IMPROVEMENT OF THE GEOSPATIAL ACCURACY OF MOBILE TERRESTRIAL LiDAR DATA

MICHAEL LESLAR

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

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

NOVEMBER 2016

ABSTRACT

Many applications, such as topographic surveying for transportation engineering, have specific high accuracy requirements which MTL may be able to achieve under specific circumstances. Since high rate, immersive (360° FOV), MTL is a relatively new device for the collection and extraction of survey data; the understanding and correction of errors within such systems is under researched. Therefore, the goal of the work presented here is to quantify the geospatial accuracy of MTL data and improve the quality of MTL data products.

Quantification of the geospatial accuracy of MTL systems was accomplished through the use of residual analysis, error propagation and conditional variance analysis. Real data from two MTL systems was analyzed using these methods and it was found that the actual errors exceeded the manufacturer's estimates of system accuracy by over 10mm. Conditional variance analysis on these systems has shown that the contribution by the interactions among the measured parameters to the variances of the points in MTL point clouds is insignificant. The sizes of the variances for the measurements used to produce a point are the primary sources of error in the output point cloud.

Improvement of the geospatial accuracy of MTL data products was accomplished by developing methods for the simultaneous multi-sensor calibration of the system's boresight angles and lever arm offsets, zero error calibration, temperature correction, and both spatial and temporal outlier detection. Evaluation of the effectiveness of these techniques was accomplished through the use of two test cases, employing real MTL

data. Test case 1 showed that the residuals between a control field and the MTL point cloud were reduced by 4.4cm for points located on both horizontal and vertical target surfaces. Similarly, test case 2 showed a reduction in the residuals between control points and MTL data of 2~3cm on horizontal surfaces and 1~2cm on vertical surfaces. The most accurate point cloud produced through the use of these calibration and filtering techniques occurred in test case 1 (27mm ± 26mm). This result is still not accurate enough for certain high accuracy applications such as topographic surveying for transportation engineering (20mm ± 10mm).

ACKNOWLEDGMENTS

I would like to acknowledge the financial assistance provided by the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Ministry of Transport Ontario (MTO). I would also like to acknowledge Optech Incorporated for providing the equipment and LiDAR Data used in this dissertation. Finally, I would like to acknowledge the significant amount of assistance and advice I received from both Dr. Jian-Guo Wang, P.Eng and Dr. Baoxin Hu, P.Eng.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABREVIATIONS

σ, STD – One Sigma Standard Deviation
ASCII – American Standard Code for Information Interchange
BLAGC – Boresight angles and Lever Arms from Ground Control
BLAOS – Boresight angles and Lever Arms from Opposing Strips
BLATS – Boresight angles and Lever Arms from Two Sensors
CFD – Constant Fraction Discrimination
DG – Direct Georeferencing
DGPS – Differential Global Positioning System
DMI – Distance Measurement Indicator
ECEF – Earth Centered Earth Fixed
GPS – Global Positioning System
GNSS – Global Navigation Satellite System
IARTK – Inertially Aided Real Time Kinetic
ICP – Iterative Closest Point
IMU – Inertial Measurement Unit
INS – Inertial Navigation System
FOV – Field Of View
LiDAR – Light Detection And Ranging
LOF – Local Outlier Factor
MAL – Mobile Airborne LiDAR
MTL – Mobile Terrestrial LiDAR
PAM – Partition Around Medoids
PDOP – Position Dilution Of Precision
PP – Post Processing
PPK – Post Processing Kinematic
PPP – Precise Point Positioning
PSF – Polynomial Surface Fitting
PRF – Pulse Repetition Frequency
RMS – Root Mean Square
RTK – Real Time Kinetic
SBAS – Space Based Augmentation System
SBET – Smoothed Best Estimated Trajectory
SONAR – SOund Navigation And Ranging
SP – Single Point
STL – Static Terrestrial LiDAR
VRS – Virtual Reference Station
WAAS – Wide Area Augmentation System

1. Introduction

Within the last decade Mobile Terrestrial LiDAR (MTL) has progressed from using adapted low rate Static Terrestrial LiDAR (STL) sensors with limited fields of view (FOV) to using purpose built high rate LiDAR sensors capable of 360° scanning. Multiple purpose built MTL systems are commercially available today, including the Lynx Mobile Mapper (http://www.teledyneoptech.com/), StreetMapper (http://www.3dlasermapping.com/), the VMX 450 (http://www.riegl.com/), the MX8 (http://www.trimble.com/), the Pegasus (http://leica-geosystems.com/) and the IP S3 (https://www.topconpositioning.com/). These MTL systems generally consist of multiple high rate (500kHz or more) LiDAR line sensors attached to a modern Direct-Georeferencing (DG) system. Similar to the Mobile Airborne LiDAR (MAL) sensors that preceded them, these MTL sensors are designed to scan in a single axis (i.e. line sensors) and therefore rely on the forward motion of the vehicle upon which they are mounted to produce three dimensional point clouds.

1.1 Motivation

MTL is used in multiple applications around the world as a means of rapidly measuring terrain or the geometry of a scene from a vehicle. MTL sensors have been installed and used from boats, trains, all-terrain vehicles and trucks. The high rate at which these sensors operate allows for the rapid measurement of all the objects on a project site. The applications for which MTL is used vary widely in their scope, purpose and accuracy requirements. Some applications, such as categorizing trees or inspecting railway ties for damage do not require geometrically or spatially accurate data. Other applications

however, such as topographically surveying a concrete runway or highway asphalt to estimate their thickness or shape do require highly accurate spatial data.

Even the most accurate MTL systems, such as the Lynx Mobile Mapper with an absolute accuracy as specified by the manufacturer of ±50mm [1, 2], the VMX-450 with an absolute accuracy as specified by the manufacturer of ±20-50mm [3] or the Pegasus with and accuracy as specified by the manufacturer of ±28mm [4], currently do not meet the minimum accuracy requirements for certain applications. For example, the British Columbia Ministry of Transportation lays out a horizontal and vertical engineering survey point accuracy requirement of 20mm ± 10mm for all survey points of pavement and/or urban detail [5]. This accuracy specification from the British Columbia Ministry of Transportation is typical of the accuracy specifications from transportation departments across North America. The consequence of this accuracy specification is that these MTL systems do not meet the engineering accuracy requirements set by the British Columbia Ministry of Transportation. Therefore, to be able to use MTL systems for certain high accuracy applications, method(s) must be found to reduce the absolute positional error of the points in the MTL data.

Since high rate, immersive (360° FOV), MTL is a relatively new device for the acquisition and retrieval of geospatial information; the understanding and correction of errors within such systems is under researched [6 – 12]. Specifically, the generation of high quality, geospatially accurate MTL data remains a major topic of investigation. To date, much of the available research was produced to understand, quantify and correct the

errors in MAL or STL systems. The problem with using algorithms designed for MAL systems in MTL work is that they make assumptions about the observation angles and ranges which are not valid for the majority of MTL systems [13, 14]. On the other hand, many algorithms designed for STL systems do not include consideration for the inclusion of a DG system with the LiDAR [15].

Previous work [15 – 19], has produced through descriptions of the types and consequences of various errors in pulsed time of flight STL sensors. Other work [13, 20] has gone into the performance of MAL systems, while [14, 21 – 24] has focused on the interactions between MAL sensors and the DG system. Yet further work [25 – 30] has focused on the integration and development of calibration and error models for STL systems attached to a DG system. Little work exists describing algorithms designed to identify and remove outlier data in terrestrial LiDAR point clouds [31].

1.2 Objectives

As previously stated, the most accurate MTL systems (20mm to 50mm) [1 – 4] provide error specifications which are adequate for some types of surveys, such as engineering surveys of open terrain (50mm ± 20mm) [5], but inadequate for other types of surveys, such as engineering surveys of pavement or urban detail (20mm ± 10mm) [5]. Therefore the objective of the work presented here is twofold. Firstly, it is essential to establish methods to determine if the accuracy estimates stated by the MTL manufacturers are correct or whether they are overly optimistic. Secondly, once the accuracy of the MTL data is established, it is necessary that methods for the reduction of errors in all phases of data acquisition and processing be established. To this end, the work presented here

describes several new methodologies for analyzing and improving the accuracy of MTL data. Since MTL systems are comprised of many separate and varied parts, multiple methods for improving the quality of the end results are needed.

1.3 Outline

A review of the current literature on LiDAR sensors and Mobile LiDAR systems was conducted to understand the available techniques for quantifying the geospatial accuracy and improving the quality of MTL data products. The results of this literature review are presented in Chapter 2. The material in Chapter 2 begins with descriptions of the components of typical MTL systems, including some of the design options available to manufacturers when building these systems. The chapter then proceeds to list and describe the known error sources possible in MTL, along with a review of the current techniques available for mitigating them.

Based on this literature review, it was decided to break the problem into three separate tasks. The first task involved the development of methods for assessing the accuracy estimates of MTL systems. Three different methods were developed, implemented and applied to MTL data from a Lynx Mobile Mapper. The simplest method of estimating the error in MTL point clouds is by creating control targets using an accurate static method such as static GPS or a Total Station, and comparing MTL observations to this data. A more complicated, but common method of estimating the error in MTL point clouds is to use error propagation to estimate the variances of the individual points. A third technique involves the use of conditional variance analysis and Monte Carlo simulation to apportion the error in MTL point clouds among the various measurement

inputs. Using conditional variance analysis on MTL data is a unique idea in the literature. All three techniques contained in the first task are presented here in Chapter 3.

The second task aimed to develop methods for reducing the identified errors in MTL data. Reducing the errors inherent in MTL systems partially means reducing the errors of the measurements taken by individual components of these systems. It also means reducing the errors generated by the interaction of dispirit components of MTL systems through calibration. To this end, a new technique for measuring and correcting the temperature drift in MTL range finders is presented in Chapter 4. Chapter 4 also introduces a new technique for calculating the zero error for MTL range finders under laboratory conditions. Finally, and most importantly, Chapter 4 examines techniques for calibrating MTL sensors to the DG system and introduces a new calibration technique based on stereo pairs of LiDAR sensors.

The third task involved developing methods to identify and filter erroneous data from MTL point clouds. Even after all the errors from the components and their integration into the system have been controlled, errors from the MTL's interaction with its environment will still occur. This could be due to the properties of the targets being scanned, the weather conditions during the scan, or the presence of interfering radiation in the vicinity of a scan. Therefore, Chapter 5 discusses experiments conducted with different outlier removal techniques in MTL data. The outlier removal techniques considered in Chapter 5 make use of both the spatial and temporal data available from the processed MTL point clouds. Using the locations of the individual points in the point

cloud, the spatial techniques considered here fit a polynomial surface to patches of the data and compare individual points to that surface. One method in the spatial domain relies on the residuals from the surface fitting itself as a means of comparison between the point cloud and the surface. The other spatial technique considered uses the student t test statistic to evaluate the position of individual points with respect to the surface. The most novel method introduced in Chapter 5 involves the use of an α-β-γ Kalman smoother and the temporal data associated with the points in the point cloud to predict an expected location for individual points based on the positions of their neighbours. Comparing the actual positions of said points with their estimated positions provides a means for identifying points outside their neighbourhood.

As a means of determining the amount of improvement to the quality of MTL data which is possible from the methods mentioned above, real MTL data from two different Lynx Mobile Mappers was used. Two test scenarios were devised, the first using data from a commercial office building and the second from a typical street scene. By analyzing the point clouds from these MTL data sets before and after corrections have been applied to the measurements and all outliers have been filtered, the effectiveness of the correction techniques described above was tested. Chapter 6 discusses the implementation and results of these techniques in detail.

2. The Basics of Mobile Time-Of-Flight LiDAR and Literature Review

The first task in dealing with errors in pulsed time-of-flight MTL systems involves understanding their components and how they interact with one another. MTL systems from different manufacturers perform very similar tasks, but employ related but distinct methods for accomplishing these tasks. Every mobile LiDAR has to consist of the same 6 general components, the laser, the rangefinder, the deflection unit, the DG system, the controlling/recording equipment and the external position correction methodology [32]. Some of these components, such as the laser range finder, deflection unit and DG system have to be rigidly mounted to the vehicle; other components such as the data recorder and correction station do not have to be rigidly fixed to the vehicle. Figure 2.1 shows a block diagram of these components and how they connect together.

From the components, their interactions with each other and their interactions with their environment, several potential sources for error are created in pulsed time of flight MTL. These errors can be broken down into four categories, instrument errors, target errors, positioning errors and environmental errors. Instrument errors include random errors, time walk, temperature drift, zero error, scale error, mixed pixels, dynamic track error, velocity error and assembly balancing issues [13, 15]. Target errors can include object reflectance and laser beam incidence angle [15]. Positioning errors can include inertial measurement unit (IMU) errors, global navigation satellite system (GNSS) errors,

boresight angle errors and lever arm offset errors [13]. Environmental errors include atmospheric propagation errors [13, 15].

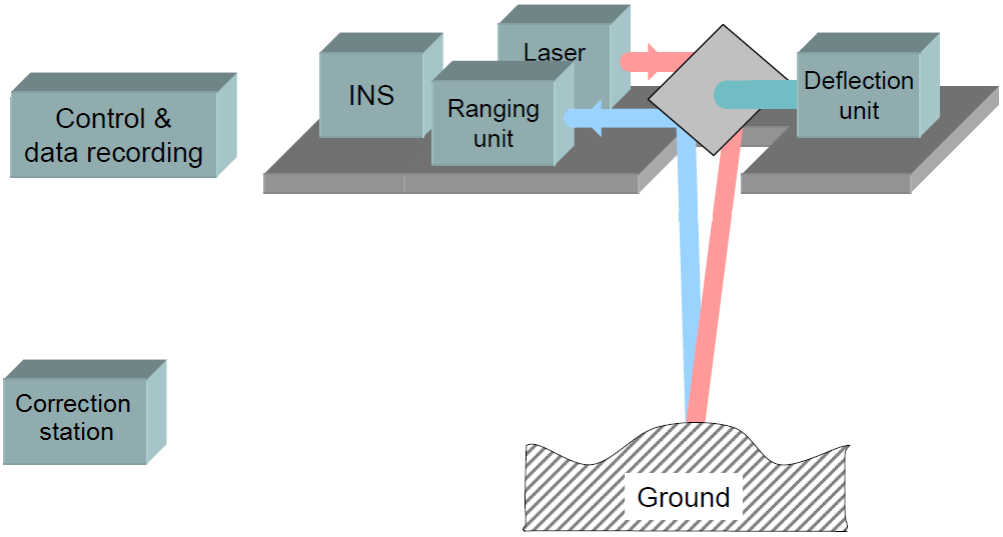

Figure 2.1: Basic components of an MTL system (modified from [32]).

2.1 Components of Mobile Time-Of-Flight LiDAR Systems

2.1.1 The Laser

The most important part of a LiDAR sensor is the laser device at the heart of the range finder sub-assembly. The rate at which the laser discharges and more specifically, the signal generated by the laser device indicating a pulse has occurred (t_0), initiates the actions of the other LiDAR sensor sub-systems, setting the pace at which the LiDAR sensor operates. Lasers are generally classified by their wavelength and their pulse repetition frequencies (PRF) which, incidentally, are usually used to describe the entire LiDAR sensor. The design of an MTL system must also take into account the width of the laser's pulse and the divergence of the laser's beam.

The first index used to classify a MTL system is the laser wavelength. The choice of wavelength is important as it dictates the ability of a LiDAR sensor to operate within different medium (atmosphere, water, etc..), the type of targets and the ranges at which those targets can be observed, the eye-safety rating of the LiDAR sensor and the cost of the laser hardware [33]. The reflectance properties of various materials with respect to the wavelength of light used to irradiate them are shown in Figure 2.2. At certain wavelengths, common materials such as asphalt, trees, snow and water will absorb most if not all light directed at them and would therefore be invisible to a LiDAR sensor operating at those wavelengths.

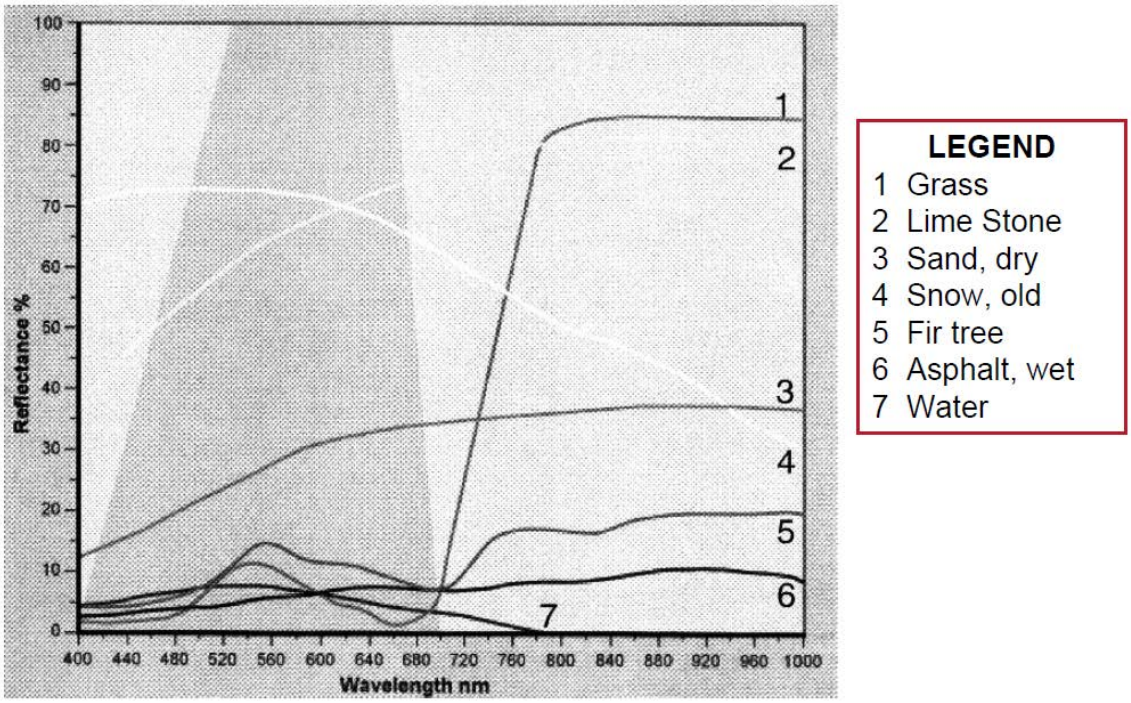

Figure 2.2: The reflectance of common materials compared with the wavelength of light used to illuminate them [33].

One of the most important considerations when working with any laser equipment is how to keep people working around the equipment safe from injury. In most cases the risk

with working around lasers is to a person's vision. Laser products are classified based on their potential to harm the vison of people exposed to the beam. The classifications are fixed by the International Electronics Commission [34] and are based on the wavelength of light being emitted by the laser and the amount of power being output. Figure 2.3 shows the laser classes considered "eye safe", the wavelengths these classes can operate within and the maximum allowed output power for each class.

Figure 2.3: Revised IEC 60825 Standard for Laser Safety Classes [34].

Many time-of-flight terrestrial based LiDAR systems incorporate lasers which operate in the near infrared (1535nm ~ 1550 nm) [1, 2, 3, 35, 36]. From Figure 2.3, it is clear that the advantage to using this wavelength is that a higher output power can be achieved while maintaining a class 1 "eye safe" classification. The disadvantages include not being able to see water or water saturated targets very well, ice and snow. Some terrestrial LiDAR systems make use of lasers operating at wavelengths closer to the visible spectrum (785nm ~ 1064nm) [35 – 38]. The advantage to using these laser

wavelengths are increased range for the same input power and the ability to scan certain targets such as ice and snow. The major disadvantage from Figure 2.3 is that higher output power at the emitter will cause the laser product to be classified as “non-eye safe”. Usually, this “non-eye safe” classification will only apply to a certain distance from the emitter since the pulse power diminishes as the collimated laser beam travels away from its emitter. Being classified “non-eye safe” is a major disadvantage for laser devices which are intended to be used in built up areas or areas where the general public is in attendance.

The second index used to classify an MTL laser is the pulse repetition frequency (PRF). This index gives the rate at which the laser can discharge light energy pulses. It is this index that determines the overall speed of the time-of-flight terrestrial LiDAR system. In many cases, the rate at which the laser pulses also has the side effect of determining the amount of outgoing power each pulse will contain. Unless input power to the laser is increased, the faster you pulse the laser, the shorter the amount of time the capacitors within the laser have to build up a charge, and therefore the lower the power in the outgoing laser pulse. Currently an MTL system such as the Riegl VMX 450 provides sensors with a maximum PRF for 550kHz [3]. The Optech Lynx Mobile Mapper SG1 provides sensors with a maximum PRF of 600kHz [2]. The Optech Lynx Mobile Mapper M1 only has a maximum PRF of 500kHz [1]. Of course all of these sensors also provide for the ability to reduce the PRF during a scan, causing the sensors to operate slower, allowing the user to control the amount of data collected. Some mobile systems such as

the ILRIS HD MC have a fixed PRF of 10kHz [35]. This causes slower scan speeds, but allows for much greater maximum ranges to be achieved.

The third index that characterizes the laser onboard an MTL system is the laser's pulse width. While the term pulse width sounds like a measurement of linear distance, it is usually described as a unit of time. Since time-of-flight LiDAR uses time to measure distance, measurements of time and distance are sometimes conflated. A laser's pulse width is the length of time it takes to emit the light energy of a single laser pulse. For example, the Velodyne HDL-64E has an advertised pulse width of 10ns [38].

The last index that characterizes the laser onboard an MTL system is the beam divergence. A laser is considered to be a collimated beam of light. While this is the goal, no laser is 100% capable of maintaining the beam diameter it had when it exited the laser emitter, over a given range. For example, the ILRIS HD has an advertised beam divergence of 150μrad and the ILRIS LR has an advertised beam divergence of 250μrad [35]. This means that the beam diameter of the ILRIS HD laser at 1000m from the instrument is expected to be 0.30m plus the beam diameter at the exit aperture of the LiDAR system. Similarly, the beam diameter of the ILRIS LR laser at 1000m from the instrument is expected to be 0.50m plus the beam diameter at the exit aperture of the LiDAR system.

2.1.2 The Range Finder

The laser range finder of a time-of-flight LiDAR sensor is the combination of the laser, receiver(s) and timing mechanism. Every time the laser pulses (t_0), a timing trigger is

sent to the timing mechanism to start the clock. The receiver opens, measuring all incoming radiation at or near the frequency of the emanated laser pulse. When radiation of sufficient intensity is detected by the receiver (t_1), a range calculation is performed based of the formula in Equation (2.1).

$$R = \frac{c}{2} \cdot (t_1 - t_0) \tag{2.1}$$

Where R is the calculated range to target and c is the speed of light at sea level. All terrestrial LiDAR sensors operate on the expectation of hitting a non-cooperative target that will scatter the incoming light ray equally in all directions (Figure 2.4, Lambertian reflectance). This means that the expected amount of laser energy returned to the receiver is a tiny fraction of the light energy originally emitted from the range finder.

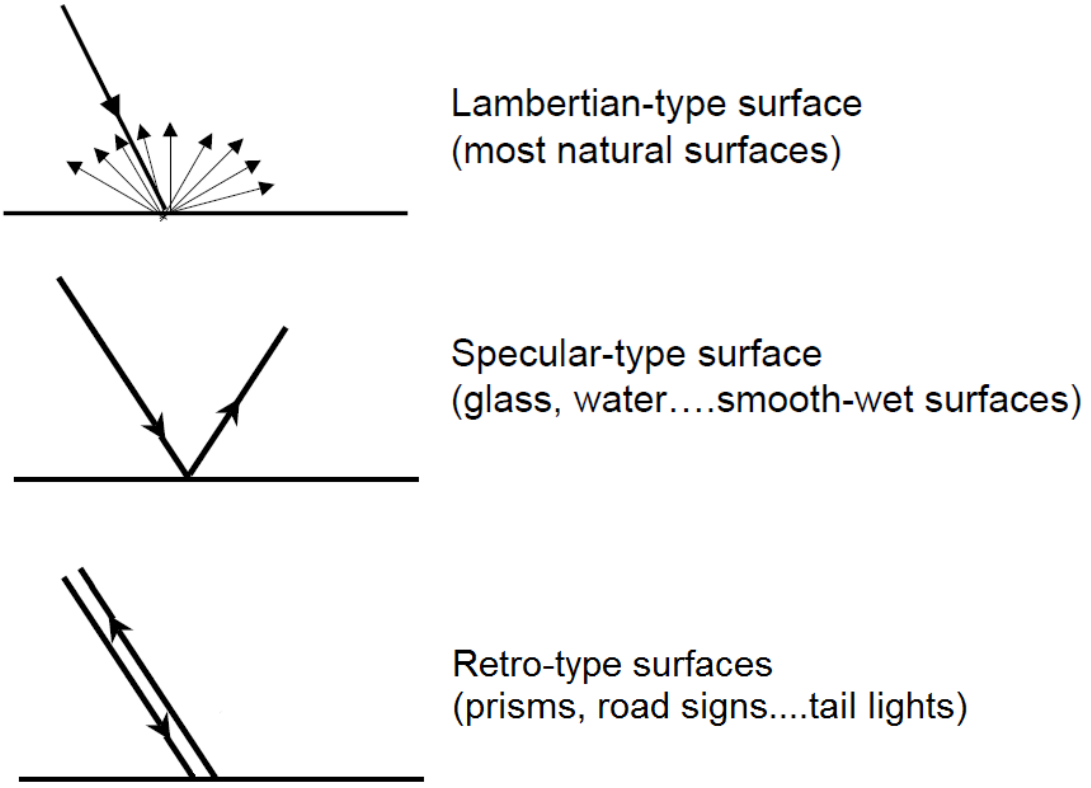

Figure 2.4: Behavior of a light pulse as it interacts with a target (modified from [33]).

The amount of light returned to the receiver is governed by the laser range equation, which was originally derived from the microwave radar range equation. The general form of the laser range equation is shown in Equation (2.2)

$$P_R = \frac{P_E \cdot G_T}{4 \cdot \pi \cdot r^2} \cdot \frac{\sigma}{4 \cdot \pi \cdot r^2} \cdot \frac{\pi \cdot D^2}{4} \cdot \eta_{Atm} \cdot \eta_{Sys} \tag{2.2}$$

Where P_E and P_R are the emitted and received laser power, G_T is the antenna gain of the transmitter, σ is the effective cross sectional area of the target, r is the range to target, D is the aperture diameter of the LiDAR and the η-terms are the system and atmospheric transmission factors [39].

When a specular target (Figure 2.4) is encountered, either the majority of the light pulse will be scattered into space causing no range to be recorded by the range finder (known as a drop out), or the pulse will reflect off the specular surface, hit an object and return to the range finder via the specular surface. This results in a low P_R value being measured by the receiver and a range which is the sum of the ray lengths of all paths travelled by the beam. Alternatively, when a retro surface (Figure 2.4) is encountered, the P_R value measured at the receiver will be much higher than the receiver's maximum measureable value (known as saturation) causing an inaccurate range measurement to be recorded.

2.1.3 The Deflection Unit

Except for some Flash LiDAR systems, the majority of pulsed time-of-flight LiDAR sensors use a mechanical means of directing the laser beam so as to scan a target. The mechanism used to direct the laser beam is required to have a highly accurate means of measuring the angle at which the pulse is directed. This is usually accomplished by

means of a device such as a rotary encoder, or galvanometer. Galvanometers give the precise and accurate ability to point the laser beam in any direction; however, they have a limited field of view. Rotary encoders provide a 360° field of view, but are limited in the precision with which they can be used to direct the laser beam.

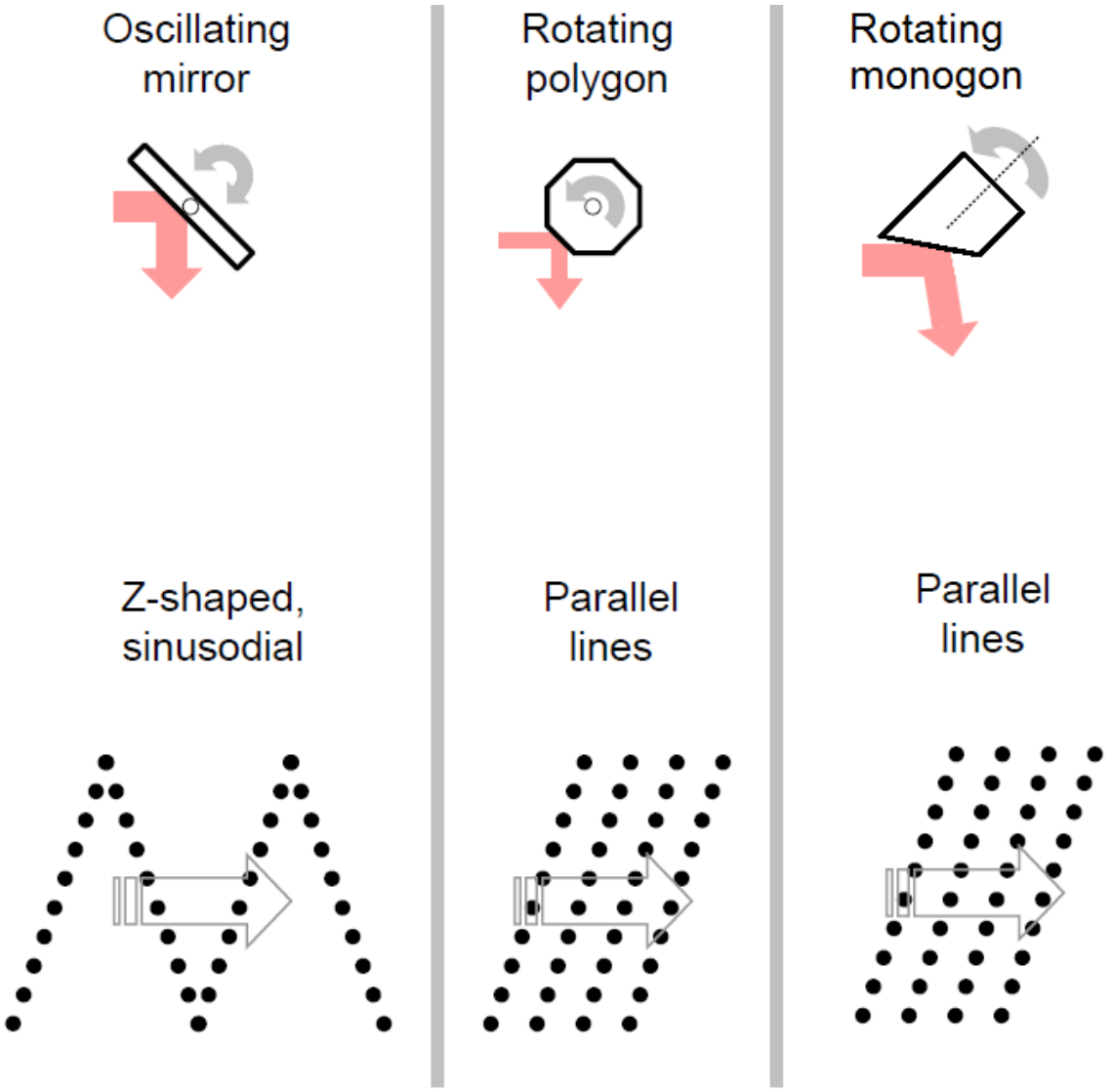

Figure 2.5: Common scanning mechanisms used in MTL systems and their associated ground patterns (modified from [32]).

The scanning pattern of the LiDAR sensor is derived from the type of deflection system used. While some LiDAR sensors use other deflection methods, the most widely used methods for MTL sensors are those shown in Figure 2.5. The oscillating mirror type of

scanning mechanism is usually a result of a deflection mechanism which employs a galvanometer. While it is possible to produce such a scan pattern by oscillating an encoder based prism, LiDAR sensors such as the ILRIS MC (www.teledyneoptech.com) typically employ a galvanometer based oscillating mirror design. Another LiDAR sensor, the VZ-1000 (www.riegl.com), employs a rotating polygon attached to a rotary encoder as the deflection unit. Most of the high rate MTL systems such as the Lynx Mobile Mapper (www.teledyneoptech.com), the VMX-1HA (www.riegl.com) and the MX8 (www.trimble.com) prefer to use the rotating monogon type of scanning mechanism attached to a rotary encoder.

2.1.4 The Control and Data Recorder

Most MTL sensors do not have the control and storage facilities onboard the LiDAR sensors themselves, but within a separate unit. This is usually done for practical reasons, as the type and amount of data being collected would usually overwhelm any storage devices built into the sensors. A separate control unit also makes the synchronization and management of more than one sensor, be it a camera or LiDAR sensor, possible. For some MTL systems like the Applanix LANDMark Marine [40], which incorporates a LiDAR sensor, digital camera and an optional SONAR system, the control unit is a laptop or desktop computer. For higher rate sensors such as the Riegl VMX-450 or the Optech Lynx Mobile Mapper, specialty control devices are required to manage and store the flow of data from the two LiDAR sensors and the multiple cameras. Figure 2.6 shows the control units for the VMX-450 and the Lynx. These are typical of the control units that are used with the other high rate MTL systems on the market.

Figure 2.6: Examples of control and data storage units for the Riegl VMX-450 [3] and the Lynx Mobile Mapper (www.teledyneoptech.com).

2.1.5 The DG System

Just as the laser forms the heart of the LiDAR sensor, a GNSS aided Inertial Navigation System (INS) forms the heart of the MTL system. The INS, also called a Direct Georeferencing (DG) system, is used to directly georeference the kinematic platform upon which the LiDAR sensors are attached. The DG system provides the position and orientation of the system at any time during the acquisition, as well as providing the timing synchronization needed to combine the various sensors of a MTL system.

Several options exist for choice of DG system to use in a MTL system. Among these options are the Applanix POS (www.applanix.com), Novatel SPAN (www.novatel.com) and the iXblue ATLANS (www.ixblue.com) etc. All of these DG systems contain the same components, an IMU, one or more GNSS receivers and a microcomputer. Some DG system variants also come with an optional distance measurement indicator (DMI), which may simply be a rotary encoder that attaches to the back wheel of a land vehicle

and can measure the linear distance traveled. The expected accuracy of a DG system depends on the individual performance of the components mentioned above. Section 2.2.3 describes the types of errors and bias these DG system components are typically prone to experience.

Table 2.1: Specifications for the Applanix LV DG system [41].

POS LV		Performance with GNSS				Performance after 60s GNSS Outage			
		X,Y Position (m)	Z Position (m)	Roll and Pitch (°)	True Heading (°)	X,Y Position (m)	Z Position (m)	Roll and Pitch (°)	True Heading (°)
210	PP	0.02	0.05	0.02	0.05	0.32	0.13	0.06	0.06
210	IARTK	0.035	0.05	0.02	0.1	1.27	0.35	0.06	0.1
210	DGPS	0.3	0.5	0.02	0.2	2.51	0.61	0.06	0.2
220	PP	0.02	0.05	0.02	0.025	0.24	0.13	0.06	0.03
220	IARTK	0.035	0.05	0.02	0.05	0.69	0.35	0.06	0.07
220	DGPS	0.3	0.5	0.02	0.05	0.88	0.61	0.06	0.07
420	PP	0.02	0.05	0.015	0.02	0.12	0.1	0.02	0.02
420	IARTK	0.035	0.05	0.015	0.02	0.34	0.27	0.02	0.03
420	DGPS	0.3	0.5	0.015	0.02	0.45	0.56	0.02	0.03
510/520	PP	0.02	0.05	0.005	0.015	0.1	0.07	0.005	0.015
510/520	IARTK	0.035	0.05	0.008	0.02	0.3	0.1	0.008	0.02
510/520	DGPS	0.3	0.5	0.008	0.02	0.42	0.53	0.008	0.02
610/620	PP	0.02	0.05	0.005	0.015	0.1	0.07	0.005	0.015
610/620	IARTK	0.035	0.05	0.005	0.02	0.28	0.1	0.005	0.02
610/620	DGPS	0.3	0.5	0.005	0.02	0.41	0.51	0.005	0.02

Table 2.1 lists the expected accuracy of the Applanix POS LV. The accuracy with which the POS LV can calculate the position (X, Y, Z) and orientation (Roll, Pitch, True Heading) is greatly dependent on the type of IMU used, the number of GNSS receivers available and the processing method used to produce the trajectory. This is reflected in the first column of Table 2.1. In Table 2.1, the DG systems are listed in the order of the accuracy of the IMU option used. The DG systems containing the more accurate IMU options are located at the bottom of the table (POS LV 610/620, POS LV 510/520) and the DG systems containing less accurate IMU option are located at the top (POS LV

210). The 20 in 220, 420, 520 and 620, indicates that these units have two onboard GNSS receivers, designed to assist the IMU in maintaining the True Heading of the system. The designators PP (Post-Processing), IARTK (Inertially Aided Real Time Kinematic) and DGPS (Differential Global Positioning System) indicate the type of processing to which the trajectory information is subjected. Table 2.1 shows that the best possible trajectory is expected to occur when it is collected with a 520 or 620 IMU employing two GNSS receivers with satellite lock and produced by post-processing the data after collection.

In contrast, Table 2.2 lists the expected accuracy of various models of the Novatel SPAN System [42]. Just as with the Applanix POS, the accuracy with which the SPAN systems can calculate their position (X, Y, Z) and orientation (Roll, Pitch, Heading) is greatly dependent on the type of IMU used, the number of GNSS receivers available and the processing method used to produce the trajectory. The different models of the SPAN system represent some of the different IMU options available from Novatel. Each of the Novatel models listed in Table 2.2 comes with a single GNSS receiver but can be upgraded to employ two GNSS receivers as a means of aiding heading determination during low dynamic applications. In Table 2.2, the designators PP (Post-Processing), RTK (Real Time Kinematic) and SP (Single Point, i.e. No External Correction) indicate the type of processing to which the trajectory information is subjected. Since, Table 2.2 provides details on system performance after only a 10s GNSS outage, it is difficult to compare performance to the Applanix POS, however, with PP under normal GNSS availability, the positioning specs on the SPAN system seem to outperform the Applanix

POS by 2~3cm both horizontally and vertically, while the orientation accuracy of the Applanix POS seems higher by 0.01º.

Table 2.2: Specifications for the SPAN GNSS/INS combined systems [42].

Span Model	Outage Duration	Positioning Mode	POSITION ACCURACY (M) RMS		VELOCITY ACCURACY (M/S) RMS		ATTITUDE ACCURACY (DEGREES) RMS		
			Horizontal	Vertical	Horizontal	Vertical	Roll	Pitch	Heading
PwrPak7-E1	0 s	RTK	0.020	0.030	0.020	0.010	0.030	0.030	0.100
		SP	1.000	0.600	0.020	0.010	0.030	0.030	0.100
		PP	0.010	0.020	0.020	0.010	0.010	0.010	0.040
	10 s	RTK	0.350	0.130	0.100	0.021	0.060	0.060	0.150
		SP	1.300	0.700	0.100	0.021	0.060	0.060	0.150
		PP	0.010	0.020	0.020	0.010	0.010	0.010	0.040
SPAN-CPT	0 s	RTK	0.020	0.030	0.015	0.010	0.020	0.020	0.060
		SP	1.000	0.600	0.020	0.010	0.020	0.020	0.060
		PP	0.010	0.020	0.015	0.010	0.008	0.008	0.035
	10 s	RTK	0.250	0.180	0.045	0.025	0.030	0.030	0.080
		SP	1.200	0.750	0.050	0.025	0.030	0.030	0.080
		PP	0.020	0.020	0.015	0.010	0.008	0.008	0.035
SPAN-IGM-A1	0 s	RTK	0.020	0.030	0.020	0.010	0.035	0.035	0.150
		SP	1.000	0.600	0.020	0.010	0.035	0.035	0.150
		PP	0.010	0.020	0.020	0.020	0.012	0.012	0.074
	10 s	RTK	0.460	0.130	0.100	0.021	0.072	0.072	0.210
		SP	1.410	0.700	0.100	0.021	0.072	0.072	0.210
		PP	0.020	0.020	0.020	0.010	0.012	0.012	0.074
SPAN-IGM-S1	0 s	RTK	0.020	0.030	0.020	0.010	0.015	0.015	0.080
		SP	1.000	0.600	0.020	0.010	0.015	0.015	0.080
		PP	0.010	0.020	0.020	0.010	0.015	0.015	0.080
	10 s	RTK	0.270	0.140	0.051	0.017	0.025	0.025	0.095
		SP	1.220	0.710	0.051	0.017	0.025	0.025	0.095
		PP	0.020	0.020	0.020	0.010	0.015	0.015	0.080

A third option, the iXblue, has the expected accuracy of three available models listed in Table 2.3 [43]. Just as with the Applanix POS and Novatel SPAN, the accuracy with which the ATLANS or AIRINS systems can calculate their position (X, Y, Z) and

orientation (Roll, Pitch, Heading) is greatly dependent on the type of IMU used, the number of GNSS receivers available and the processing method used to produce the trajectory.

Table 2.3: Specifications for the iXblue INS-GNSS systems [43].

Environment	ATLANS-C		AIRINS
	Land	Air	Air
True heading accuracy:			
DGPS	N/A	0,100 deg	0,020 deg
RTK	0,020 deg	N/A	N/A
PPK	0,020 deg	0,020 deg	0,005 deg
60 sec outage duration for RTK	0,020 deg	N/A	N/A
60 sec outage duration for PPK	0,020 deg	N/A	0,010 deg
Position accuracy (X, Y):			
DGPS	N/A	0,600 m	0,600 m
RTK	0,035 m	N/A	N/A
PPK	0,020 m	5 cm + 1 ppm	0,150 m
60 sec outage duration for RTK	0,350 m	N/A	N/A
60 sec outage duration for PPK	0,150 m	N/A	0,100 m
Position accuracy (Z):			
DGPS	N/A	0,900 m	1,500 m
RTK	0,050 m	N/A	N/A
PPK	0,050 m	10 cm + 1 ppm	0,050 m
60 sec outage duration for RTK	0,300 m	N/A	N/A
60 sec outage duration for PPK	0,100 m	N/A	0,070 m

The two models of DG system available from iXblue, the ATLANS and the AIRINS, represent the two IMU options available from this manufacturer. Both of the iXblue models listed in Table 2.3 come with a single GNSS receiver but unlike the POS or SPAN, they do not have an option to add a second antenna and their embedded software relies far more heavily on the IMU than GNSS data. In Table 2.3, the designators PPK (Post-Processing Kinematic), RTK (Real Time Kinematic) and DGPS (Differential Global Positioning System) indicate the type of processing to which the trajectory information is subjected. Comparing Table 2.3 to Table 2.1, the position accuracy during

normal GNSS availability seems comparable, however, the iXblue systems apparently experience greater IMU drift than the Applanix systems after a 60s GNSS outage. For example, Table 2.1 indicates the Applanix POS 520 has position accuracies of 0.1m horizontally and 0.07m vertically after a 60s GNSS outage, while Table 2.3 indicates the iXblue ATLANS-C has position accuracies of 0.15m horizontally and 0.1m vertically after a 60s GNSS outage. Again, doing this type of comparison with the SPAN system is harder as Table 2.2 reports accuracy after only a 10s GNSS outage, however, during normal GNSS availability the SPAN appears to be 2~3cm better in accuracy, both horizontally and vertically, than the iXblue systems.

2.1.6 The Correction Station

An important part of achieving accurate point clouds involves the method used to correct the trajectory provided by the DG system. There are a few different methods available that can be used to correct the DG system, all of which require either real-time transmission of data to the DG system or post process combination with external correction data. The most common correction method, which requires no interaction on the user's part is the space based augmentation system (SBAS). The free SBAS correction method available to the DG system is dependent upon where in the world the DG system is operating. Figure 2.7 shows SBS systems and their coverage around the world. The accuracies achievable using an SBS, specifically WAAS are on the order of ±1.5m~±3.0m [44].

Methods for correcting DG system trajectories, which require user interaction, include post process double differencing with a base station (±0.02m~±0.05m) [44], real time

kinetic (RTK) correction from a single based station (±0.02m~±0.10m) [44], differential correction with a base station (DGPS) (±0.5m~±2m) [44], precise point positioning using a precise ephemeris (PPP) (±0.1m) [44] and correction with a virtual reference station (VRS) network (±0.1m~±0.5m) [44].

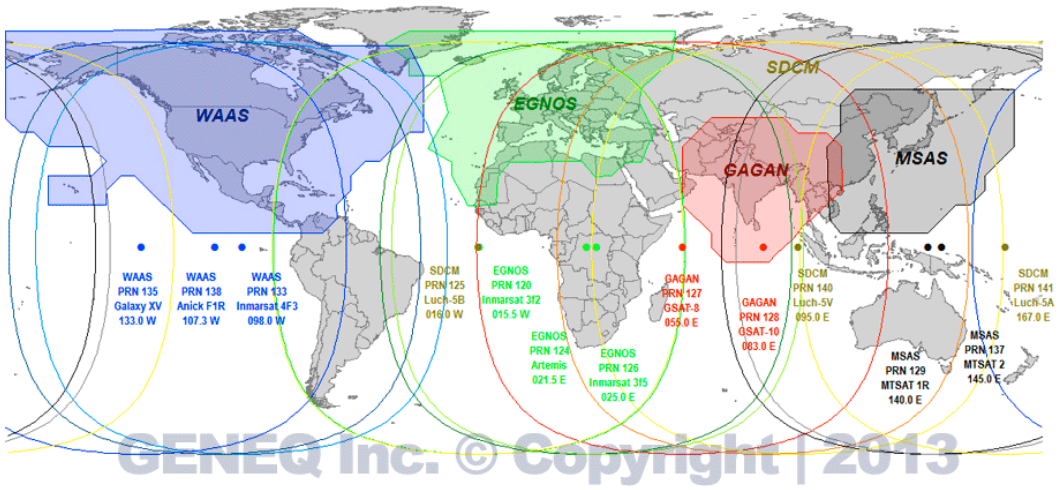

Figure 2.7: SBAS systems and coverage around the world [45].

2.2 Errors in Mobile Time-Of-Flight LiDAR Systems

As shown in the previous section, a MTL system consists of multiple independent components working together to form a larger system. Being that any MTL system involves the marriage of a LiDAR scanner and a GNSS/DG system with an external correction station, the overall accuracy of the system is dependent on the errors from each component and the measurements used to link these disparate systems together.

Measurements for the position and orientation of the MTL system are generated by the DG system for any time t along the vehicle's path. The DG system usually provides estimates of the errors for these position and orientation measurements. These error

estimates are calculated real time by the DG system and are usually given in the form of root mean square (RMS) errors or one sigma standard deviations (σ), depending on the DG system in question. On the other hand, the measurements made by the LiDAR provide information on the location of any point p in range of the system. It is usual for the LiDAR to have fixed error quantities determined by the manufacturer or the surveyor in a controlled environment. These error estimates are almost always given as one sigma standard deviations. Many of the types of errors possible in a terrestrial pulsed time-of-flight LiDAR system will be further discussed in this section.

The errors associated with the integration of the LiDAR with the DG system are usually the least understood and generally the hardest to quantify. While multiple methods have been proposed for LiDAR to DG system boresighting, few offer a reliable or rigorous approach for determining error. Only the algorithm based methods really give a means for determining error estimates for the LiDAR to DG system calibration parameters. The documented methods for determining sensor to DG system calibration parameters are detailed in Section 2.2.3.2.

Some work has already been done to analyze the errors inherent in MTL systems. The first and most common method for analyzing errors in MTL systems, as outlined in the literature, is error propagation. A test MTL was assembled in [25] and the authours used error propagation to analyze the estimated data errors in this system. Similarly, [26 – 28] use error propagation to study the effect of various types of error on both MAL and MTL systems.

Another, less rigorous approach to measuring the inherent errors in a MTL system is through comparison to some form of control field. This approach is adopted by [29] as well as [46]. In both cases coordinates were affixed to a reference façade in a typical urban street scene, using a more accurate measurement method than the MTL. In [46] a total station and static GPS observations were used to create the control field, while [29] relied on existing 3D city models as their control field. In both cases, statistics were generated from the direct vector comparisons, made between the MTL point clouds and the control field.

A third approach to the problem combined these two methods, performing both error prorogation and direct comparison [30]. Here, the authours chose to use error propagation to come up with a theoretical error budget and then compared this theory against statistics generated by comparing different MTL strips with each other and with a control field. In this case the control field was established using Real Time Kinetic (RTK) Global Positioning System (GPS) observations.

Chapter 3 will introduce a new way of analyzing the errors inherent in MTL, but before a new error analysis method is described, it is first necessary to understand the types of errors present in MTL. The remainder of this section will detail the types of errors inherent in MTL and some of the strategies currently used to correct them.

2.2.1 Instrument Errors

2.2.1.1 Random Errors

The random errors for time-of-flight LiDAR are well understood, having been first described by [47, 48] and then reproduced in several works such as [15]. The accuracy of a time-of-flight laser range finder may be estimated by Equation (2.3).

$$\sigma_r = \frac{c}{2} \cdot \sigma_t \tag{2.3}$$

Where c is the speed of light and σ_t is the jitter of the timing moment. The jitter of the timing moment can be estimated by Equation (2.4).

$$\sigma_t = RMS_n \cdot \left(\frac{\partial U}{\partial t} \right)^{-1} \tag{2.4}$$

Where RMS_n is the root mean square noise amplitude at the input of the time discriminator and $\partial U / \partial t$ is the slope of the timing pulse at the moment of timing, approximated by Equation (2.5).

$$\frac{\partial U}{\partial t} \approx \frac{U_s}{t_{rise}} \tag{2.5}$$

Where U_s is the peak value of the signal and t_{rise} the rise time of the pulse [48]. Therefore, the range accuracy for a time-of-flight laser range finder can be written as is shown in Equation (2.6) [15].

$$\sigma_r \approx \frac{c}{2} \cdot \frac{\sigma_n \cdot t_{rise}}{U_s} = \frac{c}{2} \cdot \frac{t_{rise}}{SNR} \tag{2.6}$$

Alternatively, if the range is determined by averaging n independent range measurements, the range accuracy can be computed by Equation (2.7).

$$\sigma_r = \frac{c}{2} \cdot \frac{t_{rise}}{SNR\sqrt{n}} \tag{2.7}$$

For a time-of-flight range finder which employs a hard stop threshold in the receiver for determining a potential return, the signal-to-noise ratio of a given pulse detection is given by Equation (2.8).

$$SNR = \frac{U_s}{\sigma_n} \tag{2.8}$$

If the timing mechanism does not use a single hard stop threshold, but instead determines an instantaneous threshold based on the incoming signal level (constant fraction discrimination), then the signal-to-noise ratio for the pulse detection is given by Equation (2.9).

$$SNR = \frac{i_{SIG}^2}{i_{SN}^2 + i_{BK}^2 + i_{TH}^2 + i_{DK}^2} \tag{2.9}$$

where i_{SIG}^2 is the mean square signal current, i_{SN}^2 is the signal induced or Poisson noise, i_{BK}^2 is the background induced noise, i_{TH}^2 is the thermal or Johnson noise in the receiver and i_{DK}^2 is the dark noise current in the detector[15, 48].

The signal-to-noise ratio is affected by the amount of received power from a laser pulse. Due to the relationship between the signal-to-noise ratio and the range accuracy, the amount of power received therefore also effects range accuracy. Among other things, the amount of received power is dependent on the amount of transmitted power from the laser. The relationship between the amount of received power to the amount of transmitted power is given by the laser range Equation (2.2).

The signal-to-noise ratio is also affected by the receiver characteristics [15], specifically, the responsiveness of the detector at the laser's wavelength. This property of the receiver is given by the detectitvity of the receiver, the equation for which is given in Equation (2.10).

$$D^* = \frac{\sqrt{A}}{NEP} \tag{2.10}$$

Where A is the area of the detector and NEP is the noise equivalent power or the amount of power required to overcome noise in the detector.

Finally, an angular random error that is always present due to the laser geometry involves the uncertainty in the angular location of the range measurement within the laser footprint. As was shown in [16], the actual measurement point can be located anywhere within the footprint of the laser beam. The standard deviation of the measurement point can be calculated by Equation (2.11).

$$\sigma_{beam} = \pm\frac{\gamma}{4} \tag{2.11}$$

Where γ is the beam divergence of the transmitted laser pulse.

2.2.1.2 Timing Errors

Since a time-of-flight LiDAR unit measures distances by measuring the time it takes light pulses to travel to and return from a target, the timing mechanism must be capable of extremely precise timing. For example, for a time-of-flight LiDAR to measure a distance

of 1mm, then the timing mechanism must be able to accurately measure 6.7 picoseconds (see Equation (2.12)).

$$t = \frac{2 \cdot R}{c} = \frac{2 \cdot 0.001m}{2.997 \cdot 10^{8} m/s} \approx 6.67 \cdot 10^{-12} s \approx 6.7 ps \tag{2.12}$$

The effects from timing issues in LiDAR can be dramatic. Figure 2.8 shows what can happen when the timing mechanism in a time-of-flight LiDAR fails.

