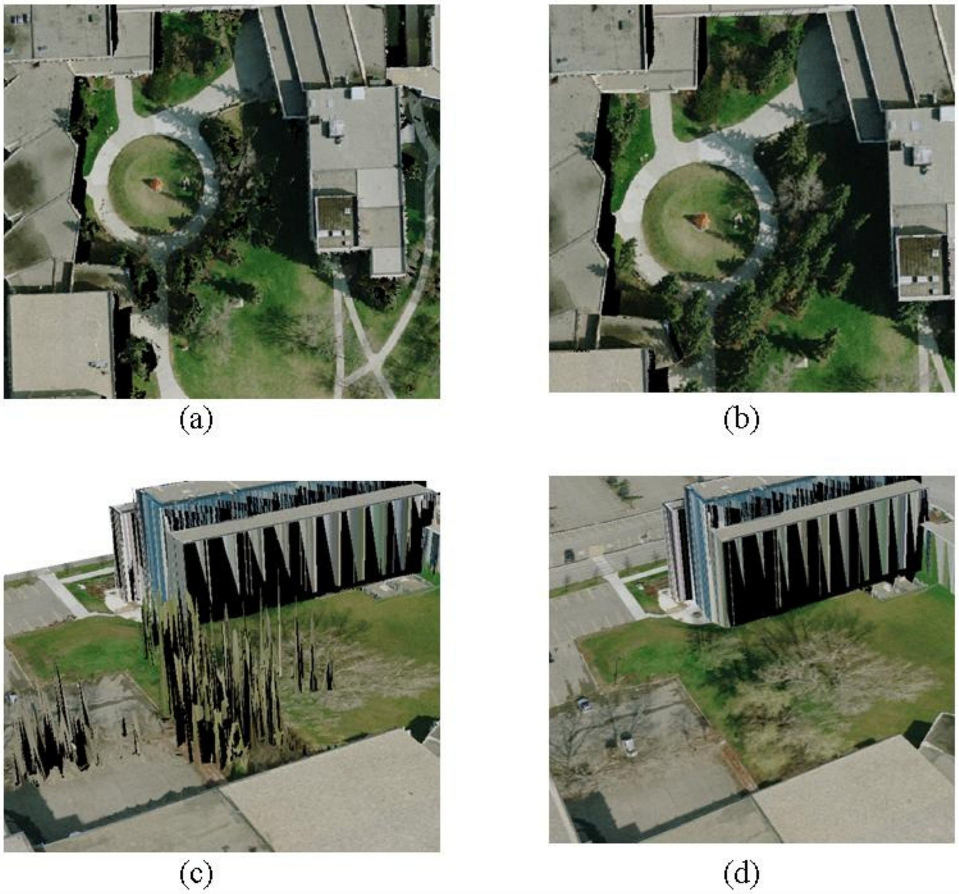


Figure 5.22: An orthophoto obtained by adding the DBM to the original TIN surface model (a), an orthophoto obtained from a refined DSM (DBM + ground TIN) (b), the 3D model obtained using the orthophoto in figure a (c), and the 3D model obtained under the orthophoto in figure b (d).