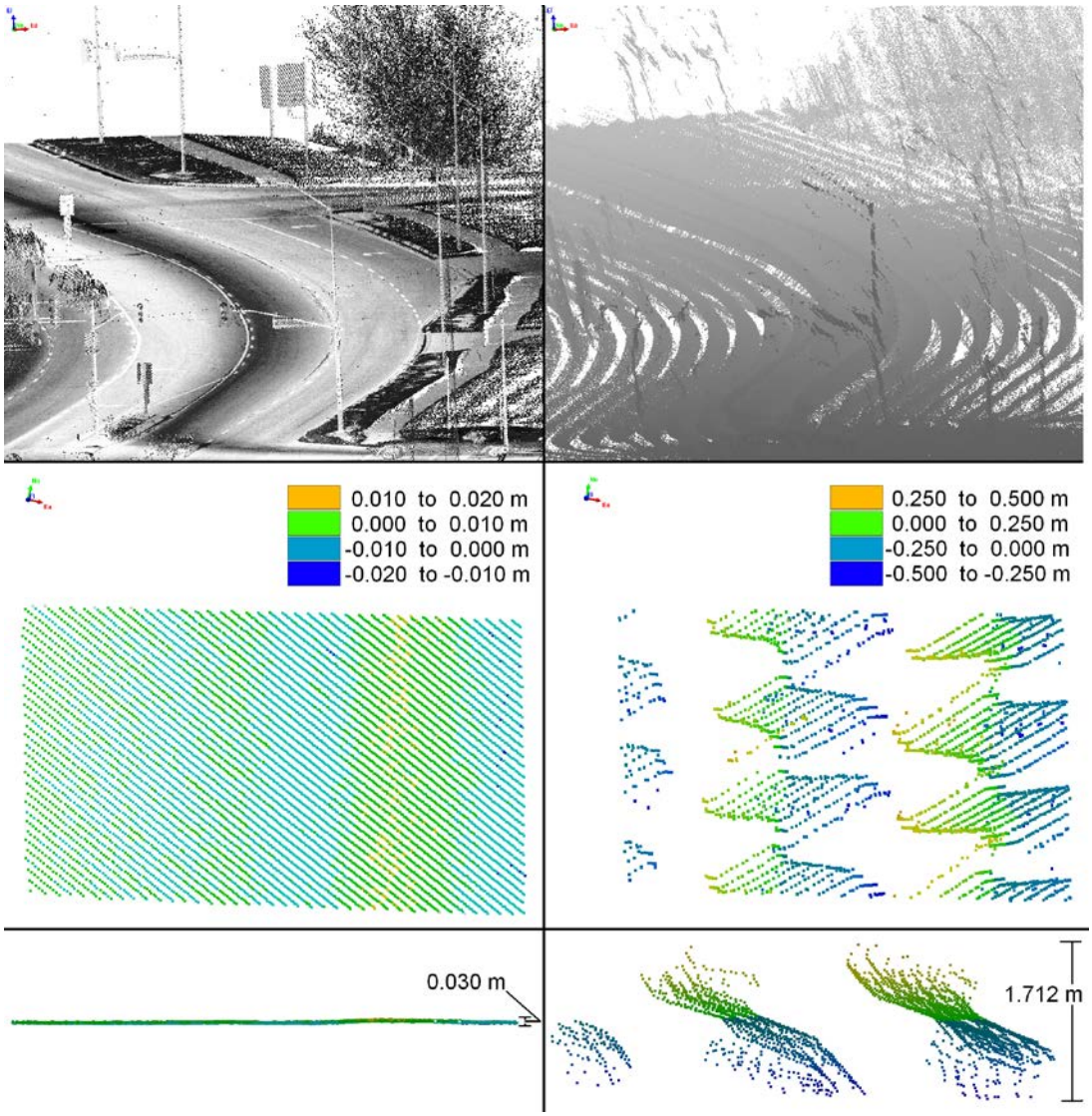

Figure 2.8: Data from two Lynx Mobile Mapper sensors. Data on the left is from a LiDAR with a calibrated and functioning timing board. Data on the right is from a LiDAR with a malfunctioning timing board.

The images in Figure 2.8 demonstrate the importance of the timing board in the overall health of the point cloud. The scene depicted here was scanned with two sensors mounted on the same vehicle. The image on the left shows the scene as it should be, the timing board was calibrated and healthy. The section of the road depicted in the middle left panel, shows that the deviation to this surface from a best fit comparison plane is less than ±2cm. The lower left panel confirms this, showing a maximum surface deviation from the comparison plane of exactly 3 cm. Contrasting this with the images on the right of Figure 2.8, it is noticeable that the familiar street scene now resembles a plowed field. The road section (the same road section as depicted in the middle left panel), shown in the middle right panel now more closely resembles a helix instead of typical scan lines. The thickness of the road, shown in the bottom right panel, indicates that the road surface is almost 2m thick.

Without a healthy and well calibrated timing mechanism, time-of-flight LiDAR does not function.

2.2.1.2.1 Calibration/Malfunction

Being that the timing board in a time-of-flight LiDAR is the core technology that all manufactures rely on to make their systems operate, the operation and calibration of these timing boards is treated as closely guarded proprietary information. The best that can be accomplished by users of this technology is to study the effect of timing errors in the resultant point cloud and incorporate these observations into the LiDAR point error estimate.

First, it would be prudent to establish what a well behaved and calibrated, rotating monogon type LiDAR system should produce. This can be done by scanning a known flat surface, in this case a wall, and extruding the two dimensional data produced into the third dimension. Extruding the two dimensional data is accomplished through the use of an artificially calculated trajectory.

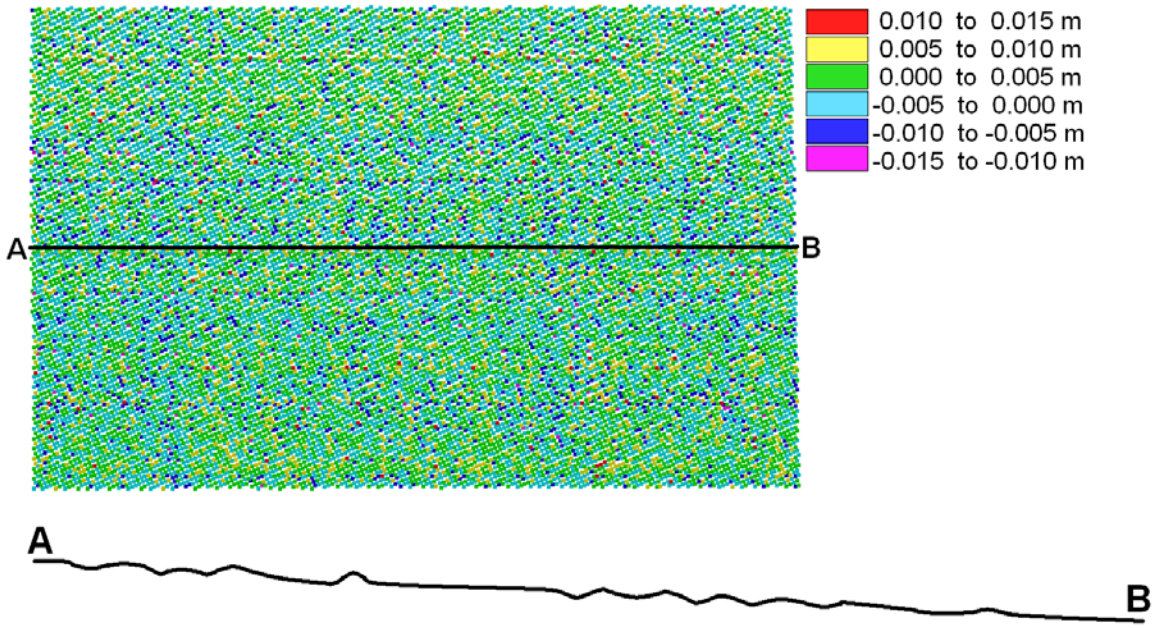

Figure 2.9: Error map of a flat wall, produced from an extruded Lynx Mobile Mapper scan.

Figure 2.9 shows an error map of such an extruded scan. The colourized error map is comparing the individual Lynx data points to the best fit plane through the wall section. Figure 2.9 shows that the majority of points lie less than 5 mm, either side of the best fit plane. Indeed, the calculated one sigma standard deviation for the data set indicates that they actually lie ± 3 mm from the best fit plane. The cross section shown in the same figure shows a relatively flat profile, with a minimal amount of deviation.

Contrasting this with the same sensor, artificially stressed by high temperature (+70°C), the timing board calibration breaks down. It should be noted that this temperature is outside the manufacture's operating specification and the sensor is close to automatic thermal shutdown.

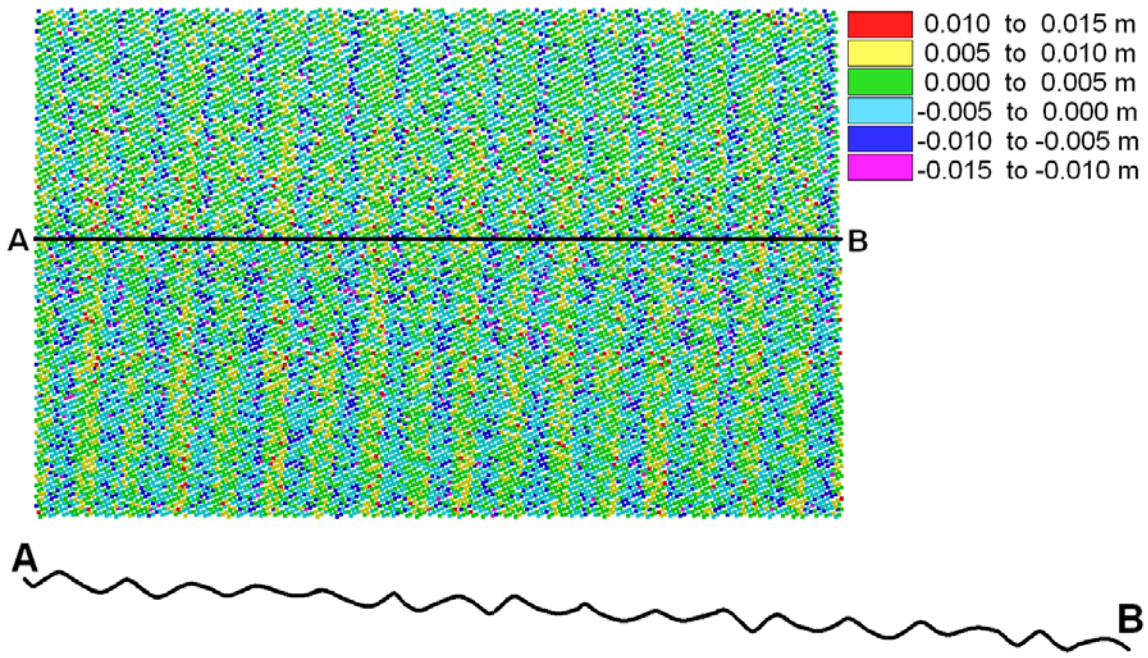

Figure 2.10: Error map of a flat wall, produced from an extruded Lynx Mobile Mapper scan, where the sensor has been stressed by temperature.

Figure 2.10, shows the results of such a test. Again fitting a best fit plane to a small section of extruded flat wall and then comparing the data to that plane generates an error map as shown in Figure 2.10. Notice the cyclical pattern in the coloured data points. This pattern is being produced by an alternating timing bias error. This bias error is changing in a regular pattern; giving times shorter than actual time to longer than the actual time. Examining the cross section shown in the same figure, it is noticeable that a cyclical bias appears as a type of sinusoidal wave throughout the data section. The scale on the error map indicates that the extents of the wave are around ±1cm, making a total bias of around 2 cm.

This bias pattern is even more pronounced by viewing the results generated from an uncalibrated timing board. Figure 2.11, shows a scan of the same flat section of wall, also extruded using an artificial trajectory. The bias in this case is still approaching ±1 cm, but no long looks like a sinusoidal pattern. The cross section of the data shown in Figure 2.11 shows that the bias in this case has very sharp transition points, with the same magnitude as the previous example.

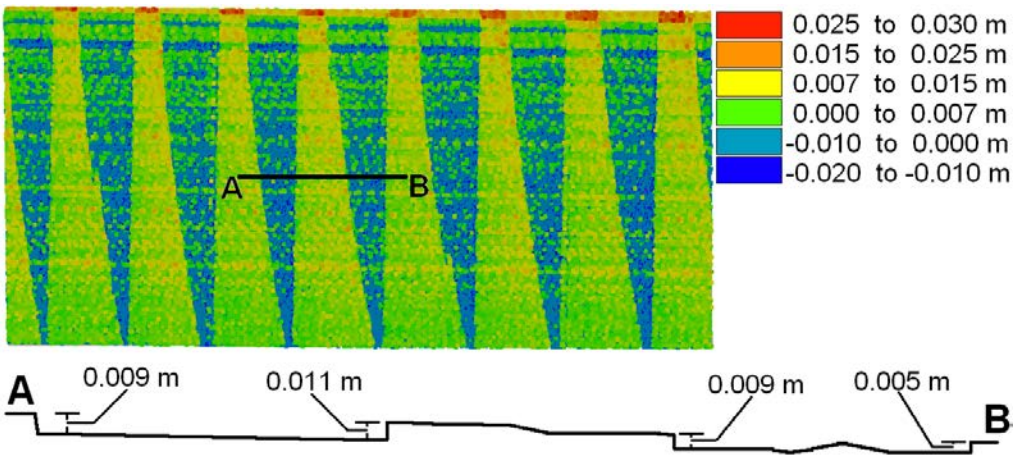

Figure 2.11: Periodic timing error, shown with scan of a flat laboratory wall, using an extruded Lynx Mobile Mapper scan.

All of these examples show that timing accuracy within the LiDAR system, can lead to bias which directly affect the health of the LiDAR data itself. Under normal operating conditions the timing onboard a LiDAR system should give range data which falls within a very tight band around the true values for sensor ranges. If the calibration of the timing mechanism is poor, its accuracy is problematic. If it is malfunctioning, the health of the LiDAR point cloud is greatly impacted. Most importantly, it must be understood by the operator of a LiDAR unit that it is important that the system be operated within the temperature range quoted by the manufacture, or else an unintended bias in the resulting data may occur.

2.2.1.2.2 Time Walk

Due to the precise timing required in the laser range finder, even small variations in the measured times can have a significant effect on the output ranges. In a system with a well calibrated timing mechanism, factors such as pulse amplitude, pulse shape and surface reflectance will still cause variations in the measured timings [49] (see Figure 2.12).

Other factors affecting the measured timings are changes in the propagation delay of the timing mechanism and potential crosstalk in said mechanism [17]. Reducing the effect of time walk involves determining an instantaneous threshold based on the incoming signal level (CFD) [49] and using a range intensity correction table to map signal amplitude with range correction.

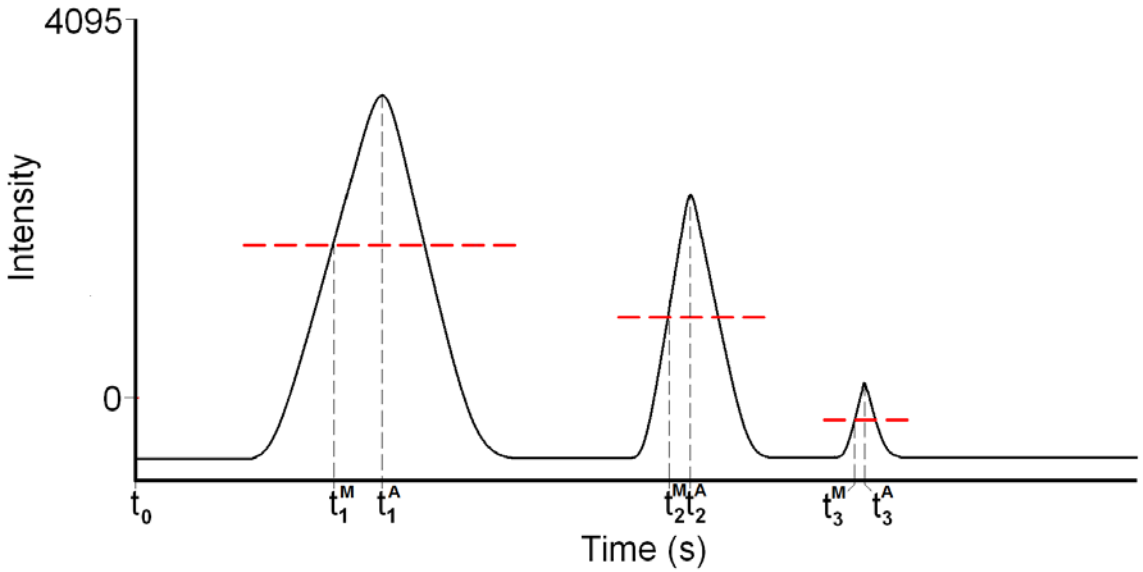

Figure 2.12: Time difference based on intensity return for three laser pulses in a time-of-flight, non-waveform digitized LiDAR with a 12-bit receiver.

Since most terrestrial based LiDAR systems don't digitize the waveform of the returning pulse, they must use a receiver intensity value to determine when to stop timing and

calculate the range to target. Most commercial LiDAR systems (that do not collect the entire waveform) use CFD (Constant Fraction Discrimination) to determine this value instantaneously depending on the incoming signal. When the receiver rises above a certain intensity value, the LiDAR system accepts that a valid range measurement has been made and calculates the range. Depending on the strength of the returning signal, the system may take this range reading at the base of the return pulse or closer to the peak. This is shown in Figure 2.12, where the difference between t^M and t^A represents the timing error caused by measuring the pulse return time at the intensity threshold and not at the return peak. This variable time measurement on pulses of different intensities may seem minuscule, but it can cause several centimeters of error in range measurements. Therefore, manufacturers of LiDAR equipment generally include a table of all possible intensity values for the LiDAR receiver along with a calibrated range correction at that intensity. Sometimes this table is not accessible to the end user; it is hidden away in the software of the LiDAR system. In the case of the Lynx Mobile Mapper, this table is provided in ASCII format along with the other calibration information and is eminently editable by the end user.

Figure 2.13 shows a graph of the range intensity corrections for two typical Lynx Mobile Mapper sensors. The corrections themselves vary depending on the variations in the timing board and receiver used in the LiDAR unit. The scale of the correction being applied is very significant. Sensor 1 in Figure 2.13 has corrections approaching 15cm for the midrange intensities. Sensor 2 has corrections of 8mm~9mm at these same intensities. This is important to note because sensor 2 has a range intensity problem.

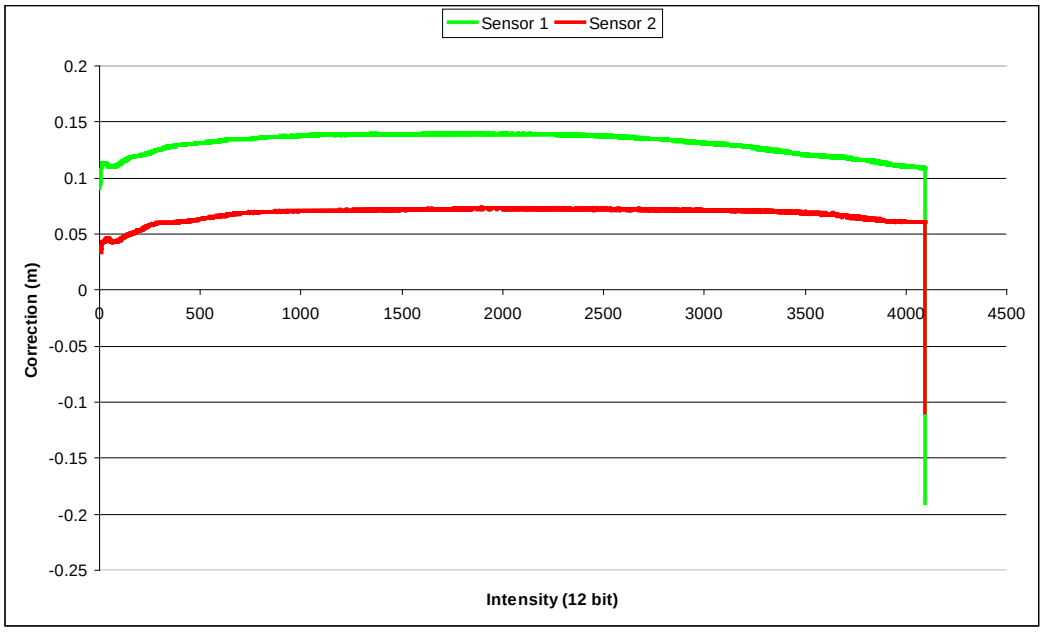

Figure 2.13: Range intensity corrections for two Lynx Mobile Mapper sensors.

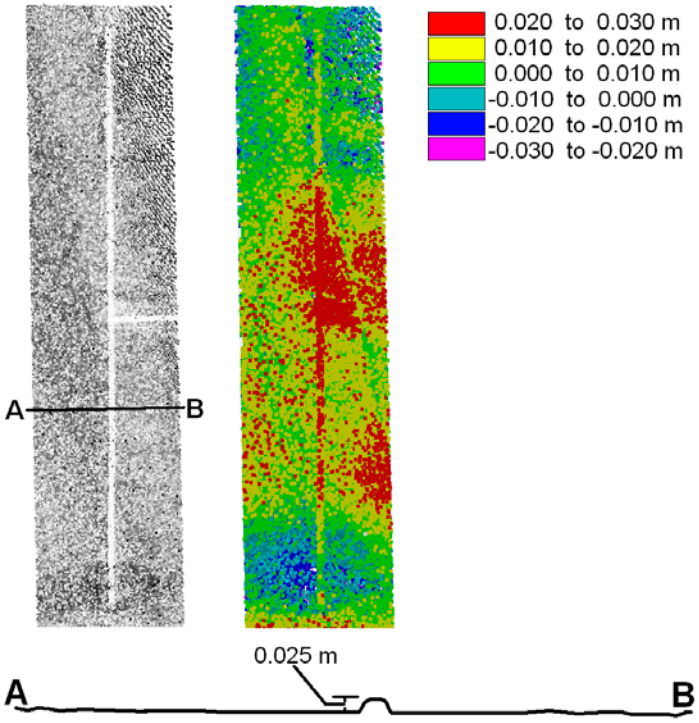

Figure 2.14: Lynx Mobile Mapper scan of a parking lot paint line, compared with the best fit plane through the data segment.

Figure 2.14 shows data of a parking lot line collected using Sensor 2. The error map shown in Figure 2.14 clearly shows that the parking lot line is higher than the

surrounding data taken in the unpainted asphalt. Looking at the cross section in Figure 2.14, the elevation difference between the painted line and the surrounding asphalt is 2.5cm.

This type of bias error can cause a serious separation of adjacent objects in the LiDAR data and must be reduced or eliminated before data collection proceeds.

2.2.1.2.3 Returned Intensity

The amplitude of the returning pulse in Figure 2.12 is governed by the laser range equation (Equation (2.2)), presented in Section 2.3. Generally, specific types of targets are used to determine the pulse amplitude effects on a LiDAR system. Therefore, the laser range equation may be simplified based a series of assumptions. It may generally be assumed that the target is larger than the footprint size of the laser beam, i.e. an extended target as opposed to a wire or point target. It can be assumed that the apparent brightness of the target surface is the same regardless of the observer's angle of view, i.e. a Lambertian reflector. Also, since most commercially available LiDAR systems have the laser emitter and receiver in close proximity, i.e. mono-static, it can be assumed that the receiver has the exact same field of view as the emitter. Based on these assumptions the laser range equation may be expressed as shown in Equation (2.13).

$$P_R = \frac{\pi \cdot P_E \cdot \rho \cdot \cos(\alpha)}{4 \cdot r^2} \cdot \eta_{Atm} \cdot \eta_{Sys} \tag{2.13}$$

Where ρ is the reflectance of the given material and α is the incidence angle of the laser beam [50]. Under laboratory conditions, when the time walk is established, the atmosphere is clear and unperturbed. This means that the atmospheric transmission

factor does not fall below 99% and can therefore be neglected. If the emitted laser power is unknown, but assumed to be constant, then the laser range equation can be further simplified as shown in Equation (2.14).

$$i = \frac{\rho \cdot \cos(\alpha)}{r^2} \cdot \eta_{Sys} \cdot C \tag{2.14}$$

Where i is the returned intensity and C is an unknown constant factor [51]. Using formulas similar to Equation (2.14) [18, 50] investigated the effect of range, surface reflectivity and angle of incidence on terrestrial based close range LiDAR. In one study [50], a metal rig containing six specially constructed targets was affixed to a standard surveyor's tripod. The targets were made of material with known reflectivities of 5%, 20%, 40%, 60%, 80% and 99%. The metal rig could be rotated with a possible accuracy of ±2º. Several experiments were conducted and it was found that the influence of the incidence angle and the target reflectivity were not separate from each other but each acted similar to a change in reflectivity. It was also found that range to target acts differently than the angle of incidence and the target reflectivity. Several empirical models were derived for the Riegl LMS-420i.

Another study [18], used various types of mineral and construction specimens to see if surface roughness, colour, wetness and range effect the accuracy of range measurements. The various specimens used were cut with a saw and placed so that their planar surface was facing the scanner. Two ranges were used 3m and 53m and the specimens were scanned dry and then wet. The study found no significant range errors between the samples, but did find varying intensity values were returned by the laser range finder.

2.2.1.4 Zero and Scale Error

All time-of-flight LiDAR equipment must provide a constant correction to each measured range. This standard range correction is used, mainly, to account for the distance traveled by the laser pulse from the laser emitter to the focusing mirror and the distance traveled by the returning pulse from the focusing mirror to the receiver. The size of these range corrections varies depending on the LiDAR design and configuration. In truth, this range correction is designed to take in all sources of range bias, the laser path out of and into the unit is simply the most significant. Other factors causing range bias could include, but are not limited to, refraction delay through the LiDAR window, refraction delay through a beam expander (if present) and reflection delay off the various mirrors inside the unit. In fact, it has been shown that a component of the zero error, which has been labelled range/reflectance crosstalk, can be linked to the dependence of the measured range on the surface reflectance [52]. The concept of zero error is explained in multiple papers and textbooks such as [53, 54]. The zero errors and their standard deviations for several time-of-flight and phase based terrestrial laser scanners were collected and summarized by [15]. These values are reproduced in Table 2.4.

Scale error is an error in the scale factor being applied to the measured distance. Not all time-of-flight LiDAR systems are calibrated to use this scale factor. The ILRIS HD and ILRIS LR do not report a scale factor as part of their specifications [35]. The Riegl LMP321 on the other hand reports a scale error of +20 ppm as part of its specification [36]. In addition, independent testing [19] has shown that the Leica HDS 2500 has a scale error as high as +400 ppm.

Table 2.4: Zero error estimates for various time-of-flight and phase based STL[15].

Scanner	Zero Error (mm)	Standard Deviation (mm)	Source
Lecia HDS 3000	6.1	No Data	[55]
Lecia HDS 3000	2.9	0.6	[56]
ILRIS 3D	-41.0	11.0	[56]
I-Site R-350	21.0	3.0	[56]
Imager 5003	4.0	No Data	[57]
Imager 5003	0.73	0.36	[58]
iQsun 880	7.4	0.1	[59]

Zero and scale errors are applied directly to laser range measurements as shown in Equation (2.15).

$$\Delta D_{ij} = D_{ij} \cdot \delta_S + K \tag{2.15}$$

Where ΔD_{ij} is the error in the measured range, D_{ij} is the measured range, K is the zero error and δ_S is the scale error.

The effect of zero error on a LiDAR unit's output point cloud is a warped planar distortion. If all ranges measured by the LiDAR unit are either too short or too long, then one can expect a planar object, perpendicular to the scanning direction, to curve around the scanner centre. This effect has been likened to a smile (or frown) by [13, 60]. Figure 2.15 shows Lynx Mobile Mapper data of a parking lot compared with a previously established control surface. The control surface, having been established from total station observation, and adjusted to the same GPS base station as the Lynx data, is expected to show a more accurate representation of the parking lot than the Lynx. The black dotted line through the scan shows the path on the vehicle through this data section. The colour scale on the error map shows that the further from the vehicle trajectory a data point lies, the more that point deviates above the control surface. Directly along the vehicle path, the point cloud to surface deviation is approximately 4mm. Along the edges

of the established cross section, the point cloud deviates over 4cm from the control surface. The deviation is approximately symmetrical on either side of the vehicle's trajectory.

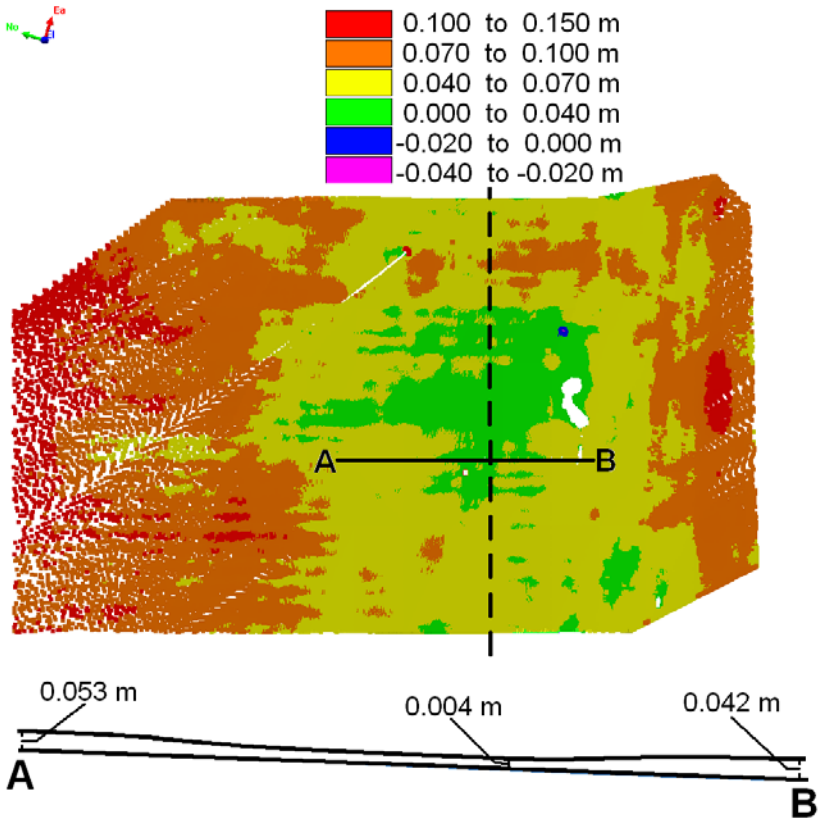

Figure 2.15: Lynx Mobile Mapper scan of a parking lot compared with a control surface established by total station observation.

The bias error depicted in Figure 2.15 is what is referred to in literature as a "Smile". This type of bias error can cause a serious distortion in LiDAR data and must be reduced or eliminated before data collection proceeds.

2.2.1.5 Temperature Drift

Temperature changes in a laser range finder occur because the two factors cited by [15]. These two factors are changes in the external ambient temperature surrounding the

instrument and internal temperature changes caused by radiant heat generated by the system's own electronics. The greatest change usually occurs during the first 20 minutes [15] after the LiDAR system has been turned on. It takes about this amount of time for the internal electronics to warm up after a cold start. Once the unit has reached a stable temperature, the majority of temperature drift will be caused by changes in the ambient temperature, external to the LiDAR sensor. This phenomenon has been described in literature for many different time-of-flight LiDAR systems. Table 2.5 summarizes the results from these various published trials.

Table 2.5: Range changes due to temperature drift from literature sources.

Scanner	Measurement Interval (Hours)	Temperature Change (ºC)	Range Change (mm)	Source
Callidus CP 3200	3.5	16	-3	[15]
Leica HDS 3000	1.5	11	-2	[15]
Lecia HDS 2500	3	15	+3	[15]
LR-2000	1/3	4	-20	[48]
Perceptron	1/2	24	-400	[52]
Imager 5003	2		2	[57]

Careful design and implementation of the laser range finder can reduce the range drift to the millimetre level [49]. Chapter 4 deals with experiments concerning the temperature drift in MTL sensors. Specifically, Section 4.2 describes a laboratory calibration method which can compensate for the effects of temperature drift in MTL data.

2.2.1.6 Mixed Pixels

Mixed pixel is a photogrammetry term used in literature [15] to describe a phenomenon which can be a serious problem in all time-of-flight LiDAR systems. It refers to the ability of a time of flight LiDAR to distinguish individual range returns from multiple surfaces in the field of view (as shown in Figure 2.16).

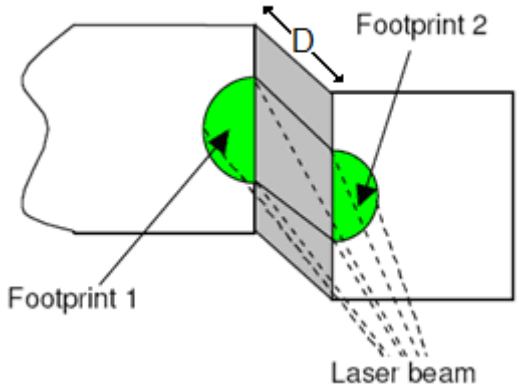

Figure 2.16: Separation of laser beam footprint due to a surface edge [61].

For many LiDAR systems it is expected that the minimum point pair separation is half the pulse width of the outgoing laser beam [56]. When the distance, D, in Figure 2.16 is larger than the minimum point pair separation for the LiDAR, then two ranges will generally be recorded as separate return values. If the distance, D, in Figure 2.16 is smaller than the minimum point pair separation distance for the LiDAR, the receiver will not be able to distinguish between returned pulses [56]. In this event the returned range will not be to any one surface but instead be derived from a combination of all surfaces within the footprint of the laser beam. This error may range from a few millimetres to several decimeters [62].

2.2.1.7 Dynamic Track Error

This angular error comprises the total mechanical angular variation of the laser beam orthogonal to the scan line. Typically this error is caused by a combination of regular and irregular deviations in the mirror surface, deviation of the mirror surface from its design angle and random non-repeatable errors caused by the bearing support system [63]. This error is depicted in Figure 2.17, were the angle α represents the dynamic track error.

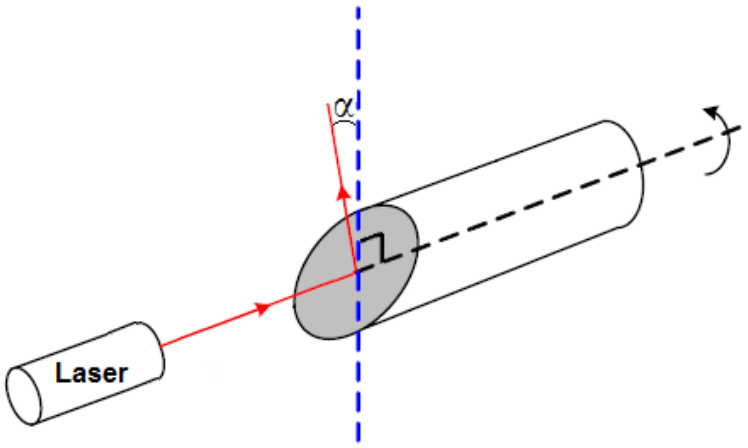

Figure 2.17: Dynamic track error in a monogon based laser scanning system.

2.2.1.8 Velocity Errors

Velocity errors include jitter, rotational speed variation and synchronization errors. Jitter is a high frequency variation in laser spot placement along a scan line [15]. This variation can be random or in a repeating pattern. Rotational speed variation occurs within a scan line over multiple revolutions. These first two velocity errors occur due to external vibration or shocks to the sensor mount, the inertia of the mirror, stability of the electric motor, oscillation of the motor speed, bearing resistance and mirror surface roughness [15]. Jitter and rotational speed variations can be seen in Figure 2.18.

Synchronization errors occur at the beginning and end of a scan. While the scanner is accelerating and decelerating the monogon, positional errors for laser shots increase. This type of error generally lasts 3 to 60 seconds [15].

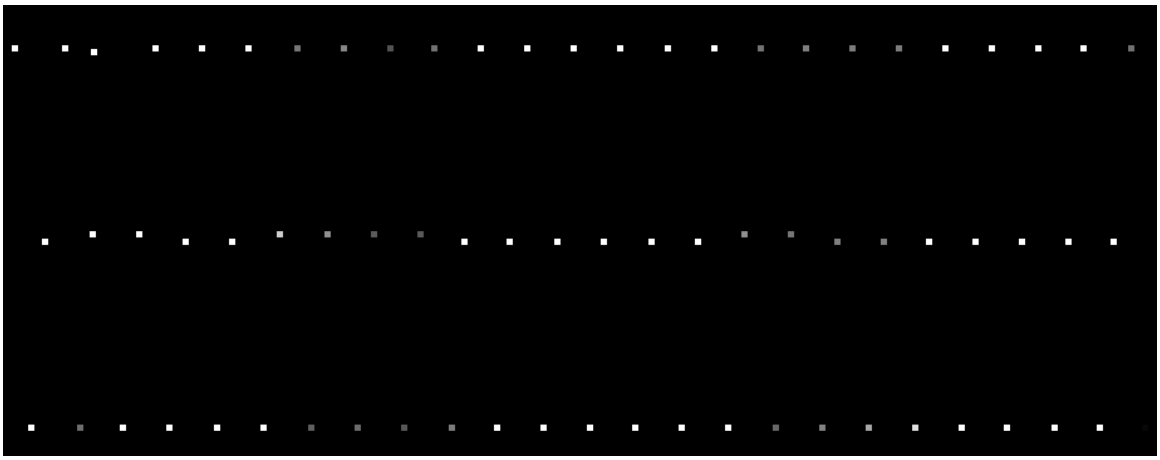

Figure 2.18: Placement variation in Lynx Mobile Mapper data due to velocity errors.

2.2.1.9 Scanner Assembly Balance

Any object spinning at high velocity is subject to balancing issues. Weight inconsistencies in the scanner assembly can cause vibration in the rotating monogon [15]. This in turn can result in positional errors in laser placement during the scan. Proper balancing of the scanner assembly can reduce this error, but generally does not eliminate it.

2.2.2 Environmental Errors

2.2.2.1 Atmospheric Propagation Errors

The atmosphere affects the laser beam by distorting [64] and attenuating the pulse. Attenuation of the pulse is a result of Rayleigh type (air molecules) and Mie type (aerosol particles) scattering as well as absorption due to water vapor, carbon dioxide and/or ozone [15]. The amount of attenuation is dependent on the laser wavelength, the range to target, ambient temperature, atmospheric pressure, gaseous composition of the atmosphere, weather conditions and particulate matter in the air.

Attenuation was described by [54] and is shown in Equation (2.16).

$$I_R = \frac{I_0}{R^2} \cdot T_a \tag{2.16}$$

Where I_R is the transmitted laser radiant intensity, I_0 is the emitted laser radiant intensity, R is the range to target and T_a is the atmospheric transmittance. Atmospheric transmittance can be determined by Equation (2.17) [65].

$$T_a = e^{-\gamma_{att}^R} \tag{2.17}$$

Where R is again the range to target and γ_{att} is the attenuation coefficient. The attenuation coefficient is calculated by Equation (2.18) [66].

$$\gamma_{att} = \alpha_m + \beta_m + \alpha_a + \beta_a \tag{2.18}$$

Where α_m is the molecular absorption coefficient, α_a is the aerosol absorption coefficient, β_m is the molecular scattering coefficient and β_a is the aerosol scattering coefficient.

Many MTL units currently in operation use lasers with wavelengths of 1500nm. This means that for all these units the atmospheric transmittance (T_a) is close to 100% [54]. Since the typical operational range of an MTL system is less than 200m [1], the effects of atmospheric attenuation are much reduced from those effecting total stations. Combined with the monochromaticity of this type of LiDAR, little dispersion occurs in air and the dispersion which does occur may be modeled more closely [15].

2.2.2.2 Adverse Weather Conditions

Attenuation of the laser beam is wavelength dependent. For the common MTL wavelength of 1500 nm, at sea level and a range of 100 m, the attenuation coefficient for clear atmospheric conditions (15km visibility) changes from 0.28km^{-1} ($T_a = 97\%$) to 0.16km^{-1} ($T_a = 98\%$) [65]. During adverse weather conditions such as rain, fog, haze or snow, attenuation is mainly caused by scattering. When the size of the particulate matter in the air matches the laser wavelength, such as when haze or fog is present, Mie-scattering occurs. Haze and fog therefore lead to a significant increase in the variation of the attenuation coefficient γ_{att}. During medium haze (5km visibility) the attenuation coefficient for the same 1500 nm wavelength laser at sea level, changes from 0.85km^{-1} to 0.47km^{-1} [65]. In fact, [66] shows that dense fog is more problematic for laser ranging than heavy rain.

During scanner operation, dropout ranges and false returns commonly occur during adverse weather due to laser beam attenuation [67]. Dropout ranges are those laser pulses whose returned intensity is too weak to trigger a range detection and therefore no range is recorded. False returns occur when beam scattering due to raindrops, aerosols or particulate matter, cause the receiver to pass the detection threshold and record multiple ranges. Some laser energy will invariably reflect from the airborne particulates, while others will reflect off the target surface. Experiments carried out by [67] have shown a linear relationship between the number of dropout ranges/false returns and the rainfall rate. It was found that the number of dropout ranges greatly exceeds the number of false

returns for the same rainfall rate. It was also found that the number of dropout ranges/false returns exponentially increases as the visibility decreases in fog and rain.

2.2.2.3 Other Atmospheric Conditions

Since pulsed time-of-flight LiDAR is dependent on highly precise time measurements, the propagation of the light beam through the air is an important factor in determining accurate range measurements. The velocity of the laser beam through the air depends on the refractive index of the air. The refractive index is in turn dependent on the temperature, pressure, relative humidity and carbon dioxide content of the air.

Several models exist for the refractive index of air. One of the more recent models, proposed by [68, 69], is used by the Leica software package Cyclone [15] to correct atmospheric effects. The model proposed by [68] is given in Equation (2.19).

$$n_{prop} - 1 = \frac{\rho_a}{\rho_{astd}} \cdot \left(n_{astd} - 1\right) + \frac{\rho_w}{\rho_{wstd}} \cdot \left(n_{wstd} - 1\right) \tag{2.19}$$

Where n_{prop} is the group refractive index of air, ρ_a is the actual density of dry air, ρ_{astd} is the standard density of dry air, ρ_w is actual density of pure water vapour, ρ_{wstd} is the standard density of pure water vapour, n_{astd} is the group refractive index of dry air under standard conditions and n_{wstd} is the group refractive index of water vapour under standard conditions. The standard conditions of dry air are given in [68] as 15°C and 1013.25mbar. The standard conditions of water vapour are given in [68] as 20°C and 13.33mbar.

To compute the indexes of dry air and water vapour under standard conditions, [68] provides the formulas shown in Equations (2.20) and (2.21).

$$10^8 \cdot (n_{astd} - 1) = \left(\frac{k_1 \cdot (k_0 + \sigma^2)}{(k_0 - \sigma^2)^2} + \frac{k_3 \cdot (k_2 + \sigma^2)}{(k_2 - \sigma^2)^2} \right) \cdot (1 + 0.534 \cdot 10^{-6} \cdot (x_c - 450)) \quad (2.20)$$

$$10^8 \cdot (n_{wstd} - 1) = 1.022 \cdot (w_0 + 3 \cdot w_1 \cdot \sigma^2 + 5 \cdot w_2 \cdot \sigma^4 + 7 \cdot w_3 \cdot \sigma^6) \quad (2.21)$$

Where x_c is the carbon dioxide content of the air in parts per million. It was estimated [68] that the uncertainty for these refractive indexes is on the order of 0.06ppm~0.15ppm. The coefficients for these equations are listed in [68] and have been reproduced in Table 2.6.

Table 2.6: Coefficients for the indexes of dry air and water vapour under standard conditions [68].

Coefficient	Numerical Value
k_0	238.0185 μm^{-2}
k_1	5792105 μm^{-2}
k_2	57.362 μm^{-2}
k_3	167.917 μm^{-2}
w_0	295.235
w_1	2.6422 μm^{2}
w_2	-0.032380 μm^{4}
w_3	0.004028 μm^{6}

Compared to the other standard models, the above model has the advantages of being valid for the range of wavelengths employed by all modern laser scanners and it includes the ability to account for the variation of carbon dioxide content in the equations. Due to the relatively short ranges (> 200m) measured by MTL it is sufficient to only consider atmospheric parameters at the instrument itself [54].

The refractive index of air can vary under the influence of turbulence. Refractive index fluctuations due to atmospheric turbulence have been computed by [66] and the model is given in Equation (2.22).

$$\delta n = \frac{79 \cdot 10^{-6} \cdot p}{(\gamma' - 1) \cdot T^2} \cdot \delta T \tag{2.22}$$

Where $\gamma' = 1.4$ for air, T is temperature in Kelvin, p is pressure in mbar and δT is the temperature fluctuation. The influence of this error cannot be ignored, especially in high accuracy work, since [70] showed that the accuracy of a scan can be decreased by up to 4 mm due to atmospheric turbulence.

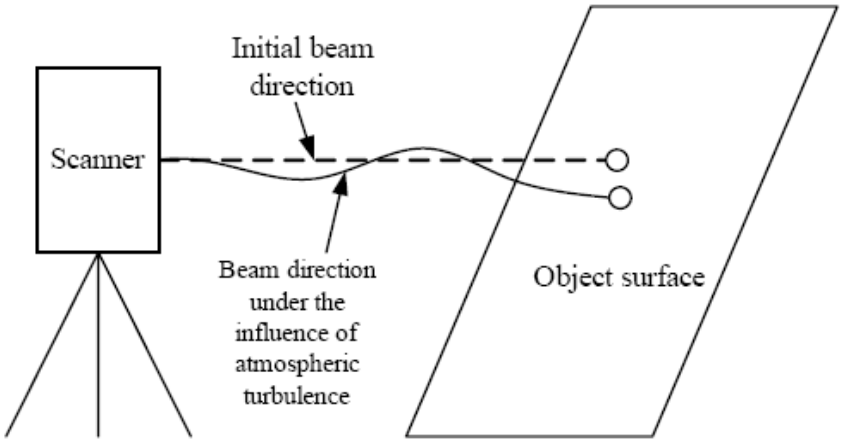

Figure 2.19: Beam refraction through the atmosphere [66].

Atmospheric turbulence can cause other effects as well. Notably, refraction or the random displacement of the laser beam from its original path [66] while the beam footprint remains constant. The variance of this beam refraction has been described in [66] by Equation (2.23)

$$\sigma_r^2 = 1.83 \cdot C_n^2 \cdot \lambda^{-1/6} \cdot R^{17/6} \tag{2.23}$$

Where σ_r^2 is the variance of the beam wander, C_n is the refractive index coefficient which depends on the magnitude of the turbulence, λ is the laser wavelength and R is the range to target. Typical values for C_n include $5 \cdot 10^{-7}$ for strong turbulence, $4 \cdot 10^{-8}$ for medium turbulence and $8 \cdot 10^{-9}$ for weak turbulence [66]. Figure 2.19 illustrates beam refraction

through at turbulent atmosphere. Previous research has found that the amount of refraction at 100m for various terrestrial based LiDAR scanners is fairly constant [15]. It was found that with the Leica HDS 3000, the beam refraction was approximately 1.5 mm at 100m.

Beam scintillation or beam intensity fluctuations can also be caused by a turbulent atmosphere. This type of effect can cause the Gaussian footprint of the laser pulse to breakup into smaller "hot spots" [15]. It is dependent on the temperature differential between the scanner and the target, as well as, the height of the instrument above the ground. The greater the temperature differential or the lower the instrument is to the ground, the more likely that scintillation will occur.

2.2.2.4 Interfering Radiation

External radiation from natural (sunlight) or artificial (lamps) sources can cause inaccurate range measurements if the external radiation is considerably stronger than the detected laser energy [62]. The amount of external radiation can be greatly reduced by placing a narrow-band optical filter between the lens and the detector [48]. Under most circumstances the problem of external radiation might be avoided by scanning at night. Some problems such as the loss of points on target surfaces have been reported when scanning at night [71].

2.2.2.5 Scanner Instability and Vibration

During MTL operations, vehicle vibration or mount instability can degrade the accuracy of the data collected. Vibration during scanning may cause mirror phase errors which in

turn induce errors in the measured angles [52]. This causes point clouds to become distorted and complicates data extraction and registration [72, 73].

The most common way to deal with shock or vibration is to use an isolator. An isolator absorbs or reduces external forces before they have a chance to affect the sensor. Isolators come in several forms, including elastometric pads, housed springs and pneumatic systems. Each of these isolators can reduce vibration with natural frequencies of 5 to 20 Hz (pads), 1 to 6 Hz (springs) and 0.5 to 5 Hz (pneumatic systems) [15].

Another approach involves compensation for vibration during data processing. One approach, presented in [74], uses a rough, distorted model of the moving object. It then employs a recursive optimization algorithm using the acquired range information to better estimate the relative motion between object and scanner. The procedure is based on the iterative closest point algorithm (ICP). Tests presented in [74] show that accuracy of the obtained 3D model was comparable to the case where the scanner was not vibrating at all.

2.2.3 Instrument Position Errors

2.2.3.1 Trajectory Errors

Trajectory errors are produced in the DG system used to position the LiDAR sensors. Position errors are largely due to the GNSS receivers built into the DG system. Most of the time the positioning system is a combination of the GNSS receivers and the IMU, however, during GPS outages, the positioning solution is entirely dependent on the IMU.

The causes of these errors are atmospheric delay, cycle slips, erroneous ambiguity resolution (multipath), IMU bias and IMU drift [13].

Attitude errors are largely due to errors in the IMU. The IMU, consisting of accelerometers and gyros, is subject to time dependent drift. This drift becomes more pronounced when the IMU is not kept constantly in motion. The reported attitude of the LiDAR sensors depends on the quality and frequency of the IMU and the method by which the IMU and GNSS data is combined [13]. The Kalman filter used to produce DG system trajectories can be loosely coupled or tightly coupled. A loosely coupled solution means that the IMU and GNSS data are processed independently and then combined after processing using time stamp interpolation. A tightly coupled solution means that the IMU and GNSS data are processed together through the Kalman filter, providing a better solution overall. The Kalman filter itself is also constantly being updated and improved. In fact, [75] did a quality assessment between two versions of the same post-process trajectory software (POSPac 4.4 and POSPac 5.0). They were able to obtain error estimates for POSPac 5.0 that were almost 50% smaller for positional accuracy and 2.87% smaller for orientation accuracy, for the same DG system hardware.

2.2.3.2 Boresight and Lever Arm Errors

The terms boresight and lever arms are used to describe the fixed offsets between the LiDAR sensor and the DG system. Figure 2.20 shows a typical MTL system setup along with the positions and orientations of the various coordinate frames involved. These angular and vector offsets play a critical part in the calibration of the MTL system. Boresight and lever arm errors in MTL data are caused by inaccurate position and

orientation information for the LiDAR sensor within the DG system coordinate frame. Depending on the local coordinate frames of the sensor and the DG system, as established by their respective manufacturers, the terms heading, pitch and roll might swap meaning. In other words what is called a pitch error in one MTL system may be called a roll error in another MTL system. The effects of these errors, however, will remain constant and will present themselves in the output point cloud in similar ways.

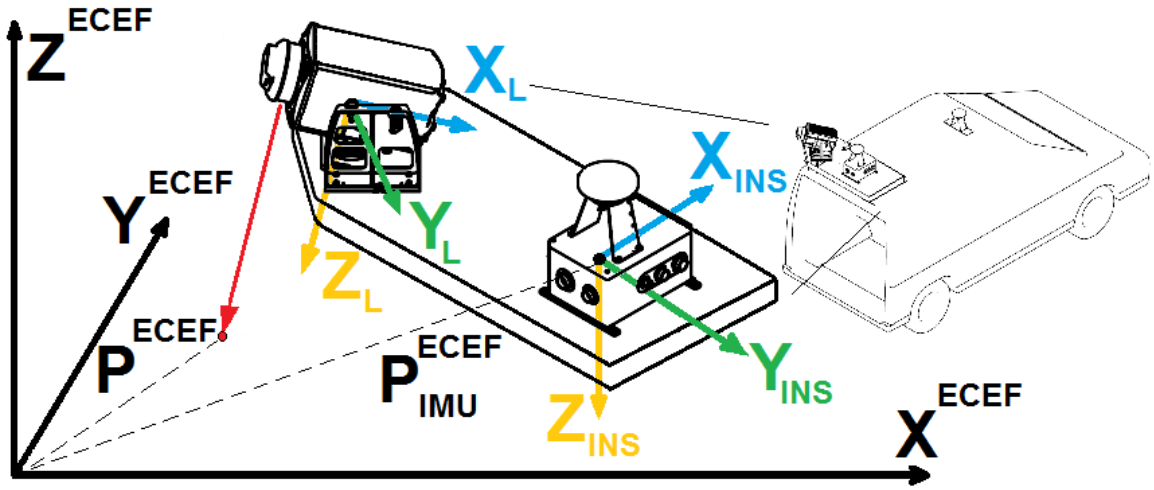

Figure 2.20: LiDAR and DG system coordinate systems for the Lynx Mobile Mapper and how they relate to the Earth Centered Earth Fixed (ECEF) coordinate frame.

2.2.3.2.1 Heading Errors

Heading errors generally refer to errors in the fixed Euler rotation angle around the vertical axis of the DG system coordinate frame. In practically all cases this means a rotation around the Z axis of the DG system. For all MTL systems that use a spinning monogon mirror, an error in this boresight parameter will cause flat surfaces to loop back on themselves when scanned from more than one point along the vehicle's trajectory. This data artifact has been termed a "fishhook" by some literature sources [76].

Figure 2.21: Heading error in Lynx Mobile Mapper data which causes a flat surface to loop back upon itself. This is known in literature as a "fishhook" [76].

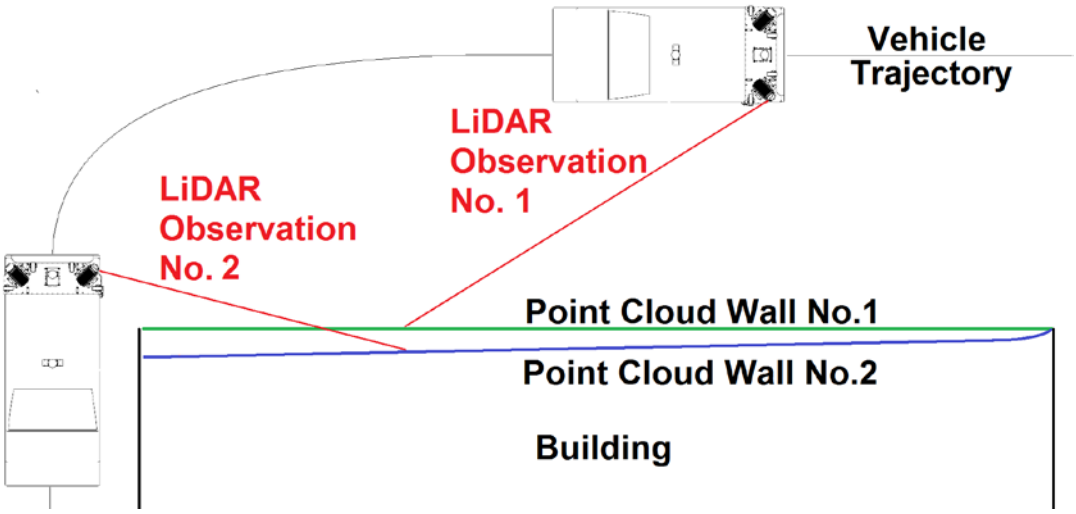

Figure 2.22: Creation of fishhooks in LiDAR data from multiple observations of a planar surface.

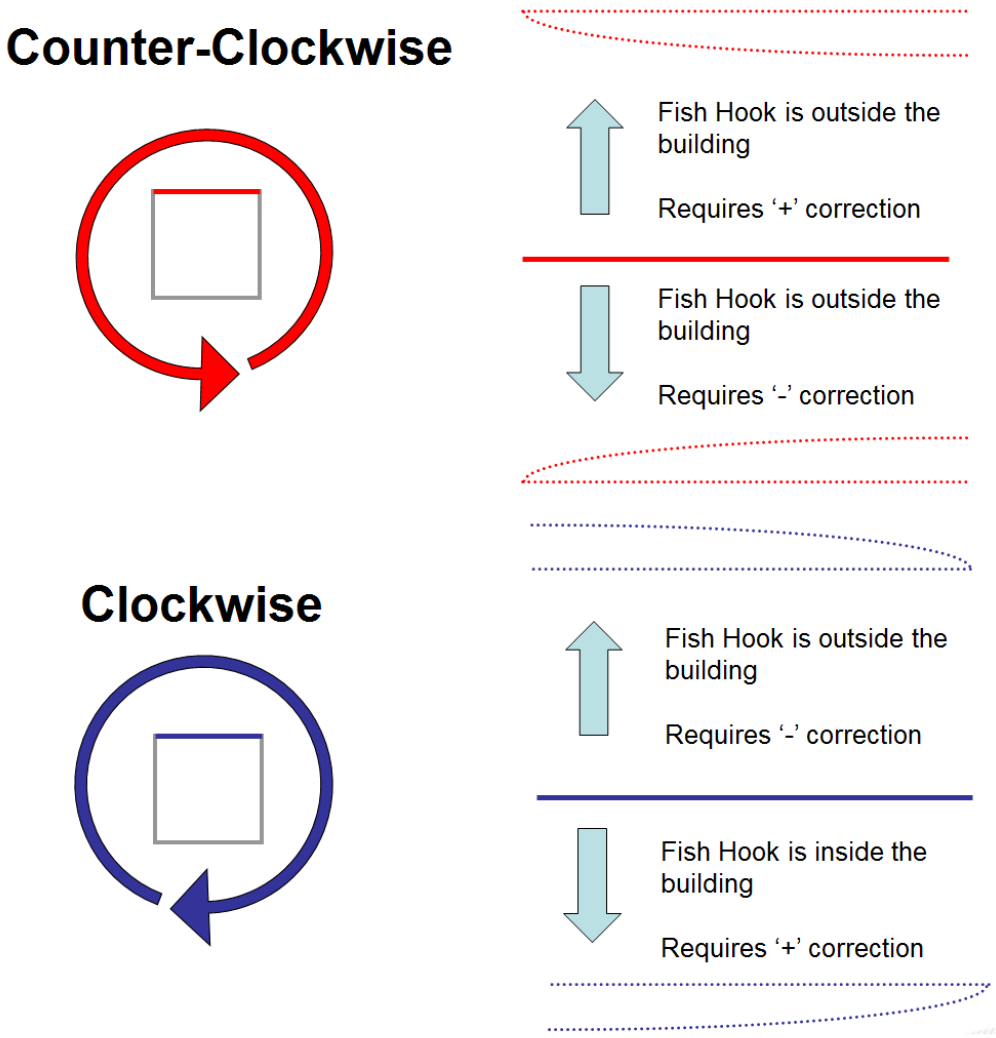

Figure 2.23: Relationship between the direction the vehicle is driven around a building, the sign of the heading error and the type of fishhook [77].

Figure 2.21 shows a fishhook in Lynx Mobile Mapper data. The fishhook in Figure 2.21 was caused when the Lynx system scanned a flat building wall, first as it passed the wall, and then again as it turned a building corner. Turning the building corner caused the LiDAR to get a view of the wall a second time at a longer range. The error in the boresight heading angle caused the measured LiDAR points from the first observation of the wall to be offset from the measured LiDAR points from the second observation of the wall (Figure 2.22). The direction that the vehicle was traveling around the building

combined with the size and sign of the heading error in question, will determine which walls will present a fishhook, the size of the fishhook and whether the fishhook will appear outside the building or inside the building. Figure 2.23 shows this relationship and lays out the different types of fishhooks that may be encountered based on the heading error.

2.2.3.2.2 Roll and Pitch Errors

Depending on the coordinate system of the MTL system in question the roll and pitch errors maybe interchanged. Roll errors are generally recognized as errors in the fixed Euler rotation angle about the DG system coordinate axis in the direction of motion of the vehicle. Pitch errors generally refer to errors in the fixed Euler rotation angle about the DG system axis perpendicular to the direction of motion of the vehicle. For MTL systems like the Lynx Mobile Mapper, the boresight roll angle occurs around the X axis of the DG coordinate system and the boresight pitch angle occurs around the Y axis of the DG coordinate system. Errors in both the roll and pitch parameters will be exposed in the point cloud when opposing drive passes of an object from the same sensor are compared. This technique, sometimes referred to as a "Patch Test", is standard for correcting roll and pitch errors in MTL systems and is documented in [6, 7, 76 – 83].

Figure 2.24 illustrates the effect of roll errors on planar surfaces passed by the Lynx system. If the system is driven in opposing direction (read and blue arrows) planar surfaces perpendicular to the direction of motion of the vehicle will shift as shown in Figure 2.24.

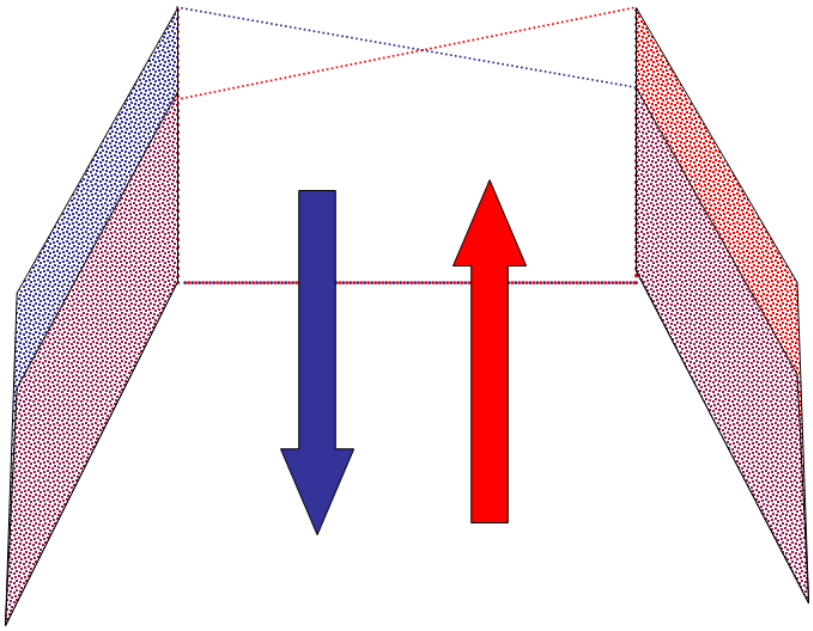

Figure 2.24: The effect of a roll error in Lynx Mobile Mapper data on vertical surfaces [77].

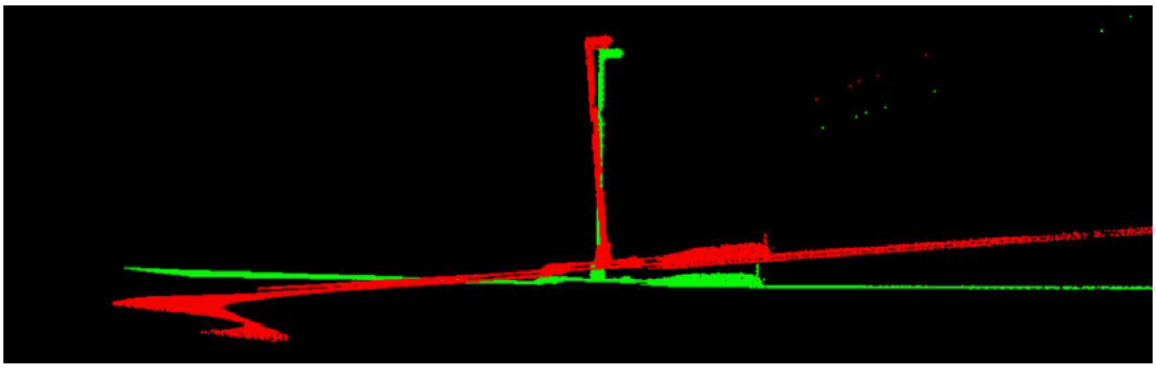

Figure 2.25: Roll error in Lynx Mobile Mapper data which causes a separation in opposing drive passes of a flat road surface [76].

The effect of a roll error in real LiDAR data can be seen in Figure 2.25. In Figure 2.25 Lynx data from opposing drive passes in a parking lot is shown. The crossed point clouds in Figure 2.25 intersect at the vehicle trajectory and the distance between the point clouds linearly increases with range.

Figure 2.26 illustrates the effect of a pitch error on vertical planar surfaces collected on opposing drive passes by a Lynx system. A pitch error generally presents itself as a lean

in flat vertical objects when two opposing drive passes from the same sensor are compared.

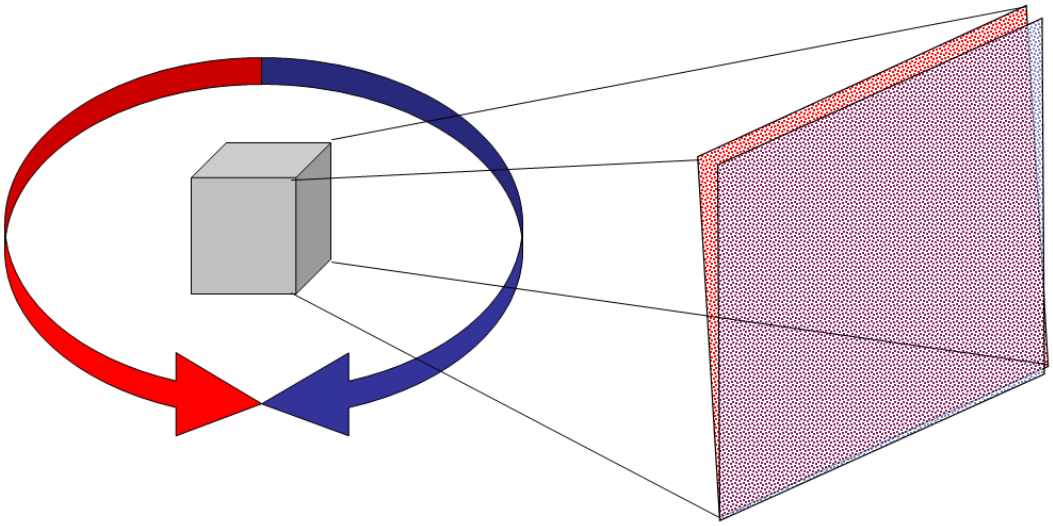

Figure 2.26: The effect of a pitch error in Lynx Mobile Mapper data on vertical surfaces [77].

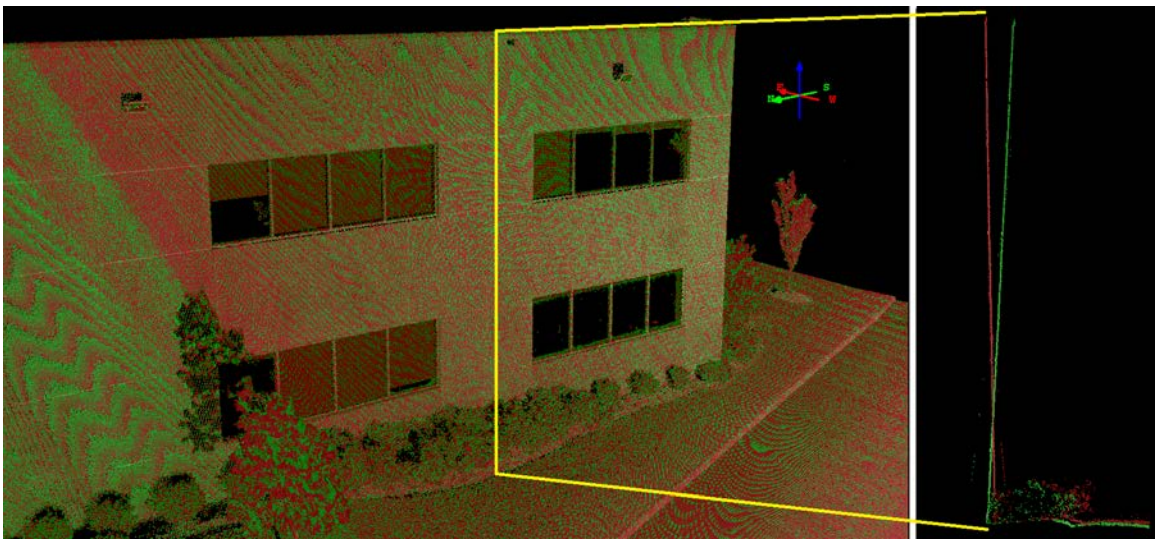

Figure 2.27: Pitch error in Lynx Mobile Mapper data which causes a separation in opposing drive passes of a flat surface [76].

The effect of a pitch error in real LiDAR data can be seen in Figure 2.27. As the LiDAR is driven around the building in Figure 2.27, first clockwise and then counterclockwise, the pitch error in the sensor causes the point cloud of the imaged building walls to lean either toward the sensor or away from the sensor depending on the direction traveled.

When the two data strips are viewed together the V shape pattern in a cross section of the building wall can be seen.

2.2.3.2.3 System Lever Arm Errors

A MTL system contains several offsets that are generally referred to as lever arms. Lever arm offsets are required for the offset between the LiDAR and the IMU, the GNSS antenna and the IMU and the DMI and the IMU. Several methods exist for determining the lever arms for a MTL system, but the most common is direct measurement with a Total Station or tape measure. For some MTL systems the lever arms are provided as a series of fixed numbers determined by the manufacturer [84]. Being that lever arms are linear offsets from the system's coordinate origin, errors in the lever arms cause linear position changes in point cloud data. Figure 2.28 shows the effect of a lever arm error on a random point in a LiDAR point cloud. If any particular lever arm value contains an error of size H and the MTL system is currently oriented within the survey coordinate frame at an angle of α, then a LiDAR point P will contain errors C and S, according to Equations (2.23) and (2.24).

$$S = H \cdot \sin(\alpha) \tag{2.23}$$

$$C = H \cdot \cos(\alpha) \tag{2.24}$$

The errors S and C will cause the position of point P, in Figure 2.28, to be incorrectly calculated at point P'. In some systems secondary effects of lever arm errors may occur if S and C are larger than a few centimeters. Surfaces scanned while the MTL system is changing direction can exhibit bulges when lever arm errors are large. These bulges are caused by the off centre positioning of the LiDAR sensor with respect to the IMU.

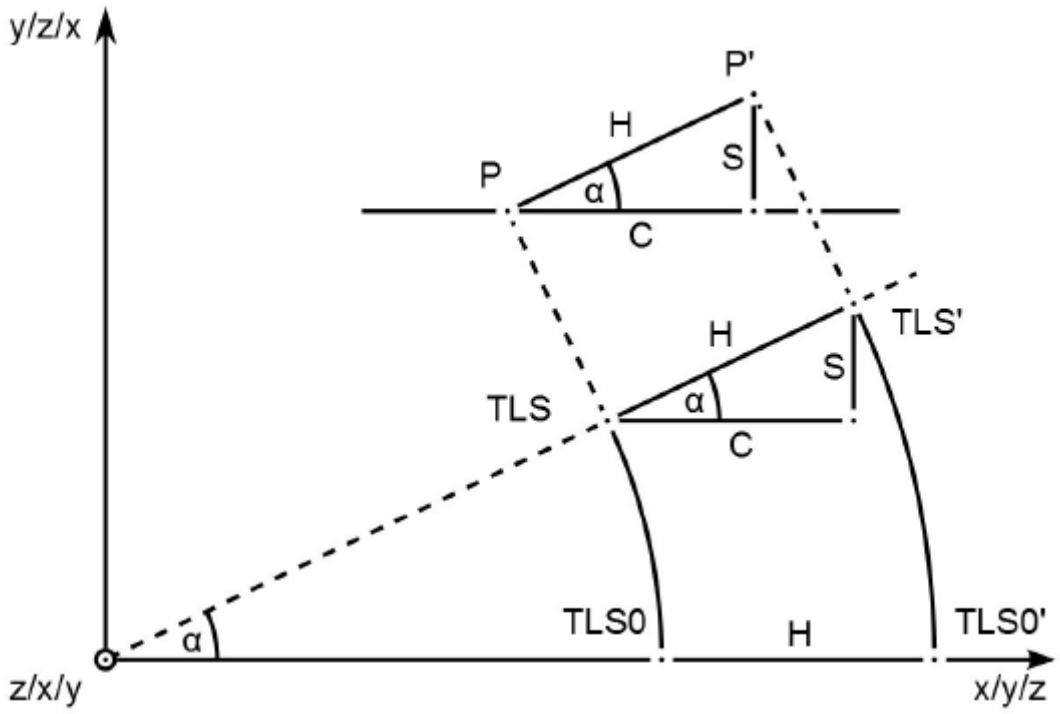

Figure 2.28: The effect of lever arm errors on an individual point in a MTL point cloud [84].

2.2.3.2.4 Methods for Computing Boresight Angles and Lever Arms

The documented methods for determining sensor boresight and lever arm parameters include manual visual methods [76], software assisted visual methods [78 – 82] and algorithm based methods. This section will focus on the rigorous algorithm based methods of computing LiDAR to DG system boresight angles and lever arms.

Much work has gone into developing algorithms for determining the boresight angles and lever arms between airborne LiDAR sensor(s) and a DG system [8, 14, 22 – 24, 85]. Much of this work has been done using Mobile Airborne LiDAR (MAL). MAL differs significantly from MTL in the ranges being considered (MAL 50-1100m [86], MTL 0-250m [2]), the scanner field of view (MAL 50° [86], MTL 360° [2]) and the accuracies required (MAL 30-70mm [86], MTL 50mm [2]). In many instances MAL is separated

from its target objects, giving little relief between objects and poor geometry for calibration purposes. MTL by contrast is usually immersed in its target field, providing diverse geometry for use in calibration.

Recently several algorithm based methods for boresighting MTL systems have been proposed in [6, 7, 9 – 12, 83, 87]. The techniques for boresighting MTL and MAL share the assumption that the lever arm offsets for the LiDAR sensor are better determined through some other means, usually mechanical drawings. Mechanical drawings may not always be provided, and even when they are, it is possible that small defects in assembly or changes in the DG system reference point may make these drawings obsolete. If reference points are not provided on the scanner itself, accurate physical measurement may be difficult or impossible. In these cases, it would be useful to obtain good estimates of the lever arms in another way.

The first approach to calibrating MAL systems was introduced in [14]. Further testing of this approach was conducted in [22]. This method proposed two separate algorithms for calculating the boresight angles, 1) comparing MAL data to ground control, 2) comparing overlapping strips of MAL data, collected from different perspectives in order to get the necessary geometric dispersion. The mathematical theories behind these approaches are described in Sections 4.3.2.1 and 4.3.2.2 respectively. Common points between the MAL data and the control field or between opposing MAL strips were extracted and used as observations in the adjustment. Lever arm estimates were excluded from this calculation

and the algorithms assume that the LiDAR is oriented such that the boresight angles should all be close to zero.

A second approach to calibrating a mobile LiDAR system [6] extracts planar features from overlapping strips of data and performs a similar adjustment to that described by [14]. Some literature [7] has taken this further, extracting both planar and catenary features from overlapping scans of MTL data. In the case of [6], the processed described only one LiDAR sensor, however, [7] describes preforming this process on multiple LiDAR sensors at once. The functional models given in [7] were separately established for the individual LiDAR sensors, i.e. the adjustment of the individual sensors was done piecemeal, and in this way makes the boresight adjustment similar to [6]. In addition, no consideration is made for the lever arms as these are assumed to be better estimated through direct measurements.

A third approach for calibrating the boresight angles of LiDAR to a DG system involves using multiple LiDARs from a static position [9, 12]. By being static, the positioning components, both at the component level and the system level can be removed from consideration. This method is also based on planar extraction and comparison, but in this case the point clouds come from different LiDAR sensors and not from multiple passes of an object with a single sensor. A similar approach was adopted by [83], in that a series of static scans would be collected under laboratory conditions and used to calibrate the boresight angles between a single LiDAR sensor and the DG system. However, in this case only one LiDAR system was used and the static scans were conducted from multiple

different perspectives to the elliptical calibration target. The methodology in [83] assumes that the LiDAR system is a two axis scanner capable of creating 3D point clouds without the aid of motion from a vehicle.

A fourth approach to calibrating MTL systems [87], adjusted a series of LiDAR sensors on a mobile device to a reference MTL point cloud. The reference MTL point cloud would have been calibrated in a standard way using multiple passes of an object by the same sensor. The authors stated that "the reference LiDAR is regarded as a target for another LiDAR, so the calibration error of the reference LiDAR will pass to the calibrated LiDAR". They also assumed that the trajectory of the vehicle would be straight, so that the calibration would remain an invertible affine transformation.

A fifth approach [10], also used a reference LiDAR scanner to calibrate multiple other LiDAR systems. This approach was presented as a two-step process where the first step involved differencing the sensors between a reference LiDAR and multiple other LiDAR scanners under laboratory conditions. The second step involved calibrating the reference LiDAR sensor with respect to the INS under real world conditions by relying on a mixture of techniques, some of which require multiple passes of a calibration object.

A sixth approach [11] uses planar geometry from multiple passes of a calibration object to simultaneously adjust the internal scanner parameters of a Velodyne HDL-64E S2 scanner and calculate its boresight offsets from the DG system to which it's attached. Calibration of the Velodyne HDL-64E S2 scanner is required, since unlike other LiDAR

line scanners, this scanner collects 64 lines of point data simultaneously by splitting its outgoing laser beams and then using an array of receivers to collect the returning laser beams. Like most other methods, this method ignores the determination of the lever arms.

A study of the effect of bias in the lever arms of the linear scanner type of MAL system was undertaken in [85]. Simulated data was used to test out both boresight and lever arm biases in the calibration parameters. No control was considered and as such the vertical lever arm component was excluded from analysis as there was insufficient geometry provided by the multiple flight lines considered to estimate this parameter. In addition, the functional model used in [85] was designed for MAL systems and under the assumption that the system was vertical (i.e. Pitch and Roll were almost zero) and the boresight angles were all close to zero.

2.2.4 Target Errors

2.2.4.1 Object Reflectance

The percentage of incident light returned by an object greatly affects the ability of the LiDAR to make accurate range measurements between the scanner and said object. It affects the signal to noise ratio to a great extent [88] and induces a time walk error in the sensor for different returning light intensities. The reflectance of an object is mainly determined by the material properties of that object, such as electric permittivity, magnetic permittivity and conductivity. Other factors affecting the reflectance of an object include the colour, roughness, temperature and moisture content of the object

surface, as well as the wavelength of the laser used and the incidence angle of the laser beam.

The relationship between the returned laser intensity and surface reflectance is given by [89] for both Lambertian and specular reflectance. The relationship for Lambertian reflectance is given by Equation (2.25).

$$I \propto \frac{\rho \cdot \cos\beta}{R^2} \tag{2.25}$$

Where I is the returned intensity, ρ is the surface albedo, β is the incidence angle and R is the range to target. The relationship for specular reflection is given by the Torrance and Sparrow model shown in Equations (2.26) and (2.27).

$$I = k \cdot \rho \cdot \left(C \cdot \cos\beta + \left(\frac{1-C}{\cos\beta} \right) \cdot e^{-\beta^2 / 2\cdot\sigma^2} \right), \quad 0 \le \beta \le \frac{\pi}{4} \tag{2.26}$$

$$I = k \cdot \rho \cdot \cos\beta \cdot \left(C + 2 \cdot (1-C) \cdot e^{-\beta^2 / 2\cdot\sigma^2} \right), \quad \frac{\pi}{4} \le \beta \le \frac{\pi}{2} \tag{2.27}$$

The Torrance and Sparrow model adds terms for surface roughness (σ), the percentage of laser energy reflected diffusely (C) and a constant value (k).

In many time of flight LiDAR systems that use a threshold method of determining ranges, a range-intensity table is used to correct for this error [90].

2.2.4.2 Laser Beam Incidence Angle

The angle at which a laser beam interacts with a target can cause a systematic range bias [59]. The effect of the incident angle on the measured range can be computed by Equation (2.28).

$$\Delta R = \frac{R \cdot \gamma}{2} \cdot \tan \beta \tag{2.28}$$

Where ΔR is the range error, R is the measured range, γ is the beam divergence angle and β is the incidence angle. The geometry of this range bias is shown in Figure 2.29.

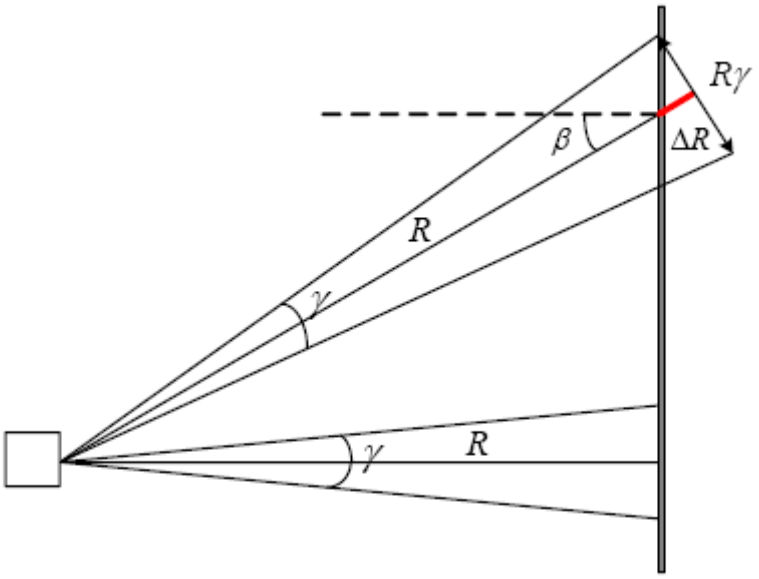

Figure 2.29: Range error due to laser beam incidence angle [15].

If the surface is highly reflective a large incidence angle may cause a multipath error. When this occurs the laser beam can bounce from one surface to another, causing an entirely inaccurate range measurement to occur. Alternatively, under the right conditions (laser beam frequency and material encountered) the laser beam can penetrate some target surfaces. Penetration of the target material causes a refraction delay in the range measurement. Surfaces like wood, marble, styrofoam, plastic or glass can allow some

amount of penetration for some laser beam frequencies. One source, [91], recorded range errors of 15mm for laser penetration into styrofoam and 5mm for laser penetration into wood.

2.2.4.3 User Pick Errors

When extracting objects from LiDAR point clouds, the accuracy with which the extracted object is placed within the global reference frame is a combination of the point cloud accuracy and the accuracy of the methodology used to extract the object. Since LiDAR collects clusters of points on a target, with no guarantee that features of interest on that target will be included in the cluster, methods for locating and extracting features of interest are needed. This is especially true when control points or other control objects need to be extracted from a point cloud. Extraction of control points could involve the selection of the nearest LiDAR point to the control point, computing the average coordinate from a cluster of LiDAR points, or computing the coordinate from the intersection of other various extracted primitive shapes (i.e. polylines, vectors, planes, cylinders, spheres, etc…).

No matter the method used to extract features from LiDAR point clouds, the density and distribution of points in the point cloud becomes the biggest factor in determining the accuracy with which the feature can be extracted. For example, if shadowing or systematic point errors render parts of the target unusable, then the extraction method used to locate the centre of a target will likely be biased toward the densest part of the point cloud. The use of primitive geometry is a very effective method for the accurate extraction of 3D targets such as the corners of buildings. It can be accomplished

manually, or through the use of automated extraction routines such as plane growing algorithms.

Another option for increasing the accuracy with which feature points can be extracted from LiDAR point clouds would be the use of custom designed targets. Custom designed targets provide a means through which control points can be more easily searched within a larger point cloud. They also allow for the development of specific methods of extraction designed to more precisely and accurately locate the target centre. The downside to using custom targets is that they must be setup before LiDAR collection commences and may require a property owners consent to be deployed.

2.3 Outlier Detection

As the previous section shows, errors can be generated in MTL data for a variety of reasons. Some of these reasons, such as boundaries of occlusion, surface reflectance and multi-path reflection are described in [92]. Instrument errors, environmental errors, positioning errors and target errors all have the potential to create stray points within a point cloud. In addition, terrestrial based scenes are collected faster than ever before, firstly because they are being collected from a moving platform and secondly, because collection speeds have greatly increased (600,000 points per second [2]). This increase in the number of terrestrial based data points collected during a survey, means greater and greater amounts of data are being produced faster. To further complicate matters, due to the fact that the scanners are now immersed in the scene being scanned, instead of flying high above it, the geometry contained in these massive data files is more complex than

those encountered previously. This makes filtering of the data harder than previously encountered, but even more necessary.

Several sources [92 – 94], provide definitions for the term outlier. Simply stated, an outlier is a point which differs from its neighbours or neighbourhood significantly. The determination of what the term significantly means is, of course, up to the individual user of the data. Of course, the significance threshold will change depending on the method used to detect the outlier.

Most methods for performing outlier detection in LiDAR data remain wholly in the spatial domain. These methods can be broken into the general categories of distribution-based, depth-based, distance-based, clustering-based and density-based [95]. In Chapter 5, three new ways of filtering outliers in MTL will be introduced. The first outlier detector introduced in Chapter 5 will combine the general principals of distance-based outlier detection with data segmentation and 10 parameter polynomial surface fitting. The second outlier detector introduced in Chapter 5 will combine the general principals of distribution-based outlier detection with data segmentation and the same 10 parameter polynomial surface fitting. The third outlier detector in Chapter 5 will incorporate the temporal data available in MTL data through the use of a α-β-γ Kalman smoother combined with the principals of a distance-based outlier detector.

2.3.1 Distribution-Based Outlier Detection

Distribution-based approaches to outlier detection in LiDAR use statistical distribution models to test whether the statistical discordancy of the points belongs to the model. By

fitting the data to a probability distribution model, such as the normal distribution, points which deviate from the distribution model can be flagged as outlier points [95]. Such a strategy is detailed in [96], where two types of algorithms, one based on a Gaussian mixture model and another based on a kernel mixture model were used to score the statistical likelihood of the input data being an outlier. The advantages of this type of model are that it's adaptive to non-stationary sources of data, it provides a clear statistic with theoretical meaning, it is computationally inexpensive and it can handle both discrete and continuous variables [96].

A distribution-based approach to outlier detection was tried specifically on MAL data in [97]. To use this form of outlier detector in LiDAR data, the point cloud was segmented and a kernel density estimation model was employed to estimate the probability that a point was an outlier. It was reported that this form of outlier detector performed well on MAL data especially when it was paired with a distance-based outlier detector [97].

2.3.2 Depth-Based Outlier Detection

Depth-based approaches to outlier detection in LiDAR are based on organizing the data into a k-dimensional data space, from which each point has a given depth within that space compared to the other points. Those points with smaller depths, i.e. those points which are more "exposed" within the data space, are more likely to be outlier points [95]. Detecting outliers in MAL data using a k-d tree approach was studied by [31]. Combined with a simple distance-based point filter, [31] first constructs kd-trees from the remaining points and then computes the distances between points within that k-dimensional space. Outliers are then identified through a threshold value set before the routine is executed.

2.3.3 Distance-Based Outlier Detection

Distance-based approaches to outlier detection in LiDAR use the direct spatial distance of a point from its neighbours. Assuming that there are less than m points within d distance from a test point q, then point q would be considered an outlier [95]. This type of approach is used in [98] and [99] on MAL data. In [98], points are connected to their nearest neighbours following the rules of Delaunay triangulation. The vectors between the points are then examined and the points are considered outliers if at least one of the vector's lengths are above some threshold set by the user.

In [99], the k-nearest neighbours to a test point are used to interpolate the expected position of a test point and then the difference between the actual point and its predicted position is tested. If the difference between the predicted and actual point is greater than some threshold, the point is classified as an outlier and removed.

2.3.4 Clustering-Based Outlier Detection

Clustering-based approaches to outlier detection in LiDAR gather points or objects which resemble each other into groups or clusters. The approach generally groups data into clusters of different density. Points in smaller clusters are flagged as candidate outliers and candidate points which lay farther than a certain distance from non-candidate points are removed as outliers.

In a survey of outlier techniques presented in [100], several examples of these types of techniques were discussed. In one method, the data is divided into k clusters, each of which provides a local model of the data. In this case, the user of this algorithm would be

expected to supply the optimal number of clusters before executing the routine. Each of the k clusters is represented as a prototype (average) vector with attribute values equal to the mean across all points in that cluster [100]. Each cluster in this algorithm then has a calculated radius which is the distance between the prototype vector and the most distant point in the cluster. If new points assigned to the cluster lay outside this radius, the points are discarded as outliers [100].

In another example, the PAM (Partition Around Medoids) algorithm represents each cluster using an actual point and a radius instead of a prototype (average) point and radius [100, 101]. The PAM algorithm is otherwise similar to the previous algorithm. PAM has the advantages of being robust to outliers, less susceptible to local minima, data order independent and provides better class separation than the previous clustering method [100]. It is however, more computationally expense, taking exponentially more running time than the previous clustering method [100].

A hybrid cluster method is discussed in [102], wherein the algorithm combines the theories of fuzzy logic with cluster-based outlier detection methods to filter both discrete and continuous variables.

2.3.5 Density-Based Outlier Detection

Density-based approaches to outlier detection in LiDAR rely on the calculation of a local outlier factor (LOF), which is a measure of the local density of a point from its k nearest neighbours. The distances between the individual pairs of k nearest neighbours are used to estimate the local density of the cluster. By comparing the local density of a point to

the local densities of its neighbours, outliers can be found as they are expected to have substantially lower densities than their neighbours.

Detecting outliers using LOF ratios was studied by [92] on MAL and STL data. It was found that the LOF algorithm was unable to identify outlier points in both data sets with point densities higher than the point density threshold value (MinPts) [92]. Also, [92] concluded that the algorithm had a challenge with clustered outliers, but might be dealt with through the addition of image information.

A density-based approach was also discussed in [93], where a k-dimensional neighbourhood of points was setup and three tests were proposed. The first algorithm in [93] removes outliers based on the LOF ratio and then iteratively updates the neighbourhood function and performs the LOF ratio calculation again. This continues until no more LOF ratios meet the density criteria of an outlier or a specific number of outlier detections have occurred. The second function in [93] works in a similar manner to the first, but examines the LOF ratio for only the z component of the spatial data under consideration. The third function in [93] also works similarly to the first, but defines the neighbourhood of the spatial data by means of calculating the median average of the points instead of the mean.

In the algorithms presented in [92, 93], the user sets the neighbourhood size (k) before the algorithm begins, however in [94], the algorithm uses Delaunay Triangulation to automatically localize the neighbourhood for a given point. This approach accounts for

the problem of spatial autocorrelation in the LOF ratios. The conclusion in [94] was that this algorithm could effectively detect more outliers than previous density-based approaches.

3. Error Analysis of MTL Systems

3.1 Introduction

The current techniques for performing error analysis in LiDAR data were discussed in Chapter 2, Section 2.2. Building on the techniques discussed in Section 2.2, this chapter introduces the idea of using Monte Carlo simulation in conditional variance analysis to estimate the errors in MTL point clouds and more specifically, to apportion those errors among the various measurements used in the data's creation. The analysis is compared to the types of analysis discussed in Section 2.2, namely, error propagation and control field comparison.

To accomplish this, it is first necessary to examine the discrepancies between a number of calibrated Lynx Mobile Mapper systems and a control field. This will indicate the error envelope that can be expected from modern MTL systems. Next, the results of a series of simulations will be presented. These simulations introduced known errors into the observations of the LiDAR and then examined the changes in the output point cloud. Error prorogation will be used to examine the errors within the MTL point clouds and then conditional variance analysis will be performed on those same point clouds, allowing the results between these two methods of error analysis to be compared.

Much of the content in this chapter has been published as "Extraction of geo-spatial information from LiDAR-based mobile mapping system for crowd control planning" [46] and "Error Analysis of a Mobile Terrestrial LiDAR System" [103].

3.2 The Mathematical Models

3.2.1 LiDAR to Geocentric Coordinate Conversion

MTL systems consist of two distinct components, the DG system and the laser ranging system, as shown by a typical system configuration in Figure 2.20 from Section 2.2.3.2. As shown in Figure 2.20, the coordinates of point p in the Earth Centered Earth Fixed (ECEF) coordinate system P^{ECEF} are a function of the laser range vector, the relative location and orientation of the LiDAR sensor to the DG system and the position and orientation of the DG system. The base equation of this relationship can be expressed by Equation (3.1).

$$P^{ECEF} = P_{INS}^{ECEF} + L_{INS-LiDAR}^{ECEF} + L_{LiDAR-p}^{ECEF} \tag{3.1}$$

In Equation (3.1) P_{INS}^{ECEF} is the position of the DG system, $L_{INS-LiDAR}^{ECEF}$ is the vector from the DG system to the LiDAR sensor and $L_{LiDAR-p}^{ECEF}$ is the vector from the LiDAR to the target point p. $L_{INS-LiDAR}^{ECEF}$ and $L_{LiDAR-p}^{ECEF}$ can be further deconstructed as expressed in Equations (3.2) and (3.3).

$$L_{INS-LiDAR}^{ECEF} = R_1(B,L) \cdot R_2(r,p,h) \cdot l_{LiDAR}^{INS}(l_X, l_Y, l_Z) \tag{3.2}$$

$$L_{INS-LiDAR}^{ECEF} = R_1(B,L) \cdot R_2(r,p,h) \cdot R_3(\theta_X, \theta_Y, \theta_Z) \cdot l_p^{LiDAR}(\alpha, \beta, d, K) \tag{3.3}$$

Where l_{LiDAR}^{INS} is the lever arm vector in the DG system body frame from the DG system to the origin of the LiDAR body frame, l_p^{LiDAR} is the laser range vector between the LiDAR and the target point p, R_1 is the rotation matrix between the local geodetic coordinates and the ECEF coordinate frame, R_2 is the rotation matrix between the DG system body

frame and the local geodetic frame and R_3 is the rotation matrix between LiDAR body frame and the DG system body frame.

Certain constants can be known, such as the offset from the LiDAR coordinate center to the scanning mirror. Most manufacturers disseminate or publish this information as it is needed to make proper measurements of the LiDAR sensor lever arms. For the Lynx Mobile Mapper the coordinates of the target points in the LiDAR body frame can be calculated as follows.

$$l_p^{LiDAR} = \begin{bmatrix} -0.175 + (d+K)\sin(\alpha)\cos(\beta) \\ (d+K)\cos(\alpha)\cos(\beta) \\ -(d+K)\sin(\beta) \end{bmatrix} \tag{3.4}$$

Where α represents the horizontal angle measured by the LiDAR in the LiDAR body frame, β represents the vertical angle measured by the LiDAR in the LiDAR body frame, d represents the distance measured by the LiDAR to the target point p and K represents the zero error of the LiDAR instrument. In the case of a MTL, only the vertical angle is measured, so the horizontal angle is fixed at an appropriate value. In the case of the Lynx, this value is zero.

Negating the Z term in Equation (3.4) creates the right-handed LiDAR coordinate system depicted in Figure 2.20 from Section 2.2.3.2. This mean that the rotation matrix (R_1) can be defined as shown in Equation (3.5).

$$R_1 = \begin{bmatrix} -\sin(B)\cos(L) & -\sin(L) & -\cos(B)\cos(L) \\ -\sin(B)\sin(L) & \cos(L) & -\sin(B)\sin(L) \\ \cos(B) & 0 & -\sin(B) \end{bmatrix} \tag{3.5}$$

Where *B* represents the geodetic latitude of the DG system and *L* represents the geodetic longitude of the DG system. A third variable that does not occur in Equation (3.5) but that is linked to the latitude and longitude can be defined as the ellipsoidal height, *h*.

The rotation matrices for the orientation of the DG system (R_2) and the boresight angles of the LiDAR (R_3) can be defined as shown in Equation (3.6) and Equation (3.7).

$$R_2 = \begin{bmatrix} \cos(p)\cos(h) & \sin(r)\sin(p)\cos(h) - \cos(r)\sin(h) & \cos(r)\sin(p)\cos(h) + \sin(r)\sin(h) \\ \cos(p)\sin(h) & \sin(r)\sin(p)\sin(h) + \cos(r)\cos(h) & \cos(r)\sin(p)\sin(h) - \sin(r)\cos(h) \\ -\sin(p) & \sin(r)\cos(p) & \cos(r)\cos(p) \end{bmatrix} \quad (3.6)$$

$$R_3 = \begin{bmatrix} \cos(\theta_Y)\cos(\theta_Z) & \sin(\theta_X)\sin(\theta_Y)\cos(\theta_Z) - \cos(\theta_X)\sin(\theta_Z) & \cos(\theta_X)\sin(\theta_Y)\cos(\theta_Z) + \sin(\theta_X)\sin(\theta_Z) \\ \cos(\theta_Y)\sin(\theta_Z) & \sin(\theta_X)\sin(\theta_Y)\sin(\theta_Z) + \cos(\theta_X)\cos(\theta_Z) & \cos(\theta_X)\sin(\theta_Y)\sin(\theta_Z) - \sin(\theta_X)\cos(\theta_Z) \\ -\sin(\theta_Y) & \sin(\theta_X)\cos(\theta_Y) & \cos(\theta_X)\cos(\theta_Y) \end{bmatrix} \quad (3.7)$$

Where r is the roll, p is the pitch and h is the heading of the DG system with respect to the local geodetic coordinate frame and θ_X is the rotation about the x-axis of the DG system body frame, θ_Y is the rotation about the y-axis of the DG system body frame and θ_Z is the rotation about the z-axis of the DG system body frame.

It is also possible to apply a form of differential rotation to Equation (3.7). Equation (3.7) can be replaced with Equation (3.8).

$$R_3 = \begin{bmatrix} 1 & -\left(\theta_Z^0 + d\theta_Z\right) & \left(\theta_Y^0 + d\theta_Y\right) \\ \left(\theta_Z^0 + d\theta_Z\right) & 1 & -\left(\theta_X^0 + d\theta_X\right) \\ -\left(\theta_Y^0 + d\theta_Y\right) & \left(\theta_X^0 + d\theta_X\right) & 1 \end{bmatrix} \quad (3.8)$$

Where θ_X^0, θ_Y^0 and θ_Z^0 are the initial approximates to the boresight angles and $d\theta_X$, $d\theta_Y$ and $d\theta_Z$ are the small changes applied to these angles. The advantage to using differential rotation in ridged body transformations is that the order of rotation no longer

matters. However, it must be remembered that $d\theta_1$, $d\theta_2$ and $d\theta_3$ have to be vanishingly small and therefore θ_X^0, θ_Y^0 and θ_Z^0 have to be quite close to their true values. In practical terms, the values of $d\theta_X$, $d\theta_Y$ and $d\theta_Z$ should not exceed 0.02º to maintain a transformation difference below 5cm at ranges of 100m.

As shown in Equations (3.4) to (3.8), the vectors l_{LiDAR}^{INS}, l_p^{LiDAR} and the matrices R_1, R_2, R_3 are themselves products of various measurements. These measurements can be grouped based on their dependencies. Group 1 consists of the DG system position in Cartesian ECEF coordinates (X_{INS}^{ECEF}, Y_{INS}^{ECEF}, Z_{INS}^{ECEF}). The Group 1 parameters are converted from the geodetic latitude (*B*) longitude (*L*) and ellipsoidal height (*h*) measured by the DG system. Group 2 consists of r, p, h, which represent the roll, pitch and heading of the DG system with respect to the local geodetic coordinate frame. Group 3 consists of l_X, l_Y, l_Z, which represent the components of the lever arm vector between the DG system and the LiDAR. Group 4 consists of $\theta_X, \theta_Y, \theta_Z$, which represent the x, y and z Euler rotations of the LiDAR coordinate frame in the DG system frame. Group 5 consists of the LiDAR measurements (α, β, d, K)

Each of the 18 aforementioned variables, have an associated error estimate usually in the form of an RMS or standard deviation. It is usual to assume that each of the measurements is normally distributed with a mean of zero. Under these circumstances the further assumption that RMS values equal standard deviations is valid and all error

estimates can be assumed to be one sigma standard deviations (σ). These 18 standard deviations can be assigned to the same groups previously discussed.

3.2.2 Error Propagation Analysis

Error prorogation analysis is a standard technique used to estimate errors in calculated values. The equations for error propagation are fairly well understood and used across many disciplines [27, 28, 104, 105]. Performing error prorogation first requires that the non-linear equation presented in Equation (3.1) be linearized as shown in Equation (3.9).

$$\begin{aligned} P^{\mathrm{ECEF}} \approx P_0^{ECEF} &+ \frac{dP^{ECEF}}{dX_{INS}^{ECEF}} \cdot \Delta X_{INS}^{ECEF} + \frac{dP^{ECEF}}{dY_{INS}^{ECEF}} \cdot \Delta Y_{INS}^{ECEF} + \frac{dP^{ECEF}}{dZ_{INS}^{ECEF}} \cdot \Delta Z_{INS}^{ECEF} \\ &+ \frac{dP^{ECEF}}{dB} \cdot \Delta B + \frac{dP^{ECEF}}{dL} \cdot \Delta L + \frac{dP^{ECEF}}{dr} \cdot \Delta r + \frac{dP^{ECEF}}{dp} \cdot \Delta p + \frac{dP^{ECEF}}{dh} \cdot \Delta h \\ &+ \frac{dP^{ECEF}}{dlx} \cdot \Delta lx + \frac{dP^{ECEF}}{dly} \cdot \Delta ly + \frac{dP^{ECEF}}{dlz} \cdot \Delta lz + \frac{dP^{ECEF}}{d\theta_X} \cdot \Delta\theta_X + \frac{dP^{ECEF}}{d\theta_Y} \cdot \Delta\theta_Y + \frac{dP^{ECEF}}{d\theta_Z} \cdot \Delta\theta_Z \\ &+ \frac{dP^{ECEF}}{d\alpha} \cdot \Delta\alpha + \frac{dP^{ECEF}}{d\beta} \cdot \Delta\beta + \frac{dP^{ECEF}}{dd} \cdot \Delta d + \frac{dP^{ECEF}}{dK} \cdot \Delta K \end{aligned} \tag{3.9}$$

In Equation (3.9), P_0^{ECEF} represents Equation (3.1) evaluated at the initial approximates ($X_{INS}^{ECEF(0)}$, $Y_{INS}^{ECEF(0)}$, $Z_{INS}^{ECEF(0)}$, B^0, L^0, r^0, p^0, h^0, l_X^0, l_Y^0, l_Z^0, θ_X^0, θ_Y^0, θ_Z^0, α^0, β^0, d^0, K^0) for the 18 measurements previously identified. Using the first derivatives from Equation (3.9), evaluated at each component (X,Y,Z) of the ECEF position, the design matrix (U) for a least square adjustment may be formed as shown in Equation (3.10).

$$U = \begin{bmatrix} \left(\frac{\delta P^{ECEF}}{\delta X_{INS}^{ECEF}}\right)_X & \left(\frac{\delta P^{ECEF}}{\delta X_{INS}^{ECEF}}\right)_Y & \left(\frac{\delta P^{ECEF}}{\delta X_{INS}^{ECEF}}\right)_Z \\ \vdots & \vdots & \vdots \\ \left(\frac{\delta P^{ECEF}}{\delta K}\right)_X & \left(\frac{\delta P^{ECEF}}{\delta K}\right)_Y & \left(\frac{\delta P^{ECEF}}{\delta K}\right)_Z \end{bmatrix} \tag{3.10}$$

Using the additional information provided from the individual error estimates of the 18 input parameters and assuming no correlation between these variables, the variance-covariance matrix (Q) may be form as shown in Equation (3.11).

$$Q = \begin{bmatrix} \sigma^2_{X_{INS}^{ECEF}} & 0 & \cdots & 0 \\ 0 & \ddots & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \sigma^2_K \end{bmatrix} \tag{3.11}$$

This allows the formation of the error propagation model shown in Equation (3.12).

$$\sigma^2_{P^{ECEF}} = U \cdot Q^{-1} \cdot U^T \tag{3.12}$$

3.2.3 Conditional Variance Analysis

Conditional variance analysis is a method of global sensitivity analysis. Global sensitivity analysis is primarily used to study how the uncertainty in the output of a model can be apportioned to different sources of uncertainty in the model inputs [106]. Preforming conditional variance analysis involves rewriting the terms of the Equation (3.1) in terms of generic variables; Equation (3.1) can be expressed as shown in Equation (3.13).

$$P^{ECEF} = f(\cdots \quad X_i \quad \cdots) \quad for \quad 1 \le i \le 18 \tag{3.13}$$

Variance based sensitivity analysis is carried out by using Monte Carlo simulation to generate N random draws from a given probability distribution, for each of the 18 variables (X_i) in Equation (3.13) [107, 108]. The number of random draws N, must be sufficient to provide a statistically meaningful sample of probable outcomes for the equation being analyzed. In the case of the analysis being performed on Equation (3.13),

N equals 1700 random samples, generated for each variable (X_i) according to the normal distribution.