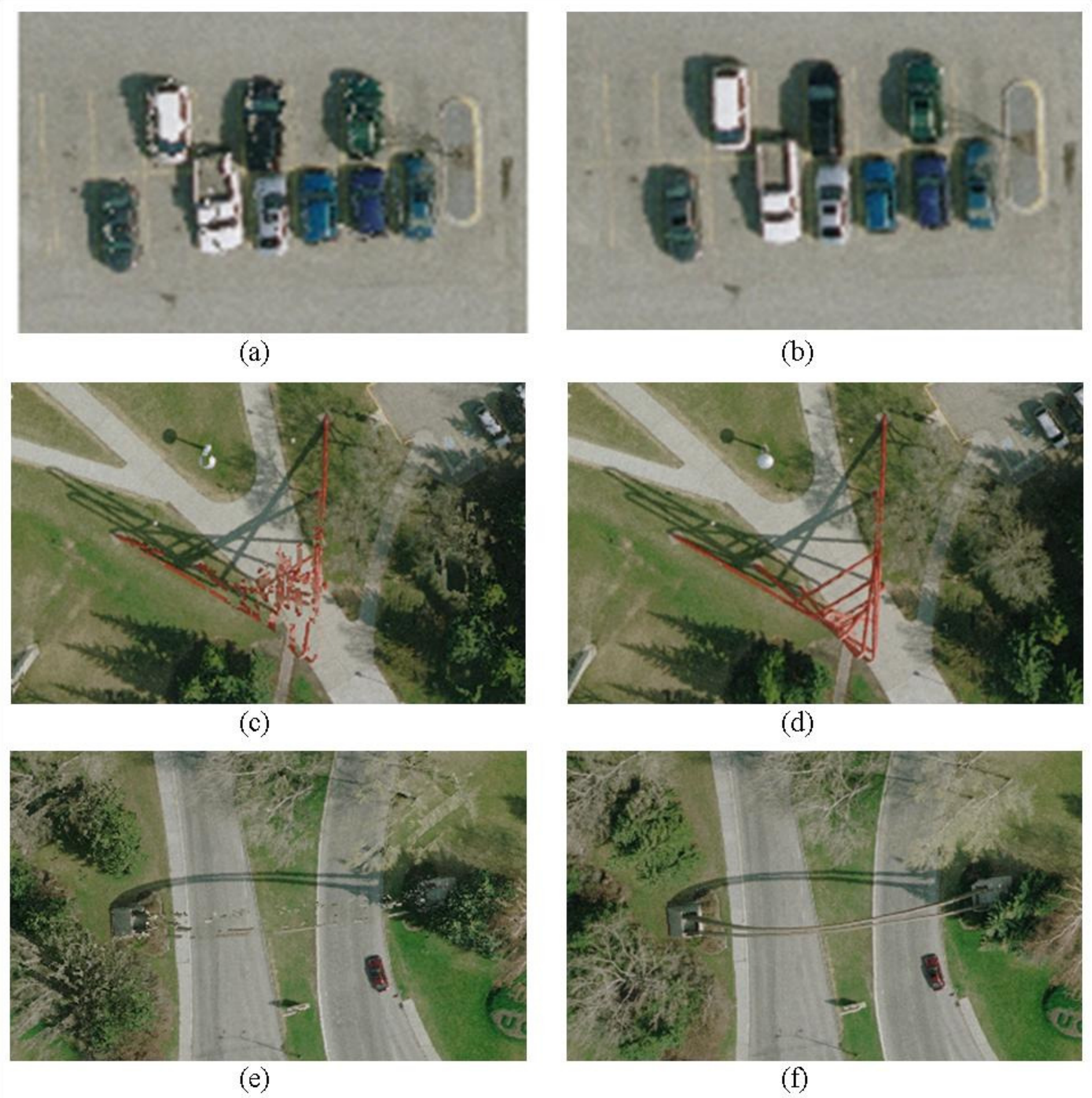


Figure 5.23: Figures (a), (c), and (e) are true-orthophotos from a DSM that is obtained by adding the DBM to the original LiDAR TIN, while figures (b), (d), and (f) are true-orthophotos generated from a DSM that is obtained by adding the DBM to a ground TIN.

From these examples, it is evident that it is a better option to simply ignore the modeling of trees, cars and other structures on the surface and simply display them on the ground. Figure 5.24 shows the final 3D urban model of the University of Calgary campus, obtained through the proposed methodology. This concludes the qualitative analysis of the produced output, and the following section will outline a quantitative analysis performed on the output DBM.

Figure 5.24: Final 3D model obtained through the proposed methodology.

5.4.2 Quantitative Evaluation

This section performs a quantitative analysis of the accuracy of the DBM produced through the proposed procedure. This was conducted by computing the RMSE of the corner points of the DBM with the corner points of a manually generated DBM, which was derived through a photogrammetric reconstruction procedure using the image dataset. Since the manually generated DBM will be used as the ground truth, it is important to assess the quality of this DBM before comparing it to the DBM produced in Section 5.4.1. To assess the consistency of coordinates derived manually, six operators independently generated a DBM from imagery. The resulting coordinates for the building corner points obtained by the six operators were used in an RMSE analysis, where the results obtained by one of the more experienced operators were compared against the other five. The computed RMSE values are displayed in Table 5.4. Looking at these results, several conclusions can be drawn. The mean values are all quite small, which indicates that there is no bias or systematic discrepancies between the six sets of measurements. The standard deviation indicates the magnitude of the random errors in the operator measurements. From the results in Table 5.4, the planimetric and vertical standard deviations range between 9cm-36cm, which is roughly 1.5-6 pixels in the imagery (recall the image GSD is 6cm), and the main cause for this difference is due to the difficulty in accurately locating the precise corner points in the imagery. The total RMSE values range from about 22cm to 56cm, and the average RMSE is around 33cm. Thus, a general conclusion that can be taken from these results is that the variability in coordinates obtained from the manual generation of a DBM is about 33cm. Table 5.5 shows a break down of the RMSE in terms planimetric and vertical components, where the average planimetric RMSE is around 0.25m and the average vertical RMSE is around 0.21m.

Table 5.4: RMSE Analysis between results obtained from six operators, where all units are in meters.

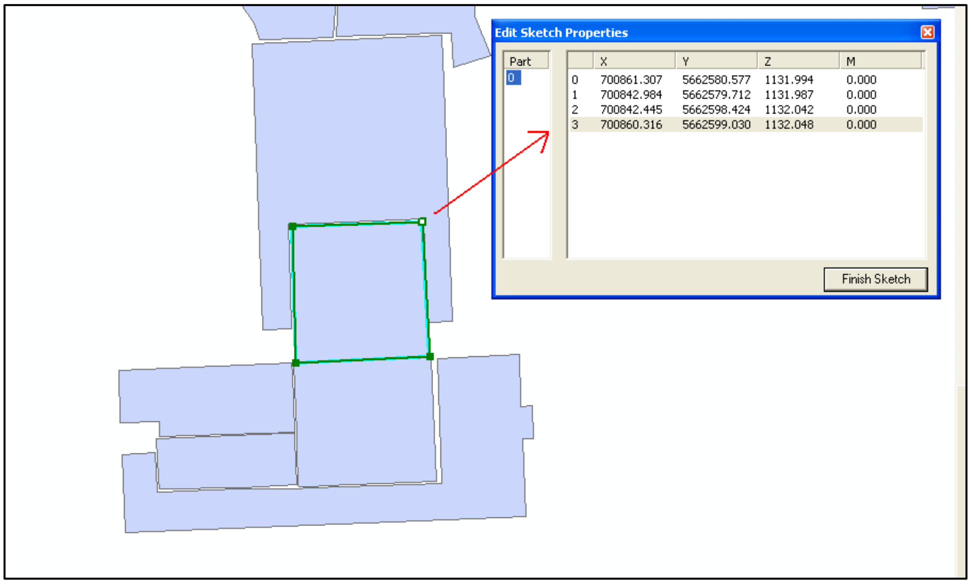
	Mean X	Mean Y	Mean Z	σX	σY	σZ	RMS X	RMS Y	RMS Z	RMS Total
Op_2 Vs. Op_1	0.010	0.043	0.003	0.119	0.132	0.192	0.119	0.138	0.191	0.264
Op_3 Vs. Op_1	0.086	0.008	0.091	0.349	0.364	0.239	0.357	0.362	0.255	0.569
Op_4 Vs. Op_1	0.027	0.035	0.021	0.172	0.124	0.210	0.174	0.128	0.210	0.301
Op_5 Vs. Op_1	0.009	0.061	0.003	0.090	0.125	0.152	0.090	0.139	0.151	0.224
Op_6 Vs. Op_1	0.013	0.034	0.038	0.149	0.124	0.252	0.149	0.128	0.253	0.321

Table 5.5: A comparison of the Planimetric and Vertical RMSE (in meters)

	Planimetric RMSE	Vertical RMSE
Op_2 Vs. Op_1	0.182	0.191
Op_3 Vs. Op_1	0.508	0.255
Op_4 Vs. Op_1	0.216	0.210
Op_5 Vs. Op_1	0.166	0.151
Op_6 Vs. Op_1	0.196	0.253

The results from the six operators were averaged and used to compute an RMSE between the manually generated DBM and the DBM generated through the proposed procedure.

The coordinates of the building corners for the DBM generated in ArcGIS are stored in the rooftop polygons and can be obtained by clicking on the building vertices (Figure 5.25).



	X	Y	Z	M
0	700861.307	5662580.577	1131.994	0.000
1	700842.984	5662579.712	1131.987	0.000
2	700842.445	5662598.424	1132.042	0.000
3	700860.316	5662599.030	1132.048	0.000

Figure 5.25: Building vertex coordinates stored in the rooftop polygon layer.

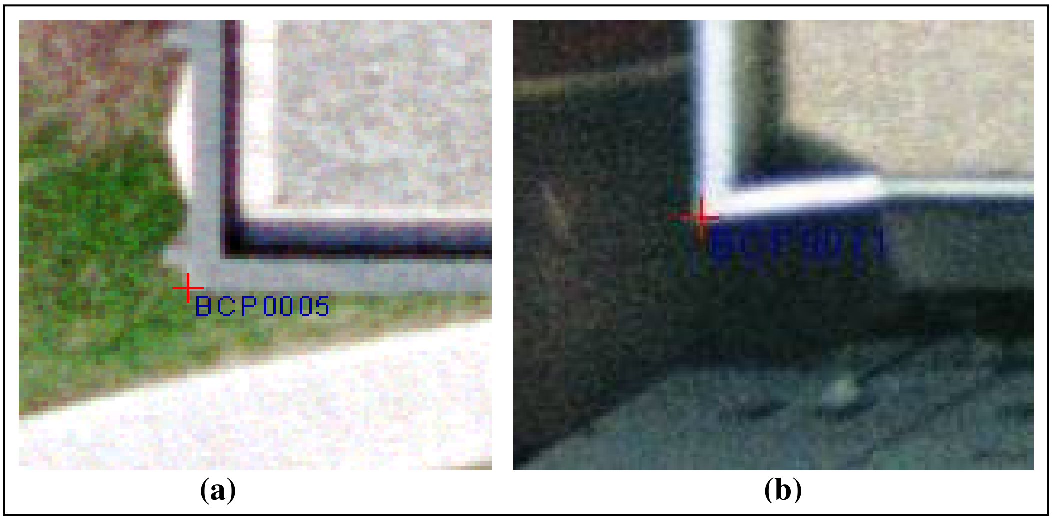
Table 5.6 lists the results from an RMSE analysis conducted to compare the DBM generated through the proposed procedure with the manually generated DBM. The second row in Table 5.6 shows the values obtained when comparing the averaged results from the six operators with the DBM generated using the developed program. The mean values in the X and Y directions are quite small, which indicates no bias between the two sets of building coordinates. The mean Z value on the other hand is quite large (about 53cm). This indicates the presence of a vertical bias between the compared DBMs. The reason for this is that the Z-coordinates of the DBM generated through the proposed procedure are obtained from the rooftop plane equation, which does not consider the height of the walls or “fences” that surround the rooftops of many buildings in the dataset. Two sample buildings from the image dataset are shown in Figure 5.26, where the fence on the building roof in Figure 5.26a has a height of around 0.45m above the roof plane, while the fence shown in Figure 5.26b has a height greater than 1m above the roof plane. The Z-coordinates obtained manually from imagery, in contrast to vertices for the generated DBM shown in Figure 5.25, were obtained for the outer-top corners of each building, and thus the points selected in imagery were located on top of these fences. The reason for this type of point selection in the imagery was due to the fact that the outer walls are more clearly visible in overlapping imagery. Returning to the analysis of the second row in Table 5.6, the standard deviations in the X, Y, and Z directions are 35cm or less, while the RMSE in X and Y are around 20cm. These RMSE values are in the same range as the RMSE values between the different operators shown in Table 5.4. This is therefore an indication of a very good accuracy for the generated DBM. As expected, the RMSE Z is quite large due to the large Mean Z value. In the third row of Table 5.6, the compared manual DBM as well as the DBM generated through the proposed method were both derived by the same operator, and the results are quite similar to those in the second row of the table. Table 5.7 reports the planimetric RMSE values obtained for the semi-automatically generated DBM. These values are consistent with the average planimetric RMSE obtained from the manual DBM generation results, which was 0.25m. In Chapter 2, results from Elaksher et al. (2003) were outlined. In their work, multiple images were used for automated DBM generation. The reported planimetric accuracy was 0.9m, for an image GSD of 12cm. The results from the semi-automated DBM generation reported in Table 5.7 reveals that the proposed method has improved upon this accuracy by about a factor of three, and thus the results are very satisfactory.

Table 5.6: RMSE analysis of the semi-automatically generated DBM (in meters)

RMSE	Mean X	Mean Y	Mean Z	σ X	σY	σZ	RMS X	RMS Y	RMS Z	Total RMS
OP_1 DBM vs. Avg Manual	-0.077	0.094	0.537	0.221	0.179	0.352	0.233	0.202	0.642	0.712
OP_1 DBM vs. OP_1 Manual	-0.057	0.064	0.567	0.275	0.160	0.402	0.280	0.172	0.693	0.768

Table 5.7: Planimetric RMSE analysis for the semi-automatically generated DBM (in meters)

	Planimetric RMSE
OP_1 DBM vs. Avg_Manual	0.308
OP_1 DBM vs. OP_1 Manual	0.329



(a) (b)

Figure 5.26: Manually selected vertices for the photogrammetrically reconstructed DBM, with a building fence height of 45cm (a), and fence height of over 1m (b).

To assess the repeatability of the proposed DBM generation procedure, the DBMs generated by the same operator are compared in the second row of Table 5.8, and the third row compares two DBMs that are generated by two different users. The mean values for both rows are quite small, indicating no systematic discrepancies between the compared DBMs. The standard deviations and RMSE values are all within the same ranges as those obtained by the six operators in Table 5.4. This indicates that a resulting DBM obtained from the proposed procedure, performed by the same operator, will obtain very similar results when repeating the DBM generation process, and that the resulting DBM is independent of the operator. One difference worth noting in Table 5.8 is the mean, standard deviation, and RMSE values for the derived Z-coordinates, which are smaller than the respective values for the X and Y coordinates. The explanation for this is that regardless of the person generating the DBM, the Z-values for a given building-primitive will always come from the same plane equation, and therefore are expected to be very similar.