Holding variable (X_i) fixed at the mean value of its randomly generated samples, while allowing the other variables to vary, N random values can be generated from Equation (3.13). The variance of the outputs of Equation (3.13) ($V\left(P^{ECEF} \mid X_i\right)$) can be calculated. If this variance is then averaged over all possible values of X_i, the expectation of the variance becomes $E\left(V\left(P^{ECEF} \mid X_i\right)\right)$. Based on linear algebra the main effect of X_i on P^{ECEF} ($V\left(E\left(P^{ECEF} \mid X_i\right)\right)$) can be computed from Equation (3.14) [106].

$$V\left(P^{ECEF}\right) = E\left(V\left(P^{ECEF} \mid X_i\right)\right) + V\left(E\left(P^{ECEF} \mid X_i\right)\right) \tag{3.14}$$

The larger $V\left(E\left(P^{ECEF} \mid X_i\right)\right)$ is the more influential X_i is on the output of Equation (3.13). Therefore, from the relationship shown in Equation (3.14), the first order Sobol index for the variable X_i can be computed according to Equation (3.15) [106].

$$S_i = \frac{V\left(E\left(P^{ECEF} \mid X_i\right)\right)}{V\left(P^{ECEF}\right)} \tag{3.15}$$

Higher order indexes are computed by holding more than one variable fixed as shown in Equation (3.16) [106].

$$S_{i,j,k,\cdots} = \frac{V\left(E\left(P^{ECEF} \mid X_i, X_j, X_K, \cdots\right)\right)}{V\left(P^{ECEF}\right)} - S_i - S_j - S_k \cdots \tag{3.16}$$

Computing all the higher order terms becomes inefficient due to the large number of combinations for models with a large number of variables. For the 18 variables in Equation (3.1), there will be 48,620 ninth order Sobol indexes alone. It is therefore

desirable to calculate the total effect for a variable (S_{t_i}). This can be done by calculating the variance of the expected values when all the variables in the model are held fixed while allowing only X_i to vary ($V\left(E\left(P^{ECEF} \mid X_{\sim i}\right)\right)$). The total effect can be computed as shown in Equation (3.17) [106].

$$S_{t_i} = 1 - \frac{V\left(E\left(P^{ECEF} \mid X_{\sim i}\right)\right)}{V\left(P^{ECEF}\right)} \tag{3.17}$$

First order sensitivity indexes are a quantitative measure of the importance of the input variables on the calculated results. They apply to additive models. The more sensitive the calculated results are to small variances in any particular input variable, the greater the first order sensitivity index. The total effect indexes are also a quantitative measure of the importance of the input variables on the calculated results. Unlike first order indexes however, they are applicable to all types of models independent of their model characteristics [109].

Additional properties of the first order Sobol indexes include

For additive models $\sum_{i=1}^{n} S_i = 1,\ S_{t_i} = S_i$

For non-additive models $S_{t_i} > S_i$

3.3 Comparing MTL to Control

Another method for evaluating the accuracy of any mapping system is to compare it to a more accurate reference. Therefore, to evaluate the average performance of MTL systems, 5 Lynx Mobile Mapper MTL systems were used to scan a control field. The control field was placed on identifiable features on the walls and ground around a two

story office building. It was surveyed using a total station with an angular accuracy of 2 arc-seconds and a range accuracy of 2mm+2ppm. A four point traverse was constructed to tie the surveyed control points together. Absolute position of the control network was determined by referencing a local base station and several control monuments maintained by the City of Vaughan. The adjusted coordinates for the control points were determined to have positional accuracies of ±5mm or less.

The Lynx Mobile Mapper is designed to consist of either one or two LiDAR sensors. In this case, the Lynx systems used, consisted of two LiDAR sensors each. For the purpose of testing the accuracy of the mobile LiDAR system used here, tests were conducted over several days. During these trials, 9 or more satellites were visible and PDOP values of 3 or less were maintained. The area in which the tests were conducted was quite open, with no obstructions to cause canyoning and only one building close by that could serve as a source for multipath. The GNSS/DG system that was used was an Applanix POS LV 420 (www.applanix.com), which is considered by the industry to contain a medium grade IMU. It has an Applanix quoted drift rate of 0.02^{o} per minute [44]. The method of correction was a single GPS only reference station, used to correct the trajectory post-process. In all of these test cases the same GPS only base station was used to correct the GNSS/DG. The base line between the mobile LiDAR system and this reference station never exceeded 500m during these tests.

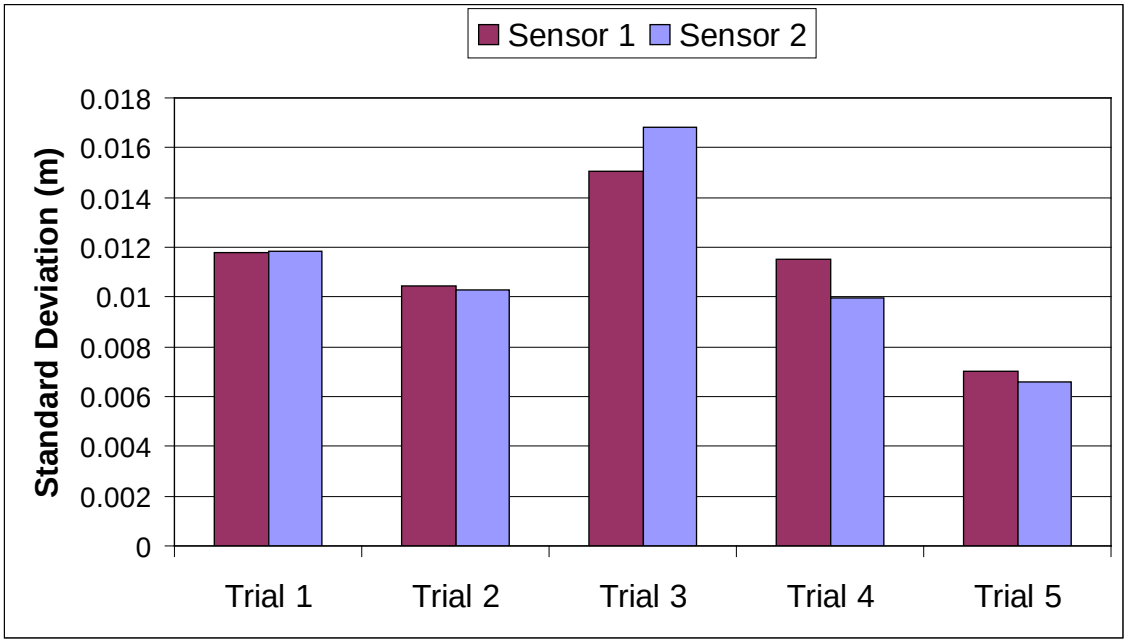

Figure 3.1: One sigma standard deviations for the horizontal residuals resulting from the comparison of 5 MTL systems to a pre-determined control field.

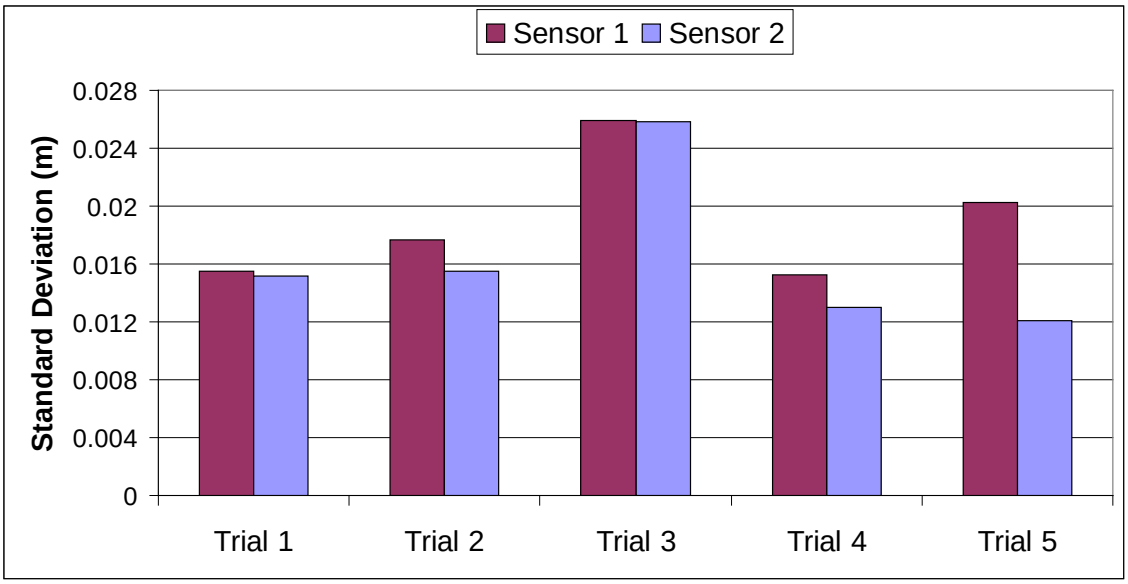

Figure 3.2: One sigma standard deviations for the vertical residuals resulting from the comparison of 5 MTL systems to a pre-determined control field.

To account for the variable nature of the GNSS constellation and the variable properties of the ionosphere and tropospheric delays that are inherent in any GNSS system, multiple data sets where collected on different days and in different weather conditions. The

control point locations were then identified in each point cloud and coordinates for these locations were extracted. To ensure that the control points were extracted as accurately as possible from the point cloud, only window corner points were used. Each of these window points was surrounded by at least three planar surfaces with differing orientations. Planar surface fitting was used to extract the surfaces and the intersection of the extracted planes was used to identify the control points. Residuals were formed by comparing the point cloud extracted coordinates with the known coordinates for the control field. Computing the standard deviation for each set of residuals, it was found that the order of magnitude of the errors were fairly consistent, with maximum standard deviations of 1.7cm in the horizontal and 2.6cm in the vertical. The results for each of the 5 Lynx systems tested are shown in Figure 3.1 and Figure 3.2.

Figure 3.1 and Figure 3.2 show, in some places, significant variation in the magnitudes of the standard deviations calculated for each sensor in the same trial. Since the position of the LiDAR points, depend on the lever arms and boresight angles of the sensors with respect to the GNSS/IMU reference frame, one would expect that the standard deviations for Sensor 1 and Sensor 2 would be equal for each trial. However, the 6 parameters which make up the lever arm and boresight angle values also contain errors, causing this assumption to be false. Since these 6 parameters are fixed values, the next expectation would be that given similar collection conditions, the standard deviations for each sensor would be offset by similar amounts for each trial. Due to variations in the performance of each LiDAR sensor and the interaction of the laser beam with the target at varying

angles of incidence, as well as, limitations on the precision with which the control points could be extracted from the point cloud, this is simply not the case.

3.4 Sensitivity Testing

Using the mathematical model presented in Equation (3.1) several simulations were run to gauge the effect of errors in the boresight, lever arm and zero error calibration numbers. The data used came from a single 360º rotation of a real MTL sensor. The data was first run with known calibration values for the boresight, lever arm and zero error. It was then run again with a varying known quantity added to each parameter in turn. The error offsets used for the boresight parameters include -1 º, -0.5º, -0.2º, -0.1º, -0.01º, 0.01º, 0.1º, 0.2º, 0.5º and 1º. The zero error and lever arms used error offset values including -0.5m, -0.1m, -0.05m, -0.02m, -0.01m, 0.01m, 0.02m, 0.05m, 0.1m and 0.5m. Several variables can affect the comparison of the results. By choosing a point out of the comparison pairs the vertical angle and range can be set to fixed values. The direction of the error and the magnitude of the individual ECEF residuals do not mean much in this comparison as they will change depending on the orientation of the DG system within the local tangential plane coordinate system. Therefore, only the overall magnitude of the error is of interest in this analysis. Figure 3.3 shows the maximum error in the ECEF coordinate point cloud for each error offset in each calibration parameter for a point in the point cloud that was 76.62m from the sensor and roughly in line with the Y axis of the sensor.

In Figure 3.3 it can be noticed, as expected, at long range, as the offset introduced into the boresight parameters increases, the roll and heading appear to become the most

significant parameters. At long range, when the offsets in the boresight parameters are kept small, the significance of the roll and pitch parameters become less significant. Interestingly enough, the amount of error introduced by the boresight angles is not equal. The pitch angle seems to contribute little error to the solution at range. In fact the pitch error is less significant than the linear offsets such as the lever arms or zero error when those values have large offsets. The linear parameters themselves contribute the induced offset directly into the point cloud. For example, when an offset of 0.5m was applied to each of the lever arms, the amount of deviation of the point cloud was exactly 0.5m from its zero position.

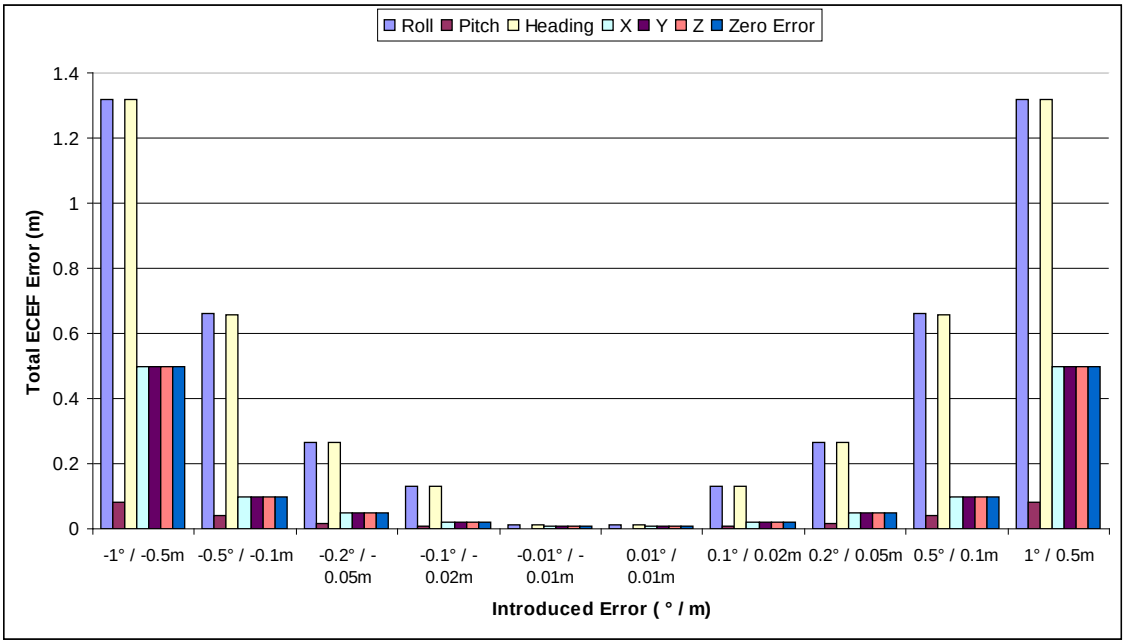

Figure 3.3: Maximum error in the ECEF point cloud at a point 75.62m from the sensor and at an angle of 3.371 degrees to the Y axis of the LiDAR coordinate frame.

Looking at another example (Figure 3.4), using a point with a distance of 2.481m and an angle to the Y axis of the sensor of 270.249°, it can be noticed that the zero error and lever arm terms have exactly the same amount of error as in the previous example. As expected, the angular boresight parameters are not contributing much error to the point

cloud at such a short range. Even with a one degree boresight offset for each of the parameters, the amount of observed deviation is less than 5mm.

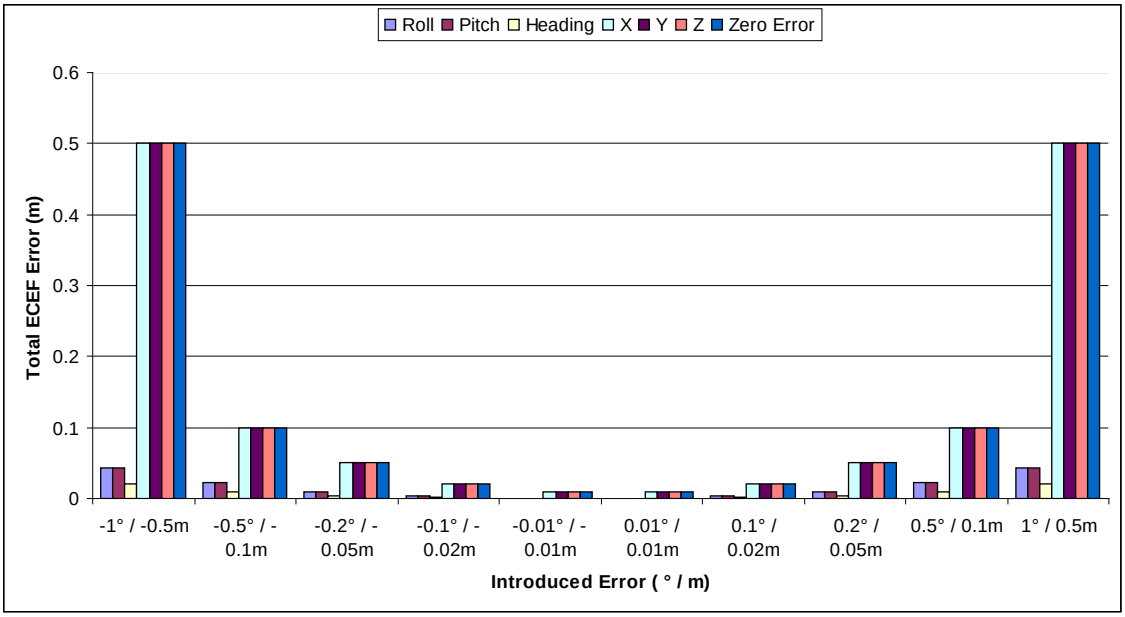

Figure 3.4: Maximum error in the ECEF point cloud at a point 2.481m from the sensor and at an angle of 270.249 degrees to the Y axis of the LiDAR coordinate frame.

3.5 Error Propagation

The error propagation described in Section 3.2.2 was implemented in Microsoft Visual C++ 6.0. For testing purposes, MTL data was collected using the Lynx Mobile Mapper. This data was collected in a mostly empty parking lot to allow the LiDAR to reach its maximum range. The data used is shown in Figure 3.5. The northern section of the Lynx data shown in Figure 3.5 contains a tree line that is located approximately 50m from the LiDAR sensor. The southern section of the data contains an open parking lot with some hedges and a lamp post. A building and two cars exist in the parking at long range for the LiDAR sensor. They are not seen in Figure 3.5 as they are at extreme long range for the LiDAR system, but random shots where collected off of them.

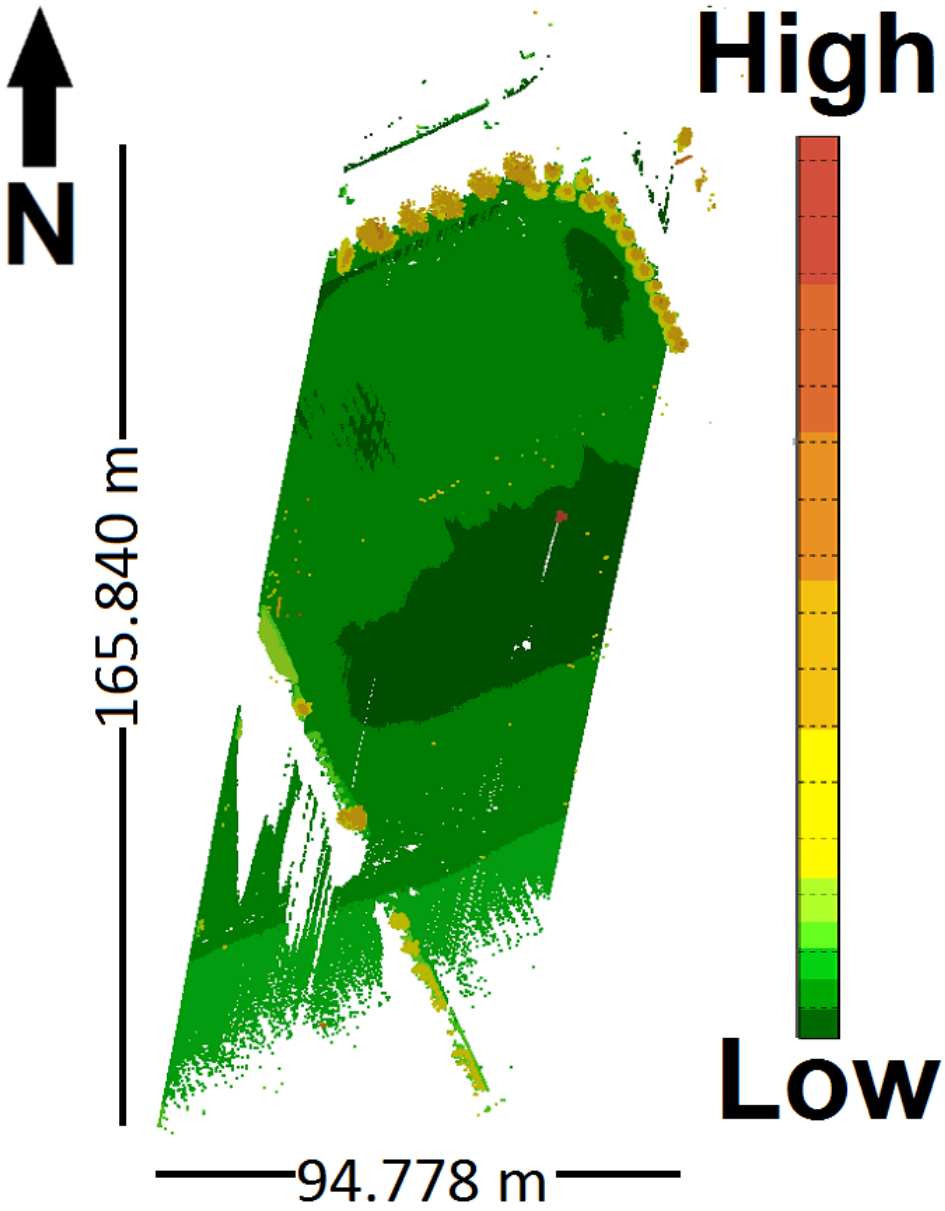

Figure 3.5: Lynx data collected in a parking lot in Vaughan Ontario Canada on May 19th 2012.

To analyze the effect of measurement errors in the 18 parameters of Equation (3.1) on the output point cloud, it is first necessary to collect the error estimates for each parameter. The error estimates concerning the position and orientation of the MTL system in the ECEF mapping frame are generated in real time by the DG system. In the case of the Lynx system the DG system used is an Applanix POS. Out of the five variables related to the position of the DG system, it was found that the latitude and longitude contribute

less than a millimetre of error to the final solution. For the purposes of testing, they were therefore excluded from further consideration since their contribution to the overall error is insignificant. The manufacturer states that the 1 sigma range accuracy of the LiDAR is 0.005m [1]. Examining planar surfaces in multiple points taken by the MTL system reveals that the range accuracy of the system is generally better than this, usually around 0.003m. Therefore these numbers will be compared to see the effect of each.

That leaves the 6 errors associated with the integration of the LiDAR and DG system, the so called boresight and lever arm parameters. Typically, the boresight operation for many MTL and multibeam SONAR systems is done using non-rigorous methods such as the so-called "Patch Test" [78]. Multiple manufacturers of this equipment and the software used to process the data use these methods for determining boresight parameters for MTL [78 – 82]. Table 3.1 gives a breakdown of the 18 parameters from Equation (3.1), the groups to which they have been assigned, the measurement errors expected from the MTL based on non-rigorous calibration methods and the ideal error values based on rigorous calibration methodologies and observed performance of a MTL system.

One literature source agrees that the expected lever arm uncertainties listed in Table 3.1 are typical for most systems [27]. However, the same source indicates that the typical manual boresight method should facilitate an uncertainty of 0.005° for roll and pitch and 0.008° for heading [27]. The same source further states that the expected results from a least squares adjustment should have a typical uncertainty of 0.001° for roll and pitch and 0.004° for heading.

Table 3.1: Expected and ideal uncertainties in MTL parameters.

Parameter	Group	Expected	Ideal
$\sigma_{X_{INS}^{ECEF}}$, Position X [m]	1	Estimated from DG system	Estimated from DG system
$\sigma_{Y_{INS}^{ECEF}}$, Position Y [m]	1	Estimated from DG system	Estimated from DG system
$\sigma_{Z_{INS}^{ECEF}}$,Position Z [m]	1	Estimated from DG system	Estimated from DG system
σ_{r}, Platform Roll [degrees]	2	Estimated from DG system	Estimated from DG system
σ_{p}, Platform Pitch [degrees]	2	Estimated from DG system	Estimated from DG system
σ_{h}, Platform Heading [degrees]	2	Estimated from DG system	Estimated from DG system
σ_{lx}, LiDAR X Lever Arm [m]	3	0.02	0.004
σ_{ly}, LiDAR Y Lever Arm [m]	3	0.02	0.004
σ_{lz}, LiDAR Z Lever Arm [m]	3	0.02	0.004
σ_{θ_X}, LiDAR Roll [degrees]	4	0.02	0.001
σ_{θ_Y}, LiDAR Pitch [degrees]	4	0.02	0.001
σ_{θ_Z}, LiDAR Heading [degrees]	4	0.02	0.001
σ_{α}, LiDAR Horizontal Angle [degrees]	5	0.0055	0.0055
σ_{β}, LiDAR Vertical Angle [degrees]	5	0.0055	0.0055
σ_{d}, LiDAR Distance [m]	5	0.008	0.003
σ_{K}, LiDAR Zero Error [m]	5	0.01	0.005

Since the manufacturer of the Lynx Mobile Mapper does not provide references for a user to locate the centre of the LiDAR sensor and since the DG system only has an imprecise sticker, indicating the general centre and approximate axes of the DG coordinate system, manually measuring the lever arms between the DG system and the LiDAR is extremely difficult. Therefore, the lever arm uncertainty estimates stated by [27] and which are reflected in Table 3.1, are plausible for the MTL being used here and conform to the accuracy estimates observed by the authour during experimentation with the 5 Lynx Mobile Mappers used in Section 3.3.

On the other hand, estimating the boresight angles based on non-rigorous patch test methods depends on the user manually adjusting the input boresight angles and observing

the results in the point cloud. This technique relies on the user looking for visual cues within the point cloud and adjusting the boresight angles until these cues disappear. Based on the boresights performed on the 5 MTL systems in Section 3.3, these visual ques tend to disappear within Lynx Mobile Mapper data while adjusting the boresight angles at the second decimal place. Therefore, the boresight uncertainties stated by [27] seem optimistic as they are given to three decimal places. When performing a manual boresight on a Lynx Mobile Mapper, a more likely value for the boresight roll, pitch and heading uncertainty is 0.02°, as reflected in Table 3.1.

MTL data was run through the error propagation in Section 3.2.2 for both the expected error estimates and the ideal error estimates. The results were converted to a colour scale and applied to the point cloud. The error propagation analysis for the expected error estimates are show in Figure 3.6.

The colour scales applied to the point clouds in Figure 3.6 show that the largest source of error in the resultant point cloud is consistently caused by the DG system positional errors. The poor quality of the lever arm estimates in this case makes the lever arms between the LiDAR and the DG system the second most consistent source of error in the system. The angular errors from the DG system orientation variables, as well as the LiDAR to DG system boresight variables take on increasing significance as the range from the LiDAR sensor increases. The LiDAR measurements themselves comprise the least source of error in the point cloud.

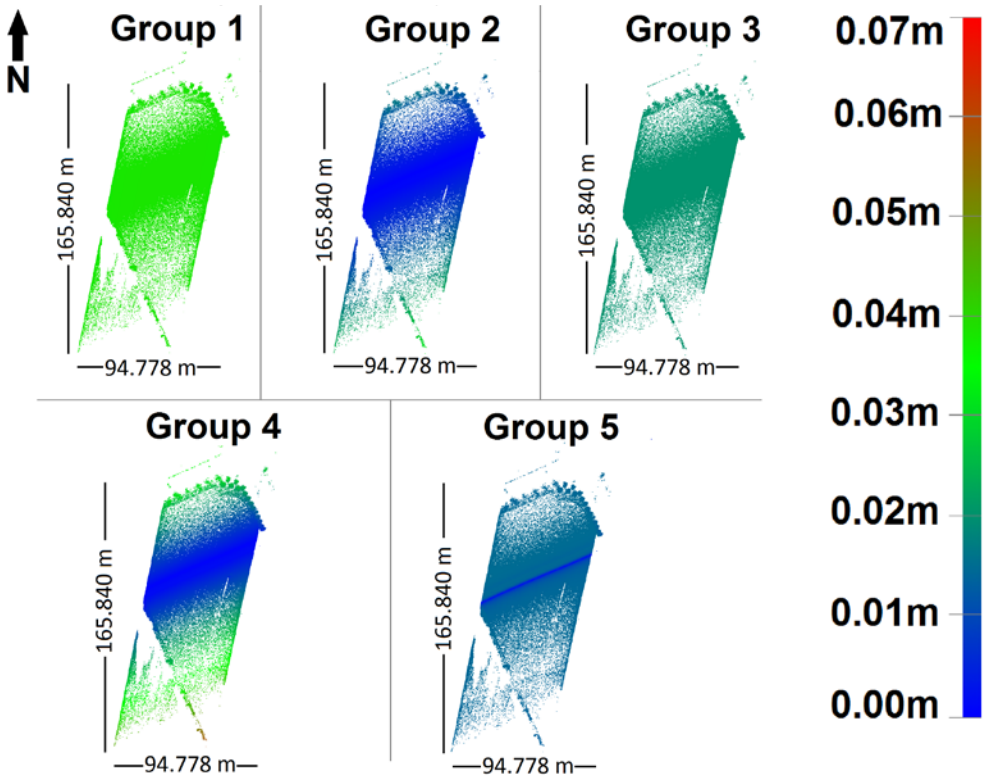

Figure 3.6: Results of error propagation, using the expected error estimates. Each of the five identified groups of variables was isolated and the error propagation performed.

The results from the error propagation for the ideal error estimates are shown in Figure 3.7. Figure 3.7 shows that the ideal error estimates produce a point cloud containing errors which are heavily influenced by the errors inherent in the DG system. The most significant source of error in Figure 3.7 is in the DG system position (Group 1) at a constant 3 to 4 cm. The second largest source of error according to Figure 3.7 is given by the DG system orientation data (Group 2). This error is insignificant at the LiDAR and increases as the range from the LiDAR increases. The LiDAR to DG system lever arms, the rotations of the LiDAR coordinate frame in the DG system frame and the LiDAR measurements themselves are shown to contribute an insignificant amount of error in Figure 3.7.

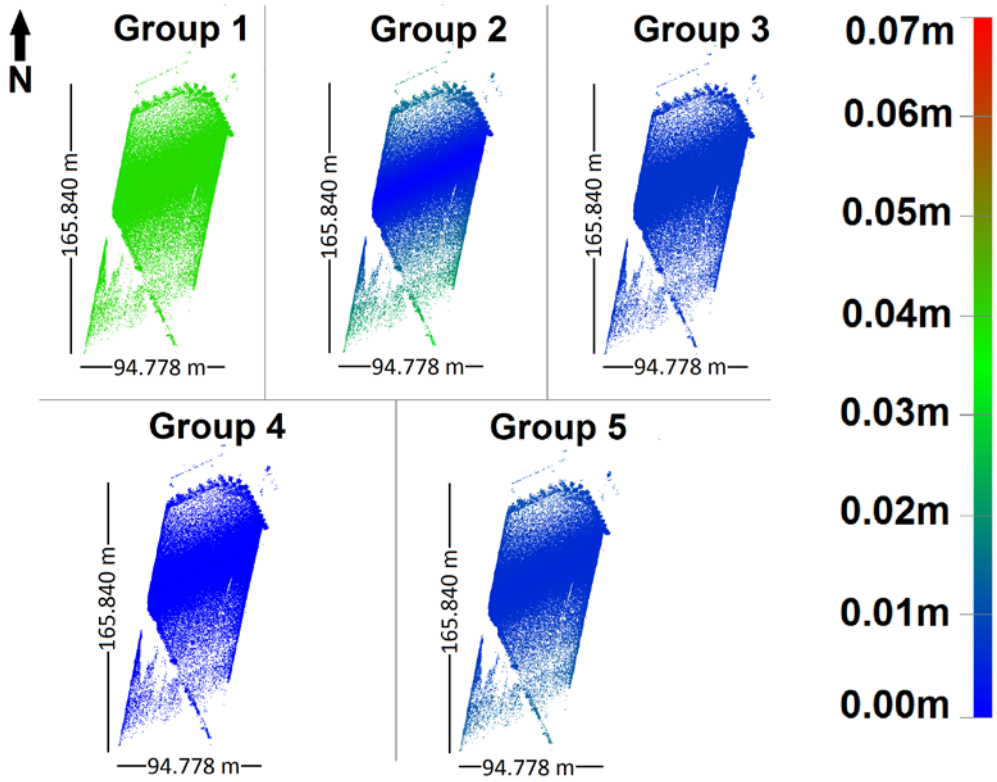

Figure 3.7: Results of error propagation, using the ideal error estimates. Each of the five identified groups of variables was isolated and the error propagation performed.

3.6 Conditional Variance Analysis

The conditional variance analysis described in Section 3.2.3 was implemented in MATLAB R14. The same data set (Figure 3.5) and the same set of expected and ideal measurement errors (Table 3.1) used in the previous section were processed using conditional variance analysis. The colour mapped data from the first order conditional variance analysis for the expected error estimates is shown in Figure 3.8. The colour maps in Figure 3.8 show that when the expected error estimates exist in the point cloud, the largest proportion of the error in the MTL point cloud is given by the LiDAR to DG system lever arm values in Group 3. The rotations between the LiDAR coordinate frame and the DG system coordinate frame (Group 4) in turn take on increasing significance as the range from the LiDAR sensor increases.

Since the color maps in Figure 3.8 refer to groups of variables instead of individual variables, the Sobol Indexes for each variable were calculated. It was found that the lever arm almost exclusively responsible for the results from Group 3 is the Z lever arm. The scatter plots in Figure 3.9 show that at close ranges the Z lever arm from the LiDAR to DG system contributes up to 90% of the error in the point cloud. Figure 3.9 also shows that as the range from the sensor increases the significance of the Z lever arm tappers off and is replaced by the LiDAR to DG system roll angle. At the maximum range of the MTL sensor, the effect of roll under these conditions becomes the single most significant source of error in the point cloud. Figure 3.9 shows that at the maximum range of the MTL sensor the error generated by the LiDAR to DG system roll angle is upwards of 70%.

In addition to the first order indexes of the conditional variance analysis, the total effect indexes for the 5 groups were also computed. These total effect indexes were converted to a colour scale and applied to the point cloud. Figure 3.10 shows the total effect indexes for the expected error estimates. Figure 3.11 shows the two most important scatter plots of the total effect indexes for the expected error estimates. The colour mapped data in Figure 3.10 show that when the expected error estimates exist in the point cloud, the total effect indexes from the conditional variance analysis are practically identical to the first order effect.

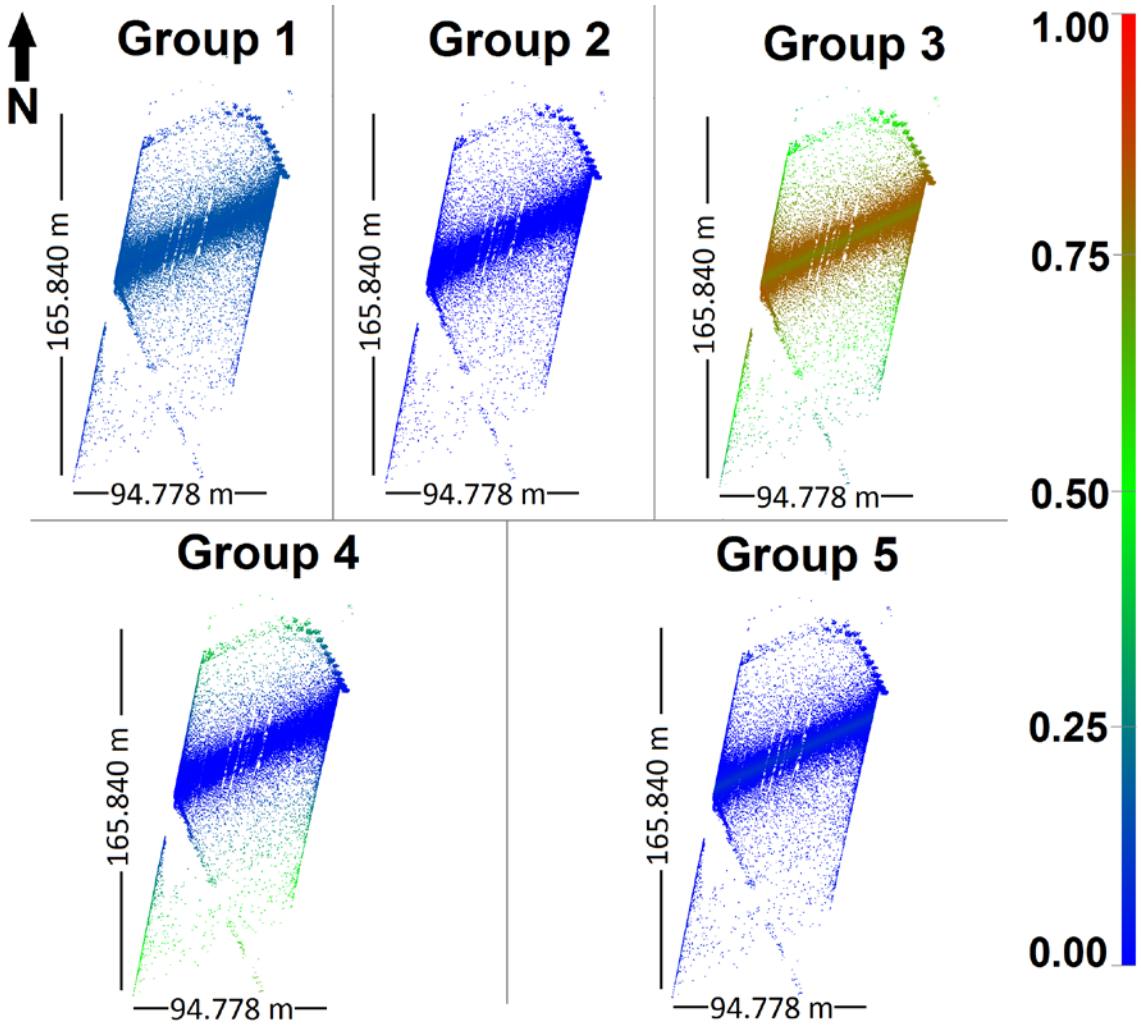

Figure 3.8: Results of first order conditional variance analysis using the expected error estimates. Each of the five identified groups of variables was isolated and the Sobol indexes computed.

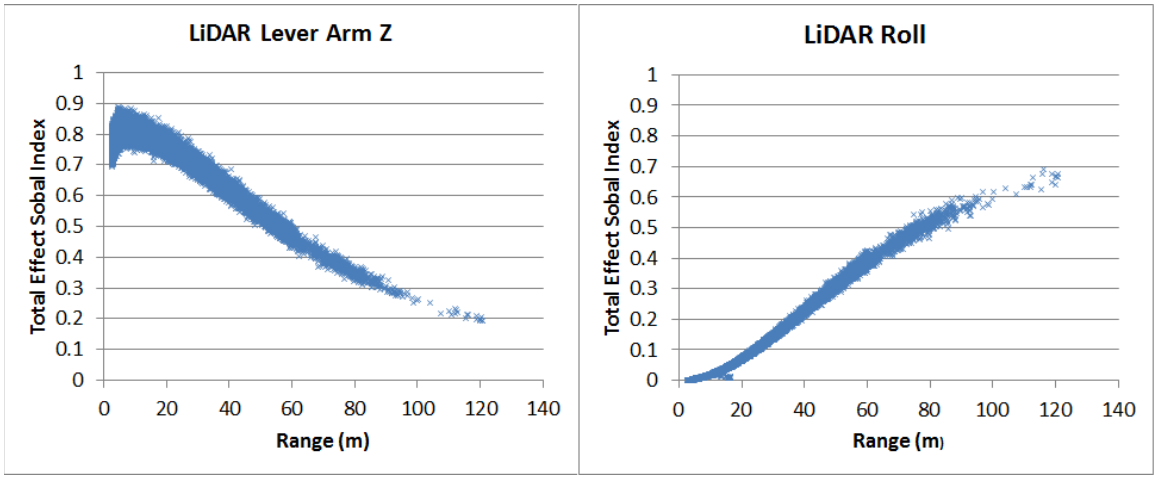

Figure 3.9: Scatter plot of first order Sobol indexes for Z lever arm and boresight roll in Lynx Mobile Mapper data.

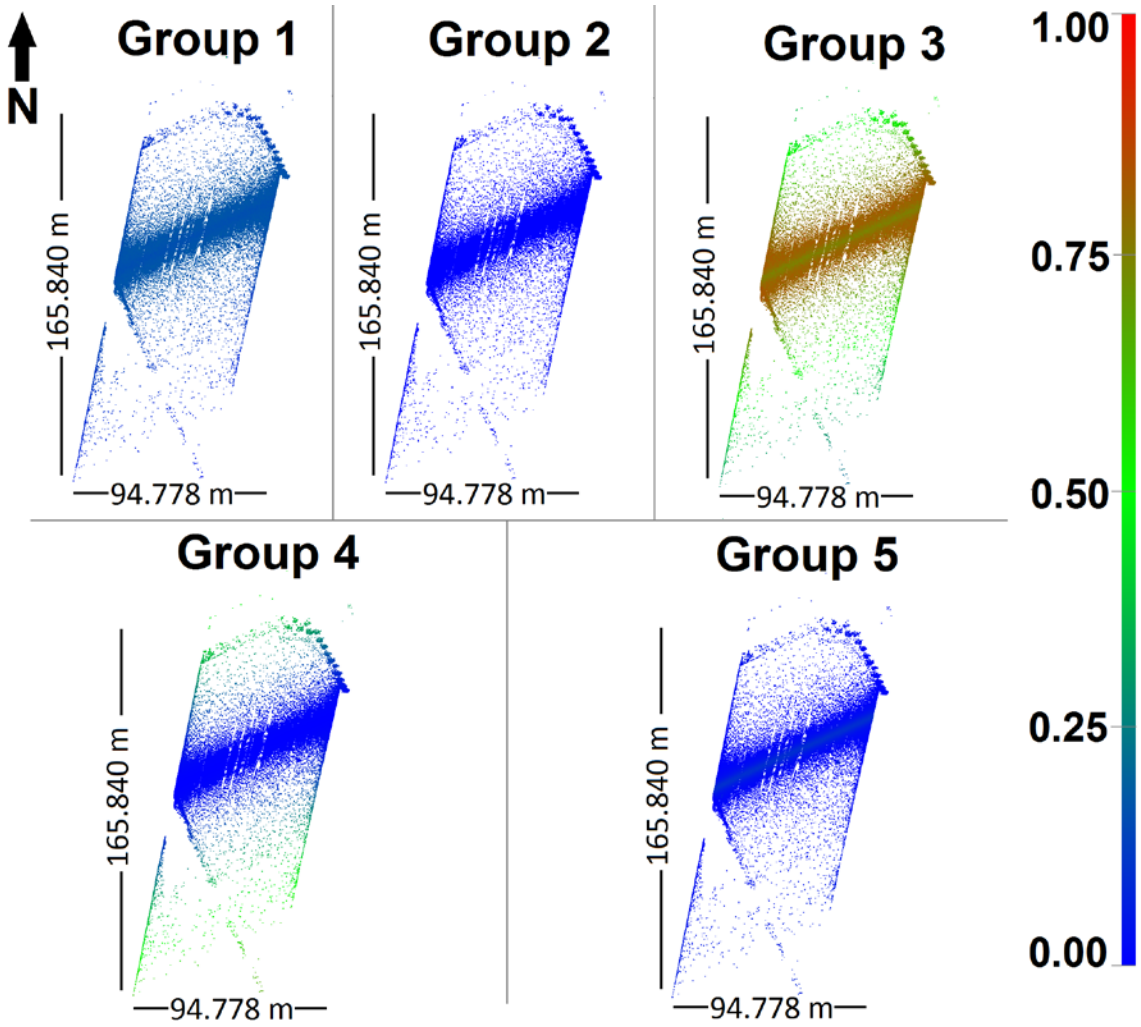

Figure 3.10: Results of the total effect conditional variance analysis using the expected error estimates. Each of the five identified groups of variables were isolated and the Sobol indexes computed.

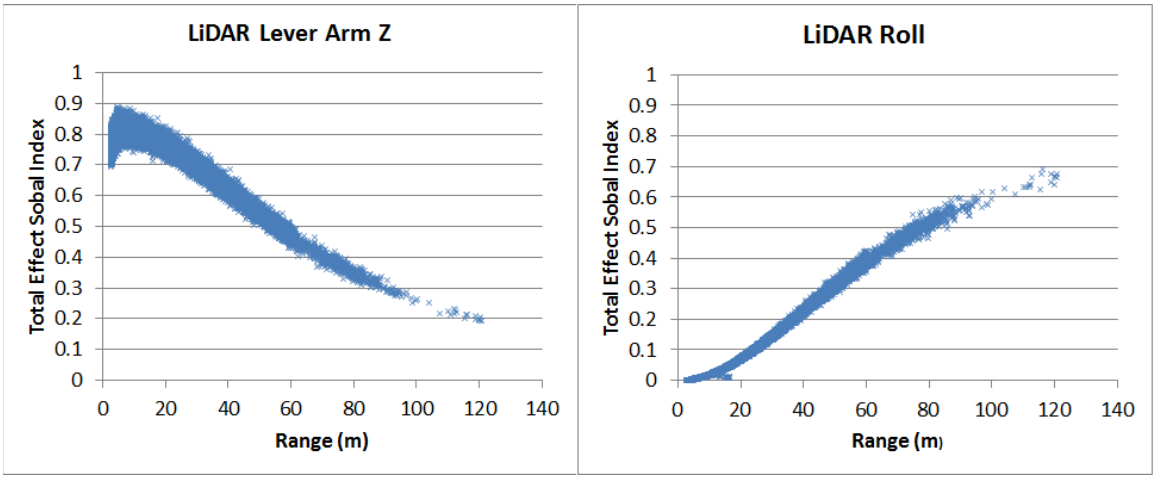

Figure 3.11: Scatter plot of total effect Sobol indexes for Z lever arm and boresight roll in Lynx Mobile Mapper data.

The fact that the first order sensitivity indexes in Figure 3.8 add up to one and that the first order indexes closely mirror the total effect indexes shown in Figure 3.10, indicates that Equation (3.1) is an additive model. This implies that the majority of the total effect can be attributed to the individual variables, i.e., the first order effect. Cross-effects caused by the various variable combinations are minimal and can be regarded as insignificant. The higher order indexes from Equation (3.16) can therefore be disregarded from further error analysis.

Breaking down the groups into individual variables as was done with the first order indexes, again it can be noticed that the LiDAR to DG system Z lever arm and the LiDAR to DG system roll angle are the most significant parameters in terms of their proportional effect on the point cloud. The scatter plots shown in Figure 3.11 indicate that the LiDAR to DG system Z lever arm contributes upwards of 90% of the error at ranges close to the LiDAR sensor and declines as the range increases. Figure 3.11 also shows that the LiDAR to DG system roll angle has little significance close to the LiDAR sensor, but makes up almost 70% of the error at maximum range.

The colour maps in Figure 3.12 show that when the ideal error estimates exist in the point cloud, the largest proportion of error in the MTL point cloud is given by the DG system position parameters of Group 1. The LiDAR measurement parameters of Group 5 contribute the next highest percentage of error in the MTL point cloud, while the other groups disappear into insignificance.

The colour mapped data in Figure 3.13 show that when the ideal error estimates exist in the point cloud, the total effect indexes from the conditional variance analysis are also very similar to the first order effect. The DG system position (Group 1) consistently contributes the largest proportion of error to the MTL point cloud, with the LiDAR measurements (Group 5) contributing the next largest proportion of error to the MTL point cloud. The other groups vanish into insignificance.

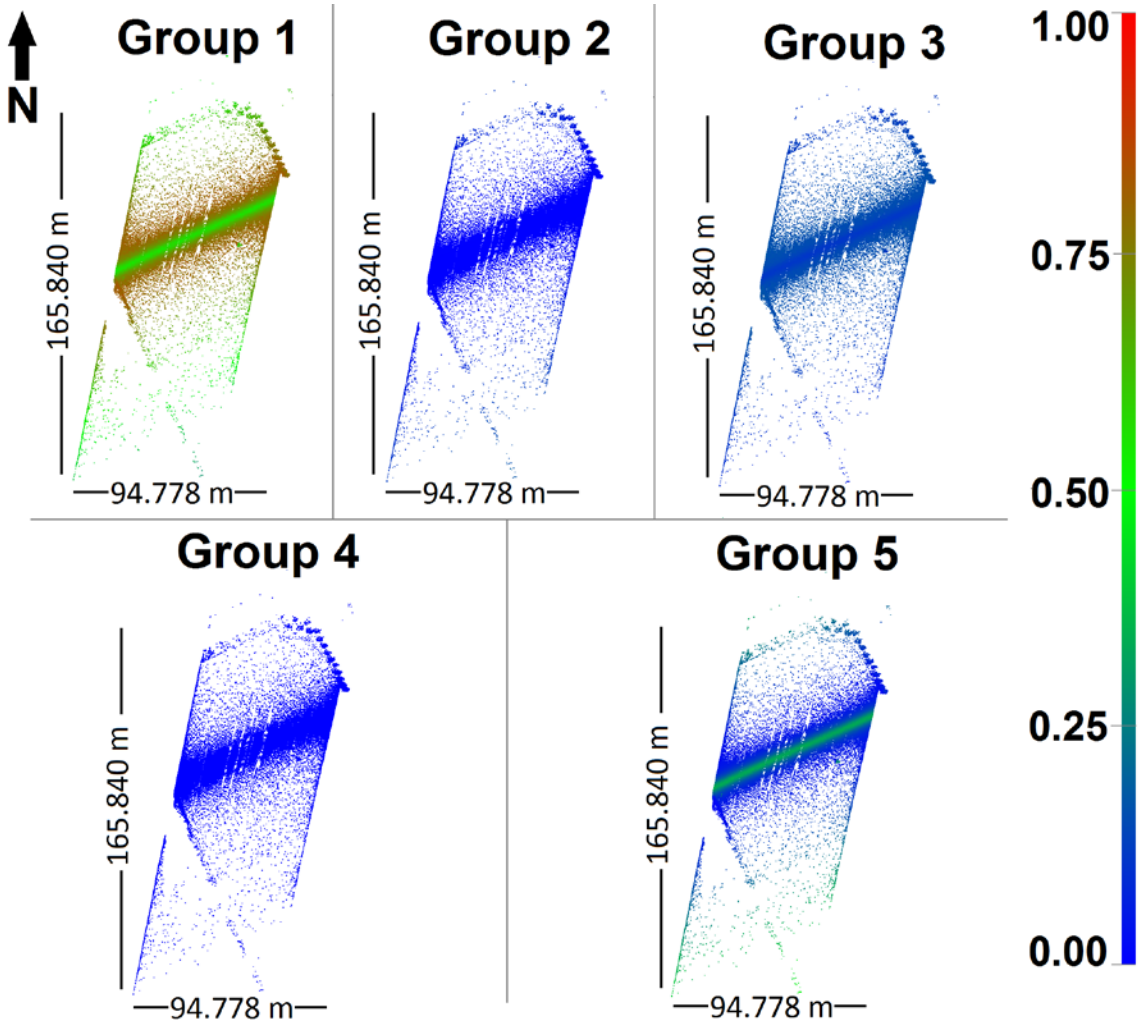

Figure 3.12: Results of first order conditional variance analysis using the ideal error estimates. Each of the five identified groups of variables was isolated and the Sobol indexes computed.

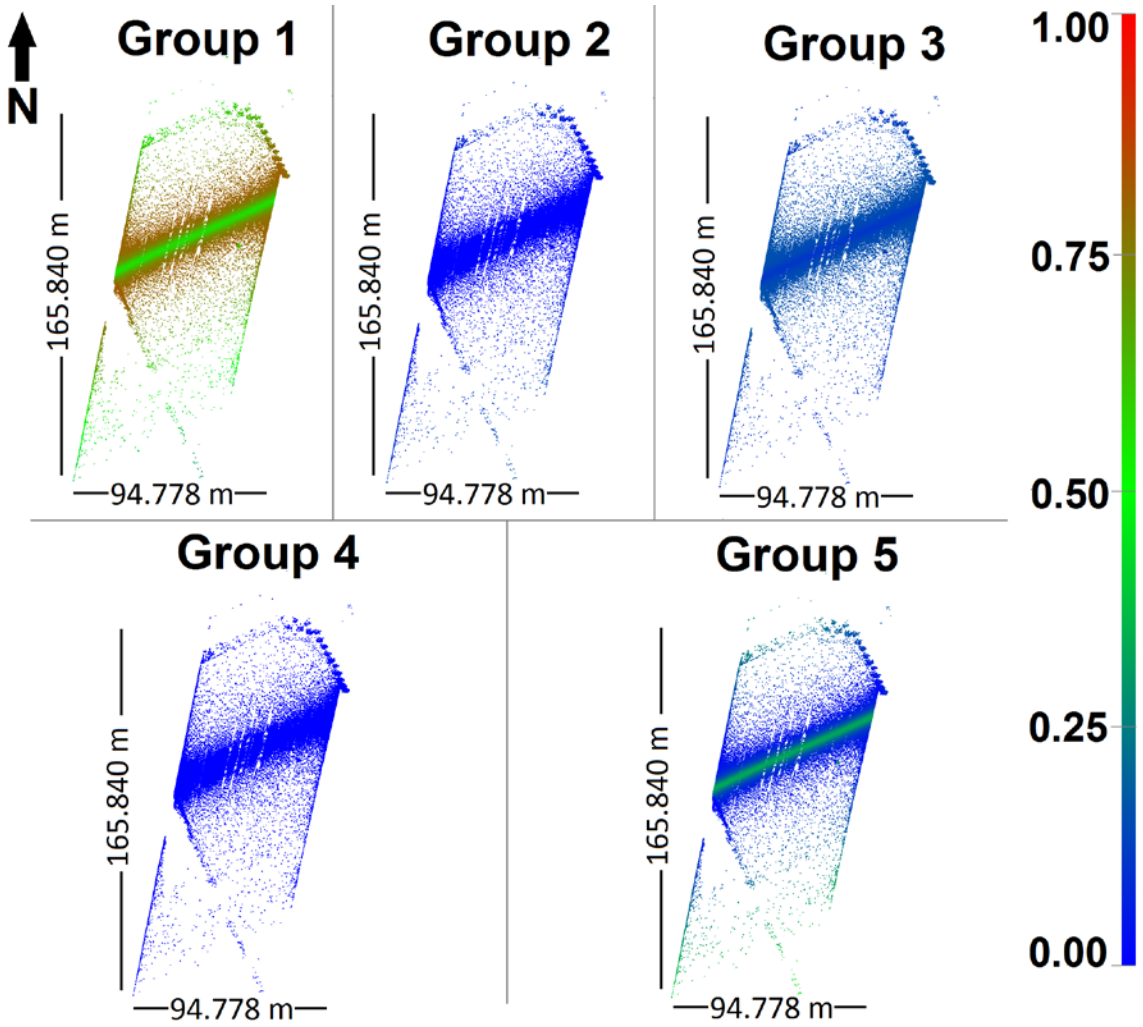

Figure 3.13: Results of total effect conditional variance analysis using the ideal error estimates. Each of the five identified groups of variables was isolated and the Sobol indexes computed.