Table 5.8: RMSE analysis to assess repeatability of proposed methodology, where the second row compares two DBMs generated by the same user with a time gap of over two weeks, and the third row compares two DBMs generated by two different operators.

RMSE	Mean X	Mean Y	Mean Z	σ X	σY	σZ	RMS X	RMS Y	RMS Z	Total RMS
OP_1 DBM1 vs. OP_1 DBM2	-0.034	-0.043	-0.009	0.178	0.207	0.029	0.180	0.211	0.030	0.279
OP_1 DBM vs. OP_2 DBM	0.065	0.029	-0.00	0.328	0.148	0.001	0.333	0.150	0.001	0.364

To perform a quantitative analysis on the produced true-orthophotos, the GPS-surveyed points used in the experiments outlined in section 5.3 can be used. In particular, an RMSE between GPS control points and the planimetric coordinates of the GCP located within the true-orthophoto mosaic can be computed. Since LiDAR-derived linear features were used as the source of control for the image georeferencing, it is expected that the results of the RMSE analysis should contain the bias that was observed in section 5.3. Only two GPS-surveyed GCP were visible in the mosaic of the area of interest, and future work will target the expansion of the mosaic for a more comprehensive RMSE analysis. Nonetheless, the differences in the X and Y coordinates of these two points can be discussed. The computed differences in the planimetric coordinates of the first point were 0.51m and 0.27m, in X and Y respectively. For the second point, the X and Y coordinate differences were 0.38m and 0.87m. These large values confirm the bias between the GPS and LiDAR reference frames that was previously detected (i.e., recall that the $\boldsymbol{Mean_{\Delta X}}$ and $\boldsymbol{Mean_{\Delta Y}}$ reported in Table 5.1 were 1.0m and 0.76m, respectively).

5.5 Summary

Through the qualitative and quantitative analyses performed on the DBMs generated through the proposed methodology, several conclusions can be summarized. The developed software that is integrated within the ArcGIS platform allows a user to generate a DBM through the use of a single custom toolbar. In addition, several existing tools available within the GIS library are utilized, and the output products benefit from the display and visualization capabilities of the GIS. In terms of the qualitative analysis, it was concluded that the addition of the DBM to a ground TIN produced the most appealing visualization of an urban area. Furthermore, the true-orthophotos obtained using the refined DSM show accurate and crisp building boundary delineation. The quantitative analysis has shown that the accuracies achieved by producing a DBM through the proposed procedure are comparable to those achieved using traditional manual photogrammetric techniques. Furthermore, the produced DBM has been shown to be operator independent. For these reasons it can be confidently concluded that the proposed methodology and developed software can be utilized to obtain an accurate 3D realistic visualization of an urban environment. The following chapter outlines the conclusions, limitations, and areas of future work.

CHAPTER SIX: CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

6.1 Introduction

This work has outlined the integration of photogrammetric and LiDAR data, within a GIS, for the accurate reconstruction of a 3D realistic urban model. Moreover, through the integration of these two data sources, in a semi-automated procedure, accurate boundaries of the building-primitives can be obtained from a single image. This chapter will summarize the work that has been done, comment on the limitations, and propose areas of future work.

6.2 Conclusions and Limitations

To avoid social and environmental problems that can arise from rapid urbanization, accurate and current geospatial information is required in a form that is simple to understand and analyze. Moreover, there is a need for practical and accurate tools for exploiting geospatial data acquired by modern remote sensing systems. Urban environment reconstruction can be a difficult procedure due to the complexity of the involved scenes: dense buildings, variety in building types, etc. The proposed DBM generation procedure aims at providing a solution to the often complex task of 3D urban model reconstruction, to cater to the growing needs for accurate and realistic 3D visualizations of urban environments. In particular, the developed program allows a user with little or no prior experience in DBM generation or in the use of ArcGIS geoprocesing techniques to create an accurate and realistic 3D city model. Although there are some current 3D visualization platforms, such as Microsoft Virtual Earth and Google

Earth that are freely available, there are significant differences between these platforms and the developed ArcGIS program. For instance, the developed program allows the user to input their own LiDAR and image data and perform the DBM generation process from start to end all within the same program. In addition, the accuracy of the two prior mentioned commercial platforms are not of engineering quality, as their main purpose is to provide a visually pleasing 3D model. Figure 6.1 shows an area of Chicago taken from Google Maps that displays buildings that are “leaning into” one another. This quality of model would not be suitable for flood simulations or engineering design projects, and thus there is a need for products such as the GEO.DPRG.TOOLS.EXTENDED program that has been developed in this work, for numerous and various applications.

Figure 6.1: Noticeable inaccuracies in Google Earth building models.

A review of the topics discussed in each chapter and their key findings will now be summarized. It has been outlined that to obtain an accurate and realistic reconstruction of an urban environment, the techniques for DSM, DBM, and true-orthophoto generation must be investigated. Therefore, a review of previous work conducted in the areas of orthophoto, DSM, and DBM generation were conducted. It was found that many existing true-orthophoto generation techniques did not perform very well in urban environments, and the new angle-based techniques proposed in Habib et al. (2007a) were selected as the best option, as the performance of such techniques does not require a DBM of the area to correctly identify occluded DSM cells, and are independent of the DSM cell size and image GSD. Moreover, the angle-based techniques do not depend on the attitude/orientation of the involved image. Regardless of which true-orthophoto generation procedure is used, the result is affected by the quality of the DSM, in terms of positional accuracy and detail. A review of different DSM generation procedures was thus investigated, but it was found that a DSM obtained from LiDAR data did not sufficiently model the buildings in an urban environment while an image-based DSM contain problems regarding the level of automation. Above ground features such as trees and vegetation are often not included in a DSM, and it is common for a DSM to be composed of only the ground and buildings. Therefore, the procedures for enhancing the DSM target the refinement of the description of the buildings in the surface model, through DBM generation procedures. If the buildings can be accurately modeled and added to the DTM of the area, a refined DSM can be obtained for an urban area, and thus improved true-orthophotos can also be produced for the same area. A review of the different data sources used for DBM generation in previous literature was outlined, and it was found that due to the complimentary nature of LiDAR and photogrammetric data, an integration of these two sources will yield a more accurate and complete DBM. The use of these two sources required several pre-requisites. To insure a high quality of output products (orthophoto, DSM, DBM), QA and QC procedures must be performed on the photogrammetric and LiDAR systems and derived products. Following a discussion of the QA/QC procedures, the co-registration of the two datasets was reviewed. When integrating data sources, it is required that they are registered to a common reference frame. Some experimental results were provided from an independent study which revealed that LiDAR-derived control features produced the best results for georeferencing the involved imagery relative the LiDAR reference frame.

The proposed integration of LiDAR and photogrammetric data, within a GIS platform, was then outlined in detail, where the customized modifications of the ArcGIS functionality were performed both in ArcObjects and C#. It was found that by implementing the proposed procedure through the creation of a custom toolbar within ArcGIS, several benefits could be obtained. The powerful functionality available within ArcGIS was modified to suite the desired application, and thus not every algorithm that was used had to be fully implemented from scratch. In addition, the visualization capability of ArcGIS could be utilized. Furthermore, many users are already familiar with this popular GIS platform, and thus the use of the developed product would be very intuitive to many new users. Previously developed programs for the classification and segmentation of LiDAR data were added into the custom toolbar, and thus no previous knowledge about the individual programs these procedures came from is required by a user. The developed program implements the DBM generation procedure through a semi-automated process. In the first step, the initial building-primitives are obtained from LiDAR, through LiDAR data classification and segmentation, and displayed to the user. The user can then select a given building-primitive, which commences a series of steps. First the best image from which to extract the boundaries of the selected building-primitive is located, and image processing techniques are applied to the image, after which extracted lines are displayed for the user. These extracted lines represent the detected line segments located within a buffer distance from the initial LiDAR-derived building-primitive boundary. The user then manually selects the line segments they wish to project into 3D. If any lines are visible in the image but have not been located automatically, the user has the option to manually digitize any missing lines. If this process had been fully automated, these missing lines would not have been located, and thus this user interaction insures that the reconstructed buildings are as complete as possible. After manual editing is performed on the image, the selected line segments and digitized lines are projected into 3D through a monoploting procedure. Through this process, 3D information is obtained from a single image. After inspecting the 3D lines, if any lines are missing the user has an option to extract lines for the given building-primitive from a second image. If this option is chosen, the 3D lines are projected onto the second image, along with the detected lines from the second image. The user again has the option to perform manual editing in the image space, and then must select which lines to project into 3D. By allowing the user to check through multiple images for any missing boundary lines, it can be assured that the utmost number of boundary lines is extracted. A snapping option can be selected, in the object space, to close any gaps between neighbouring lines. In addition, if after checking through every image in the dataset there still remain some missing building boundaries, the user can manually digitize the boundaries in 3D while using the LiDAR building-primitive as a guide, after which the final building-primitive rooftop polygon is obtained and displayed in the GIS interface. Once all building rooftop polygons are obtained, a function is used to add the polygons as buildings to a DTM. It was concluded that in order to avoid distracting displays of above-ground structures, it was best to add the DBM to a ground TIN, and not the original DSM. This produces a refined DSM of the urban model, and is exported to be used by the angle-based true-orthophoto generation procedure to obtain improved orthophotos over the urban area. When the improved orthophotos are draped over the refined DSM, a 3D realistic urban model is obtained.

Qualitative and quantitative analyses were then performed on the produced 3D model. In the qualitative analysis, it was found that the ortho-imagery obtained using the refined DSM produced accurate results: no traces of buildings were found on the ground, and the building boundaries were clearly defined. In addition, since the DBM was added to a ground TIN as opposed to a TIN surface model, there were no distracting displays of inaccurately depicted above-ground features, such as trees, cars, etc. It was found that the produced DSM and ortho-imagery was a significant improvement on those obtained using the original LiDAR-derived DSM, as well as the results obtained when the DBM was added to the original DSM. The results from the quantitative analysis found that the generated DBM was equivalent to the results obtained by manually generating a DBM from imagery. Furthermore, it was shown that the generated DBM was not dependent on the operator, which is an important aspect for any developed system. The proposed methodology has thus outlined a successful approach for the detection and reconstruction of complex buildings, through a small amount of user interaction, through the integration of the developed program in ArcGIS.

Although the proposed methodology has provided satisfactory results, there are some limitations that are now discussed. The first limitation is the level of automation, in particular in obtaining a closed building-primitive boundary in the object space. The current 3D pruning function joins neighbouring lines based only on a threshold. Therefore, in many cases when there is a large gap between neighbouring lines that should be joined, these lines are currently not joined. If the threshold for joining lines based on distance separation is increased, however, some lines may be incorrectly joined. A possible solution for this limitation is discussed in section 6.3. A second limitation occurs when a boundary of a building-primitive is not visible in any of the perspective images. In the current implementation, manual digitizing must be performed in 3D to obtain this line segment. Although the initial LiDAR-derived building-primitive boundary can be used as a guide for manual digitizing in 3D, the digitized boundary may not be very accurate. Section 6.3 will also discuss a suggestion to solve this problem.

6.3 Future Work

One of the more noticeable areas of future work that is visible in the final 3D model is the wall texture. In the current program, no wall texturing is performed. As a result, a uniform coloured buffer wall is simply created around the roof polygons.

Texture information is vital to produce as realistic city model as possible. Oblique images and terrestrial images are more suited for building texturing in comparison to vertical imagery, as they provide better side views of the building structures (Wang et al., 2008). Another option is the use of artificial texturing applied to the building faces (Halaa et al., 1998). A final addition that could be made to enhance the 3D visualization is the addition of street furniture that is readily available with the majority of the commercially available GIS (Figure 6.2).

Figure 6.2: Samples of the street furniture available in ArcGIS.

To address the 3D pruning limitation stated in the previous section, the initial building-primitive boundary could be used to locate neighbouring lines that should be joined. For example, in Figure 6.3, a threshold used to join the neighbouring lines may accidentally join Line 1 with Line 3, instead of joining Line 1 with Line 2. A better solution would be to use the LiDAR information to determine which lines are truly neighbouring boundary segments that should be merged.