Figures 3.12 and 3.13 reinforce the implication that the majority of total effect can be attributed to the first order effect. The higher order effects from Equation (3.16) can therefore be treated as insignificant.

3.7 Discussion

Comparing the data from several MTL systems to the positional coordinates of the control points has shown that under ideal GPS conditions and at ranges of approximately

15m the MTL can achieve the standard deviations of ±1.7cm horizontally (Figure 3.1) and ±2.6cm vertically (Figure 3.2). Simulating errors in the boresight angles and lever arms at both short and long ranges (Figure 3.3) indicate that these results are highly dependent on the quality of the boresight angles for heading and roll. Indeed the heading and roll seem to be more important than the pitch or any of the lever arm components.

Error propagation, on the other hand, with the expected error estimates (Figure 3.6) indicates that the largest source of error in the point cloud comes from the DG system position. The LiDAR to DG system boresight angles and lever arms do show a significant contribution to the error in the point cloud, however, this contribution is far less significant than the error being generated by the DG system position. In comparison, the results obtained using the ideal error estimates in the error propagation model show that the DG system position is the main source of error in the point cloud (Figure 3.7). In Figure 3.7 only the DG system position (Group 1) and orientation (Group2) errors play a significant role in the final point cloud error. All other groups of variables show error estimates less than 1cm in the ideal case. In both cases (expected and ideal error estimates) the LiDAR errors themselves comprise the least amount of error in the MTL system.

The first order conditional variance analysis using the expected error estimates for the MTL measurements tells a different story from the error propagation. It supports the observations made during the simulations. Figure 3.8 indicates that the DG system position only accounts for about 12% of the total error in the point cloud. The largest

source of error is actually the LiDAR to DG system lever arms, not the DG system position. The fact that both Figures 3.8 and 3.9 show the Z-lever arm and the LiDAR to DG system roll angle contributing upwards of 90% and 70% to the total error at various times highlights the significance of these two parameters. This was not apparent from the error propagation model. The fact that the Z lever arm is one of the most significant parameters makes sense, since the position of the MTL system is mostly determined by the onboard GNSS, which in itself measures its position in the gravity direction less accurately due to the bias caused by all the GNSS satellites being located on the skyward side of the antenna.

Examining the results of conditional variance analysis using the ideal error estimates, it can now be noticed that the DG system position errors make up more than 50% of the errors throughout the point cloud (Figure 3.12). The conditional variance analysis now agrees with the error propagation in that the DG system position is the most significant factor. Figure 3.12 shows that the DG system orientation, LiDAR lever arms and the LiDAR boresight collectively make up a small percentage of the total error. In addition, Figure 3.12 shows that the second most significant source of error is from the LiDAR and that it occurs directly along the MTL system's path. This is due to effects in the LiDAR receiver caused by the extreme short ranges measured by the system. The rest of the error from the LiDAR increases as the range increases. This indicates that the angular uncertainties in the LiDAR's encoder combined with range and zero error uncertainties become more pronounced in the resulting point cloud at range.

3.8 Summary

In this chapter methods for analyzing the accuracy of points in MTL point clouds were discussed. For the first time, the concepts of conditional variance analysis were used to analyze the accuracy of real MTL data. It revealed that the cross effects between the measurements were insignificant contributors to the uncertainty in the MTL point cloud and that the measurements could be dealt with as individual error sources. Conditional variance analysis also revealed that the greatest source of error in the tested MTL point clouds was the system calibration parameters, which position and orient the LiDAR sensors within the DG system coordinate frame. Comparing the uncertainty condition when the patch test calibration parameters were used with the uncertainty condition achievable through a rigorous calibration, conditional variance analysis revealed that the next two sources of error in the point cloud come from the DG system position and the LiDAR sensor's range measurements. Therefore, improvement of the accuracy of the MTL point clouds requires the development of rigorous calibration methods for the LiDAR sensors to the DG system and the correction of the laser ranges from the LiDAR sensors to the targets being scanned. To this end, the next chapter discusses the procedures developed to correct the LiDAR's laser ranges for zero error and temperature drift and a rigorous method for calculating the boresight angles and lever arms for dual sensor MTL systems.

4. Calibration of MTL Systems

As shown in Chapter 3, the output of a high accuracy point cloud from a MTL system is dependent on a number of measurements made by the various components of the MTL system. The LiDAR component of the MTL system has a number of biases and systematic errors inherent in it that can affect the output point cloud. In this chapter novel methods for correcting three of these errors will be examined. First a new method for determining the size and accuracy of the zero and scale errors for the LiDAR sensor is presented, then a new method for identifying and correcting range errors caused by temperature changes inside the LiDAR sensor is described and finally a new method for simultaneously determining the position and orientation of the LiDAR sensors with respect to the DG system is presented. Much of the content in the section dealing with the determination of the position and orientation of the LiDAR sensors with respect to the DG system has been published as “Boresight and Lever Arm Calibration of a Mobile Terrestrial LiDAR System” [112].

4.1 Zero Error of an MTL Sensor

4.1.1 Introduction

Among the biases inherent within any LiDAR sensor is the zero and scale errors for the laser range finder. The concepts behind zero and scale error were previously discussed in Section 2.2.1.4 As stated in Section 2.2.1.4, the zero error is a standard range correction used, mainly, to account for the distance traveled by the laser pulse from the laser emitter to the focusing mirror and the distance traveled by the returning pulse from the focusing

mirror to the receiver. Depending on the design of the LiDAR, the magnitude of this systematic bias can be in the metres. In addition, when the PRF of the laser is variable but the input power to the laser remains constant, the laser range finder zero error will vary in accordance with the selected PRF.

In this section, a new method for calculating the zero error of a MTL sensor is presented. The sensor used to test this method came from a Lynx Mobile Mapper; however, this methodology would work for any LiDAR sensor that uses a spinning mirror to scan a 360° profile. The Lynx sensor lends itself to this type of testing since the calibration numbers are provided by the manufacture in ASCII text file format. Unlike some other systems, where the calibration is stored in proprietary binary formats, the Lynx calibration values are open to the end user.

4.1.2 Mathematical Model and System Setup

The relationship between the zero error and the scale error was presented as Equation (2.15) in Section 2.2.1.4. For most instruments, the standard relationship between zero and scale error is shown in Equation (4.1).

$$\Delta D_{ij} = D_{ij} \cdot \delta_S + K \tag{4.1}$$

Where ΔD_{ij} is the error in the measured range, D_{ij} is the measured range, K is the zero error and δ_S is the scale error. Traditionally, the measured range in Equation (4.1) was obtained by locating the LiDAR at one end of a pre-surveyed line. A target would be placed at the other end of this line and the LiDAR would measure a range to the target. This type of setup is not ideal for the current generation of MTL systems since locating

the center of the mirror at one end of a survey line can be exceedingly difficult. The 360º, planar, measurement produced by all the purpose built MTL systems currently on the market provides an alternative.

By placing a LiDAR sensor between two surveyed targets (as shown in Figure 4.1), the tricky problem of accurately locating the LiDAR system can be avoided. By using measurements on either side of the LiDAR system the range between the targets can be reconstructed quite simply.

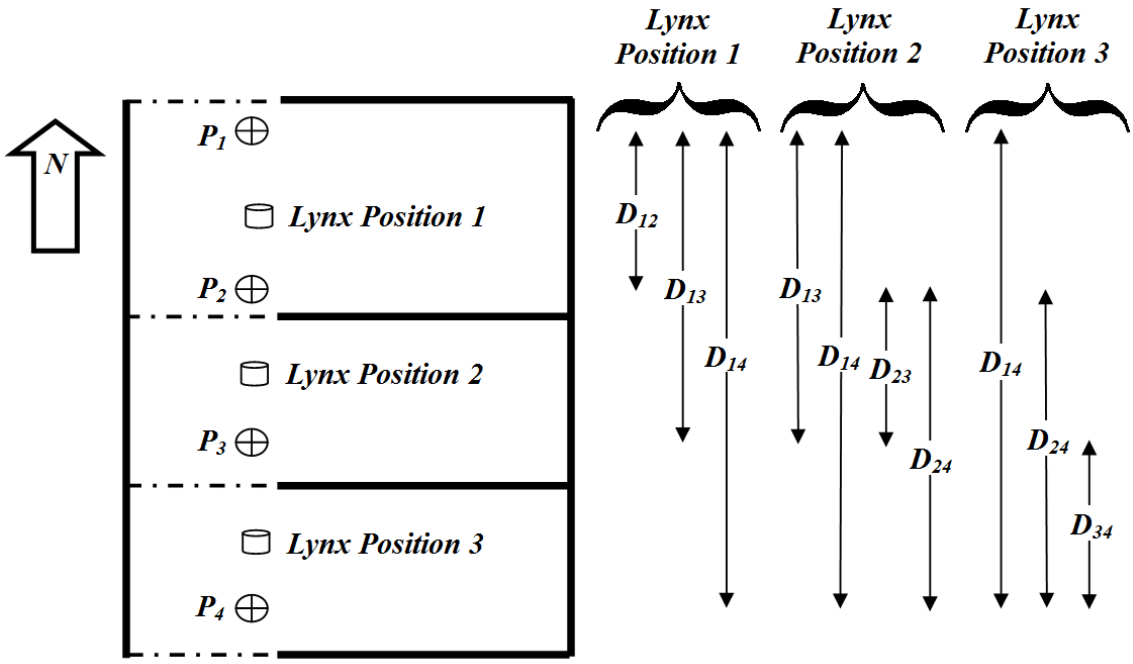

Figure 4.1: Positions of Lynx sensors and fixed targets during zero and scale error calibration.

Based on these measurements, the distances between the targets were simply the sum of the two sensor distances as shown in Equation (4.2).

$$D_{ij} = D_{ij}^{(1)} + D_{ij}^{(2)} \tag{4.2}$$

A simple least square adjustment ($Ax - L = 0$) can be used to compute the zero and scale errors for the two sensors. The normal equations for this adjustment are given in Equation (4.3).

$$\begin{bmatrix} 2 & \left(D_{12}^{(1)} + D_{12}^{(2)}\right) \\ \vdots & \vdots \\ 2 & \left(D_{ij}^{(1)} + D_{ij}^{(2)}\right) \end{bmatrix} \cdot \begin{bmatrix} K \\ \delta_S \end{bmatrix} - \begin{bmatrix} \overline{D}_{12} - D_{12}^{(1)} - D_{12}^{(2)} \\ \vdots \\ \overline{D}_{ij} - D_{ij}^{(1)} - D_{ij}^{(2)} \end{bmatrix} = 0 \tag{4.3}$$

The residuals for this adjustment can be determined according to Equation (4.4).

$$v_{ij} = \overline{D}_{ij} - D_{ij}^{(1)} - D_{ij}^{(2)} - \left(D_{ij}^{(1)} + D_{ij}^{(2)}\right) \cdot \delta_S - 2 \cdot K \tag{4.4}$$

Some LiDAR sensors such as the Lynx do not use a scale error in their range estimates; therefore, it may also be justified to estimate the zero error without a scale error component. Equations (4.3) and (4.4) may be modified to exclude the scale error as shown in Equations (4.5) and (4.6).

$$\begin{bmatrix} 2 \\ \vdots \\ 2 \end{bmatrix} \cdot K - \begin{bmatrix} \overline{D}_{12} - D_{12}^{(1)} - D_{12}^{(2)} \\ \vdots \\ \overline{D}_{ij} - D_{ij}^{(1)} - D_{ij}^{(2)} \end{bmatrix} = 0 \tag{4.5}$$

$$v_{ij} = \overline{D}_{ij} - D_{ij}^{(1)} - D_{ij}^{(2)} - 2 \cdot K \tag{4.6}$$

The accuracy of the computed zero and scale errors can be computed from the least squares matching as shown in Equations (4.7), (4.8) and (4.9).

$$\sigma_0^2 = \frac{\sum_{i=1,j=1}^{m} v_{ij}^2}{n-2} \tag{4.7}$$

$$\sigma_K = \sigma_0 \cdot \sqrt{Q_{11}} \tag{4.8}$$

$$\sigma_{\delta_S} = \sigma_0 \cdot \sqrt{Q_{22}} \tag{4.9}$$

Where m is the number of targets, n is the number of Equations and Q_{11} and Q_{22} are the diagonal terms of the inverse of the coefficient matrix.

4.1.3 Experimental Results

Using the methodology outlined in the previous section, the horizontal distance between the four targets, labeled $\boldsymbol{P_1}$ to $\boldsymbol{P_4}$ in Figure 4.1, were measured by total station, a MTL sensor was calibrated by placing it in a set of arbitrary positions between the targets and recording the distances from the sensor to each target. In the setup used for these trials, the maximum range between the targets employed ($\boldsymbol{P_1}$ to $\boldsymbol{P_4}$) was 29.8252m. To ensure that the sensor was collecting data in the centre of the appropriate targets an InGas camera, capable of detecting 1500 nm light was used to view the laser crossing the target. The sensor was adjusted until the laser was striking the appropriate targets on opposite sides of the sensor, through the target centres. The Lynx M1 LiDAR sensor [1] is capable of pulsing at rates of 500kHz, 250kHz, 125kHz and 75kHz. During this experiment, data was collected at each laser PRF for each sensor position on each target.

Having collected the raw data in the lab, the data was processed in the Optech software package Dashmap. A fake SBET (Smoothed Best Estimated Trajectory) was created so that the collected data could be extruded along the sensor's X axis. Using the extruded Cartesian data produced by Dashmap, each target was found and the associated scanner angles recorded by the system were identified. Using these angles to search an un-extruded version of the point cloud, ranges between each set of targets were found and average values for the slope distances from the sensors to the point cloud were computed. The angles recorded by the sensor where used to compute the horizontal distances from

the sensor to each target. From the internal temperature sensors within the Lynx sensor, it was noted that the average internal temperature of the sensor during the zero error collected was 60°C.

For the Lynx sensor being tested here, the manufacturer's calibrated values for the zero error are given in Table 4.1.

Table 4.1: Manufacturer's zero error estimates for Lynx Mobile Mapper sensor SN131104.

PRF (kHz)	Zero Error (mm)
75	-543.36
125	-924.138
250	-1239.56
500	-1268.489

Table 4.2: Zero and scale errors for Lynx sensor SN131104 at a laser PRF of 500kHz.

Data Set	Mirror Speed (Hz)	Zero Error (mm)	Sigma (mm)	Scale Error (mm/m)	Sigma (mm/m)
Trial 1 22-10-2011	200	-1271.150	2.408	0.504	0.203
Trial 2 18-03-2012	200	-1272.225	3.589	0.520	0.303
Trial 3 18-03-2012	200	-1271.006	3.432	0.418	0.290
Trial 4 22-04-2012	200	-1271.642	3.942	0.359	0.333
Trial 5 22-04-2012	200	-1271.605	2.400	0.471	0.203
Trial 1 22-10-2011	80	-1272.390	3.097	0.568	0.245
Trial 2 18-03-2012	80	-1272.598	2.505	0.442	0.211
Trial 3 18-03-2012	80	-1273.229	3.725	0.517	0.315
Trial 4 22-04-2012	80	-1270.982	2.488	0.400	0.210
Trial 5 22-04-2012	80	-1271.987	2.933	0.470	0.248

Table 4.3: Zero and scale errors for Lynx sensor SN131104 at a laser PRF of 250kHz.

Data Set	Mirror Speed (Hz)	Zero Error (mm)	Sigma (mm)	Scale Error (mm/m)	Sigma (mm/m)
Trial 1 22-10-2011	200	-1255.393	6.360	0.641	0.538
Trial 2 18-03-2012	200	-1252.165	5.023	0.247	0.425
Trial 3 18-03-2012	200	-1255.490	7.441	0.699	0.629
Trial 4 22-04-2012	200	-1252.393	5.863	0.264	0.496
Trial 5 22-04-2012	200	-1255.519	6.691	0.510	0.566
Trial 1 22-10-2011	80	-1256.586	6.455	0.784	0.546
Trial 2 18-03-2012	80	-1252.003	5.386	0.299	0.455
Trial 3 18-03-2012	80	-1255.070	8.264	0.555	0.699
Trial 4 22-04-2012	80	-1255.993	5.591	0.579	0.473
Trial 5 22-04-2012	80	-1256.320	5.246	0.613	0.444

Table 4.4: Zero and scale errors for Lynx sensor SN131104 at a laser PRF of 125kHz.

Data Set	Mirror Speed (Hz)	Zero Error (mm)	Sigma (mm)	Scale Error (mm/m)	Sigma (mm/m)
Trial 1 22-10-2011	200	-1286.473	9.259	3.288	0.783
Trial 2 18-03-2012	200	-1288.106	14.148	3.575	1.197
Trial 3 18-03-2012	200	-1288.604	13.711	3.694	1.160
Trial 4 22-04-2012	200	-1289.516	5.739	3.564	0.485
Trial 5 22-04-2012	200	-1288.165	4.888	3.560	0.414
Trial 1 22-10-2011	80	-1287.634	9.698	3.582	0.820
Trial 2 18-03-2012	80	-1289.926	17.840	3.668	1.509
Trial 3 18-03-2012	80	-1288.879	17.941	3.548	1.518
Trial 4 22-04-2012	80	-1288.912	7.986	3.576	0.676
Trial 5 22-04-2012	80	-1289.115	6.845	3.551	0.579

Five independent data collects on three separate dates were performed using the MTL sensor. Using the information gathered during these data collects a zero and scale error was computed based on Equations (4.3) and (4.4). This was done for two different rotational mirror speeds of 200Hz and 80Hz. The results for each laser PRF are listed in Table 4.2, Table 4.3, Table 4.4 and Table 4.5, respectively.

Table 4.5: Zero and scale errors for Lynx sensor SN131104 at a laser PRF of 75kHz.

Data Set	Mirror Speed (Hz)	Zero Error (mm)	Sigma (mm)	Scale Error (mm/m)	Sigma (mm/m)
Trial 1 22-10-2011	200	-1246.286	12.330	4.059	1.047
Trial 2 18-03-2012	200	-1245.979	9.413	4.121	0.799
Trial 3 18-03-2012	200	-1245.730	6.602	3.996	0.561
Trial 4 22-04-2012	200	-1246.325	8.153	4.085	0.692
Trial 5 22-04-2012	200	-1244.222	8.529	3.695	0.724
Trial 1 22-10-2011	80	-1246.419	16.475	3.829	1.399
Trial 2 18-03-2012	80	-1244.438	4.871	3.929	0.414
Trial 3 18-03-2012	80	-1245.577	5.800	3.896	0.492
Trial 4 22-04-2012	80	-1246.167	13.057	4.063	1.109
Trial 5 22-04-2012	80	-1244.379	11.929	3.835	1.013

While values for the scale error computed here seem high compared to those of a total station, [19] reported scale errors of +400ppm for the Leica HDS 2500 laser scanner. The variation in the scale errors given in Tables 4.2 to 4.5 is more than could be expected by other devices such as a total station. The length of the base line used in testing (29.8252m) is 1/3 the maximum distance of the LiDAR sensor and should, therefore, provide a sufficient range for accurate scale factor error calibration. Since the zero and

scale error numbers generated have a high relative uncertainty and considering that the base line should be sufficient for calibration, the question must be considered as to whether the scale error needs to be computed at all.

Table 4.6: Zero errors for Lynx sensor SN131104 at a laser PRF of 500kHz.

Data Set	Mirror Speed (Hz)	Zero Error (mm)	Sigma (mm)
Trial 1 22-10-2011	200	-1265.502	0.983
Trial 2 18-03-2012	200	-1266.402	1.289
Trial 3 18-03-2012	200	-1266.327	1.183
Trial 4 22-04-2012	200	-1267.619	1.295
Trial 5 22-04-2012	200	-1266.327	0.954
Trial 1 22-10-2011	80	-1265.614	1.324
Trial 2 18-03-2012	80	-1267.651	0.956
Trial 3 18-03-2012	80	-1267.442	1.322
Trial 4 22-04-2012	80	-1266.498	0.921
Trial 5 22-04-2012	80	-1266.723	1.084

Table 4.7: Zero errors for Lynx sensor SN131104 at a laser PRF of 250kHz.

Data Set	Mirror Speed (Hz)	Zero Error (mm)	Sigma (mm)
Trial 1 22-10-2011	200	-1248.230	2.121
Trial 2 18-03-2012	200	-1249.401	1.577
Trial 3 18-03-2012	200	-1247.680	2.457
Trial 4 22-04-2012	200	-1249.439	1.834
Trial 5 22-04-2012	200	-1249.816	2.158
Trial 1 22-10-2011	80	-1247.826	2.225
Trial 2 18-03-2012	80	-1248.658	1.700
Trial 3 18-03-2012	80	-1248.862	2.638
Trial 4 22-04-2012	80	-1249.524	1.872
Trial 5 22-04-2012	80	-1249.461	1.795

Table 4.8: Zero errors for Lynx sensor SN131104 at a laser PRF of 125kHz.

Data Set	Mirror Speed (Hz)	Zero Error (mm)	Sigma (mm)
Trial 1 22-10-2011	200	-1249.708	5.081
Trial 2 18-03-2012	200	-1248.143	6.308
Trial 3 18-03-2012	200	-1247.316	6.329
Trial 4 22-04-2012	200	-1249.667	4.894
Trial 5 22-04-2012	200	-1248.372	4.801
Trial 1 22-10-2011	80	-1247.589	5.469
Trial 2 18-03-2012	80	-1248.921	7.210
Trial 3 18-03-2012	80	-1249.211	7.135
Trial 4 22-04-2012	80	-1248.937	5.194
Trial 5 22-04-2012	80	-1249.413	5.011

Table 4.9: Zero errors for Lynx sensor SN131104 at a laser PRF of 75kHz.

Data Set	Mirror Speed (Hz)	Zero Error (mm)	Sigma (mm)
Trial 1 22-10-2011	200	-1201.100	6.435
Trial 2 18-03-2012	200	-1200.111	6.019
Trial 3 18-03-2012	200	-1201.245	5.506
Trial 4 22-04-2012	200	-1200.851	5.802
Trial 5 22-04-2012	200	-1203.083	5.412
Trial 1 22-10-2011	80	-1203.781	7.053
Trial 2 18-03-2012	80	-1200.702	5.251
Trial 3 18-03-2012	80	-1202.202	5.300
Trial 4 22-04-2012	80	-1200.938	6.573
Trial 5 22-04-2012	80	-1201.685	6.132

Based on Equations (4.5) and (4.6), a zero error only solution was calculated. The results for calculating just the zero error from known target distances for each trial are given in Table 4.6, Table 4.7, Table 4.8 and Table 4.9, respectively. The zero error results reported in Tables 4.6 to 4.9 are noticeably different from those reported in Tables 4.2 to 4.5. This is not surprising as removing the scale error as a parameter in the adjustment means that the zero error is now solely compensating for the perceived error in the actual and measured ranges between the targets used in this test.

4.1.4 Field Verification of Results

The effect of an error in the zero error value for a LiDAR sensor is a warped planar distortion. If all ranges measured by the LiDAR unit are either too short or too long, then one can expect a planar object, perpendicular to the scanning direction, to curve around the scanner centre. This effect has been likened to a smile (or frown) by [20, 21]. A range bias error can cause a serious distortion in LiDAR data and must be reduced or eliminated before data collection proceeds. Not all sources are in agreement about this however. Some sources [13, 110] believe that the warped planar distortion is caused by a

mirror angle scale error. It is possible that this distortion is caused by a combination of these two errors.

To evaluate the effect of the calculated zero errors on real data, test data sets were collected at each laser PRF in a commercial parking lot, over which, a control surface had been surveyed. The LiDAR data was processed using no zero error correction, the manufacturer's zero error correction and the calculated zero error correction. The MTL data was compared to the survey control surface using Polyworks IMSurvey's error map feature. Cross sections perpendicular to the direction of the vehicle where cut through the data and additional measurements made.

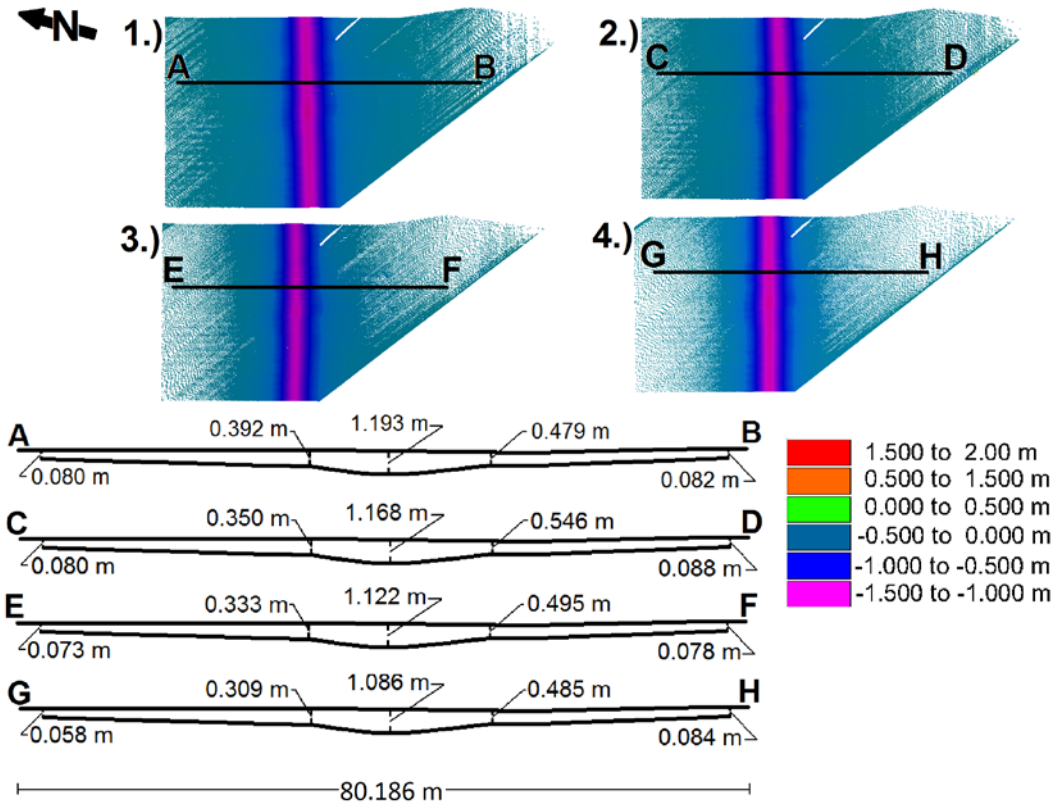

Figure 4.2: Lynx Mobile Mapper scan of a parking lot processed with no zero error applied compared with a control surface established by Total Station observation. Each error map represents the surface at a PRF of 1.) 500kHz, 2.) 250kHz, 3.) 125kHz, 4.) 75kHz.

Figure 4.2 shows the effect of setting the zero error to zero for each laser PRF collected. The purple strip through the center of all four data fragments shown in Figure 4.2, indicate the presence of a trench in the data, directly under the LiDAR sensor and in the direction of motion of the vehicle. The cross sections shown in Figure 4.2 reveal that the depth of this trench varies between 1.086m to 1.196m below the control surface.

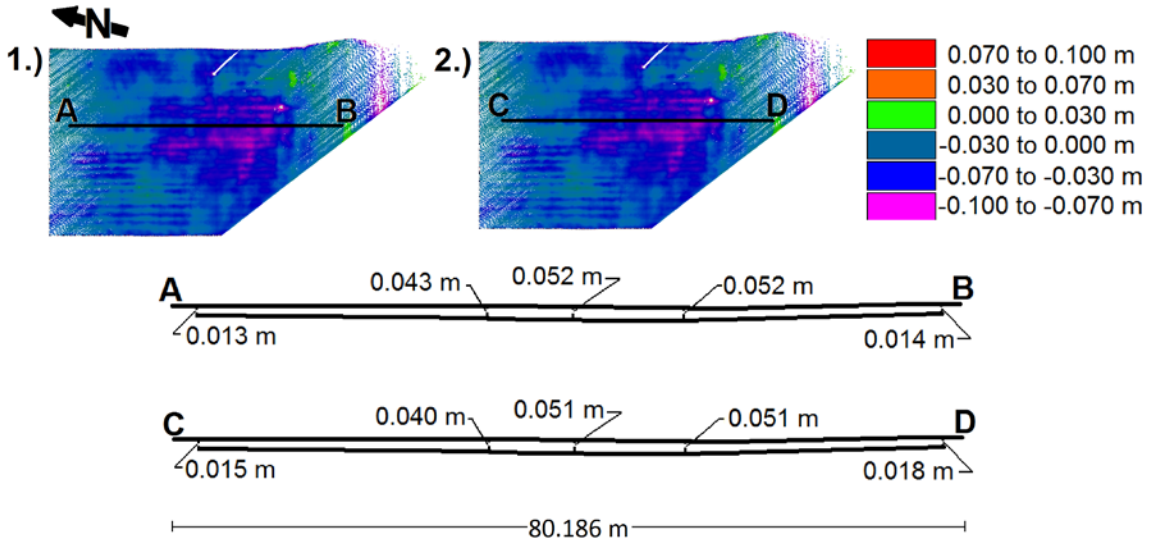

Figure 4.3: 500kHz Lynx Mobile Mapper scan of a parking lot 1.) using manufacturer derived zero error, 2.) using average zero error from values in Table 4.6, compared with a control surface established by Total Station observation.

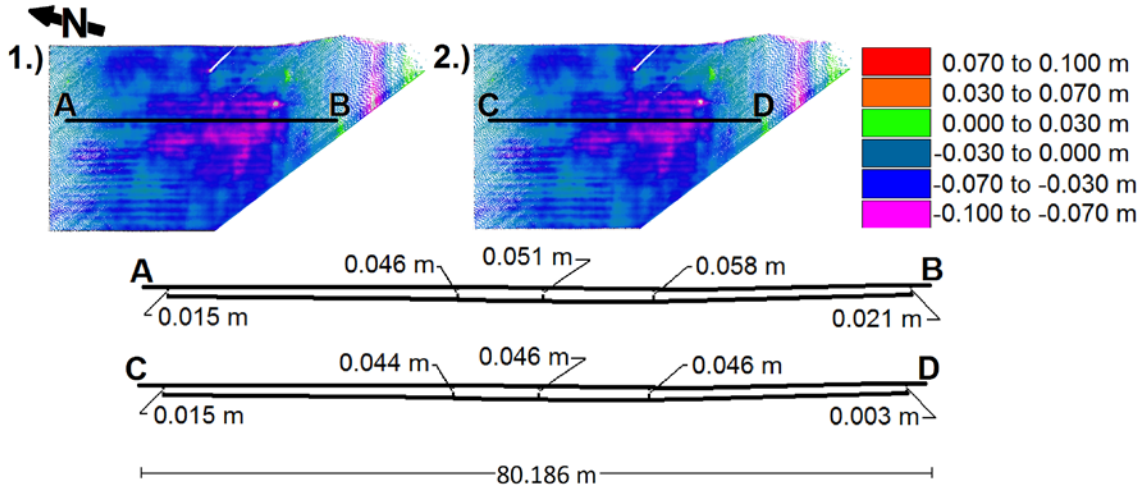

Figure 4.4: 250kHz Lynx Mobile Mapper scan of a parking lot 1.) using manufacturer derived zero error, 2.) using average zero error from values in Table 4.7, compared with a control surface established by Total Station observation.

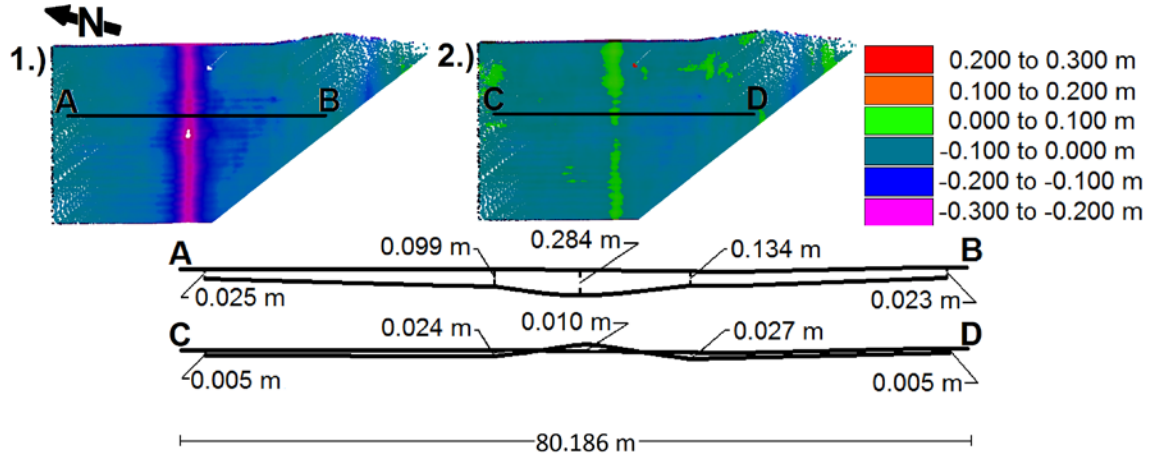

Figure 4.5: 125kHz Lynx Mobile Mapper scan of a parking lot 1.) using manufacturer derived zero error, 2.) using average zero error from values in Table 4.8, compared with a control surface established by Total Station observation.

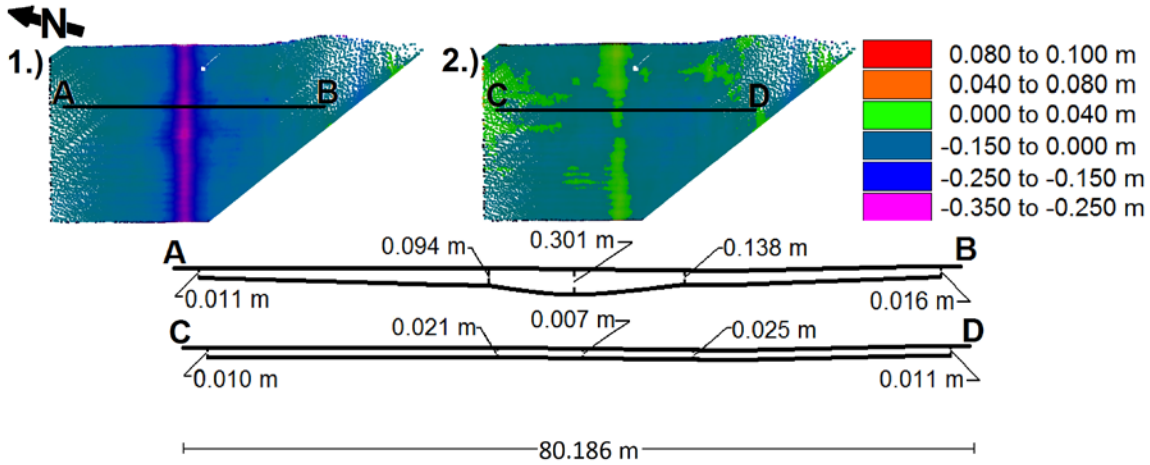

Figure 4.6: 75kHz Lynx Mobile Mapper scan of a parking lot 1.) using manufacturer derived zero error, 2.) using average zero error from values in Table 4.9, compared with a control surface established by Total Station observation.

Figures 4.3 and 4.4 show that the zero errors for 500kHz and 250kHz generally contain the same amount of deviation from the control surface whether the manufacturer's zero error is used or the zero error calculated in the field is used. This is due to the fact that the zero errors reported by the manufacturer and the zero errors calculated in the lab did closely agree with one another. Figures 4.5 and 4.6 show that the zero errors for 125kHz and 75kHz deviate widely from the control surface, while using the manufacturer's zero error, but are much better behaved with the zero errors calculated in the lab. These zero

errors were different by decimeters between the manufacturer's zero error values and the calculated zero error values. The cross sections of Figures 4.4 and 4.5 show that the manufacturer determined zero errors deviate from the control surface in the classical smile pattern discussed in [20, 21]. The calculated average zero error produced a much closer approximation to the control surface with little smile or frown present.

4.2 Temperature Changes in MTL Sensors

4.2.1 Introduction

As discussed in Section 2.2.1.5, the pulsed time-of-flight LiDAR sensor experiences a range walk as temperature changes. A commercial pulsed time-of-flight LiDAR is a precise timing instrument similar to GPS. Unlike GPS however, LiDAR does not have any access to the adjusted atomic clock data. Instead, it relies on a purely electronic means of measuring time. Modern electronics, when operating, generate a lot of heat. Sealing these electronics inside a weather proof case, such as is done with LiDAR sensors, causes heat to build up inside this case. Excessive heat can not only disturb the sensitive measurements being performed by the electronics in a laser range finder, it can damage those electronics. Fans and heat transferable paste are used to control this temperature rise and direct heat away from the electronics. This allows a LiDAR sensor to operate for long periods of time without damaging the electronics inside. [15, 48, 52, 57] have documented this phenomenon in STL systems.

As the timing board inside a LiDAR set is stressed by temperature, the time it reports will become a few nanoseconds longer or shorter. Whether the error is longer or shorter

varies from timing board to timing board. In either case, this error is directly translated into the range reported by the unit and is measurable. During the first few minutes after a LiDAR sensor is started, the temperature inside the LiDAR unit will increase until a stable temperature is reached. This stable temperature is the point at which the manufacturer designed cooling methods counteracts further heat buildup. The LiDAR system was most likely calibrated at or around this temperature, so this would be the temperature at which maximum range accuracy can be achieved. This stable temperature will vary from LiDAR startup to LiDAR startup and is affected by the ambient temperature surrounding the unit.

Some types of LiDAR equipment have built-in means of automatically estimating and applying corrections for this temperature walk. Other types of LiDAR equipment use material or mechanical means of keeping the temperature stable. In most cases it is the type and use of the LiDAR equipment which dictates the method of temperature correction. Terrestrial based static scanners such as Optech's ILRIS HD (www.optech.ca), use a series of internal targets of different intensity values, set at precisely known distances from the scanning mechanism, to estimate the temperature error at regular intervals. This methodology requires the ILRIS scanner to pause scanning at regular intervals, and spend a couple of seconds scanning the internal targets. The differences between the known target ranges and the measured target ranges are stored and applied to the range data post-process.

This is fine for a static scanner, but a MTL, cannot afford to pause scanning and conduct this type of test. Doing so would cause gaps in the resulting point cloud. Therefore, the temperature must be controlled so that temperature walk does not become too great. To do this, the timing board is surrounded by a heat absorbent material and a cooling system employing low voltage fans is deployed. If the mobile LiDAR is moving fast enough, exposure of the system to the air flowing across the vehicle will work as well. Of course, if the ambient air temperature is low and the speed is great, this could cause the sensor to operate at too cool a temperature and introduce a range error.

Among the pieces of information collected by the Lynx Mobile Mapper sensor are measurements of both the exterior and interior temperatures of the sensor. These temperatures are collected at a rate of 1 reading per second while the sensor is operating. This type of information gives a new option in correcting range measurements based on sensor temperature. In this chapter, a novel approach to temperature correction in MTL systems using observed interior temperature readings and ranges is presented. Due to the nature of the observations collected by the Lynx sensor, the noise in the observed temperature/range readings precluded the formation of a range correction table. To solve this problem, a form of the Kalman filter, known as the α-β-γ smoother was employed to smooth the noise in the range readings and produce a range correction table based on the observed temperature/range data.

4.2.2 Establishing the Amount of Temperature Walk in an MTL Sensor

4.2.2.1 Experiment 1

An experiment was conducted to understand the effect of temperature walk on MTL sensors. A MTL sensor was placed in a holder on a table in the laboratory. The sensor was placed in such a way that the laser path was roughly perpendicular with two of the room's walls. The system was started and the path of the laser beam was traced using an InGaAs camera. Two black and white circular targets were placed on the opposing laboratory walls such that they were the same height above the floor and the centre of the laser beam passed through the centre of the targets. The MTL sensor was powered off and allowed to cool.

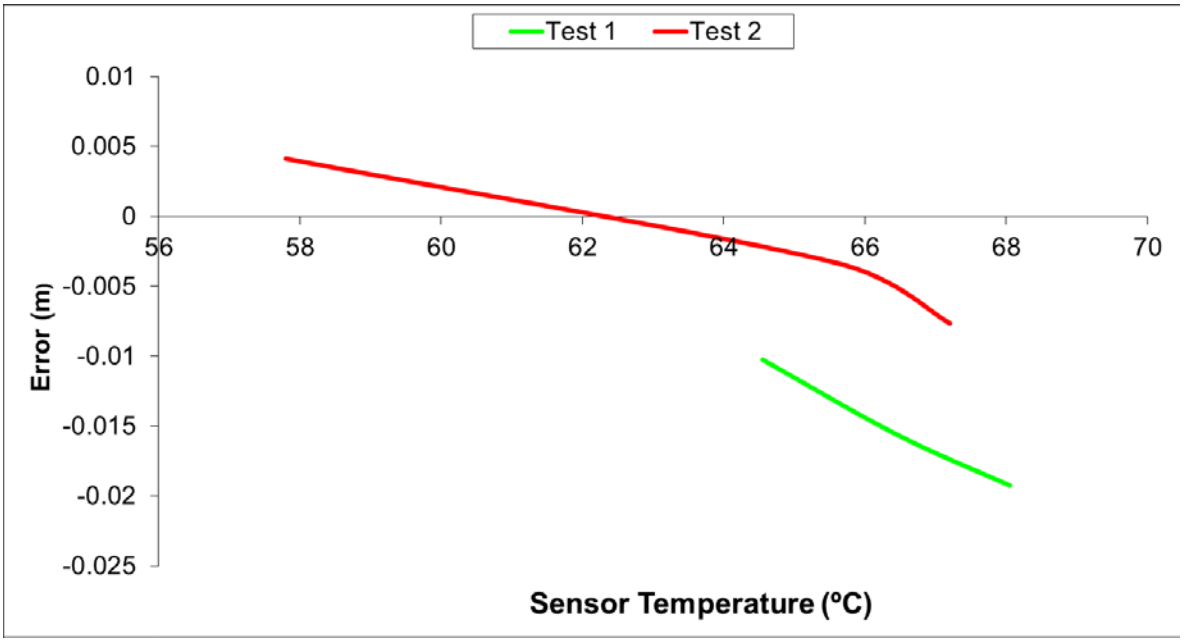

Figure 4.7: Results of laboratory temperature tests on a Lynx Mobile Mapper sensor head.

The distance between the two targets was measured, first using a surveyor's chain, and then verified using a laser range finder. The distance between the two targets was

determined to be between 8.234 m (surveyor's chain) and 8.236 m (laser range finder). The measurements were conducted several times to verify the distance.

Two test runs were conducted. The first oriented the sensor on its side; the second oriented the sensor similar to how it would mount on a vehicle. In both test cases, data was collected as soon as possible after startup. In each test, three 1 minute long data strips were collected with 10 minute idle intervals in between data collects. The data was then processed, using an artificially derived trajectory file to extrude the sensor data. The timestamps, associated with each LiDAR point, were used to trace scan lines and ensure that measurements were taken between points collected milliseconds apart. To account for white noise in the sensor, nine consecutive points of comparison were taken across each of the opposing targets. The range between the opposing pairs of consecutive points was measured from the LiDAR data and the average range was computed. The results are given in Figure 4.7.

Figure 4.7 shows that in test 1, the sensor started with a board temperature of 65.5°C and ended with a board temperature of 68.0°C. Over this interval the difference between the physically measured target to target range and the LiDAR measured target to target range became smaller by about 9mm. Test 2 showed a similar performance. This time the starting temperature was 57.8°C and the ending temperature was 67.2°C. The difference in range between the two targets became smaller by about 12mm.

4.2.2.2 Experiment 2

Two MTL sensors were subjected to temperature variation under laboratory conditions. A target was placed at a distance of approximately 12.5m from the sensors. The experiment had the LiDAR unit begin scanning the target while the exterior temperature of the sensor was maintained at 21ºC. The temperature was then decreased by 20ºC every 20 minutes, until the sensors exterior temperature reached -20ºC. The exterior temperature of the sensor was then increased at a similar rate of 20ºC every 20 minutes, until the unit reached an exterior temperature of 50ºC. This temperature range was slightly beyond the manufacturer's stated operating range of -10ºC to +40ºC [1], but within the normal operating conditions of people in the field. While the unit was scanning the target, the temperature on the LiDAR's timing board was recorded at a rate of one reading per second. After the test was complete, the outside air pressure was recorded as 102.004 kPa with an average humidity of 77%.

The LiDAR's measured angle and range data was processed so that the zero error correction and the optical model were applied to the data. The angle and range data were scanned and all range readings with measured angles of 0.002644º for Sensor 1 and 0.004128º for Sensor 2 were extracted. The timestamp associated with each of the ranges was used to identify the correct temperature for the timing board. Linear interpolation was used to compute the exact temperature from the temperature data. The temperatures were then used to bin the range data. For each unique temperature, all range data at that temperature was averaged and a standard deviation was computed. These results are show in Figure 4.8 for Sensors 1 and 2.

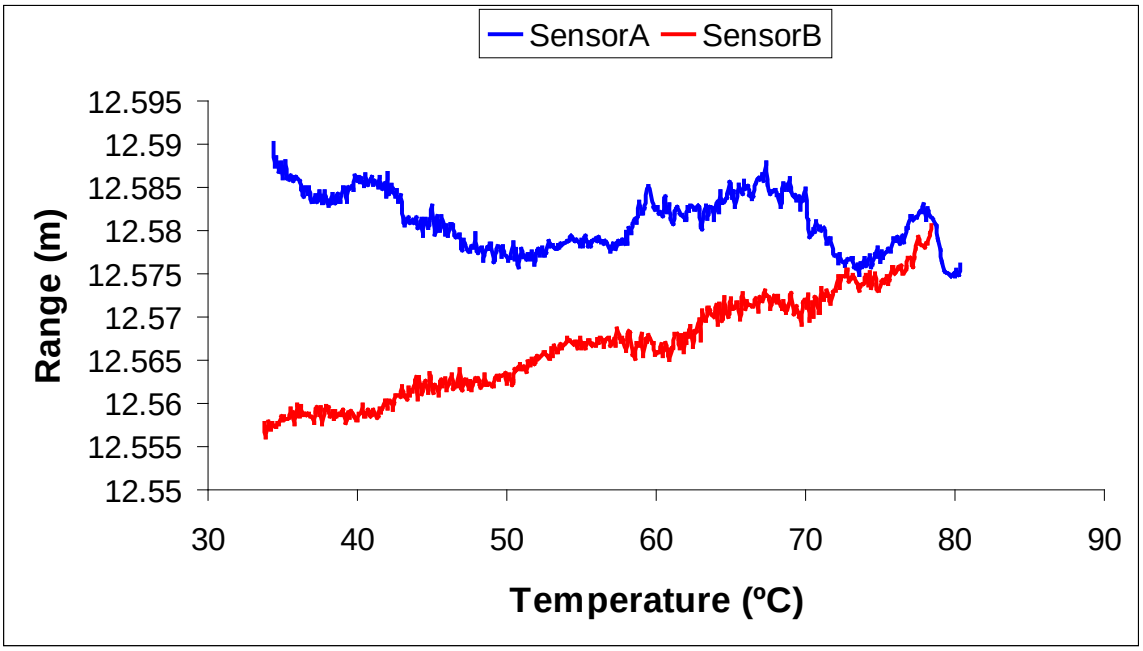

Figure 4.8: Range variation in Sensors 1 and 2 due to external temperature fluctuation. Results are for experiments conducted on April 7th, 2011.

The curves shown in Figure 4.8 clearly show that both sensors are subject to a range variation due to the interior temperature of the sensor. The amount of variation caused by the changing temperature can be seen most clearly in Table 4.10. From Table 4.10, it can be observed that the overall drift in Sensor 1 was 0.015m and that the overall drift in Sensor 2 was 0.025m. As was expected, the maximum and minimum values occurred close to the extremes of the temperature range tested.

Looking at the curves in Figure 4.8 it can also be observed that the drift due to temperature variation is not the same for each sensor. Computing the correlation values for the two curves results in a correlation value of 0.39, indicating that the curves are not very similar.

Table 4.10: Maximum and minimum data values for ranges measured during temperature testing. Results are for experiments conducted on April 7th, 2011.

	Sensor 1				Sensor 2			
	Temp (ºC)	Average Range (m)	Standard Deviation (m)	No. of Ranges	Temp (ºC)	Average Range (m)	Standard Deviation (m)	No. of Ranges
Max Range	34.4	12.5901	0.0065	266	78.4	12.5807	0.0038	151
Min Range	79.7	12.5747	0.0035	145	33.9	12.5561	0.0063	231
Max STD	39.5	12.5849	0.0080	101	36.3	12.5586	0.0074	108
Min STD	73.7	12.5763	0.0024	63	57.5	12.5675	0.0028	22
Max No. of Ranges	57.8	12.5790	0.0045	450	56.7	12.5674	0.0040	510
Min No. of Ranges	59.5	12.5852	0.0029	11	58.9	12.5667	0.0048	16

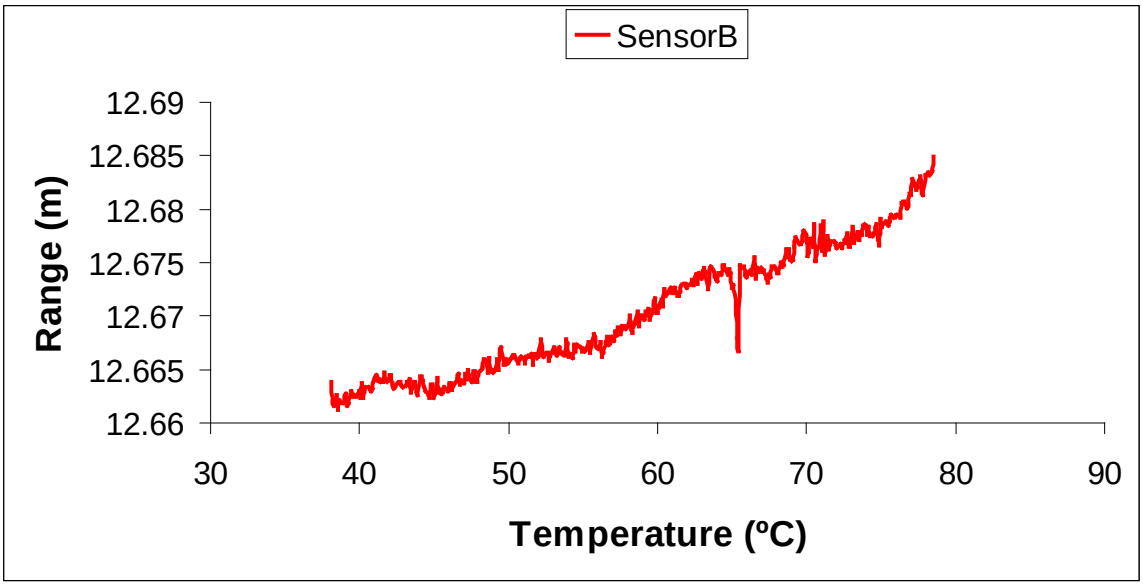

Figure 4.9: Range variation in Sensor 2 due to external temperature fluctuation. Results are for experiments conducted on June 6th, 2011.

To ensure that the results showing temperature walk were reproducible, the same test was performed on Sensor 2. Performing the test again, when the atmospheric pressure was different, was important in showing that temperature drift could be modeled. After the test was completed, atmospheric pressure was measured at 101.438 kPa with an average humidity of 70%. The results of the second test, performed on June 6th, 2011, are shown in Figure 4.9 and Table 4.11.

Table 4.11: Maximum and minimum data values for ranges measured during temperature testing. Results are for experiments conducted on June 6th, 2011.

	Sensor 2			
	Temp (ºC)	Average Range (m)	Standard Deviation (m)	No. of Ranges
Max Range	78.5	12.6849	0.0044	139
Min Range	38.5	12.6612	0.0046	127
Max STD	47.5	12.6639	0.0062	56
Min STD	72.7	12.6778	0.0030	30
Max No. of Ranges	65.4	12.6667	0.0051	508
Min No. of Ranges	71.0	12.6761	0.0036	27

Due to the arbitrary distance used when placing the sensor, the ranges collected do not match those from the previous trial. The data in Table 4.11 shows that the maximum and minimum ranges occur at the temperature extremes exactly as before. It also shows that the data trends the same way and that the difference between the maximum and minimum ranges is 0.024m which is pretty much the same as before. In addition, calculating a value of 0.95 for the correlation coefficient between the data from the two trials of Sensor 2 indicates that the two data sets closely agree. Despite different humidity and atmospheric pressure variables, the two trials of Sensor 2 show that the range walk due to temperature is repeatable on an individual MTL sensor.

4.2.3 Estimating Temperature Corrections

To create a set of correction tables, the temperature at which the zero error for each sensor was calculated was found and used as the point of zero correction. For both sensors it was determined that the zero error was set when the sensor had an internal average temperature of 60ºC. All other range readings were subtracted from the recorded range at this temperature. Several attempts to model the curves in Figure 4.8 using polynomial regression were made; however, no single model could be found which could

describe both curves. The curves previously established contained a lot of high frequency white noise, which is incompatible with creating an error graph or error reference table. It was decided to filter out the white noise using the version of the α-β-γ smoother described in Section 5.2.1. Using the ranges and the standard deviation estimates generated by the bins of temperature data, a smoothed table of values was generated.

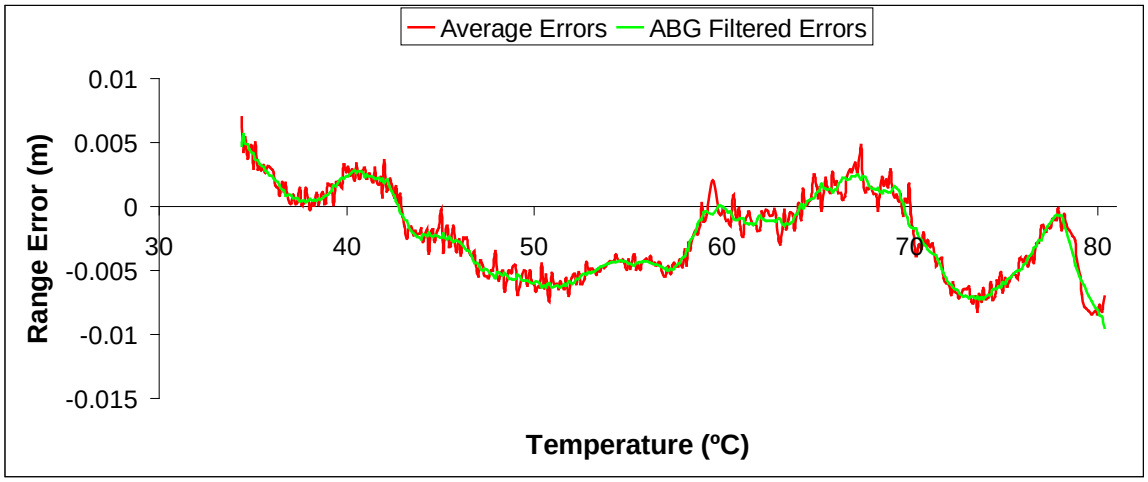

Figure 4.10: Average errors and α-β-γ smoothed errors for LiDAR ranges taken with Sensor 1. These errors are centred on a standard operating temperature of 60ºC.

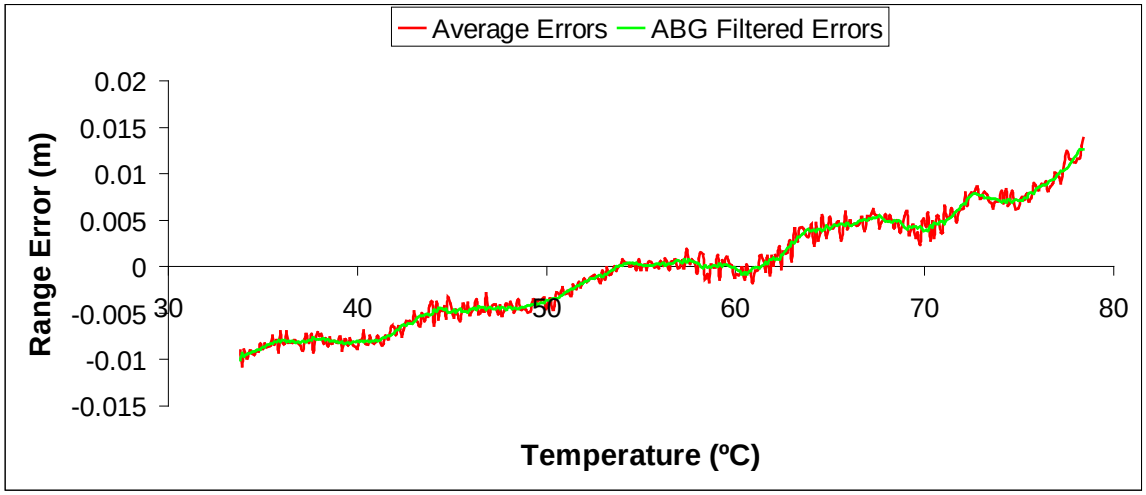

Figure 4.11: Average errors and α-β-γ smoothed errors for LiDAR ranges taken with Sensor 2. These errors are centred on a standard operating temperature of 60ºC.

Figure 4.10 shows the curves for the estimated average errors for Sensor 1 compared to the α-β-γ smoothed errors for Sensor 1. Similarly, Figure 4.11 shows the curves for the estimated average errors for Sensor 2 compared to the α-β-γ smoothed errors for Sensor 2. In both cases one can see how the white noise has been removed by the α-β-γ smoother, from the average errors for each sensor.

4.2.4 Evaluating the Results

To evaluate how well the range temperature correction table works, base lines of known length needed to be established. Since the MTL sensors are not really designed to be positioned over a known point and surveying in the scanning centre of the MTL sensors proved too difficult, a different strategy needed to be adopted. Therefore, the same setup used for calculating the instruments zero error (Section 4.1) was employed to test the range correction tables. Four targets were placed over permanent survey points and horizontal ranges were measured between these stations using a Leica TC1800 (1", 1mm + 2ppm) total station. The MTL sensors were placed between these target sets and ranges were measured to both targets using the 360° vertical rotation of the LiDAR scanner. Combining the ranges measured by the LiDAR to each target gave a LiDAR range between targets. This setup resulted in 10 different ranges measured by the LiDAR between the known targets. Figure 4.1 in Section 4.1.2 illustrates the setup of the four targets and the positions of the LiDAR in-between these targets.

Two trials were conducted using Sensor 2. The zero error, established in 4.1 was applied to all the ranges before temperature testing began. Between the two scans of the control field, the pressure, humidity and ambient temperature of the surroundings varied. The

first trial was performed on October 22, 2011 when the pressure was 994.7hPa, the relative humidity was 68% and the ambient temperature was 19.1°C. The results from the first trial are shown in Table 4.12. The second trial occurred on March 18, 2012 when the pressure was 1021.32hPa, the relative humidity was 92% and the ambient temperature was 22.0°C. The results of the second trial are shown in Table 4.13.

Table 4.12: Comparison of LiDAR measured ranges to total station measured ranges before and after temperature compensation was applied. This trial was conducted on October 22nd, 2011.

Position	Range	Average Sensor Temperature (°C)	Total Station Measurement R_{TS}(m)	LiDAR Measurement with No Temperature Correction R_{NT}(m)	LiDAR Measurement with Temperature Correction R_T(m)	Δ Between Total Station and Uncorrected Ranges [R_{TS} - R_{NT}] (m)	Δ Between Total Station and Corrected Ranges [R_{TS} - R_T] (m)
1	D_{12}	58.4	9.5407	9.533	9.533	0.007	0.007
1	D_{13}	62.2	19.4335	19.435	19.434	-0.002	-0.001
1	D_{14}	63.6	29.8252	29.825	29.822	0.000	0.003
2	D_{13}	69.6	19.4335	19.442	19.438	-0.008	-0.004
2	D_{14}	69.4	29.8252	29.825	29.820	0.001	0.005
2	D_{23}	69.8	9.8629	9.873	9.869	-0.010	-0.006
2	D_{24}	69.9	20.2545	20.270	20.266	-0.015	-0.011
3	D_{14}	69.3	29.8252	29.834	29.830	-0.009	-0.005
3	D_{24}	68.7	20.2545	20.265	20.260	-0.011	-0.006
3	D_{34}	66.8	10.3616	10.375	10.370	-0.014	-0.009

As Tables 4.12 and 4.13 show in most cases there was a substantial improvement in the ranges measured between the two targets. In many cases the temperature correction in the range accounted for more than 1cm of error. Being that the accuracy of the laser rangefinder used in the Lynx Mobile Mapper is ±5mm [1]; this correction is outside the white noise of the laser rangefinder and therefore a significant error. In one case the

temperature correction made no difference. Target D_{12} in Table 4.12 saw no significant correction applied. Target D_{14} in Table 4.12, on the other hand, shows that the temperature correction made the measured range slightly worse. In this case, the maximum correction applied to Target D_{14} in Table 4.12 is 4mm. It is safe to conclude that the effect of the temperature for target D_{14} is within the white noise of the laser range finder and therefore acceptable.

Table 4.13: Comparison of LiDAR measured ranges to total station measured ranges before and after temperature compensation was applied. This trial was conducted on March 18th, 2012.

Position	Range	Average Sensor Temperature (°C)	Total Station Measurement R_{TS}(m)	LiDAR Measurement with No Temperature Correction R_{NT}(m)	LiDAR Measurement with Temperature Correction R_T(m)	Δ Between Total Station and Uncorrected Ranges [R_{TS} - R_{NT}] (m)	Δ Between Total Station and Corrected Ranges [R_{TS} - R_T] (m)
1	D_{12}	64.25	9.5407	9.551	9.544	-0.011	-0.003
1	D_{13}	67.5	19.4335	19.441	19.430	-0.007	0.003
1	D_{14}	69	29.8252	29.838	29.830	-0.013	-0.005
2	D_{13}	71.8	19.4335	19.451	19.439	-0.017	-0.005
2	D_{14}	71.1	29.8252	29.840	29.830	-0.014	-0.005
2	D_{23}	72.4	9.8629	9.886	9.871	-0.023	-0.008
2	D_{24}	72.6	20.2545	20.274	20.258	-0.019	-0.004
3	D_{14}	72.05	29.8252	29.852	29.839	-0.027	-0.014
3	D_{24}	72.7	20.2545	20.274	20.258	-0.020	-0.004
3	D_{34}	73	10.3616	10.397	10.382	-0.036	-0.020

On the other hand, both D_{14} and D_{34} in Table 4.13 still show range discrepancies greater than 1cm after temperature correction. As previously stated, there are many factors that contribute to the error in a laser range finder. From the intensity information for the ranges collected on Target 4 from position 3 on March 18th, 2012, it becomes apparent that the receiver in the sensor was close to saturation. This is probably due to the proximity of the sensor to that target. Saturation in the receiver during measurement can

greatly add error to that range measurement. However, the temperature corrections for these measurements were applied correctly and these corrections cause the LiDAR ranges to more closely conform to the total station measured ranges.

4.3 Calibration of MTL Sensors to the DG System

4.3.1 Introduction

One of the most pressing problems for MTL users to solve is system integration between the LiDAR sensor and the DG system. As was shown in Section 3.0, the integration of LiDAR and DG system creates the need for precise and accurate alignment information between the two sensor coordinate frames. If the error in the angular alignment in any of the three boresight parameters is allowed to become 0.02° or higher, then this error will become the dominant error source in the final data set. Similarly, if the lever arm offset between the LiDAR and the DG system is allowed to become 0.02m or higher, then this error will similarly dominate the error in the final point cloud.

Methods for performing a rigorous boresight of LiDAR sensors to a DG system were discussed in Section 2.2.3.2.4. Methods for accurately selecting target objects from the point cloud have been proposed by several sources [6, 7, 23]. These methods are needed when calibrating LiDAR sensors since LiDAR data is, by its nature, discreet. This means that control points usually do not coincide with any single LiDAR point. Therefore a means for dealing with the mismatch between LiDAR and control is required lest user pick error becomes the limiting factor in calibrating a sensor. The methods for dealing with this problem generally fall into one of three categories, point based, line based and

plane based. It has been shown [111] that point-based and plane-based self-calibration of STL scanners can produce comparable results.

In this chapter, a new method for determining the boresight angles and lever arm offsets for MTL systems with multiple pairs of sensors is explored. To understand how this new method works, it is first important to derive the mathematical framework behind two accepted methods in use today, finding boresight and lever arms from ground control (BLAGC) and finding boresight and lever arms from opposing data strips (BLAOS). Finding boresight and lever arms from ground control (BLAGC) is a fairly straight forward process where a least squares adjustment is employed to calibrate a single MTL sensor by fitting the data to the predefined geometry of a field of control points. Finding boresight and lever arms from opposing data strips (BLAOS) is a method for calibrating a single MTL sensor by fitting the data from a single pass of a calibration scene or object to the data from a different pass of that same scene or object, as observed from a different viewing angle and position.

The new method, finding boresight and lever arms from data collected by two sensors operating concurrently (BLATS), extends and combines the previous two methods. It uses the geometry created by simultaneously operating pairs of LiDAR sensors to compute the calibration parameters for both sensors simultaneously. One of the benefits of using pairs of sensors in such an adjustment is that multiple passes of a group of target objects is not necessary to estimate boresight parameters for the system. Another benefit of concurrently calibrating two sensors is the nullification of much of the positioning

error inherent in the DG system [112]. In addition, this chapter will present the results of experiments to determine how well the lever arms can be determined. It will be shown how vector geometry can be used in a point based correction methodology to reduce the discrepancy between control points and LiDAR points.

4.3.2 The Mathematical Models

4.3.2.1 Finding Boresight Angles and Lever Arms from Ground Control (BLAGC)

Using the LiDAR trajectory Equation (3.1) from Section 3.2.1, an adjustment to find the three boresight angles $(\theta_X, \theta_Y, \theta_Z)$ and the three lever arm offsets (l_X, l_Y, l_Z) can be formed. By comparing the results of Equation (3.1) with control points provided by the user, the basic equation of the adjustment can be formed as is given in Equation (4.10)

$$P^{ECEF}(\theta_X, \theta_Y, \theta_Z, l_X, l_Y, l_Z) - C^{ECEF} = 0 \tag{4.10}$$

Where P^{ECEF} is the computed position of a LiDAR point and C^{ECEF} is the corresponding user provided control point. Similar to Equation (3.9), a linearized form of Equation (3.1) must be produced, as shown in Equation (4.11).

$$\begin{aligned} P^{ECEF} \approx P_0^{ECEF} &+ \frac{\partial P^{ECEF}}{\partial \theta_X} \cdot \Delta\theta_X + \frac{\partial P^{ECEF}}{\partial \theta_Y} \cdot \Delta\theta_Y + \frac{\partial P^{ECEF}}{\partial \theta_Z} \cdot \Delta\theta_Z \\ &+ \frac{\partial P^{ECEF}}{\partial l_X} \cdot \Delta l_X + \frac{\partial P^{ECEF}}{\partial l_Y} \cdot \Delta l_Y + \frac{\partial P^{ECEF}}{\partial l_Z} \cdot \Delta l_Z \end{aligned} \tag{4.11}$$

Using the linearized equation (Equation (4.11)), the least square adjustment can be formed.

$$A_1^T \cdot W \cdot A_1 \cdot X_1 = A_1^T \cdot W \cdot L_1 \tag{4.12.1}$$

Where

$$A_1 = \begin{bmatrix} \frac{\partial P^{ECEF}}{\partial \theta_X}_{(1)} & \frac{\partial P^{ECEF}}{\partial \theta_Y}_{(1)} & \frac{\partial P^{ECEF}}{\partial \theta_Z}_{(1)} & \frac{\partial P^{ECEF}}{\partial l_X}_{(1)} & \frac{\partial P^{ECEF}}{\partial l_Y}_{(1)} & \frac{\partial P^{ECEF}}{\partial l_Z}_{(1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{\partial P^{ECEF}}{\partial \theta_X}_{(N)} & \frac{\partial P^{ECEF}}{\partial \theta_Y}_{(N)} & \frac{\partial P^{ECEF}}{\partial \theta_Z}_{(N)} & \frac{\partial P^{ECEF}}{\partial l_X}_{(N)} & \frac{\partial P^{ECEF}}{\partial l_Y}_{(N)} & \frac{\partial P^{ECEF}}{\partial l_Z}_{(N)} \end{bmatrix} \quad (4.12.2)$$

$$X_{BL1} = \begin{bmatrix} \Delta\theta_X & \Delta\theta_Y & \Delta\theta_Z & \Delta l_X & \Delta l_Y & \Delta l_Z \end{bmatrix}^T \quad (4.12.3)$$

$$L_1 = \begin{bmatrix} \left(C^{ECEF} - P_0^{ECEF}\right)_{(1)} & \cdots & \left(C^{ECEF} - P_0^{ECEF}\right)_{(N)} \end{bmatrix}^T \quad (4.12.4)$$

$$W = \begin{bmatrix} \frac{1}{\sigma^2_{C\ (1)}} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{1}{\sigma^2_{C\ (N)}} \end{bmatrix} \quad (4.12.4)$$

Solving Equation (4.12.1) at the initial approximates for the boresight and lever arm values $\left(\theta_X^0, \theta_Y^0, \theta_Z^0, l_X^0, l_Y^0, l_Z^0\right)$ results in Equation (4.13)

$$X_{BL1} = \left(A_1^T \cdot W \cdot A_1\right)^{-1} \cdot A_1^T \cdot W \cdot L_1 \quad (4.13)$$

4.3.2.2 Finding Boresight Angles and Lever Arms from Opposing Data Strips (BLAOS)

One of the most popular ways of deriving boresight values for MAL and MTL is to compare multiple passes of a target object or scene against one another. Again, using the LiDAR trajectory Equation (3.1) from Section 3.2.1 and differencing data taken in the same area but at different times along the trajectory, the boresight angles can be estimated without the aid of ground control. The new basic equation of the adjustment is shown in Equation (4.14).

$$P_{P1}^{ECEF}\left(\theta_X, \theta_Y, \theta_Z, l_X, l_Y, l_Z\right) - P_{P2}^{ECEF}\left(\theta_X, \theta_Y, \theta_Z, l_X, l_Y, l_Z\right) = 0 \quad (4.14)$$

Where the label (P1) denotes the first pass of the sensor by the calibration object and the label (P2) denotes the second pass of the sensor by the same calibration object. The adjustment proceeds similarly as previously shown. Equations (4.11) to (4.13) remain the same, except for Equation (4.12.2) and Equation (4.12.4) which are replaced by Equation (4.15) and Equation (4.16).

$$A_2 = \begin{bmatrix} \left(\left(\frac{\partial P^{ECEF}}{\partial \theta_X}\right)_{P1} - \left(\frac{\partial P^{ECEF}}{\partial \theta_X}\right)_{P2}\right)_{(1)} & \cdots & \left(\left(\frac{\partial P^{ECEF}}{\partial l_Z}\right)_{P1} - \left(\frac{\partial P^{ECEF}}{\partial l_Z}\right)_{P2}\right)_{(1)} \\ \vdots & \vdots & \vdots \\ \left(\left(\frac{\partial P^{ECEF}}{\partial \theta_X}\right)_{P1} - \left(\frac{\partial P^{ECEF}}{\partial \theta_X}\right)_{P2}\right)_{(N)} & \cdots & \left(\left(\frac{\partial P^{ECEF}}{\partial l_Z}\right)_{P1} - \left(\frac{\partial P^{ECEF}}{\partial l_Z}\right)_{P2}\right)_{(N)} \end{bmatrix} \tag{4.15}$$

$$L_2 = \begin{bmatrix} \left(\left(P_0^{ECEF}\right)_{P2} - \left(P_0^{ECEF}\right)_{P1}\right)_{(1)} & \cdots & \left(\left(P_0^{ECEF}\right)_{P2} - \left(P_0^{ECEF}\right)_{P1}\right)_{(N)} \end{bmatrix}^T \tag{4.16}$$

While control points are not necessary to perform this type of boresight adjustment, they can be introduced as conditions to constrain the adjustment. To introduce these constraints on the adjustment, Equation (4.15) and (4.16) can be replaced with Equations (4.17) and (4.18), respectively.

$$B_2 = \begin{bmatrix} \left(\left(\frac{\partial P^{ECEF}}{\partial \theta_X}\right)_{P1} - \left(\frac{\partial P^{ECEF}}{\partial \theta_X}\right)_{P2}\right)_{(1)} & \cdots & \left(\left(\frac{\partial P^{ECEF}}{\partial l_Z}\right)_{P1} - \left(\frac{\partial P^{ECEF}}{\partial l_Z}\right)_{P2}\right)_{(1)} \\ \vdots & \vdots & \vdots \\ \left(\left(\frac{\partial P^{ECEF}}{\partial \theta_X}\right)_{P1} - \left(\frac{\partial P^{ECEF}}{\partial \theta_X}\right)_{P2}\right)_{(N)} & \cdots & \left(\left(\frac{\partial P^{ECEF}}{\partial l_Z}\right)_{P1} - \left(\frac{\partial P^{ECEF}}{\partial l_Z}\right)_{P2}\right)_{(N)} \\ \hline \frac{\partial P^{ECEF}}{\partial \theta_X}_{(1)} & \cdots & \frac{\partial P^{ECEF}}{\partial l_Z}_{(1)} \\ \vdots & \vdots & \vdots \\ \frac{\partial P^{ECEF}}{\partial \theta_X}_{(M)} & \cdots & \frac{\partial P^{ECEF}}{\partial l_Z}_{(M)} \end{bmatrix} \tag{4.17}$$

$$H_2 = \begin{bmatrix} \left(\left(P_0^{ECEF}\right)_{P2} - \left(P_0^{ECEF}\right)_{P1}\right)_{(1)} \\ \vdots \\ \left(\left(P_0^{ECEF}\right)_{P2} - \left(P_0^{ECEF}\right)_{P1}\right)_{(N)} \\ ---------- \\ \left(C^{ECEF} - P_0^{ECEF}\right)_{(1)} \\ \vdots \\ \left(C^{ECEF} - P_0^{ECEF}\right)_{(M)} \end{bmatrix} \tag{4.18}$$

Where both B_2 and H_2 consist of N pairs of common points and M pairs of control points. The solution to the set of linear equation then becomes Equation (4.19).

$$X_{BL2} = \left(B_2^T \cdot W \cdot B_2\right)^{-1} \cdot B_2^T \cdot W \cdot H_2 \tag{4.19}$$

4.3.2.3 Finding Boresight Angles and Lever Arms from Data from Two Sensors, Collected Concurrently (BLATS)

Since many MTL systems consist of two or more LiDAR sensors, the differences in geometry between one sensor observing a point and another sensor observing that same point can be used to compute the boresight and lever arm of each LiDAR sensor with respect to a common DG system. The base equation of the adjustment is shown in Equation (4.20).

$$P_{S1}^{ECEF}\left(\theta_{X1},\theta_{Y1},\theta_{Z1},l_{X1},l_{Y1},l_{Z1}\right) - P_{S2}^{ECEF}\left(\theta_{X2},\theta_{Y2},\theta_{Z2},l_{X2},l_{Y2},l_{Z2}\right) = 0 \tag{4.20}$$

Where the label (S1) denotes a point scanned from the first sensor on the calibration object, with $\left(\theta_{X1},\theta_{Y1},\theta_{Z1},l_{X1},l_{Y1},l_{Z1}\right)$ representing the boresight and lever arms between Sensor 1 and the DG system. The label (S2) denotes a point scanned from the second sensor on the same calibration object as S1, with $\left(\theta_{X2},\theta_{Y2},\theta_{Z2},l_{X2},l_{Y2},l_{Z2}\right)$ representing the boresight and lever arms between Sensor 2 and the DG system. Again the adjustment proceeds along the lines of Equations (4.11) to (4.13), except for Equations (4.12.2),

(4.12.3) and (4.12.4) which are replaced by Equations (4.21), (4.22) and (4.23), respectively.

$$B_3 = \begin{bmatrix} (A_1)_{S1} & (A_1)_{S2} \end{bmatrix} \tag{4.21}$$

$$X_{BL3} = \begin{bmatrix} (X_{BL1})_{S1} & (X_{BL1})_{S2} \end{bmatrix}^T \tag{4.22}$$

$$L_3 = \begin{bmatrix} \left(\left(P_0^{ECEF}\right)_{S2} - \left(P_0^{ECEF}\right)_{S1}\right)_{(1)} & \cdots & \left(\left(P_0^{ECEF}\right)_{S2} - \left(P_0^{ECEF}\right)_{S1}\right)_{(N)} \end{bmatrix}^T \tag{4.23}$$

Again, control points may optionally be introduced to the adjustment as constraints. To introduce these constraints on the adjustment, Equation (4.21) and (4.23) can be replaced with Equation (4.24) and Equation (4.25).

$$B_3 = \begin{bmatrix} (A_1)_{S1} & (A_1)_{S2} \\ (A_1)^C_{S1} & 0 \\ 0 & (A_1)^C_{S2} \end{bmatrix} \tag{4.24}$$

$$H_3 = \begin{bmatrix} L_3 \\ (L_1)^C_{S1} \\ (L_1)^C_{S2} \end{bmatrix} \tag{4.25}$$

Where $(A_1)_{S1}$, $(A_1)_{S2}$ and L_3 consist of N pairs of common points between the two sensors, $(A_1)^C_{S1}$ and $(L_1)^C_{S1}$ consist of M1 pairs of LiDAR to control comparisons from Sensor 1 data and $(A_1)^C_{S1}$ and $(L_1)^C_{S2}$ consist of M2 pairs of LiDAR to control comparisons from Sensor 2. The solution to the set of linear equation then becomes Equation (4.26).

$$X_{BL3} = \left(B_3^T \cdot W \cdot B_3\right)^{-1} \cdot B_3^T \cdot W \cdot H_3 \tag{4.26}$$

4.3.2.4 Correcting Point Selections

Points extracted from a LiDAR point cloud generally fall between MTL scan lines. This occurs as the object upon which the extracted point lies is usually not directly scanned by the MTL system. Assume there exists four three dimensional points $P_1 = [X_1, Y_1, Z_1], P_2 = [X_2, Y_2, Z_2], P_3 = [X_3, Y_3, Z_3]$ and $E = [X_E, Y_E, Z_E]$ that are not collinear. Let it further be assumed that the relative positions of these points will remain constant while the boresight and lever arm values are adjusted. These four points represent the point (E) extracted from the point cloud and the three closest LiDAR points (P_1, P_2, P_3) to this extracted point. Figure 4.12 shows how vector addition can be used to describe the location of the point extracted from the LiDAR point cloud with respect to the other three points.

In general, point E can be located with respect to the LiDAR points (P_1, P_2, P_3) through vector addition as shown in Equation (4.27).

$$E = P_1 + S_1 \cdot (P_2 - P_1) + S_2 \cdot (P_3 - P_2) + \frac{S_3}{|(P_2 - P_1) X (P_3 - P_2)|} \cdot (P_2 - P_1) X (P_3 - P_2) \quad (4.27)$$

Where S_1 is the scale factor needed to adjust the magnitude of the vector between P_1 and P_2, S_2 is the scale factor needed to adjust the magnitude of the vector between P_2 and P_3, and $S_3$is the scale factor used to adjust the normalized normal vector of the plane formed by P_1, $P_2$and P_3.

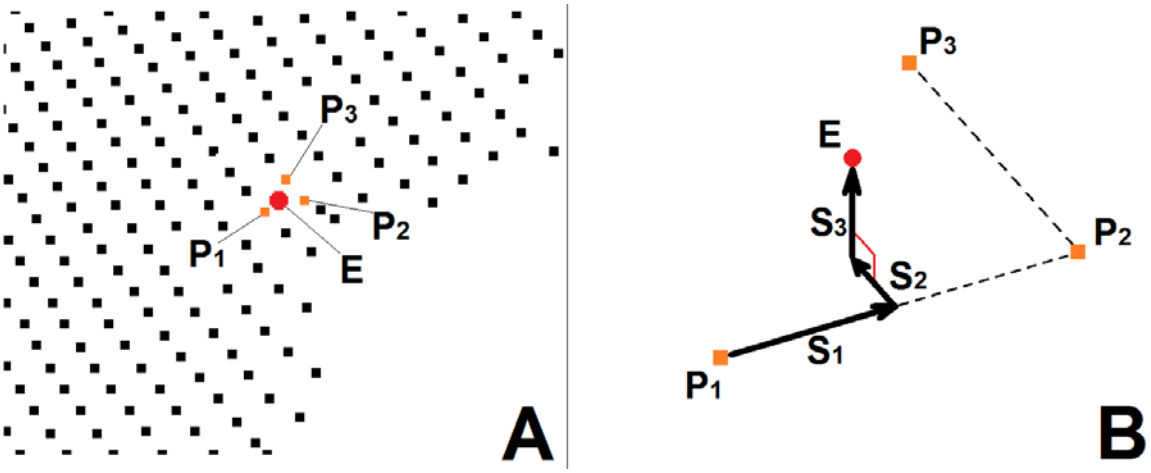

Figure 4.12: Vector addition used to describe the position of a point selected out of the LiDAR point cloud with respect to points measured in that point cloud. A) shows the extracted point within the context of the LiDAR point Cloud. B) shows the vector addition used to describe the extracted point with respect to the other measured points.

Making the further assumption that all points extracted from the point cloud will lie in the same plane as (P_1, P_2, P_3), Equation (4.18) can be simplified as shown in Equation (4.28).

$$E = P_1 + S_1 \cdot (P_2 - P_1) + S_2 \cdot (P_3 - P_2) \tag{4.28}$$

Since there are three equations and two unknowns, this system of linear equations can be written in matrix form and solved as a least square adjustment. The design matrix, observation vector and parameter vector for the adjustment are shown in Equations (4.29.1), (4.29.2) and (4.29.3), respectively.

$$A_4 = \begin{bmatrix} (X_2 - X_1) & (X_3 - X_2) \\ (Y_2 - Y_1) & (Y_3 - Y_2) \\ (Z_2 - Z_1) & (Z_3 - Z_2) \end{bmatrix} \tag{4.29.1}$$

$$L_4 = \begin{bmatrix} (X_E - X_1) & (Y_E - Y_1) & (Z_E - Z_1) \end{bmatrix}^T \tag{4.29.2}$$

$$X_S = \begin{bmatrix} S_1 & S_2 \end{bmatrix}^T \tag{4.29.3}$$

The solution to this set of linear Equations is given in Equation (4.30)

$$X_S = \left(A_4^T \cdot A_4\right)^{-1} \cdot A_4^T \cdot L_4 \tag{4.30}$$

Using (P_1, P_2, P_3) and combining Equation (4.28) with Equation (4.11) produces Equation (4.31).

$$
\begin{aligned}
E &= \left(P_0^{ECEF}\right)_1 + \frac{\partial P_1^{ECEF}}{\partial \theta_1}\cdot\Delta\theta_1 + \frac{\partial P_1^{ECEF}}{\partial \theta_2}\cdot\Delta\theta_2 + \frac{\partial P_1^{ECEF}}{\partial \theta_3}\cdot\Delta\theta_3 + \frac{\partial P_1^{ECEF}}{\partial l_X}\cdot\Delta l_X \\
&+ \frac{\partial P_1^{ECEF}}{\partial l_Y}\cdot\Delta l_Y + \frac{\partial P_1^{ECEF}}{\partial l_Z}\cdot\Delta l_Z + S_1\cdot\left(P_0^{ECEF}\right)_2 + S_1\cdot\frac{\partial P_2^{ECEF}}{\partial \theta_1}\cdot\Delta\theta_1 \\
&+ S_1\cdot\frac{\partial P_2^{ECEF}}{\partial \theta_2}\cdot\Delta\theta_2 + S_1\cdot\frac{\partial P_2^{ECEF}}{\partial \theta_3}\cdot\Delta\theta_3 + S_1\cdot\frac{\partial P_2^{ECEF}}{\partial l_X}\cdot\Delta l_X + S_1\cdot\frac{\partial P_2^{ECEF}}{\partial l_Y}\cdot\Delta l_Y \\
&+ S_1\cdot\frac{\partial P_2^{ECEF}}{\partial l_Z}\cdot\Delta l_Z - S_1\cdot\left(P_0^{ECEF}\right)_1 - S_1\cdot\frac{\partial P_1^{ECEF}}{\partial \theta_1}\cdot\Delta\theta_1 - S_1\cdot\frac{\partial P_1^{ECEF}}{\partial \theta_2}\cdot\Delta\theta_2 \\
&- S_1\cdot\frac{\partial P_1^{ECEF}}{\partial \theta_3}\cdot\Delta\theta_3 - S_1\cdot\frac{\partial P_1^{ECEF}}{\partial l_X}\cdot\Delta l_X - S_1\cdot\frac{\partial P_1^{ECEF}}{\partial l_Y}\cdot\Delta l_Y - S_1\cdot\frac{\partial P_1^{ECEF}}{\partial l_Z}\cdot\Delta l_Z \\
&+ S_2\cdot\left(P_0^{ECEF}\right)_3 + S_2\cdot\frac{\partial P_3^{ECEF}}{\partial \theta_1}\cdot\Delta\theta_1 + S_2\cdot\frac{\partial P_3^{ECEF}}{\partial \theta_2}\cdot\Delta\theta_2 + S_2\cdot\frac{\partial P_3^{ECEF}}{\partial \theta_3}\cdot\Delta\theta_3 \\
&+ S_2\cdot\frac{\partial P_3^{ECEF}}{\partial l_X}\cdot\Delta l_X + S_2\cdot\frac{\partial P_3^{ECEF}}{\partial l_Y}\cdot\Delta l_Y + S_2\cdot\frac{\partial P_3^{ECEF}}{\partial l_Z}\cdot\Delta l_Z - S_2\cdot\left(P_0^{ECEF}\right)_2 \\
&- S_2\cdot\frac{\partial P_2^{ECEF}}{\partial \theta_1}\cdot\Delta\theta_1 - S_2\cdot\frac{\partial P_2^{ECEF}}{\partial \theta_2}\cdot\Delta\theta_2 - S_2\cdot\frac{\partial P_2^{ECEF}}{\partial \theta_3}\cdot\Delta\theta_3 - S_2\cdot\frac{\partial P_2^{ECEF}}{\partial l_X}\cdot\Delta l_X \\
&- S_2\cdot\frac{\partial P_2^{ECEF}}{\partial l_Y}\cdot\Delta l_Y - S_2\cdot\frac{\partial P_2^{ECEF}}{\partial l_Z}\cdot\Delta l_Z
\end{aligned}
\tag{4.31}
$$

Collecting like terms produces Equation (4.32).

$$E = E_0 + E_{\theta 1} + E_{\theta 2} + E_{\theta 3} + E_{lx} + E_{ly} + E_{lz} \tag{4.32.1}$$

Where

$$E_0 = \left(P_0^{ECEF}\right)_1 + S_1\cdot\left(P_0^{ECEF}\right)_2 - S_1\cdot\left(P_0^{ECEF}\right)_1 + S_2\cdot\left(P_0^{ECEF}\right)_3 - S_2\cdot\left(P_0^{ECEF}\right)_2 \tag{4.32.2}$$

$$E_{\theta 1} = \left(\frac{\partial P_1^{ECEF}}{\partial \theta_1} + S_1\cdot\frac{\partial P_2^{ECEF}}{\partial \theta_1} - S_1\cdot\frac{\partial P_1^{ECEF}}{\partial \theta_1} + S_2\cdot\frac{\partial P_3^{ECEF}}{\partial \theta_1} - S_2\cdot\frac{\partial P_2^{ECEF}}{\partial \theta_1}\right)\cdot\Delta\theta_1 \tag{4.32.3}$$

$$E_{\theta 2} = \left(\frac{\partial P_1^{ECEF}}{\partial \theta_2} + S_1 \cdot \frac{\partial P_2^{ECEF}}{\partial \theta_2} - S_1 \cdot \frac{\partial P_1^{ECEF}}{\partial \theta_2} + S_2 \cdot \frac{\partial P_3^{ECEF}}{\partial \theta_2} - S_2 \cdot \frac{\partial P_2^{ECEF}}{\partial \theta_2} \right) \cdot \Delta\theta_2 \quad (4.32.4)$$

$$E_{\theta 3} = \left(\frac{\partial P_1^{ECEF}}{\partial \theta_3} + S_1 \cdot \frac{\partial P_2^{ECEF}}{\partial \theta_3} - S_1 \cdot \frac{\partial P_1^{ECEF}}{\partial \theta_3} + S_2 \cdot \frac{\partial P_3^{ECEF}}{\partial \theta_3} - S_2 \cdot \frac{\partial P_2^{ECEF}}{\partial \theta_3} \right) \cdot \Delta\theta_3 \quad (4.32.5)$$

$$E_{lx} = \left(\frac{\partial P_1^{ECEF}}{\partial l_x} + S_1 \cdot \frac{\partial P_2^{ECEF}}{\partial l_x} - S_1 \cdot \frac{\partial P_1^{ECEF}}{\partial l_x} + S_2 \cdot \frac{\partial P_3^{ECEF}}{\partial l_x} - S_2 \cdot \frac{\partial P_2^{ECEF}}{\partial l_x} \right) \cdot \Delta l_x \quad (4.32.6)$$

$$E_{ly} = \left(\frac{\partial P_1^{ECEF}}{\partial l_y} + S_1 \cdot \frac{\partial P_2^{ECEF}}{\partial l_y} - S_1 \cdot \frac{\partial P_1^{ECEF}}{\partial l_y} + S_2 \cdot \frac{\partial P_3^{ECEF}}{\partial l_y} - S_2 \cdot \frac{\partial P_2^{ECEF}}{\partial l_y} \right) \cdot \Delta l_y \quad (4.32.7)$$

$$E_{lz} = \left(\frac{\partial P_1^{ECEF}}{\partial l_z} + S_1 \cdot \frac{\partial P_2^{ECEF}}{\partial l_z} - S_1 \cdot \frac{\partial P_1^{ECEF}}{\partial l_z} + S_2 \cdot \frac{\partial P_3^{ECEF}}{\partial l_z} - S_2 \cdot \frac{\partial P_2^{ECEF}}{\partial l_z} \right) \cdot \Delta l_z \quad (4.32.8)$$

The design and observation matrices, which can be used to calculate the boresight and lever arms, can therefore be constructed as shown in Equations (4.33), (4.34) and (4.35).

$$A_5 = \begin{bmatrix} \left(E_{\theta 1}^1\right)_1 & \left(E_{\theta 2}^1\right)_1 & \cdots & \left(E_{lz}^1\right)_1 & \left(E_{\theta 1}^2\right)_1 & \left(E_{\theta 2}^2\right)_1 & \cdots & \left(E_{lz}^2\right)_1 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ \left(E_{\theta 1}^1\right)_n & \left(E_{\theta 2}^1\right)_n & \cdots & \left(E_{lz}^1\right)_n & \left(E_{\theta 1}^2\right)_n & \left(E_{\theta 2}^2\right)_2 & \cdots & \left(E_{lz}^2\right)_2 \\ - & - & - & - & - & - & - & - \\ \left(E_{\theta 1}^1\right)_1 & \left(E_{\theta 2}^1\right)_1 & \cdots & \left(E_{lz}^1\right)_1 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ \left(E_{\theta 1}^1\right)_{m1} & \left(E_{\theta 2}^1\right)_{m1} & \cdots & \left(E_{lz}^1\right)_{m1} & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \left(E_{\theta 1}^2\right)_1 & \left(E_{\theta 2}^2\right)_1 & \cdots & \left(E_{lz}^2\right)_1 \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 & \left(E_{\theta 1}^2\right)_{m2} & \left(E_{\theta 2}^2\right)_{m2} & \cdots & \left(E_{lz}^2\right)_{m2} \end{bmatrix} \quad (4.33)$$

$$L_5 = \begin{bmatrix} \left(E_0^1\right)_1 - \left(E_0^2\right)_1 \\ \vdots \\ \left(E_0^1\right)_n - \left(E_0^2\right)_n \\ (C)_1 - \left(E_0^1\right)_1 \\ \vdots \\ (C)_{m1} - \left(E_0^1\right)_{m1} \\ (C)_1 - \left(E_0^2\right)_1 \\ \vdots \\ (C)_{m2} - \left(E_0^2\right)_{m2} \end{bmatrix} \tag{4.34}$$

$$W_5 = \begin{bmatrix} \frac{1}{\sigma^2_{(E)_1}} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{\sigma^2_{(E)_{n+m1+m2}}} \end{bmatrix} \tag{4.35}$$

The solution to the set of linear equations then becomes Equation (4.36).

$$X_{BL5} = \left(A_5^T \cdot W_5 \cdot A_5\right)^{-1} \cdot A_5^T \cdot W_5 \cdot L_5 \tag{4.36}$$

4.3.2.5 Adjustment Statistics

For all the adjustments in the previous Sections, several types of statistics may be generated to aid the adjustment or evaluate its results. The most common statistic to calculate for a least squares adjustment are the residuals and the variance of unit weight as shown in Equation (4.37) and Equation (4.38).

$$V = A \cdot X_{BL} - L \tag{4.37}$$

$$\hat{\sigma}_0^2 = \frac{V^T V}{i - j} \tag{4.38}$$

Where V is the residual vector, $\hat{\sigma}_0^2$ is the variance of unit weight, i is the number of observations and j is the number of unknowns. Using the design (A) and weight (W)

matrices previously defined, some useful matrices for calculating a number of statistics can be created. If the number of unknowns is j and the number of observations is k, then the matrices shown in Equations (4.39) can be formed.

$$AA = \left(A^T \cdot W \cdot A\right)^{-1} = \begin{bmatrix} aa_{11} & aa_{12} & \cdots & aa_{1j} \\ aa_{21} & aa_{22} & \cdots & aa_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ aa_{j1} & aa_{j2} & \cdots & aa_{jj} \end{bmatrix} \tag{4.39.1}$$

$$Q = \left(W^{-1} - A \cdot AA \cdot A^T\right) = \begin{bmatrix} q_{11} & q_{12} & \cdots & q_{1k} \\ q_{21} & q_{22} & \cdots & q_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ q_{k1} & q_{k2} & \cdots & q_{kk} \end{bmatrix} \tag{4.39.2}$$

$$\alpha = W \cdot A \cdot AA = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \cdots & \alpha_{1j} \\ \alpha_{21} & \alpha_{22} & \cdots & \alpha_{2j} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{k1} & \alpha_{k2} & \cdots & \alpha_{kj} \end{bmatrix} \tag{4.39.3}$$

The estimates for the quality of the unknown parameters can be obtained from the diagonal elements of Equation (4.39.1) and the variance of unit weight according to Equation (4.40).

$$\hat{\sigma}_i = \hat{\sigma}_0 \cdot \sqrt{aa_{ii}} \quad \left(i = 1,2,\ldots, j\right) \tag{4.40}$$

Outlier detection can be implemented in the adjustment to eliminate any observations that may be biasing the adjustment. Using critical values taken from the Tau distribution, outliers may be identified as shown in Equation (4.41.1).

$$\frac{v_i}{\hat{\sigma}_0 \cdot \sqrt{q_{ii}}} \sim \tau_{(f)} \quad \left(i = 1,2,\ldots, k\right) \tag{4.41.1}$$

Where v_i are the individual a-posteriori residuals from vector V and f is the degrees of freedom of the adjustment. Since the Tau critical values are not easily accessible, the

student t critical value can be modified to conform to the Tau distribution as shown in Equation (4.41.2) [113].

$$\tau_{(f)} = \frac{\sqrt{f \cdot t_{(f-1)}}}{\sqrt{f - 1 + t^2_{(f-1)}}} \tag{4.41.2}$$

In addition to outlier detection, another form of reliability analysis that can be performed is to calculate the reliability indexes. The redundancy of the observations can be calculated as shown in Equation (4.42) [114].

$$R = Q \cdot W \tag{4.42}$$

The redundant indexes for each observation in the adjustment are the diagonal terms (r_{ii}) of the matrix R from Equation (4.42). The lower r_{ii} the more important the observation is to the solution of the boresight and lever arm parameters.

Another measure of the importance of the individual equations to the adjustment is the contribution index given by Equation (4.43.2) [114].

$$\alpha = W \cdot A_1 \cdot AA \tag{4.43.1}$$

$$c_i^j = \frac{\alpha_{ij}^2}{w_i} \tag{4.43.2}$$

In the case of the contribution index, the higher the value of each individual term, the more effect the equation has on parameter j.

Having computed the redundant indexes for each of the observations, variance component estimation can be used to re-estimate the standard deviations of the measured

quantities ($\sigma^2_{(C)_i}$) for each group of measurements. The equation for computing the new standard deviations is shown in Equation (4.44) [115].

$$\sigma^2_{(C)_i} = \frac{\sum_{l=1}^{k} v_l^2}{\sum_{l=1}^{k} r_{kk}} \quad (i = 1,2,\ldots) \tag{4.44}$$

4.3.3 Implementation and Testing

4.3.3.1 The Calibration Site

For testing purposes, MTL data was collected using the Lynx Mobile Mapper. This data was collected around a commercial office building with control points surveyed onto the face of the building and on the lines painted onto the asphalt surrounding it. The data used is shown in Figure 4.13.

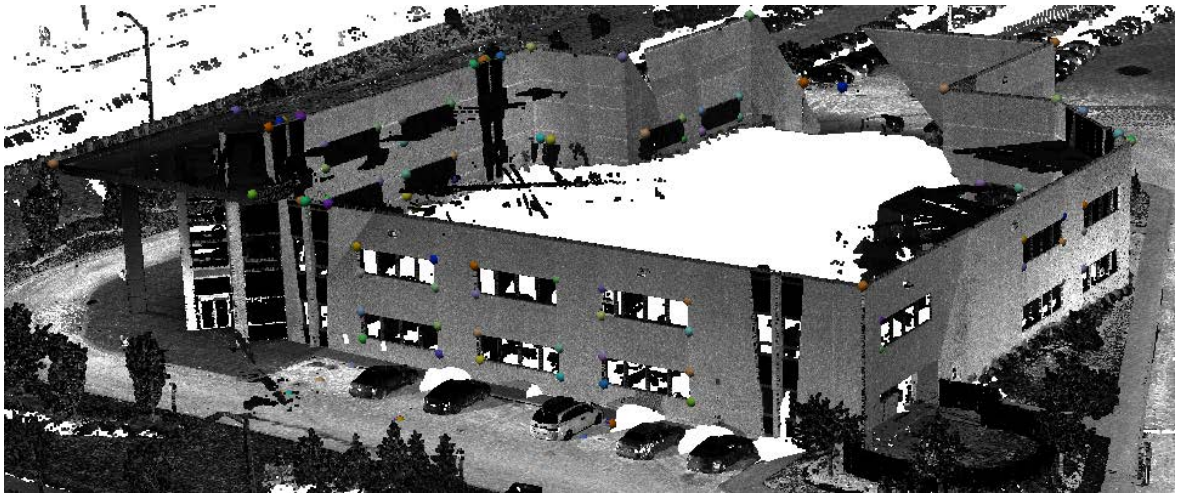

Figure 4.13: Lynx data collected around an office building in Vaughan Ontario Canada on February 5th, 2013. Control points are present on the building walls and on paint lines in the parking lot.

The LiDAR data comes with two sets of boresight and lever arm values. Table 4.14 shows the boresight and lever arm values that are indicated on the mechanical drawings for a generic Lynx system that uses an FMU P300 model IMU. Table 4.15 shows the

boresight and lever arms values provided by the manufacture of the LiDAR for this particular system. For processing purposes the point clouds were created using both sets of boresight and lever arm values. The control points on the face of the building were extracted from an initial processing of each of these point clouds.

Table 4.14: Generic boresight and lever arm values for a Lynx Mobile Mapper system.

Value	Roll (°)	Pitch(°)	Heading (°)	X (m)	Y (m)	Z (m)
Sensor 1	0	-30	35	0.035	-0.727	-0.169
Sensor 2	0	-30	-35	0.035	0.673	-0.169

Table 4.15: Manufacturer provided boresight and lever arm values for the Lynx Mobile Mapper system used in testing.

Value	Roll (°)	Pitch(°)	Heading (°)	X (m)	Y (m)	Z (m)
Sensor 1	0.07	-29.7	37.55	0.035	-0.717	-0.169
Sensor 2	-0.15	-29.85	-37.05	0.035	0.673	-0.169

The control coordinates were surveyed by traversing around the target site with a Leica TC1800 (1", 1mm + 2ppm) total station. ECEF coordinates were established by post processing static GPS observations of the control traverse and referencing them to the same base station as the Lynx Mobile Mapper data. The static GPS data was collected using a Leica 1200 GPS receiver (5mm + 0.5ppm (horizontal), 10mm + 0.5ppm (vertical)). The control coordinates established at the target site are given in Table 4.16, along with error estimates.

The control points listed in Table 4.16 are located on 3 walls of the commercial building and on the parking lines. The V200 and H200 series points are located on the Eastern walls of the building and the Eastern parking lot lines respectively. The V300 and H300 series points are located on the Northern wall of the building and the Northern parking lot lines. The V400 series points are located on the Western wall of the building.

Table 4.16: ECEF control coordinates established at the target site using Total Station.

Target ID	X [m]	Y [m]	Z [m]	σ_X [mm]	σ_Y [mm]	σ_Z [mm]	σ_{3D} [mm]
V201	838501.714	-4535000.057	4391303.864	3.0	2.7	1.9	4.4
V202	838501.356	-4534995.841	4391308.258	3.1	2.6	1.8	4.4
V203	838501.178	-4534993.735	4391310.453	3.2	2.6	1.8	4.5
V205	838501.482	-4534998.784	4391302.619	3.0	2.6	2.0	4.4
V206	838501.087	-4534994.581	4391307.034	7.4	2.5	5.7	9.7
V209	838501.182	-4534997.019	4391300.934	4.0	3.6	2.9	6.1
V211	838500.718	-4534990.707	4391307.480	10.0	5.0	8.2	13.8
V212	838500.259	-4534986.472	4391311.917	5.2	3.5	2.6	6.7
V213	838500.944	-4534995.744	4391299.689	4.1	3.5	3.0	6.1
V302	838473.501	-4534979.052	4391330.731	2.3	3.2	2.3	4.5
V303	838470.494	-4534979.482	4391330.865	2.2	3.4	2.5	4.8
V308	838473.272	-4534977.782	4391329.487	2.3	3.0	2.4	4.5
V309	838470.257	-4534978.211	4391329.619	2.2	3.3	2.6	4.7
V315	838469.933	-4534976.468	4391327.921	3.6	4.7	4.0	7.1
V316	838463.897	-4534977.288	4391328.228	3.5	5.7	4.6	8.1
V317	838460.854	-4534977.706	4391328.313	1.2	3.0	2.0	3.8
V321	838469.652	-4534975.258	4391326.678	1.4	1.9	1.8	2.9
V323	838460.632	-4534976.481	4391327.142	1.2	2.8	2.2	3.8
V324	838454.606	-4534977.266	4391327.407	5.3	7.4	6.2	11.1
V403	838445.664	-4534997.083	4391317.459	3.7	1.8	1.5	4.4
V405	838446.178	-4535003.395	4391310.890	5.2	2.9	2.4	6.4
V424	838445.541	-4534998.226	4391310.185	4.7	2.2	2.4	5.6
V425	838445.626	-4535000.375	4391307.945	5.3	2.6	2.9	6.6
H215	838508.279	-4534994.819	4391296.377	2.4	1.8	1.7	3.5
H216	838508.269	-4534994.740	4391296.463	2.4	1.8	1.7	3.5
H235	838507.915	-4534991.132	4391300.274	3.6	2.4	2.1	4.8
H236	838507.904	-4534991.060	4391300.357	3.6	2.4	2.1	4.8
H255	838507.582	-4534987.429	4391304.223	3.3	2.0	1.7	4.2
H256	838507.568	-4534987.349	4391304.307	3.4	2.0	1.7	4.2
H315	838477.939	-4534967.735	4391330.728	1.0	0.6	0.7	1.4
H316	838477.825	-4534967.749	4391330.731	1.0	0.6	0.7	1.4
H385	838458.643	-4534970.169	4391331.471	1.0	0.8	0.9	1.5
H386	838458.531	-4534970.185	4391331.479	1.0	0.8	0.9	1.5

In all cases natural building features and paint lines were used in lieu of placing predefined targets. It was decided to use existing features as these are more readily accessible to most surveyors and do not require the permission of property owners for

their placement. Using building features also means that historic or previously collected LiDAR data can be used, as pre-preparation of the target site is not required.