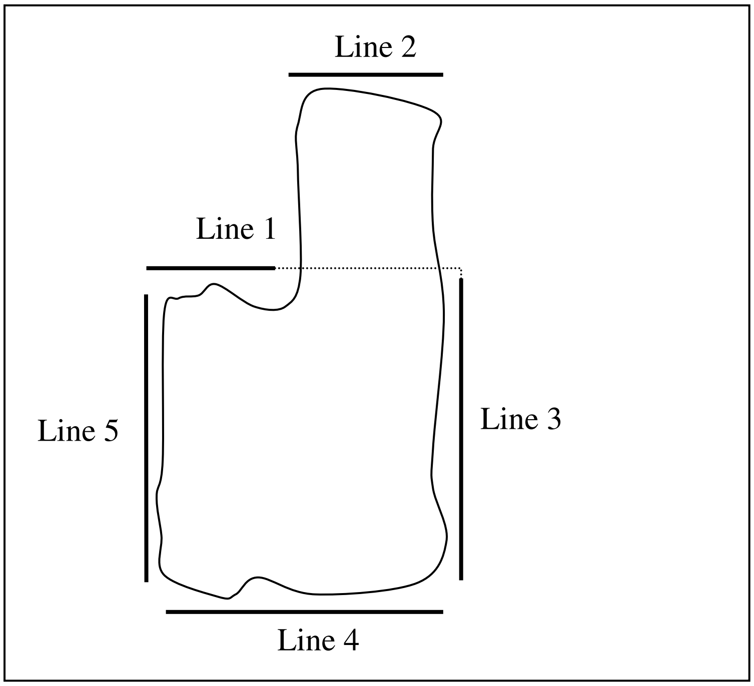
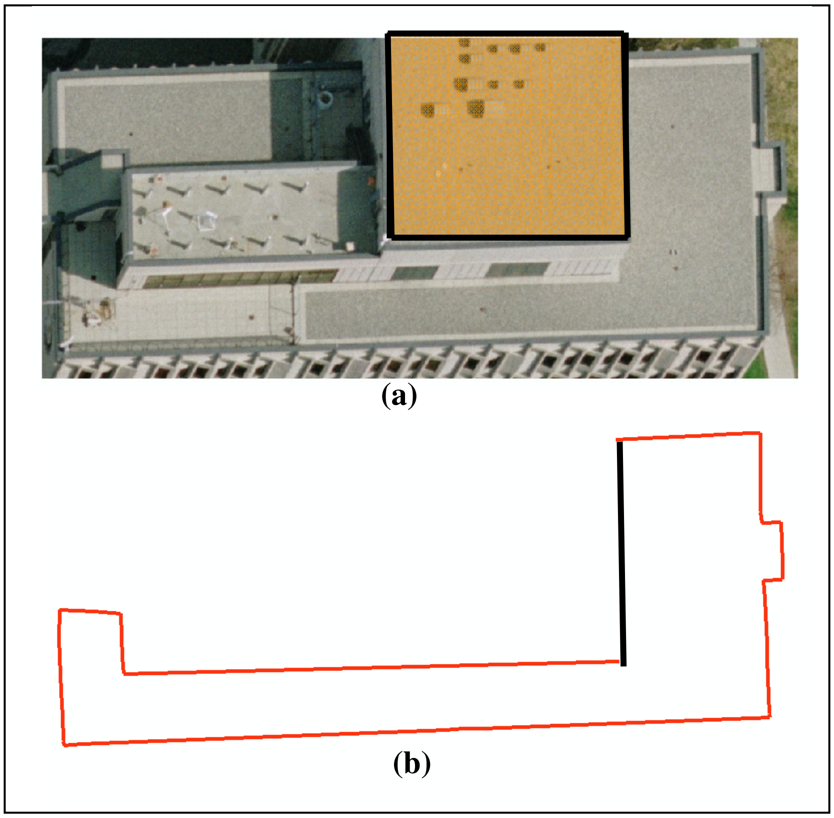


Figure 6.3: Future method for joining neighbouring lines in 3D

To address the situation when a building-primitive boundary is not visible in any of the images, the building-primitive selection should be performed in a specific order. That is, building-primitives that have a higher elevation should be selected first. This will then allow for a solution when a lower building-primitive boundary is occluded by a neighbouring (taller) building that share a vertical wall. This situation is shown in Figure 6.4. In Figure 6.4a, if precise building-primitive boundaries for the top building-primitive are obtained first (displayed in black), these boundary segments can be projected onto the lower building-primitive to locate the line segment that was not detected from imagery, for the lower building-primitive (Figure 6.4b). This will reduce the need for manual digitizing in 3D, and will also provide more accurate boundary segments than would be obtained from manual digitizing.


(a)

(b)

Figure 6.4 A building in which the top building-primitive is first investigated (a), after which the information obtained for the top building-primitive is used to complete the boundary of a lower building-primitive (b).

References

Aldelgawy, M., I. D. Detchev, and A. F. Habib. Alternative Procedures for the Incorporation of LiDAR-Derived Linear and Areal Features for Photogrammetric Geo-referencing, *ASPRS Conference Proceedings*, Portland, Oregan, April 28-May 2, unpaginated on CD-Rom, 7 pages, 2008.

Amhar, F. J. Jansa, and C. Ries. The generation of true orthophotos using a 3D building model in conjunction with a conventional DTM. *International Archive Photogrammetry and Remote Sensing,* 32: pp. 16-22, 1998.

Anderson, J. M., and E. M. Mikhail, *Surveying Theory and Practice, Seventh Edition*, USA: McGraw-Hill, 1998.

Baillard, C., C. Schmid, A. Zisserman, and A. Fitzgibbon. Automatic Line Matching and 3D Reconstruction of Buildings from Multiple Views, *Proceedings from ISPRS Conference on Automatic Extraction of GIS Objects from Digital Imagery,* IAPRS Vol. 32, Part 3-2W5, 1999.