4.3.3.2 Utility Development

The mathematical model described above was implemented in Microsoft Visual C++ 6.0 and tested using the workflow depicted in Figure 4.14.

Point clouds were generated for Sensor 1 and Sensor 2 using both the boresight and lever arm values in Table 4.14 and the boresight and lever arm values in Table 4.15. Between the data from the LiDAR sensors 33 pairs of common points were extracted from the four point clouds. The control points were then apportioned such that each control point was assigned to a particular sensor's point cloud and so as to evenly distribute control across the entire length of each sensor's point cloud. It should be noted that far fewer control points in the parking lot could be extracted from Sensor 2 due to the angles at which the targets were seen and the brightness of the intensity return from these points on this sensor.

An initial adjustment was performed on the data and outlier detection based on the Tau test statistic given in Equation (4.41.1) was performed on both sets of point clouds. Identified points that exceeded the Tau critical value as defined by Equation (4.41.2) were removed from the adjustment. Of the 33 common points between the two sensors, 29 remained after outlier detection was performed. Of the 33 control points used, 21 remained after outlier detection was performed. The redundancy for the adjustment used after outlier detection was performed was 38 points (29 + 21 – 12).

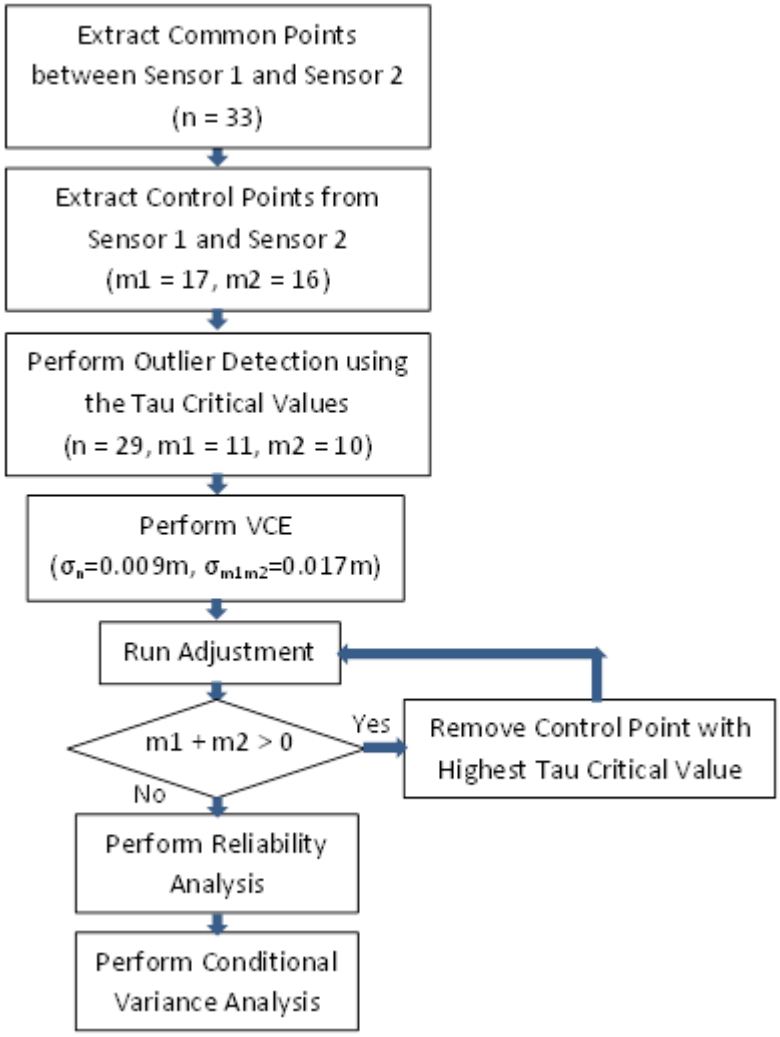

Figure 4.14: Block diagram of the utility testing workflow used to evaluate the effectiveness of the boresight and lever arm adjustment.

Variance component estimation was used on the cleaned data set to refine the estimated errors for the common picked points between the two sensors and the control points picked from each sensor. At first the common points were estimated to have an error envelope of 50mm based on the manufacturer's documentation [2]. The control points were assigned a common error envelope of 15mm to account for the error estimates in Table 4.16 and the error in the point selection. After several iterations were completed, the standard deviation of unit weight approached 1.0 and the common points had an

estimated error envelope of 9mm. The control points had a larger error envelope of 17mm.

Once the error estimates for the point comparisons were established, the data sets were run through the adjustment again and standard deviations were calculated for the adjusted boresight parameters according to Equation (4.40). Reliability analysis was performed and the redundancy vector of Equation (4.42) and a matrix of contribution indexes (Equation (4.43.2)) were computed. As an additional indication of the quality of the boresight and lever arms being calculated, conditional variance analysis was performed similar to that performed in [103].

4.3.3.3 Determining the Minimum Number of Control Points to Incorporate into the Adjustment

To get a good solution several control points, both on the asphalt and on the building, were used in the solution. It is necessary to figure out if control points are required to accurately determine the boresight and lever arms and if so, how much control and how does it need to be distributed. The adjustment was run several times using varying numbers of control points. Control points were eliminated from the solution based on their Tau critical values and in such a way as to maintain an approximately equal distribution over the collection area. Figure 4.15 and Figure 4.16 show the estimated errors for the boresight and lever arm parameters of Sensors 1 and 2 as the number of control points in the solution were reduced.

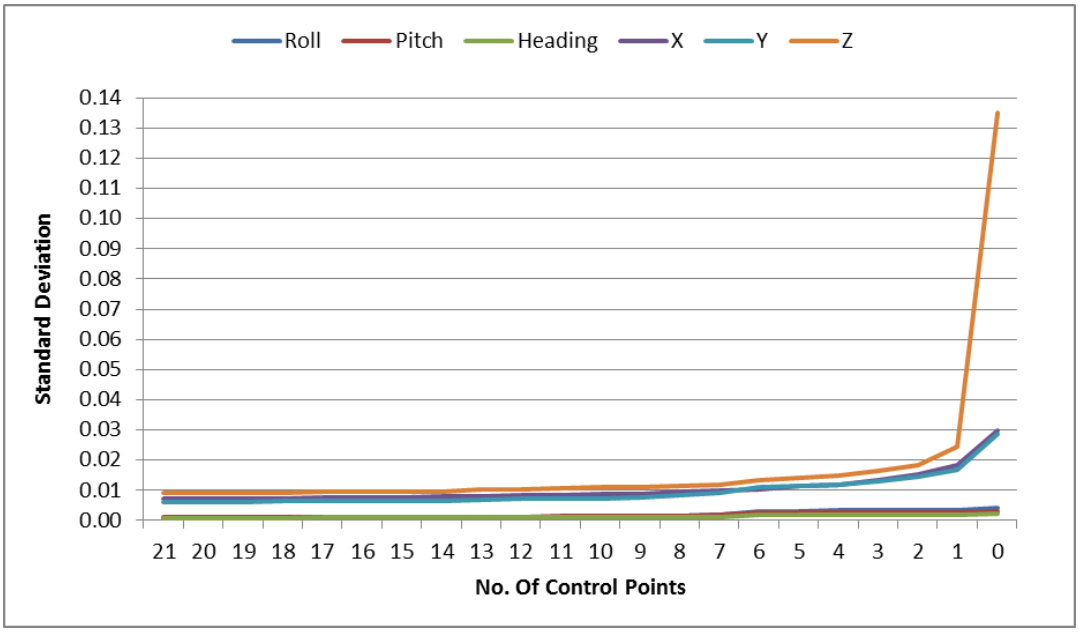

Figure 4.15: One sigma accuracy estimates for the boresight parameters of Sensor 1 when the control points were systematically removed from the adjustment. Standard deviation of roll, pitch and heading are given in degrees and X, Y, Z are given in metres.

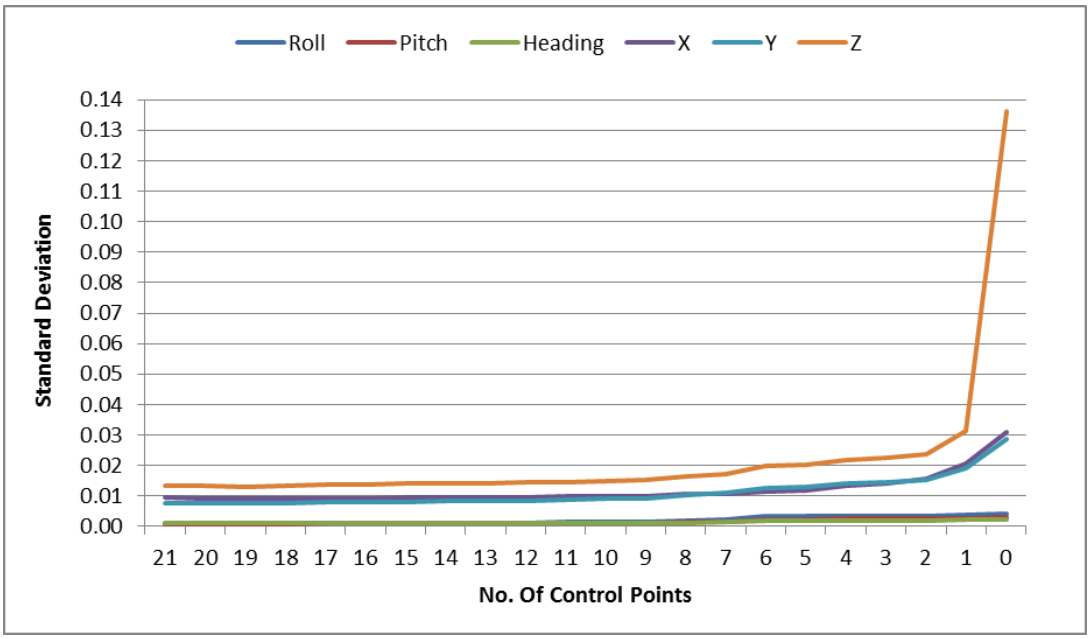

Figure 4.16: One sigma accuracy estimates for the boresight parameters of Sensor 2 when the control points were systematically removed from the adjustment. Standard deviation of roll, pitch and heading are given in degrees and X, Y, Z are given in metres.

In both Figure 4.15 and Figure 4.16, it can be noticed that a gradual increase in the estimated errors for all three lever arms occurs as the amount of control used in the solution is reduced. The boresight angles generally are not affected by the presence of

the control. The Z lever arm is the most effected by control as its complete absence causes the Z lever arm error to soar.

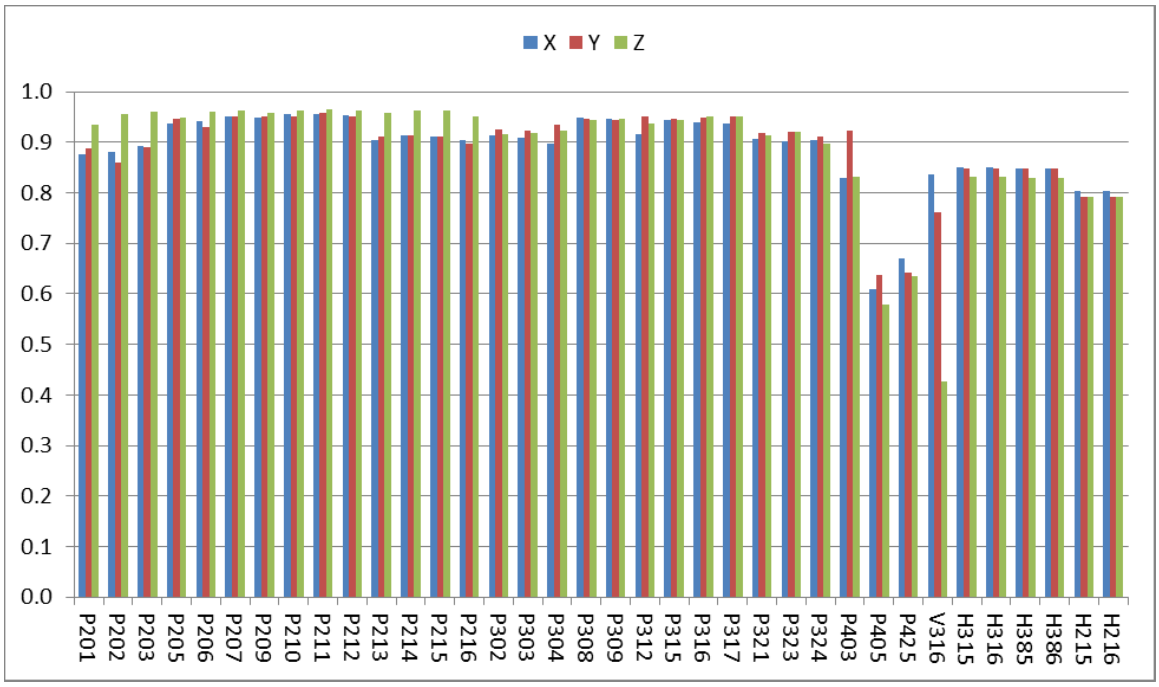

Figure 4.17: Redundancy indexes for the X, Y, Z components of the boresight and lever arm adjustment when 29 common LiDAR points and 7 control points are used in the adjustment.

The point where the most noticeable transition in Figure 4.15 and Figure 4.16 occurs is when 6 control points are used. Looking more closely at the redundancy values (Figure 4.17) just before this point, when 7 control points were used, it can be noticed that the P400 series values and the single control point left on the vertical surface are the least redundant observations in the adjustment. Looking closer at the single control point on a vertical surface, the redundancy of this observation on the Z coordinate is relatively quite small. Looking at the contribution of point V316 to the total error (Figure 4.18), it can be noticed that the Y and Z lever arms for both sensors are heavily dependent on this lone control point on a vertical surface.

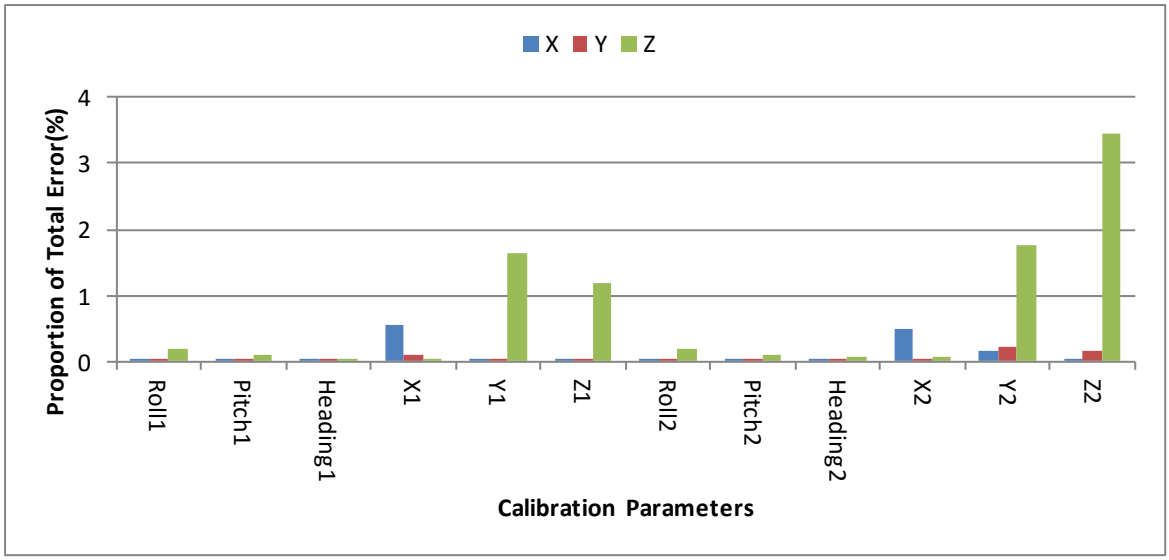

Figure 4.18: Contribution indexes for point V316 when 29 comparison points and 7 control points are used in the adjustment.

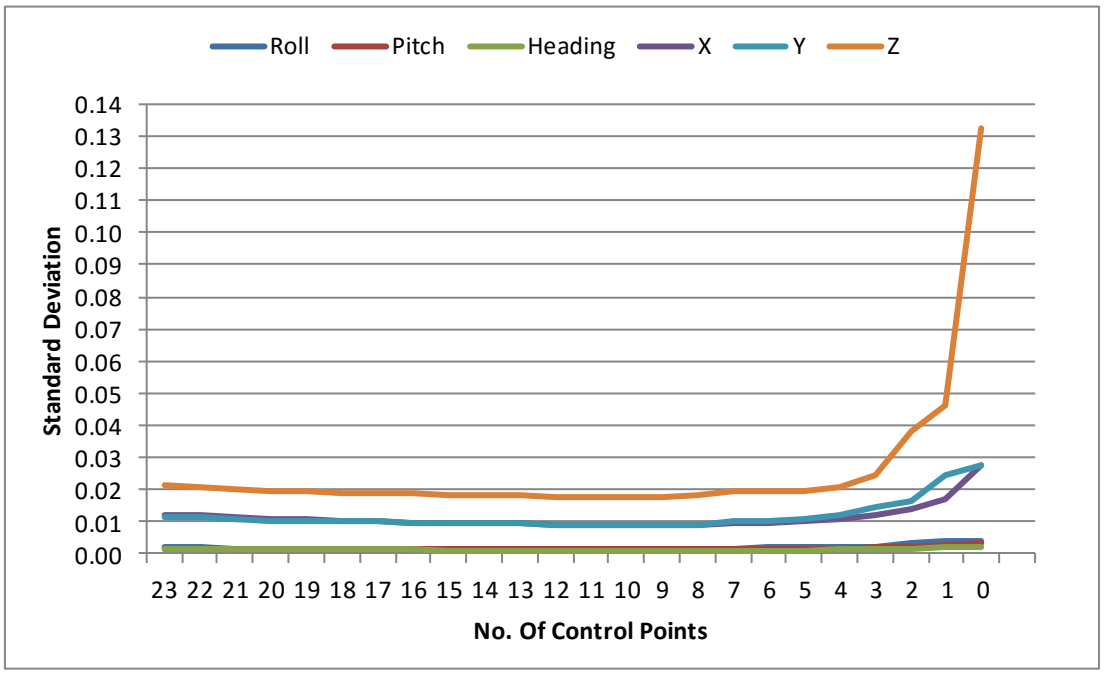

Figure 4.19: One sigma accuracy estimates for the boresight parameters of Sensor 1 when the control points were systematically removed from the adjustment. Standard deviation of roll, pitch and heading are given in degrees and X, Y, Z are given in metres.

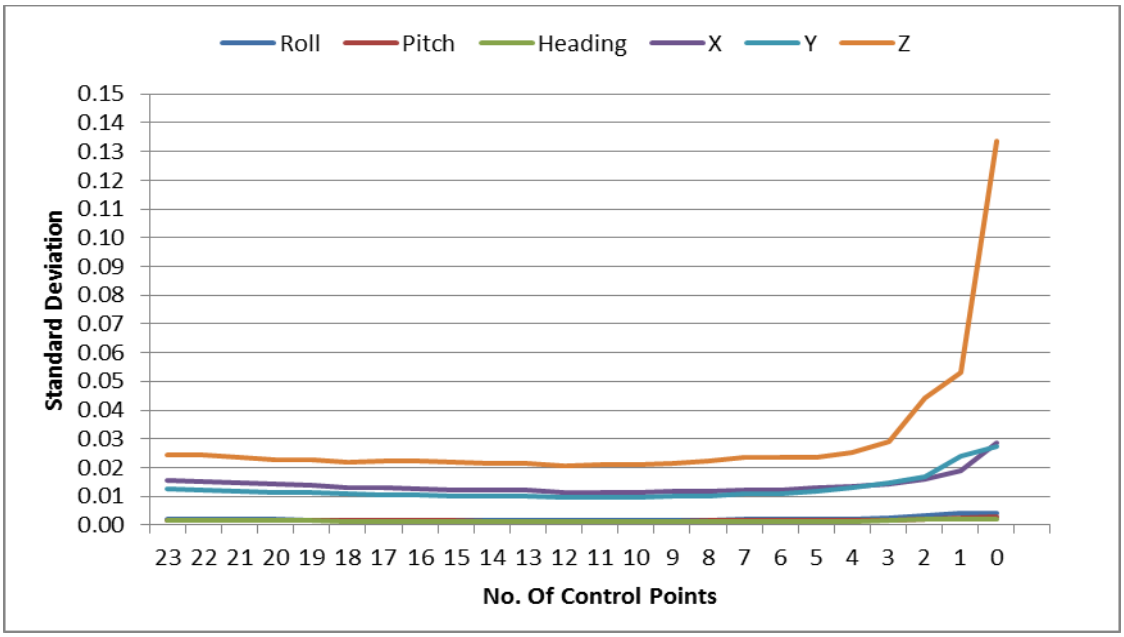

Figure 4.20: One sigma accuracy estimates for the boresight parameters of Sensor 2 when the control points were systematically removed from the adjustment. Standard deviation of roll, pitch and heading are given in degrees and X, Y, Z are given in metres.

Running the adjustment multiple times with varying levels of control points has shown that the lever arm adjustment can be quite stable with relatively few control points, However, the distribution of control so far has favored those control point collected on the horizontal road surface. This has led to a situation where the lone control point on a vertical surface has become overly significant in determining the adjusted values for two of the three lever arms for both sensors. Removing the control points located on horizontal surfaces and running the same test again with only those control points located on vertical surfaces, it can be noticed that the effect of these vertical surface control points on the end result is significant.

Figure 4.19 and Figure 4.20 show the estimated errors for the boresight and lever arm parameters of Sensors 1 and 2 using only those control points located on vertical

surfaces. The graphs in Figure 4.19 and Figure 4.20 show what happens when the number of vertical surface control points in the solution was systematically reduced.

In Figure 4.19 and Figure 4.20, the boresight angles were again not greatly affected by the presence or lack thereof of control points. The Z lever arm estimates are consistently quite a bit higher than previously determined for both sensors. The standard deviation for the lever arms remained fairly steady until the number of control point dropped below 6 and then they began to increase.

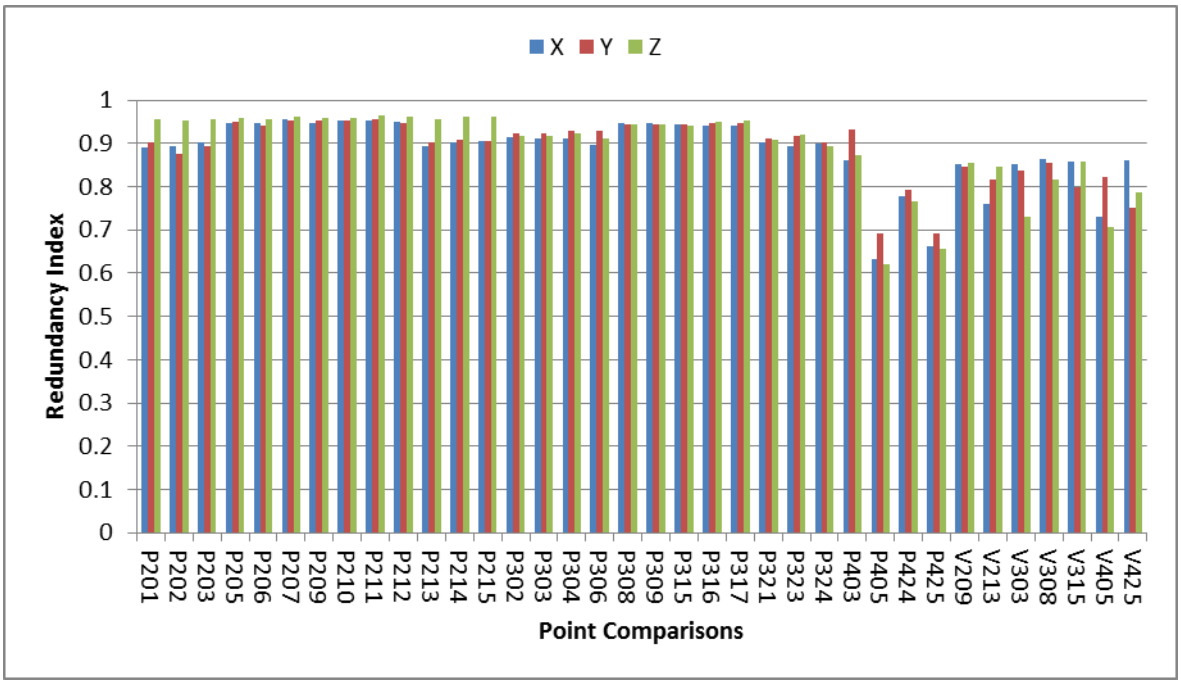

Figure 4.21: Redundancy indexes for the X, Y, Z components of the boresight and lever arm adjustment when 29 common LiDAR points and 7 control points are used in the adjustment.

Looking closer at the redundancy values (Figure 4.21) when 7 control points were used in the solution, it is clear that again the redundancy on the common points selected on the western wall is low. It is also clear that the distribution among the control is more even. Looking at the control point with the lowest redundancy values (V405), it is clear from

the contribution index for this point (Figure 4.22) that the Z lever arm is much more affected by the contributions of this control point than all the other parameters for both sensors.

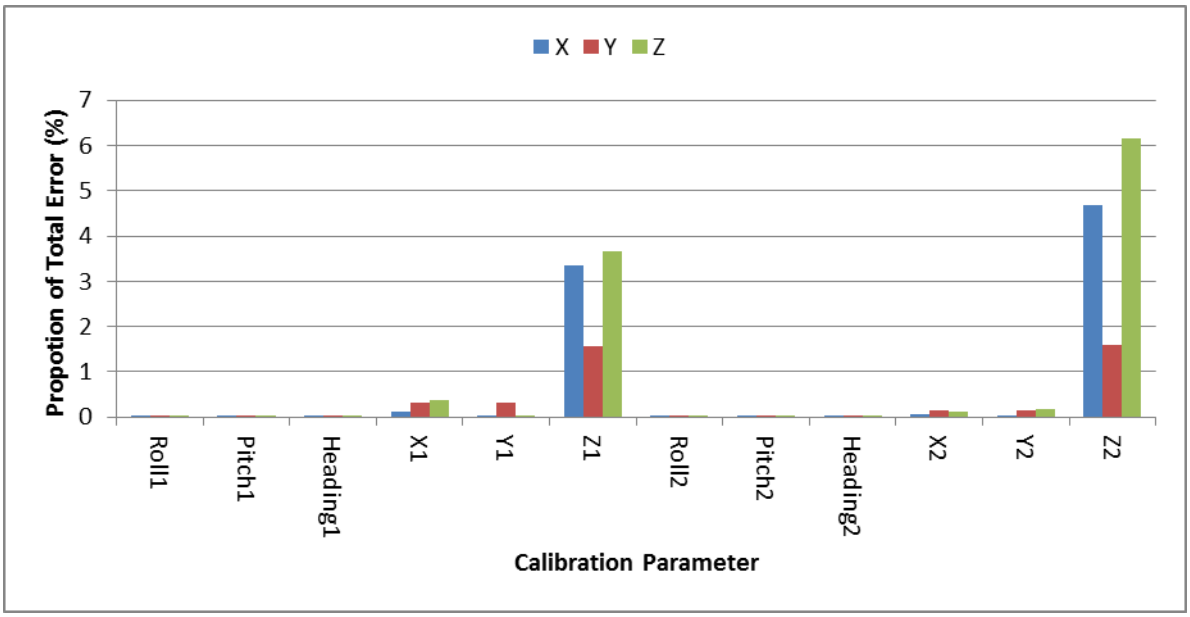

Figure 4.22: Contribution indexes for point V405 when 29 comparison points and 7 control points are used in the adjustment.

Comparing the contribution index in Figure 4.18 with the contribution index in Figure 4.22 it is clear that the Z lever arm is the calibration parameter that is most affected by the presence of the control points. It also seems that this calibration parameter is the one that most requires dispersion of the control points across both vertical and horizontal surfaces during the collect. The number of control points need not be excessive, in this case, with 29 common points between the sensors and 7 control points with good dispersion, the adjustment had sufficient redundancy (29 + 7 – 12 = 24) to get centimeter or sub centimeter accuracy on all three of the lever arm estimates in Sensor 1.

4.3.4 Results and Discussion

Since a manufacturer's calibration existed, determination of the accuracy of that calibration was performed. The common points extracted from the manufacturer's calibrated point cloud were run through a single iteration of the adjustment and the standard deviations of the boresight and lever arm parameters for both sensors were determined. These values are given in Table 4.17.

Table 4.17: Boresight and lever arm values with their standard deviations for data from a Lynx Mobile Mapper, as provided by the manufacturer.

Value		Roll (°)	Pitch(°)	Heading (°)	X (m)	Y (m)	Z (m)
Sensor 1	Calibration Values	0.07	-29.7	37.55	0.035	-0.717	-0.169
	Standard Deviation	0.0031	0.0025	0.0018	0.0103	0.0108	0.0127
Sensor 2	Calibration Values	-0.15	-29.85	-37.05	0.035	0.673	-0.169
	Standard Deviation	0.0038	0.0025	0.002	0.0121	0.0136	0.0202

To understand the amount of error incorporated within the point clouds created using the manufacturer's calibration values (Table 4.17), all surveyed control points (33 in total) were extracted from the LiDAR point clouds of both sensors of the MTL system and used as check points. Residuals were formed between the control points and the extracted check points, the results of which are displayed in Figure 4.23. The check point residuals in Figure 4.23 are separated into their horizontal and vertical components and are further separated into categories depending on whether they are located on a horizontal feature (parking lot line) or a vertical feature (building wall). The designator V and H are used on the check point residuals to indicate if the feature is a vertical (V) feature such as a building wall or a horizontal (H) feature such as a parking lot paint line.

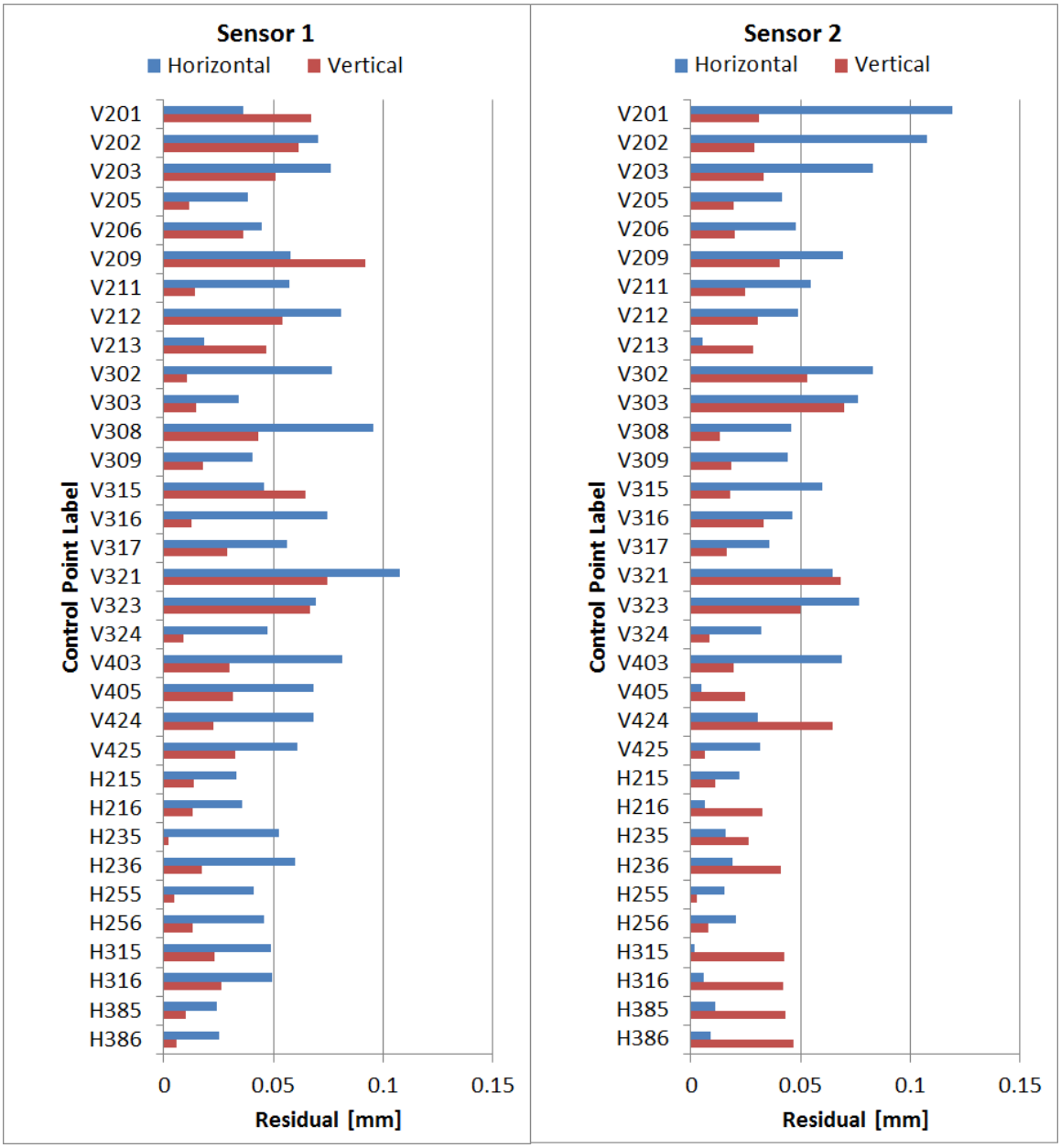

Figure 4.23: Check point residuals computed from point primitives extracted from the LiDAR point clouds of each sensor in the MTL system and compared to the control points associated with unique building features before boresight and lever arm correction.

Examining the check point residuals in Figure 4.23, it is found that the horizontal component of the residuals of Sensor 1 have a mean average of 55mm and a standard deviation of 21mm. The vertical components of the check point residuals for Sensor 1 have a mean of 31mm and a standard deviation of 24mm. Similarly, the horizontal

components of the Sensor 2 check point residuals in Figure 4.23 have a mean of 43mm and a standard deviation of 31mm, while, the vertical components have a mean of 31mm and a standard deviation of 18mm.

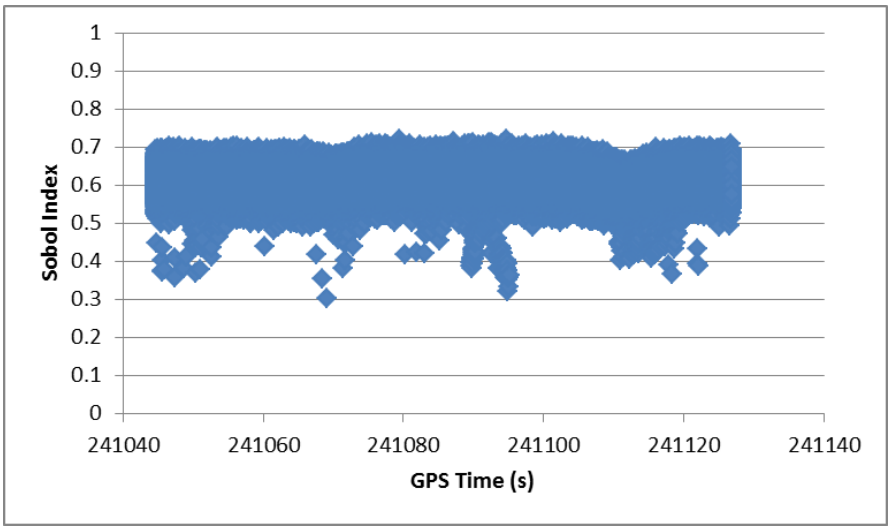

Figure 4.24: Sobol total effect indexes for the Z lever arm of all Sensor 1 points using manufacturer calibration values.

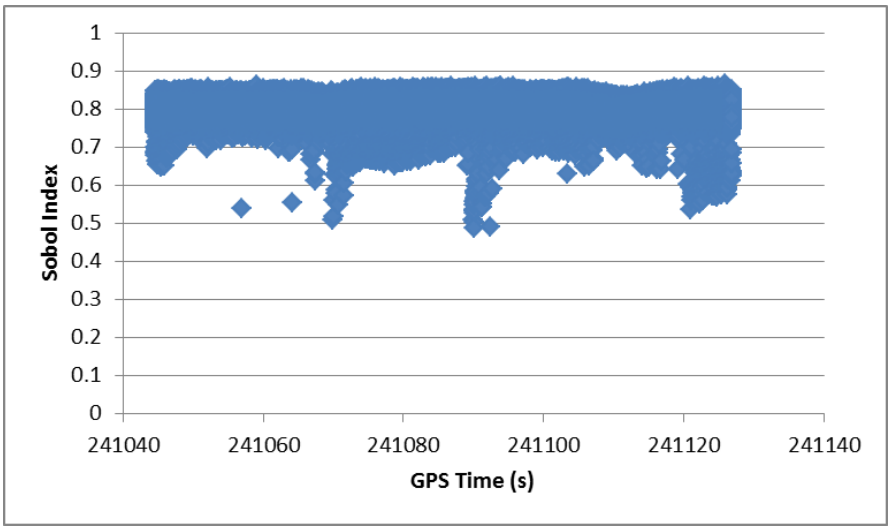

Figure 4.25: Sobol total effect indexes for the Z lever arm of all Sensor 2 points using manufacturer calibration values.

It is also possible to use conditional variance analysis to view these errors by how much they contribute to the error in the final point cloud. Running conditional variance analysis, it is found that the largest single contributor to the error for all the points in the

point cloud is the Z lever arm. Depending on the range from the LiDAR to the target the error in the Z lever arm accounts for between 30% (long range) and 75% (short range) of the total error in the point cloud from Sensor 1 (Figure 4.24). Similarly, the Z lever arm accounts for between 65% (long range) and 88% (short range) of the total error in the point cloud from Sensor 2 (Figure 4.25).

Selecting points V316, H215, H216, H315, H316, H385 and H386 as the control points which will be used in the adjustment, the algorithm was run using an additional 29 common points (24 degrees of freedom). Running the adjustment with these 36 points and using the initial estimates taken from the mechanical drawing calibration values listed in Table 4.15 produced a new set of calibration values and their estimated errors. The output calibration values are given in Table 4.18.

Table 4.18: Calculated boresight and lever arm values with standard deviation error estimates for data from a Lynx Mobile Mapper point cloud.

Value		Roll (°)	Pitch(°)	Heading (°)	X (m)	Y (m)	Z (m)
Sensor 1	Calibration Values	0.1217	-29.6692	37.4554	0.019	-0.749	-0.181
	Standard Deviation	0.0012	0.0014	0.0010	0.0065	0.0055	0.0068
Sensor 2	Calibration Values	-0.1136	-29.4622	-37.3087	0.0358	0.612	-0.175
	Standard Deviation	0.0013	0.0011	0.0010	0.0072	0.0076	0.0111

To confirm that the adjustment has improved the MTL system calibration, all surveyed control points (33 in total), including the 7 control points used in the calibration process, were extracted from the LiDAR point clouds of both sensors after the calibration was performed. Residuals were formed between the control points and the extracted check points from point clouds created for both sensors using the estimated calibration values of

Table 4.18 (Figure 4.26). The 7 control points (V316, H215, H216, H315, H316, H385 and H386) used as constraints were included in Figure 4.26 to quantify the final size of the minimized residuals and so they may be contrasted with the pre-adjustment residuals (Figure 4.23). The number of independent check points depicted in Figure 4.26 is therefore reduced to the other 26 check point residuals.

The check point residuals in Figure 4.26 are separated into their horizontal and vertical components and are further separated into categories depending on whether they are located on a horizontal feature (paint line) or a vertical feature (building wall). The designator V and H are used on residuals to indicate if the feature is a vertical (V) feature such as a building wall or a horizontal (H) feature such as a parking lot paint line.

Examining the residuals in Figure 4.26, it is found that the of the horizontal component of the residuals of Sensor 1 have a mean average of 25mm and a standard deviation of 14mm. The vertical components of the residuals for Sensor 1 have a mean of 13mm and a standard deviation of 11mm. Similarly, the horizontal components of the Sensor 2 residuals in Figure 4.26 have a mean of 22mm and a standard deviation of 17mm, while, the vertical components have a mean of 18mm and a standard deviation of 11mm.

Running conditional variance analysis again results in the Z lever arm being the most significant source of error in the point cloud. Depending on the range from the LiDAR sensor, the Z lever arm accounts for between 20% (long range) and 60% (short range) of the total error in point cloud as shown in Figure 4.27 and Figure 4.28.

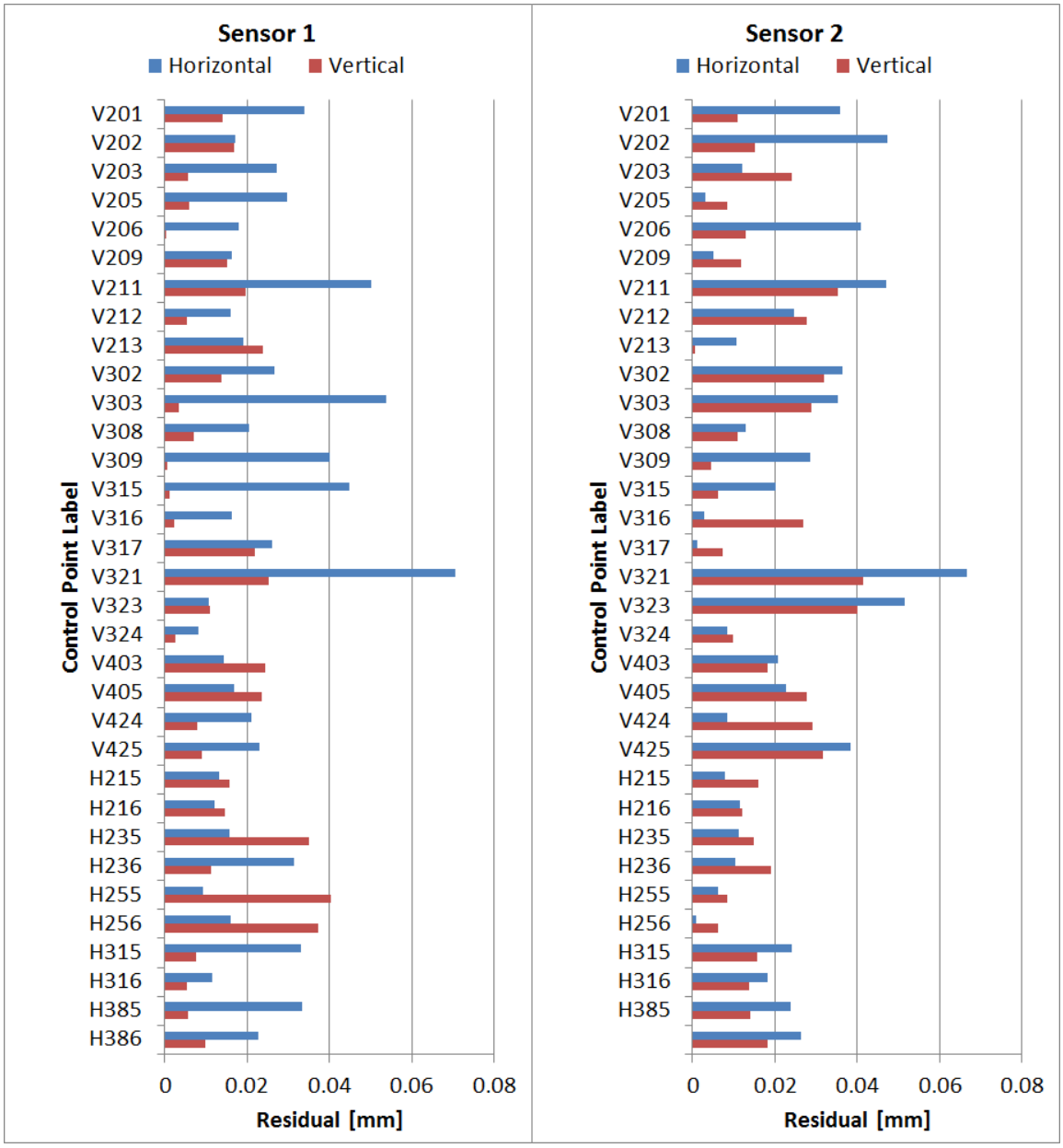

Figure 4.26: Check point residuals computed from point primitives extracted from the LiDAR point clouds of each sensor in the MTL system and compared to the control points associated with unique building features after boresight and lever arm correction. Note that points V316, H215, H216, H315, H316, H385 and H386 were used in the adjustment process leaving the other 26 points as independent check points.

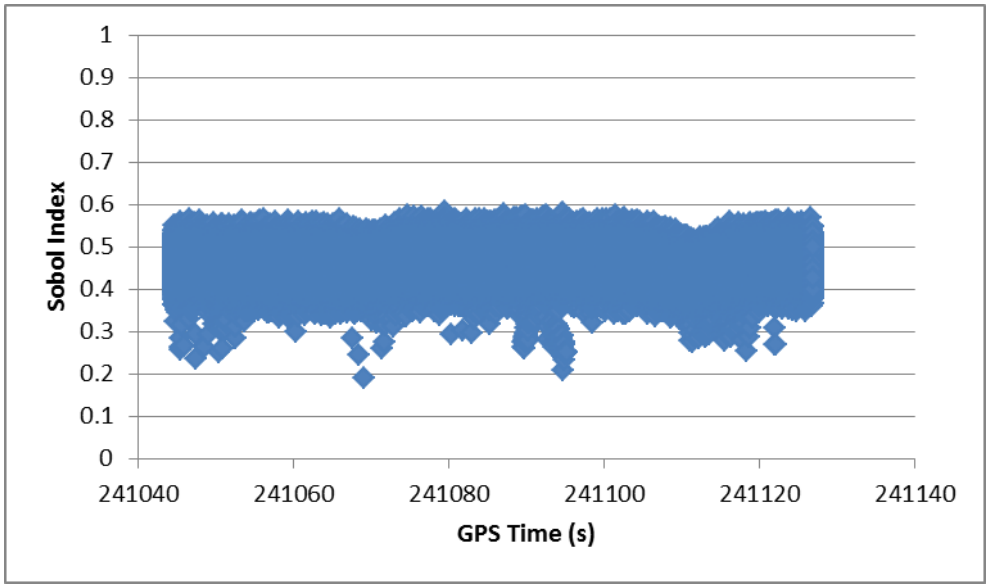

Figure 4.27: Sobol total effect indexes for the Z lever arm of Sensor 1 using calibration values determined from the routine.

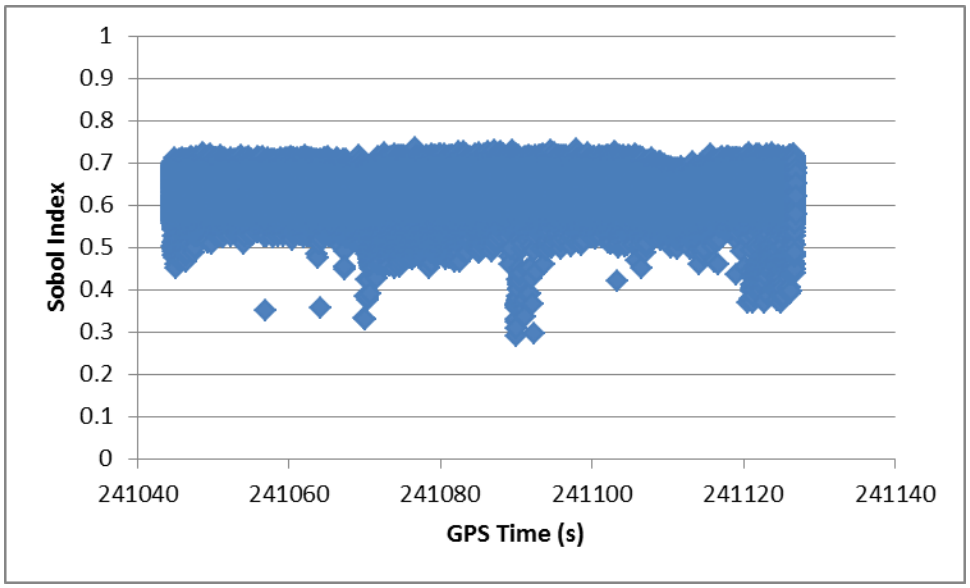

Figure 4.28: Sobol total effect indexes for the Z lever arm of Sensor 2 using calibration values determined from the routine.

The standard deviations of the boresight and lever arm values that were produced by the routine (Table 4.18) do indicate a significant improvement over those received with the system (Table 4.17). Residual comparison between control points and the point cloud produced by the manufacture's calibration and the algorithm calibration show a general accuracy improvement in both Sensors 1 and 2 of approximately 18mm. Conditional

variance analysis confirms that a significant reduction in the percentage of error caused by the most significant parameter (Z lever arm) also occurred. The lever arms between Table 4.17 and Table 4.18 indicate that the two sensors shifted 3cm toward the driver side of the vehicle from the IMU position. Sensor 1 was raised by about 12mm and Sensor 2 was raised by about 6mm. These changes in the estimated position of the sensors within the MTL system may be a result of variation in manufacturing, variation in the calibration center of the sensor or mounting tolerances. It could very well be that all these factors are contributing to this result.

4.4 Summary

In this chapter, methods for improving the accuracy of measurements from a MTL system's laser range finder and for calibrating the LiDAR sensor offset values were discussed. A new method for zero error calibration of a 360º LiDAR sensor was presented. The zero error calibration method presented in this chapter removed the requirement of precisely fixing the location of the sensor with respect to some calibration target and instead, made use of two fixed targets, allowing the sensor to freely move between them. Field testing of the zero error has shown that this calibration produced results in real MTL data that conformed to a control surface with deviations as small as 7mm. This result was better than that obtained with the zero error calibration supplied by the MTL manufacturer.

In addition, a new method for producing temperature correction tables for a MTL laser's range finder was proposed and tested. This method uses a α-β-γ smoother as a low pass filter to remove white noise and produce an accurate reproducible curve of temperature

corrections. Testing showed that the range reported by multiple MTL laser range finders drifted by 15mm~25mm as the ambient temperature varied between -20ºC and +50ºC. This temperature range was chosen as it falls within the normal operating conditions of MTL systems in the field. Applying the correction tables to test data collected in a fixed position to a stationary target showed that the estimated errors in the laser range measurements were reduce in many cases from centimeters to millimeters.

Finally, a new rigorous boresight angle and lever arm calibration method for dual sensor MTL systems was proposed and tested. Employing a new concept for extracted point correction based on vector geometry and using enough extracted information from the two sensor's point clouds (29 common points, 7 control points, 24 degrees of freedom), to produce boresight and lever arm values for the MTL with error estimates similar to the ideal error estimates discussed in Table 3.1 in Section 3.5. Comparison between the pre and post calibration MTL point clouds showed a general accuracy improvement in both Sensors 1 and 2 of approximately 18mm.

Having improved the accuracy of the laser range finder through zero error and temperature correction and having improved the accuracy of the boresight angles and lever arm values for the LiDAR sensors, two of the most significant sources of error identified by the conditional variance analysis conducted in Chapter 3 have been controlled. There are, however, other causes of errors as outlined in Chapter 2 which may cause individual LiDAR observations to become biased in some manner. To ensure that the points remaining in the MTL data are free of bias caused by such factors as solar

radiation, particulate matter in the air, specular reflection or mixed pixels, outlier detection should be perform within the processed point cloud. To this end, the next chapter discusses new outlier detection methods and details the testing performed to verify their effectiveness.

5. Outlier Detection and Removal from MTL Data

5.1 Introduction

As stated in [102], the main reason for isolating outliers is associated with data quality assurance. In LiDAR data, the removal of outliers ensures more uniformity in the positional accuracy of any points likely to be used for data extraction. Removing outliers is also likely to have a positive impact on data analysis and data mining [102]. It cleans up LiDAR point clouds, making data features easier to visualize and manipulate.

In this chapter, three new methods of outlier detection in MTL data will be compared. As was previously discussed in Section 2.3, these methods will use two separate concepts, one in the spatial domain and one in the temporal domain. The first method combines the generation of a 10 parameter quadratic polynomial surface and the general principals of distance-based outlier detection to spatially compare points to their neighbourhood. The second method also uses a 10 parameter quadratic polynomial surface and the general principals of distribution based outlier detection to spatially detect outlying data points. The third method makes use of the precise timings available from either MTL or STL in a α-β-γ Kalman smoother to predict point positions based on their neighbourhood. Using the principals of distance based outlier detection; the predicted point in the third method can be compared to the actual LiDAR point and the actual point can be removed if the deviation from the predicted position is too great.