Baltsavias, E. P. A comparison between photogrammetry and laser scanning, *ISPRS Journal of Photogrammetry & Remote Sensing,* 54: pp 83-94, 1999.

Baltsavias, E., S. Mason, and D. Stallmann. Use of DTMs/DSMs and Orthoimages to Support Building Extraction, in *Automatic Extraction of Man-Made Objects from Aerial* *and Space Images, Ascona, Switzerland* (A. Gr¨un, O.K¨ubler, and P. Agouris, Eds.), pp. 199–210, 1995.

Bang, K., A. Habib, C. Kim, and S. Shin. Comprehensive Analysis of Alternative Methodologies for True Orthophoto Generation From High Resolution Satellite and Aerial Imagery. *Proceedings from ASPRS 2007 Annual Conference*, Tampa, Florida, May, unpaginated on CD-Rom, 7 pages, 2007.

Canny, J. A computational approach to edge detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8: pp. 679–690, 1986.

Chang, Y.-C., A. F. Habib, D. C. Lee, and J.-H. Yom. Automatic Classification of LiDAR Data into Ground and Non-Ground Points, *Proceedings from ISPRS*, Beijing, pp. 457-462, 2008.

Chen, L.-C., T.-A. Teo, J.-Y. Rau, J.-K. Liu, W.-C. Hsu. Building Reconstruction from LIDAR Data and Aerial Imagery, *International Geoscience and Remote Sensing Symposium (IGARSS) 4,* no. 1525661: pp. 2846-2849, 2005.

Chen, L.-C., T.-A. Teo, C.-Y. Kuo, and J.-Y Rau. Shaping Polyhedral Buildings by the Fusion of Vector Maps and Lidar Point Clouds, *Photogrammetric Engineering & Remote Sensing* (74)9: pp. 1147-1157, 2008.

Cramer, M., D. Stallmann, and N. Haala. Direct Georeferencing Using GPS/Inertial Exterior Orientations for Photogrammetric Applications, *International Archives of Photogrammetry and Remote Sensing,* 33(B3): pp. 198-205, 2000.

Csanyi, N., and C.K. Toth. Improvement of LiDAR Data Accuracy Using Lidar-Specific Ground Targets, *Photogrammetric Engineering & Remote Sensing,* 73(4): pp. 385-396, 2007.

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

Elaksher, A. F., J. Bethel, and E. Mikhail. Roof Boundary Extraction Using Multiple Images, *Photogrammetric Record,* 18(101): pp. 27-40, 2003.

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

Fraser, C. S., E. Baltsavias, and A. Gruen. Processing of *Ikonos* imagery for submetre 3D positioning and building extraction, *ISPRS Journal of Photogrammetry & Remote Sensing,* 56: pp. 177-194, 2002.

Frueh, C., and A. Zakhor. Constructing 3D City Models by Merging Aerial and Ground Views, *IEEE Computer Graphics and Applications,* 23(6): pp. 52-61, 2003.

Gamba, P., and B. Houshmand. Digital Surface Models and Building Extraction: A Comparison of IFSAR and LIDAR data, *IEEE Transactions on Geoscience and Remote Sensing,* 38(4): pp. 1959 – 1968, 2000.

Gonzalez, R. C., and R. E. Woods, *Digital Image Processing Third Edition*, New Jersey, USA: Pearson Prentice Hall, 2008.

Haala, N., C. Brenner, and K.-H. Anders. 3D Urban GIS From Laser Altimeter and 2D Map Data, *Commission III, Working Group 4,* unpaginated, 8 pages, 1998.

Habib, A., M. Ghanma, and E. Mitishita. Co-registration of Photogrammetric and LiDAR data: Methodology and Case study, *Brazilian Journal of Cartography (RBC),* 56(1): pp.1-13, 2004a.

Habib, A., M. Morgan, E. Kim, and R. Cheng. Linear Features in Photogrammetric Activities, *ISPRS Congress, Istanbul, Turkey, PS ICWG II/IV: Automated Geo-Spatial Data Production and Updating*, pp.610, 2004b.

Habib, A., A. Pullivelli, E. Mitishita, M. Ghanma, and E. Kim. Stability analysis of low-cost digital cameras for aerial mapping using different geo-referencing techniques. *Journal of Photogrammetric Record*, 21(113): pp. 29-43, 2006.

Habib, A., E. Kim, and C. Kim. New methodologies for true ortho-photo generation, *Photogrammetric Engineering and Remote Sensing*, 73(1): pp.25-36, 2007a.

Habib, A., A. Jarvis, P. Quackenbush, G. Stensaas, D. Moe, J. Lay, E. Mitishita. Medium-Format Digital Cameras: A study into the Calibration, Stability Analysis and Application for Aerial Mapping. *Proccedings from the CIG,* Toronto, May, unpaginated on CD-Rom, 11 pages, 2007b.

Habib, A., A. Jarvis, P. Quackenbush, and G. Stensaas. Medium-Format Digital Cameras: A Study into the Calibration, Stability Analysis, and Achievable Accuracy. *Proceedings from the ASPRS Annual Conference,* Tampa, Florida, May, unpaginated on CD-Rom, 12 pages, 2007c.

Habib, A., K.I. Bang, S.-W. Shin, and E. Mitishita. Tight Integration of Photogrammetric and LiDAR Data for the Estimation of LiDAR Bore-Sighting Parameters. *Proceedings from the CIG Conference,* Toronto, May, unpaginated on CD-Rom, 12 pages, 2007d.

Habib, A., K.I. Bang, M. Aldelgawy, and S. Shin. Integration of Photogrammetric and LiDAR Data in a Multi-primitive Triangulation Procedure, *Proceedings of the ASPRS*, Tampa, Florida, May, unpaginated on CD-Rom, 7 pages, 2007e.

Habib, A., A. Jarvis, A. P. Kersing, and Y Alghamdi. Comparative Analysis of Georeferencing Procedures Using Various Sources of Control Data, *Proceedings from ISPRS*, Beijing, pp. 1147-1152, 2008a.

Habib, A., J. Kersting, T. McCaffrey, A. Jarvis. Integration of LiDAR and Airborne Imagery for Realistic Visualization of 3D Urban Environment, *Proccedings from ISPRS*, Beijing, pp. 617-624, 2008b.

Habib, A., C. Kim, K. Bang, A. Kersting, J. Kersting, A. Jarvis, R. Zhai, and I. Detchev. Semi-Automated 3D Complex Building Extraction from Multiple Aerial Images and LiDAR Data. *Submitted to the Korean Electronics and Telecommunications Research Institute (ETRI)*, (113 pages), 2008c.

Jarvis, R. A. Computing the shape hull of points in the plane, *Proceedings of IEEE Computer Society Conference Pattern Recognition and Image Process*, pp.231-261,1977.

Jensen, J. R., D. Cowen, T. Kammerer, and X. Huang. GPS and softcopy photogrammetry for large-scale digital elevation model creation and orthophoto mapping for urban applications. *Proceedings from ASPRS*, Washington, Aug. 26-29, pp. 169-178, 1994.

Kim, C., A. Habib, P. Mrstik, and K. Kusevic. New Approach for Classification and Segmentation of Airborne and Terrestrial LiDAR Data. *Proceedings from CIG*, Toronto, May, unpaginated on CD-Rom, 12 pages, 2007.

Köninger, A. 3D-GIS for Urban Purposes, *GeoInformatica* 2(1): pp. 79-103, 1998.

Kuzmin, P. Y., S. A. Korytnik, O. Long. Polygon-Based True Orthophoto Generation, *Proceedings from 20th ISPRS Congress*, July 12-23, Istanbul, pp. 529-531, 2004.

Lin, C., and R. Nevatia. Building Detection and Description from a Single Intensity Image. *Computer Vision and Image Understanding,* 72(2): pp. 101-121, 1998.

Longley, P. A., M. F. Goodchild, D. J. Maguire, and D. W. Rhind. *Geographic Information Systems and Science, 2nd Edition,* West Sussex, UK: Wiley, 2005.

Ma, R. Building Model Reconstruction from LIDAR Data and Arial Photographs, *Ph.D. dissertation, The Ohio State University,* 181 pages, 2004.

Maas, H.-G. The potential of height texture measures for the segmentation of airborne laserscanner data. *Proceedings of the Fourth International Airborne Remote Sensing Conference,* I: pp.154-161,1999.

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

Makarovic, B. Digital mono-plotters, *ITC J.,* 4: pp 583-599, 1973.

Nevatia R., C. Lin, and A. Huertas. A System for Building Detection from Aerial Images, *Proceedings in Automatic Extraction of Man-Made Objects from Aerial and Space Images (II),* Basel, Switzerland, pp. 77–86, 1997.

Nichol J., B. King, D. Quattrochi, I. Dowman, M. Ehlers, and X. Ding. Earth Observation for Urban Planning and Management: State of the Art and

Recommendations for Applications for Earth Observation in Urban Planning, *Journal of Photogrammetric Engineering & Remote Sensing*, pp: 973-979, 2007.

Nielsen, M. O. True Orthophoto Generation, *M.Sc. Thesis Technical University of Denmark*, 142 page, 2004.

Optech Inc. *ALTM 3100 System Specifications,* http://www.optech.ca/pdf/Specs/specs_altm_3100.pdf, last accessed November 02, 2008.

Palmer, T.C., and J. Shan. A comparative study on urban visualization using LiDAR data in GIS, *URISA Journal,* pp.19-25, 2002.

Ranzinger, M., and G. Gleixner. GIS Datasets For 3D Urban Planning, *Computers, Environment, and Urban Systems,* 21(2): pp. 159-173, 1997.

Rau, Jiann-Yeou. Geometrical Building Modeling and Its Application to the Ortho-Rectification for Aerial Images, *PhD Dissertation, National Central University*, 114 pages, 2002.

Rau, J., N. Chen, and L. Chen. Hidden compensation and shadow enhancement for true orthophoto generation, *Proceedings of Asian Conference on Remote Sensing,* pp. 04-08, 2000.

Rottensteiner, F., J. Trinder, S. Clode, and K. Kubik. Fusing Airborne Laser Scanner Data and Aerial Imagery For the Automatic Extraction of Buildings in Densely Built-up Areas, *International Archives of Photogrammetry and Remote Sensing*, 6 pages, 2004.

Shiode, N. 3D urban models: Recent developments in the digital modeling of urban environments in three-dimensions. *GeoJournal,* 52: pp. 263-269, 2001.

Shufelt, J. A. Performance Evaluation and Analysis of Monocular Building Extraction From Aerial Imagery, *IEEE Transactions on Pattern Analysis and Machine Inteligence,* 21(4): pp. 311 – 317, 1999.

Skarlatos, D. Orthophotograph production in urban Areas, *Photogrammetric Record*, 16(94): pp. 643-650, 1999.

Suveg, I., and G. Vosselman. 3D Reconstruction of Building Models, *IAPRS, Vol. XXXIII, Amsterdam,* Working Group IV/2, unpaginated, 8 pages, 2000.

Toutin, T. DSM generation and evaluation from QuickBird stereo imagery with 3D physical modelling. *International Journal of Remote Sensing*, 25(22): pp. 5181-5193, 2004.

Vosselman, G., and S. Dijkman. 3D Building Model Reconstruction From Point Clouds and Ground Plans, *International Archives of Photogrammetry and Remote Sensing*, Volume XXXIV – 3/W4 Annapolis, MD, Oct. 22-24, pp. 37-43, 2001.

Wang, Y., S. Schultz, and F. Giuffrida. Pictometry's Proprietary Airborne Digital Imaging Systems and Its Application in 3D City Modellling, *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences,* Vol. XXXVII. Part B1, Beijing, pp. 1065-1070, 2008.

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

Wolf, P. R., and B. A. Dewitt. *Elements of Photogrammetry with Applications in GIS. 3rd Edition*, New York, USA: McGraw-Hill, 2000.

Zlatanova, S., A. A. Rahman, and M. Pilouk. 3D GIS: Current Status and Perspectives, *International Archives of Photogrammetry and Remote Sensing*, 34(4): pp. 66-71, 2002.

Zuxun, Z., W. Jun, and Z. Jianqing. A method for straight line extraction based on line space, *Geomatics and Information Science of Wuhan University*, 29(3): pp. 189-194, 2004.