The three methods of performing outlier detection in this chapter each have their own advantages and disadvantages. The 10 parameter quadratic polynomial fitting produces a surface model from LiDAR points which is accommodating to the varying changes in the shape of the object being modeled. It can adapt to and overcome holes in the LiDAR data caused by objects of occlusion and adsorptive targets in the collection area. That being said, the 10 parameter quadratic polynomial is not expected to be forgiving of drastic changes in the LiDAR data, such as are encountered when terrain turns into a building, or the terrain drops away in the case of a cliff face. The temporal algorithm, based on the α-β-γ Kalman smoother, on the other hand, provides a means for dealing with sharp changes in the surfaces scanned by the LiDAR. Its property as a low pass filter (dampening white noise), combined with its ability to deal with rapid changes in surface direction, makes it an ideal candidate for use as an outlier filter. The α-β-γ Kalman smoother has the additional benefit of being able to work on both continuous and discrete data sets. How the α-β-γ Kalman smoother deals with gaps and holes in the LiDAR data is a question examined in this chapter.

Much of the content in this chapter has been published as “Comprehensive Utilization of Temporal and Spatial Domain Outlier Detection Methods for Mobile Terrestrial LiDAR Data” [116].

5.2 Mathematical Models

5.2.1 The α-β-γ Kalman Smoother

The mathematical basis for the α-β-γ Kalman smoother was presented in [117] and based on the concepts presented in [118]. Given a set of coordinate components (R), the standard deviations for each R (σ_R) and the precise timings associated with each LiDAR point, provided by the DG system, the Lagrange multipliers from Equation (5.1) can be calculated.

$$\begin{pmatrix} \lambda_1(k) & \mu_1(k) & \eta_1(k) \\ \lambda_2(k) & \mu_2(k) & \eta_2(k) \\ \lambda_3(k) & \mu_3(k) & \eta_3(k) \end{pmatrix} = \begin{pmatrix} \sum_{i=-n_1}^{n_2} \frac{1}{\sigma_R^2(k+i)} & \sum_{i=-n_1}^{n_2} \frac{t_{k+i}-t_k}{\sigma_R^2(k+i)} & \sum_{i=-n_1}^{n_2} \frac{(t_{k+i}-t_k)^2}{\sigma_R^2(k+i)} \\ \sum_{i=-n_1}^{n_2} \frac{t_{k+i}-t_k}{\sigma_R^2(k+i)} & \sum_{i=-n_1}^{n_2} \frac{(t_{k+i}-t_k)^2}{\sigma_R^2(k+i)} & \sum_{i=-n_1}^{n_2} \frac{(t_{k+i}-t_k)^3}{\sigma_R^2(k+i)} \\ \sum_{i=-n_1}^{n_2} \frac{(t_{k+i}-t_k)^2}{\sigma_R^2(k+i)} & \sum_{i=-n_1}^{n_2} \frac{(t_{k+i}-t_k)^3}{\sigma_R^2(k+i)} & \sum_{i=-n_1}^{n_2} \frac{(t_{k+i}-t_k)^4}{\sigma_R^2(k+i)} \end{pmatrix}^{-1} \quad (5.1)$$

Where i = $-n_1,\ldots,-2, -1, 0, 1, 2, \ldots, n_2$. Equation (5.1) requires a window size $[-n_1, n_2]$ to be chosen. This window will specify the number of neighboring points that will be used by the routine to estimate the expected range at discrete time period (k). Once the Lagrange multipliers have been calculated, the coefficients for the minimum variance unbiased estimate of the state space variables can be found as shown in Equations (5.2.1), (5.2.2) and (5.2.3), respectively.

$$a_i(k) = \frac{1}{\sigma_R^2(k+i)}\left[\lambda_1(k) + \lambda_2(k)\cdot(t_{k+i}-t_k) + \lambda_3(k)\cdot(t_{k+i}-t_k)^2\right] \quad (5.2.1)$$

$$b_i(k) = \frac{1}{\sigma_R^2(k+i)}\left[\mu_1(k) + \mu_2(k)\cdot(t_{k+i}-t_k) + \mu_3(k)\cdot(t_{k+i}-t_k)^2\right] \quad (5.2.2)$$

$$c_i(k) = \frac{1}{\sigma_R^2(k+i)}\left[\eta_1(k) + \eta_2(k)\cdot(t_{k+i}-t_k) + \eta_3(k)\cdot(t_{k+i}-t_k)^2\right] \quad (5.2.3)$$

Again, i = $-n_1,\ldots,-2, -1, 0, 1, 2, \ldots, n_2$ for Equations (5.2.1), (5.2.2) and (5.2.3). Using the results from the calculations in (5.2), the state estimates for the current coordinate can be computed as shown in Equation (5.3):

$$\hat{\varphi}(k) = \sum_{i=-n_1}^{n_2} a_i(k) \cdot R(k+i) \tag{5.3.1}$$

$$\hat{\dot{\varphi}}(k) = \sum_{i=-n_1}^{n_2} b_i(k) \cdot R(k+i) \tag{5.3.2}$$

$$\hat{\ddot{\varphi}}(k) = \sum_{i=-n_1}^{n_2} c_i(k) \cdot R(k+i) \tag{5.3.3}$$

The variances of these estimates can be computed as in Equation (5.4):

$$\sigma_{\varphi}^2(k) = \sum_{i=-n_1}^{n_2} \frac{\left[\lambda_1(k) + \lambda_2(k) \cdot (t_{k+i} - t_k) + \lambda_3(k) \cdot (t_{k+i} - t_k)^2\right]^2}{\sigma_R^2(k+i)} = \sum_{i=-n_1}^{n_2} a_i^2(k) \cdot \sigma_R^2(k+i) \tag{5.4.1}$$

$$\sigma_{\dot{\varphi}}^2(k) = \sum_{i=-n_1}^{n_2} \frac{\left[\mu_1(k) + \mu_2(k) \cdot (t_{k+i} - t_k) + \mu_3(k) \cdot (t_{k+i} - t_k)^2\right]^2}{\sigma_R^2(k+i)} = \sum_{i=-n_1}^{n_2} b_i^2(k) \cdot \sigma_R^2(k+i) \tag{5.4.2}$$

$$\sigma_{\ddot{\varphi}}^2(k) = \sum_{i=-n_1}^{n_2} \frac{\left[\eta_1(k) + \eta_2(k) \cdot (t_{k+i} - t_k) + \eta_3(k) \cdot (t_{k+i} - t_k)^2\right]^2}{\sigma_R^2(k+i)} = \sum_{i=-n_1}^{n_2} c_i^2(k) \cdot \sigma_R^2(k+i) \tag{5.4.3}$$

The estimate for the next point in the discrete time series can be estimated as in Equation (5.5):

$$\hat{\varphi}(k+1) = \hat{\varphi}(k) + \hat{\dot{\varphi}}(k) \cdot (t_{k+1} - t_k) + \frac{1}{2} \hat{\ddot{\varphi}}(k) \cdot (t_{k+1} - t_k)^2 \tag{5.5.1}$$

$$\hat{\dot{\varphi}}(k+1) = \hat{\dot{\varphi}}(k) + \hat{\ddot{\varphi}}(k) \cdot (t_{k+1} - t_k) \tag{5.5.2}$$

$$\hat{\ddot{\varphi}}(k+1) = \hat{\ddot{\varphi}}(k) \tag{5.5.3}$$

It is also possible to use this model to perform outlier detection. Knowing that the observation equation for most α-β-γ smoothers is given by Equation (5.6):

$$R(k+1) = \varphi(k+1) + \Delta(k+1) \tag{5.6}$$

Where, R is the measured range, φ is the state estimate variable and Δ is the measurement noise. By rearranging Equation (5.3.1), a model for the predicted estimate can be produced as is shown in Equation (5.7).

$$R_p(k) = \frac{1}{1 - a_0(k)} \cdot \left[\sum_{i=-n_1}^{-1} a_i(k) \cdot R(k+i) + \sum_{i=1}^{n_2} a_i(k) \cdot R(k+i) \right] \tag{5.7}$$

Finding the difference $\delta R(k) = R(k) - R_p(k)$ at discrete time period (k), the variance of this difference can be computed as in Equation (5.8):

$$\sigma_{dR}^2(k) = \sigma_R^2 + \frac{1}{\left(1 - a_0(k)\right)^2} \cdot \left[\sum_{i=-n_1}^{-1} a_i^2(k) \cdot \sigma_R^2(k+i) + \sum_{i=1}^{n_2} a_i^2(k) \cdot \sigma_R^2(k+i) \right] \tag{5.8}$$

Using the variance calculated in Equation (5.8), the standard difference can be computed and this value can be used as a statistic test to detect outliers. If the inequality given in (5.9) is true, then the range at (k) can be rejected as an outlier.

$$\frac{R(k) - R_p(k)}{\sigma_{dR}} \geq \varepsilon \tag{5.9}$$

Where ε is some threshold value determined by the user.

5.2.2 Quadratic Polynomial Surface Fitting (PSF)

The mathematical model presented here was first presented in [119]. The Generalized model for a quadric polynomial surface is given in Equation (5.10).

$$\begin{aligned} & f(a_1, \cdots, a_{10}, x, y, z) \\ & = a_1 \cdot x^2 + a_2 \cdot y^2 + a_3 \cdot z^2 + a_4 \cdot x \cdot y + a_5 \cdot x \cdot z + a_6 \cdot y \cdot z + a_7 \cdot x + a_8 \cdot y + a_9 \cdot z + a_{10} = 0 \end{aligned} \tag{5.10}$$

Fitting points to the polynomial surface in Equation (5.10) could result in multiple parallel surfaces. Due to the mathematical ambiguity in the surface determination introduced by the intercept parameter a_{10}, It has been proposed that it is necessary to constrain the 10 parameter adjustment as shown in Equation (5.11) [120].

$$C = a_1^2 + a_2^2 + a_3^2 + a_4^2 + a_5^2 + a_6^2 + a_7^2 + a_8^2 + a_9^2 + a_{10}^2 = 1 \tag{5.11}$$

After linearization of Equations (5.10) and (5.11), the adjustment problem can be modeled as a parametric adjustment with constraint. The system of normal equations for this adjustment is given in Equation (5.12).

$$\begin{pmatrix} B^T \cdot D_{LL}^{-1} \cdot B & c \\ c^T & 0 \end{pmatrix} \cdot \begin{pmatrix} \delta\hat{X} \\ \lambda \end{pmatrix} + \begin{pmatrix} B^T \cdot D_{LL}^{-1} \cdot W \\ C_0 \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix} \tag{5.12}$$

Given a set of n, (n > 10), three dimensional coordinates; the components of Equation (5.12) can be defined as follows:

$$B = \begin{pmatrix} x_1^2 & y_1^2 & z_1^2 & x_1 \cdot y_1 & x_1 \cdot z_1 & y_1 \cdot z_1 & x_1 & y_1 & z_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n^2 & y_n^2 & z_n^2 & x_n \cdot y_n & x_n \cdot z_n & y_n \cdot z_n & x_n & y_n & z_n & 1 \end{pmatrix} \tag{5.13.1}$$

$$D_{LL} = \begin{pmatrix} \sigma_{L_1}^2 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \sigma_{L_n}^2 \end{pmatrix} \tag{5.13.2}$$

$$c^T = \begin{pmatrix} a_1^{(0)} & a_2^{(0)} & a_3^{(0)} & a_4^{(0)} & a_5^{(0)} & a_6^{(0)} & a_7^{(0)} & a_8^{(0)} & a_9^{(0)} & a_{10}^{(0)} \end{pmatrix} \tag{5.13.3}$$

$$\delta\hat{X} = \begin{pmatrix} \delta a_1 & \delta a_2 & \delta a_3 & \delta a_4 & \delta a_5 & \delta a_6 & \delta a_7 & \delta a_8 & \delta a_9 & \delta a_{10} \end{pmatrix} \tag{5.13.4}$$

$$W^T = \begin{pmatrix} w_1 & \cdots & w_i & \cdots & w_n \end{pmatrix} \tag{5.13.5}$$

$$C_0 = \frac{1}{2} \cdot c^T \cdot c - \frac{1}{2} \tag{5.13.6}$$

Furthermore, for each point i ($1 \leq i \leq n$)

$$\begin{aligned} w_i = {} & a_1^{(0)} \cdot x_i^2 + a_2^{(0)} \cdot y_i^2 + a_3^{(0)} \cdot z_i^2 + a_4^{(0)} \cdot x_i \cdot y_i + a_5^{(0)} \cdot x_i \cdot z_i + a_6^{(0)} \cdot y_i \cdot z_i \\ & + a_7^{(0)} \cdot x_i + a_8^{(0)} \cdot y_i + a_9^{(0)} \cdot z_i + a_{10}^{(0)} \end{aligned} \tag{5.14.1}$$

$$\sigma_{L_i}^2 = \begin{pmatrix} a_{X_i} & a_{Y_i} & a_{Z_i} \end{pmatrix} \cdot D_{ii} \cdot \begin{pmatrix} a_{X_i} \\ a_{Y_i} \\ a_{Z_i} \end{pmatrix} \tag{5.14.2}$$

$$D_{ii} = \begin{pmatrix} \sigma_{X_i}^2 & \sigma_{X_i} \cdot \sigma_{Y_i} & \sigma_{X_i} \cdot \sigma_{Z_i} \\ \sigma_{Y_i} \cdot \sigma_{X_i} & \sigma_{Y_i}^2 & \sigma_{Y_i} \cdot \sigma_{Z_i} \\ \sigma_{Z_i} \cdot \sigma_{X_i} & \sigma_{Z_i} \cdot \sigma_{Y_i} & \sigma_{Z_i}^2 \end{pmatrix} \tag{5.14.3}$$

$$\begin{pmatrix} a_{X_i} \\ a_{Y_i} \\ a_{Z_i} \end{pmatrix} = \begin{pmatrix} 2 \cdot a_1^{(0)} \cdot x_i + a_4^{(0)} \cdot y_i + a_5^{(0)} \cdot z_i + a_7^{(0)} \\ 2 \cdot a_2^{(0)} \cdot y_i + a_4^{(0)} \cdot x_i + a_6^{(0)} \cdot z_i + a_8^{(0)} \\ 2 \cdot a_3^{(0)} \cdot z_i + a_5^{(0)} \cdot x_i + a_6^{(0)} \cdot y_i + a_9^{(0)} \end{pmatrix} \tag{5.14.4}$$

The 10 parameters $\left(a_1^{(0)}, \cdots, a_{10}^{(0)}\right)$ are the initial approximates of the polynomial surface parameters. There are quite a few methods for estimating these approximate values. The method used here involved computing the eigenvalues and the eigenvectors of Equation (5.15).

$$A = B^T \cdot B \tag{5.15}$$

Since the eigenvalues represent the sum of the squared residuals for each column vector, the eigenvector corresponding to the minimum eigenvalue calculated from matrix A in Equation (5.15), represents the best estimate of the 10 parameters. Using this vector as the initial estimates in the least squares model will allow this fit to be refined.

The standard form of the parametric adjustment with constraints defines the residual vector as:

$$-V = B \cdot \delta\hat{X} + W \tag{5.16.1}$$

Where

$$V^T = \begin{pmatrix} v_1 & \cdots & v_i & \cdots & v_n \end{pmatrix} \tag{5.16.2}$$

$$v_i = a_{X_i} \cdot v_{X_i} + a_{Y_i} \cdot v_{Y_i} + a_{Z_i} \cdot v_{Z_i} \tag{5.16.3}$$

To better scale the weight matrix, the a-priori variance of unit weight (σ_0^2) can be introduced as in Equation (5.17).

$$P_{LL} = \sigma_0^2 \cdot D_{LL}^{-1} \tag{5.17}$$

Solving Equation (5.12) for $\delta\hat{X}$, the corrections to the initial approximates contained in vector c can be computed as in Equation (5.18.1).

$$\delta\hat{X} = -N_{bb}^{-1} \cdot \left[I - c \cdot N_{cc}^{-1} \cdot c^T \cdot N_{bb}^{-1} \right] \cdot B^T \cdot P_{LL} \cdot W - N_{bb}^{-1} \cdot c \cdot N_{cc}^{-1} \cdot C_0 \tag{5.18.1}$$

Where I is the 10 x 10 identity matrix and N_{bb} and N_{cc} are defined below:

$$N_{bb} = B^T \cdot P_{LL} \cdot B \tag{5.18.2}$$

$$N_{cc} = c^T \cdot N_{bb}^{-1} \cdot c \tag{5.18.3}$$

Applying these corrections to the initial approximate vector (c) and iterating this equation until some threshold (ε) for the corrections to the initial approximates is reached; the estimates for the 10 surface parameters $\left(\hat{a}_1, \cdots, \hat{a}_{10}\right)$ can be refined.

Once the refined estimates of the surface parameters have been obtained, the cofactor matrix for the parameter vector (c) and the residual vector (v) can be determined from Equations (5.19.1) and (5.19.2) respectively.

$$Q_{XX} = N_{bb}^{-1} - N_{bb}^{-1} \cdot c \cdot N_{cc}^{-1} \cdot c^T \cdot N_{bb}^{-1} \tag{5.19.1}$$

$$Q_{VV} = P_{LL}^{-1} - B \cdot Q_{XX} \cdot B^T \tag{5.19.2}$$

The a-posteriori variance matrices for the parameter vector (c) and the residual vector (v) can now be estimated as in Equations (5.20.1) and (5.20.2), respectively.

$$D_{XX} = \hat{\sigma}_0^2 \cdot Q_{XX} \tag{5.20.1}$$

$$D_{VV} = \hat{\sigma}_0^2 \cdot Q_{VV} \tag{5.20.2}$$

Where $\hat{\sigma}_0^2$ is the a-posteriori variance of unit weight. The a-posteriori variance of unit weight can be calculated from Equation (5.21).

$$\hat{\sigma}_0^2 = \frac{V^T \cdot P_{LL} \cdot V}{n - 10 + 1} \tag{5.21}$$

From Equations (5.16) and (5.19) the test statistic for the student t probability distribution can be formed.

$$T_i = \frac{|v_i|}{\hat{\sigma}_0 \cdot \sqrt{q_{V_i V_i}}} \sim t(n - 1 - 10 + 1) \tag{5.22}$$

Where v_i is the ith residual of vector V, $\hat{\sigma}_0$ is the standard deviation computed from the a-posteriori variance of unit weight and $q_{V_i V_i}$ is the ith diagonal element of matrix Q_{VV}. Furthermore, the likelihood that one or more outliers may creep into the point cloud sample being used to form the polynomial surface, it would be a good idea to provide a statistic check on the goodness-of-fit for each calculated surface. Such a statistic can be produced by comparing the a-posteriori variance with the a-priori variance as shown in Equation (5.23).

$$\frac{V^T \cdot P_{LL} \cdot V}{\sigma_0^2} \sim \chi^2(n - 10 + 1) \tag{5.23}$$

5.3 Implementation

The methods described above were implemented using Microsoft Visual C++ 6.0. Figure 5.1 shows the basic idea of how each of these algorithms works.

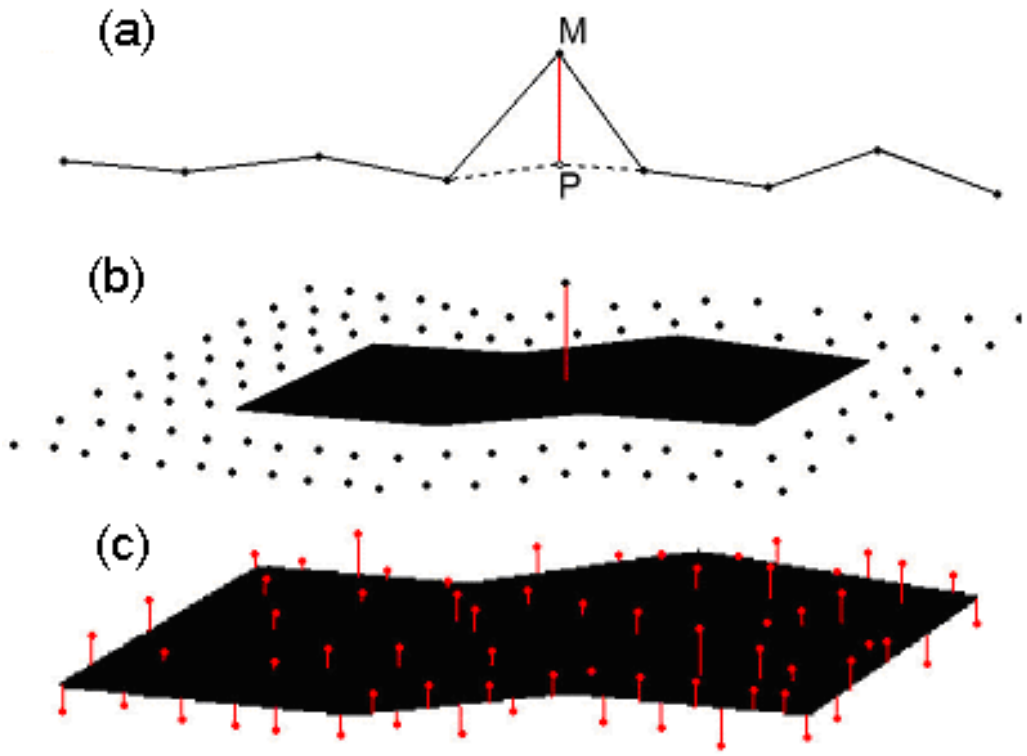

Figure 5.1: Methods for the detection of outliers. (a) Time series of points used to generate predictions (P) for measured points (M). (b) Polynomial surface patch in the immediate neighbourhood of the point being tested. (c) Spatial residuals to the best fit polynomial surface, used to test multiple points simultaneously.

The moving fixed interval prediction algorithm recognizes the fact that the point cloud can be treated as a series of lines of point data (Figure 5.1(a)). Since a significant portion of any terrestrial LiDAR scan is likely to include portions of the sky, numerous LiDAR points can be expected to be missing from the point cloud. These missing shots effectively segment the continuous line being followed by the scanner's optics, into multiple smaller line segments. Treating these smaller line segments as independent entities, allows the Kalman filter to be applied to each of these subset lines from the point

cloud. Since only a section of the total point cloud is being used to compare the results from these three methods, allowances have to be made for lines shorter that the window size ($-n_1, n_2$) and the window size has to be adjusted to accommodate points at the start and end of each line.

The quadratic curved-surface fitting algorithm generates small surface patches in the neighbourhood of each point (Figure 5.1(b)). This is an outlier detector in the spatial domain, which relies on the assumption that the points immediately adjacent to an outlier will themselves lie on the surface and not be outliers as well. The variable in implementing this method is the number of point cloud coordinates surrounding the point in question which should be used. On one hand, at least 10 points are required to derive the best fit surface, on the other hand, the larger the number of coordinates used, the greater the probability that other outliers will be incorporated into the calculation of the surface. In fact, when discussing LiDAR, the conditions, which cause an outlier, will also greatly increase the likelihood that other outliers lie close by. Therefore, care must be taken when setting a patch size.

Instead of computing a 10 parameter quadratic surface for numerous small patches, the quadratic surface can also be generated for much larger sections of the point cloud. The idea is to segment the point cloud and compute the polynomial surface for user defined sections of the point cloud. Using the residuals produced from the adjustment, one can examine the separation of each point from the surface. Using a test statistic, such as the one given in Equation (5.23), outliers can be identified. Due to the potential

discrepancies in the magnitude of any set of outliers, it might be necessary to discard identified outliers and re-compute the surface. Iteration in this manner should continue until no more outliers are identified. This should ensure that all outliers are taken into account.

It should be noted that using large number coordinate data to generate a 10 parameter quadratic surface can become computationally expensive when fitting numerous surfaces to a data set. One way of dealing with this problem is to choose a point to act as a local origin for the surface and difference all points used in the surface fitting with this local origin. This technique is the simplest way of preserving the surface orientation while providing a simple means of repositioning the computed surface back into its proper place within the larger data set.

Combining the temporal, moving fixed interval prediction algorithm, and the spatial, quadratic curved-surface fitting algorithms, can be accomplished in a variety of ways. The simplest method of combining these algorithms is to run the temporal algorithm followed by the spatial algorithm(s). During testing, this is how the algorithms were combined. For the rest of this chapter, whenever combining the temporal and spatial algorithms is discussed, it will be referring to executing the temporal algorithm (Figure 5.1 (a)) on a data set followed by the small patch version of the spatial algorithm (Figure 5.1 (b)).

5.4 Algorithm Tests

All of the case studies discussed below involve MTL data collected with the Lynx Mobile Mapper. To test the effectiveness of the outlier detection methods described in Section 5.2 and implemented in 5.3, three tests have been conducted using the data collected during each case study.

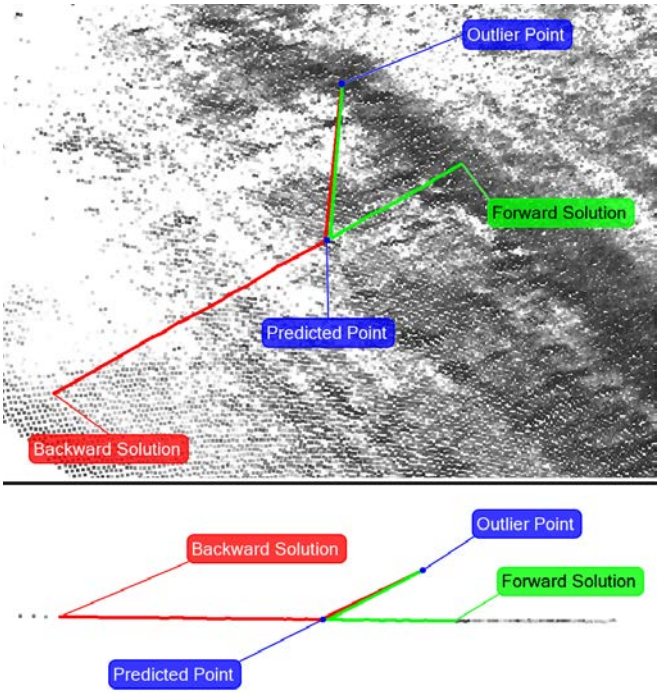

Figure 5.2: Time series calculation of a predicted point in a generic Lynx Mobile Mapper point cloud.

First, confirmation of the correct operation of the routines needed to be established. This was accomplished by observing results from a generic Lynx Mobile Mapper point cloud. Figure 5.2 shows graphically the results from one correctly identified outlier, found using the time series algorithm. In this figure, the forward and backward processes are highlighted as well as the positions of the outlier point and its expected position, derived using the time series algorithm. It is interesting to note that the backward process line appears longer than the forward process line. This is an optical illusion, since each line

contains the same number of points. The points in the backward process line are simply spaced farther apart than those of the forward process line.

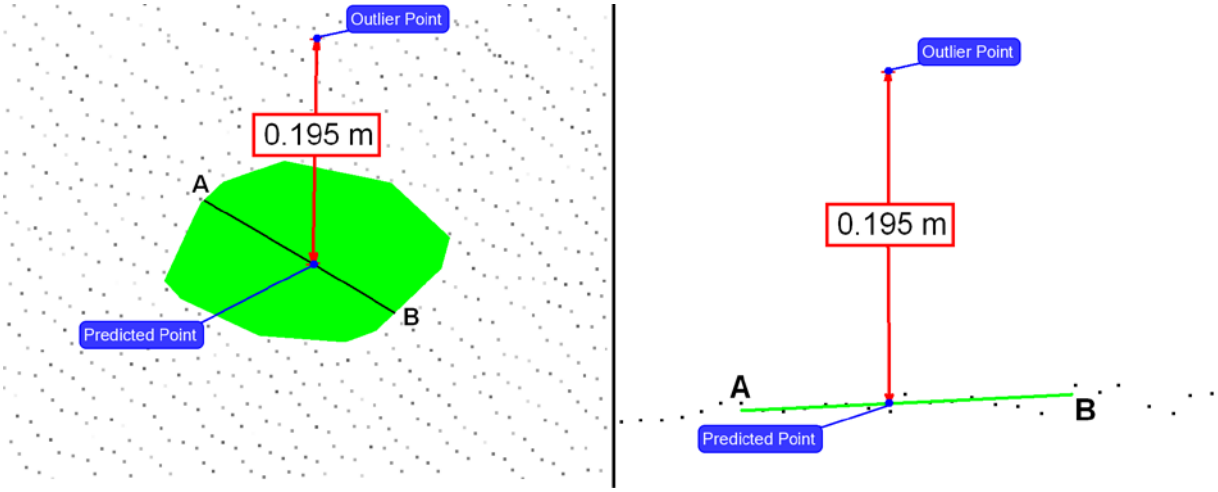

Figure 5.3: Spatial series calculation of a predicted point in a generic Lynx Mobile Mapper point cloud.

Similarly, Figure 5.3 shows graphically the results from a correctly identified outlier using the spatial series algorithm. In Figure 5.3, the best fit polynomial surface is displayed, along with the original outlier point and the shortest distance projection of the outlier point onto the surface. Examining cross section AB in this figure, it can be seen that the polynomial routine has flattened the noise in the road surface data.

From the mathematical model presented in Section 5.2.1, it is apparent that the user of this algorithm must choose an appropriate window size for the input data. This window size can be expected to vary depending on the density and arrangement of the data. Therefore, for each point cloud used in this experiment, the optimal window size must be determined. Trials were conducted where the window size was varied to see how this variable affected the reliability of the routine. The data selected by the routine was divided into correctly identified outliers and incorrectly identified non-outliers. The

results of these trials in the time series are given in Sections 5.4.1.1, 5.4.2.1 and 5.4.3.1, respectively.

Similarly, the spatial algorithm described in Section 5.2.2 requires two inputs that may need to be adjusted. The first input is the a-priori reference variance. Little needs to be done to correctly set this variance, since its value should be of the same magnitude as the point variances being input to the routine. Therefore, for all further testing, this value was set to $0.001m^2$. The other variable is the size of the patch of data used to determine the coefficients of the best fit polynomial. This value has to be at least 10 points, but cannot be allowed to grow too large, since this increases the possibility that some of the input points could, themselves, be outliers. To discover what effect patch size has on the effectiveness of the PSF routine, the patch size was varied and the data selected by the routine was again divided into correctly identified outliers and incorrectly identified non-outliers. The results of these trials in the spatial series are given in Sections 5.4.1.2, 5.4.2.2 and 5.4.3.2, respectively.

Once the correct settings for both the time and spatial series approaches have been determined, trials were conducted to establish the effectiveness of operating the routines individually and in tandem. First each routine was run individually to establish its effectiveness on each point cloud. The routines were then run in tandem to see if this improved the final result. The results of these trials in each case study are given in Sections 5.4.1.3, 5.4.2.3 and 5.4.3.3, respectively.

Finally, the outlier removal tool available in the commercial software package Polyworks (www.innovmetric.com) was used to judge the effectiveness of these routines. A few outlier routines exist within the Polyworks software package, all of them tied to another function of the software. For testing purposes the "reject outliers" function associated with the "wrap mesh" function of Polyworks IMSurvey module was used. Details about this function can be found in [121]. This "reject outliers" routine uses the spatial displacement of the points from their neighbours to determine if outliers are present. This routine, run before a triangulated mesh is imposed on the data, has two inputs that the user must specify. The first input, max point-to-point distance, is used to simply limit the distance allowed between two neighbouring data points. The second input, maximum cluster size, is used to determine if a cluster of points, each having a nearest neighbour less than or equal to the max point-to-point distance, fall outside the main body of the point cloud. The cluster size is determined by the diagonal of the bounding box surrounding the cluster.

Since the Polyworks "reject outliers" routine requires that a maximum point to point distance is set, the first object was to establish the maximum and minimum grid point spacing in each of the candidate point clouds. This was done by random sampling of points in dense and sparse areas of the point cloud. The maximum and minimum values obtained by random sampling are listed in Table 5.6, Table 5.11 and Table 5.16. Setting the max point-to-point distance to less than the minimum caused, in each case, a warning box to appear saying that outlier rejection has failed. Once the input parameters for the Polyworks routine were decided upon, each point cloud collected from the case studies

were run through this routine. The results of these trials are presented in Sections 5.4.1.4, 5.4.2.4 and 5.4.3.4, respectively.

5.4.1 Case Study 1 – Vaughan, Ontario, Canada

Data was collected of a typical parking lot in Vaughan Ontario in early November of 2008 and again in January of 2010. From this data a section of the asphalt was selected and a patch of data was extracted from each of three separate data strips of the parking lot. The content of these three strips is summarized in Table 5.1 and the three strips are shown in Figure 5.4.

Table 5.1: Specifications for point cloud sections collected in a parking lot in Vaughan Ontario and used in algorithm testing.

Point Cloud	A1	B1	C1
Total No. of Points	1098689	295147	237740
Total No. of Outliers	11035	872	31
Total % of Points Which are Outliers	1.00	0.30	0.01

Point cloud A1 contains numerous outliers in two large groups. As shown in Table 5.1, the outliers make up 1.00% of the total point cloud. This data was collected on a January day where the asphalt was wet, but the temperature was just below 0ºC. The prevailing cold wet conditions caused condensation from the vehicle's exhaust pipe to combine with varying high and low intensity returns from the standing pools of water. This caused multiple laser reflections to be recorded above the asphalt surface.

Point cloud B1 was collected later the same day as point cloud A1. There are far fewer outliers in this point cloud (0.30% from Table 5.1) and they are more spread out. Conditions were nearly the same, however the temperature had risen to just above 0ºC. A traffic barrier arm, which restricts vehicle access to the parking lot, caused the linear

outliers in the top left hand corner of point clouds A1 and B1. The LiDAR system captured data on this arm while the arm was is operation.

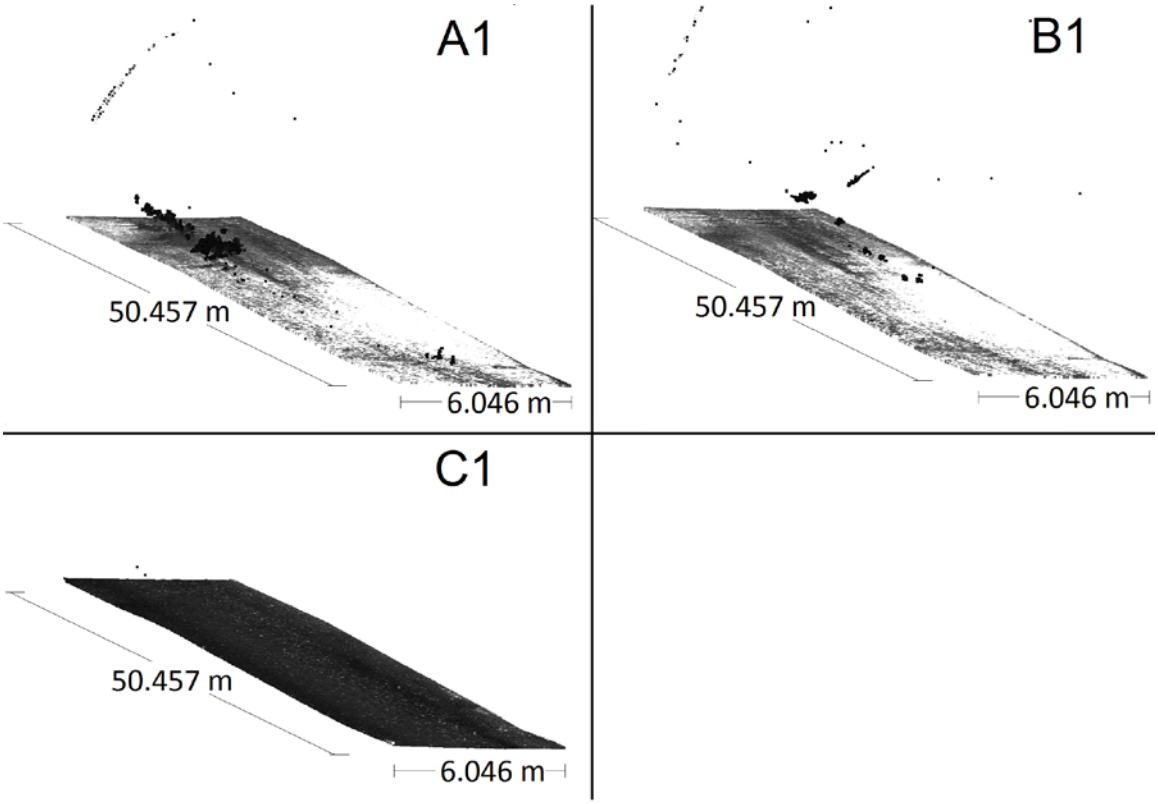

Figure 5.4: Point cloud sections collected in a parking lot in Vaughan Ontario and used in algorithm testing. Point cloud A1 contains numerous outliers clustered together in two large groups above the asphalt surface. Point cloud B1 contains numerous outliers as well; however, these outliers are more evenly distributed above the asphalt surface. Point cloud C1 contains few outliers, most of which are within centimeters of the asphalt surface.

In contrast, point cloud C1 was collected on a November day where the temperature was close to 10ºC and the pavement was dry. These conditions produced a point cloud with comparatively few outliers (0.01% from Table 5.1). Many of the outliers which do exist in this data set are within centimetres of the asphalt surface. The traffic barrier arm was not captured in operation in this scan.

5.4.1.1 Test 1 – Appropriate Window Size for the Time Series Approach

As mentioned at the beginning of Section 5.4, trials to determine the optimal window size for the time series approach (algorithm (a)) were conducted. These trials included the data collected in Vaughan Ontario.

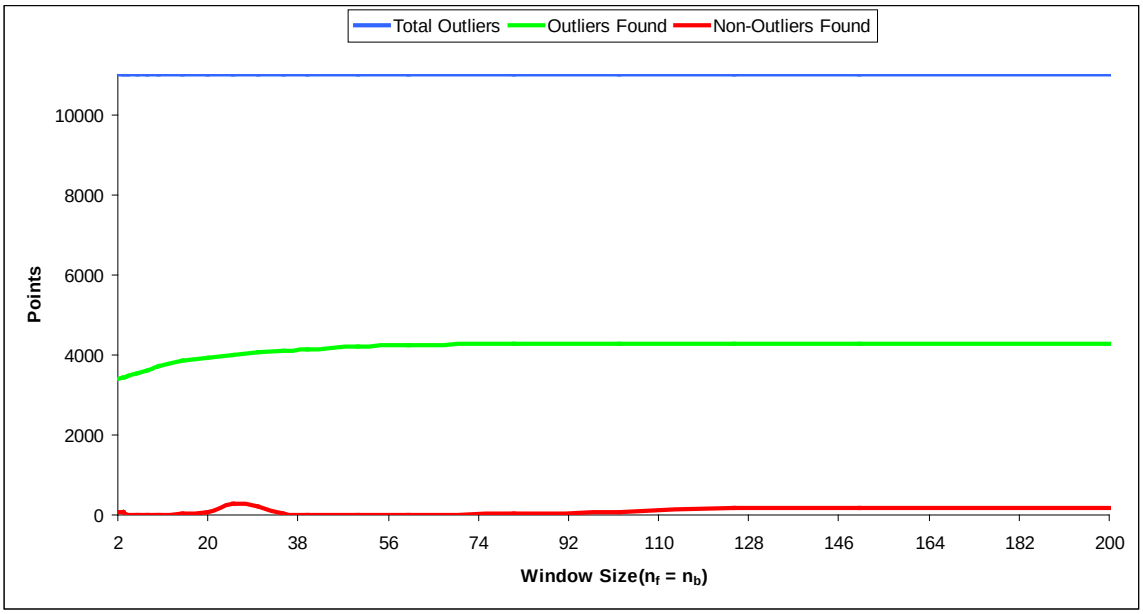

Figure 5.5: Results of varying window size in the time series approach (routine (a)) using point cloud A1.

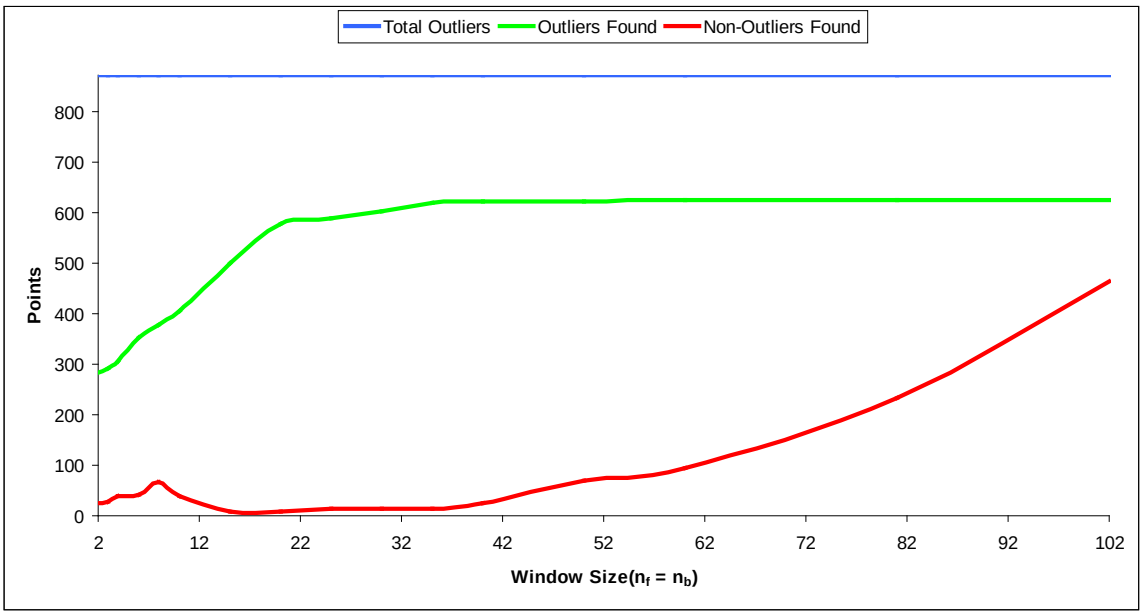

Figure 5.6: Results of varying window size in the time series approach (routine (a)) using point cloud B1.

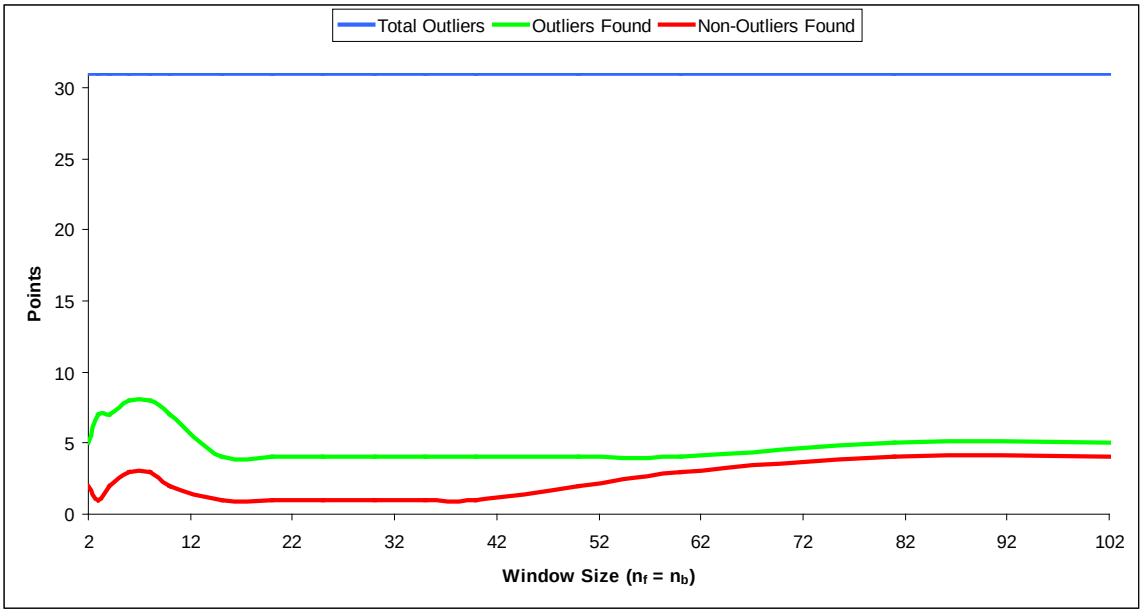

Figure 5.7: Results of varying window size in the time series approach (routine (a)) using point cloud C1.

Figure 5.5 shows the results of varying the time series window size in point cloud A1 with respect to the number of outliers and non-outliers found. These results are discussed in Section 5.5.1.

Similarly, Figure 5.6 shows the results of varying the time series window size in point cloud B1 with respect to the number of outliers and non-outliers found. Finally, Figure 5.7 shows the results of varying the time series window size in point cloud C1 with respect to the number of outliers and non-outliers found.

5.4.1.2 Test 2 – Appropriate Sample Size for the Spatial Series Approach

As mentioned at the beginning of Section 5.4, trials to determine the optimal data patch size for the spatial series approach (algorithm (b)) routine were conducted. These trials included the data collected in Vaughan Ontario.

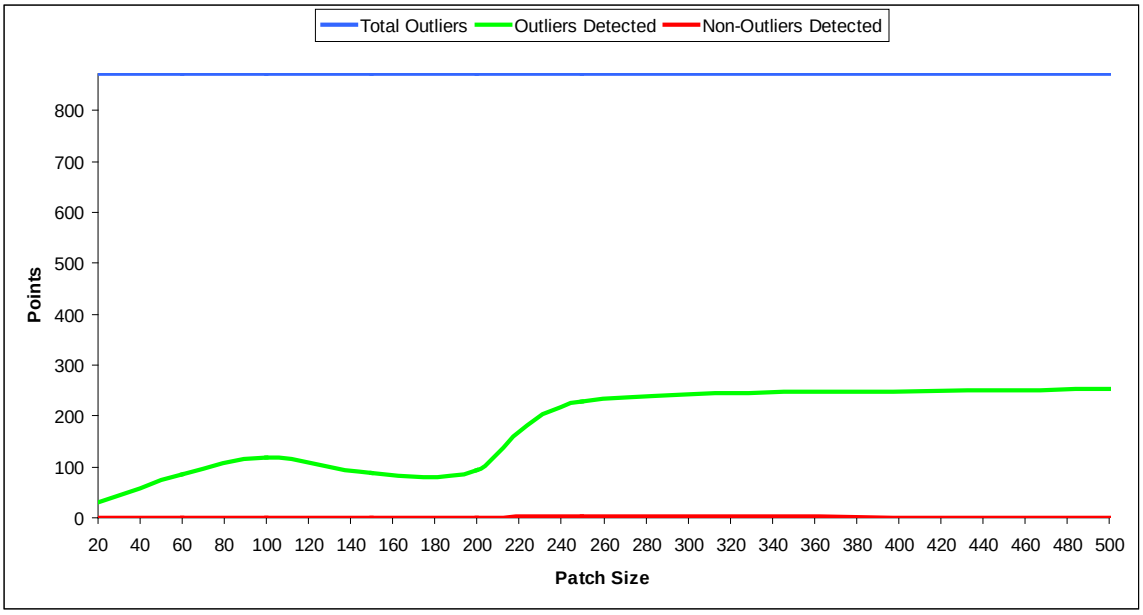

Figure 5.8: Results of varying window size in the spatial series approach (routine (b)) using point cloud B1.

Figure 5.8 shows the results for varying the spatial series patch size in point cloud B1 with respect to the number of outliers and non-outliers found. These results are discussed in Section 5.5.2.

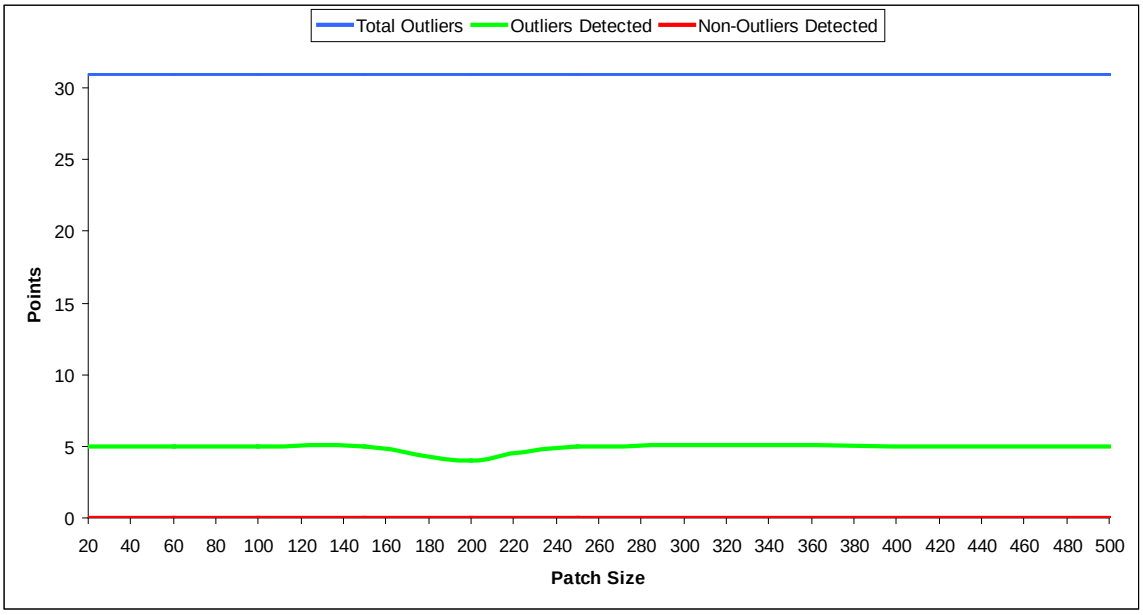

Figure 5.9: Results of varying window size in the spatial series approach (routine (b)) using point cloud C1.

Similarly, Figure 5.9 shows the results for varying the spatial series patch size in point cloud C1 with respect to the number of outliers and non-outliers found.

5.4.1.3 Test 3 – Maximum Number of Outliers Detectable by the Routines

As mentioned at the beginning of Section 5.4, trials to determine the individual and combined effectiveness of the outlier methods presented in Section 5.2 were conducted. These trials included the data collected in Vaughan Ontario.

Table 5.2: Best results from trials conducted using algorithm (a) on point clouds A1, B1 and C1.

Point Cloud	A1	B1	C1
Window Size (points)	50	35	8
No. of Outliers Identified	4209	620	8
No. of Non-Outliers Identified	0	14	3
No. of Outliers Missed	6826	252	23
% of Outliers Identified	38.14	71.10	25.81
% of Point Cloud Identified	1.00	0.30	0.01
% of Point Cloud Identified Incorrectly	0.00	0.00	0.00

Table 5.3: Best results from trials conducted using algorithm (b) on point clouds B1 and C1.

Point Cloud	B1	C1
Patch Size (points)	500	150
No. of Outliers Identified	253	5
No. of Non-Outliers Identified	0	0
No. of Outliers Missed	619	26
% of Outliers Identified	29.01	16.13
% of Point Cloud Identified	0.30	0.01
% of Point Cloud Identified Incorrectly	0.00	0.00

Table 5.4: Best results from trials conducted using algorithm (c) on point clouds A1, B1 and C1.

Point Cloud	A1	B1	C1
No. of Outliers Identified	5221	712	15
No. of Non-Outliers Identified	15431	0	193
No. of Outliers Missed	5060	160	16
% of Outliers Identified	47.31	81.65	48.39
% of Point Cloud Identified	1.88	0.24	0.09
% of Point Cloud Identified Incorrectly	1.40	0.00	0.08

Table 5.5: Results from trials conducted using algorithm (a) preceding algorithm (b) on point clouds B1 and C1.

Point Cloud	B1	C1
Window Size (Routine A)	35	8
Patch Size (Routine B)	500	150
No. of Outliers Identified (Routine A)	620	5
No. of Outliers Identified (Routine B)	23	2
No. of Non-Outliers Identified (Routine A)	14	7
No. of Non-Outliers Identified (Routine B)	0	0
No. of Outliers Missed	252	26
% of Outliers Identified	73.74	22.58
% of Point Cloud Identified	0.22	0.01
% of Point Cloud Identified Incorrectly	0.00	0.00

The results from tests conducted using point clouds A1, B1 and C1 are given in Table 5.2 for algorithm (a), in Table 5.3 for algorithm (b) and in Table 5.4 for algorithm (c). In addition, a combination of algorithms (a) and (b) was performed, where the reduced point cloud produced by algorithm (a) was input to algorithm (b). The results for this test conducted using data strips B1 and C1 are given in Table 5.5. These results are discussed in Section 5.5.3

5.4.1.4 Test 4 –Using Commercial Software to Detect Outliers

As mentioned at the beginning of Section 5.4, the three data sets collected in the parking lot in Vaughan Ontario were loaded into Polyworks IMSurvey and the “reject outlier” routine available from the “wrap mesh” function was used to isolate outliers in the three point clouds.

The results from these tests are listed in Table 5.6 below. The parameters used in the Polyworks routine are included in Table 5.6. These results are discussed in Section 5.5.4.

Table 5.6: Results from trials conducted using Polyworks IMSurvey's reject outliers routine on point clouds A1, B1 and C1.

Point Cloud	A1	B1	C1
Max Spot Space Measured (m)	0.072	0.053	0.080
Min Spot Space Measured (m)	0.012	0.023	0.024
Max Point-to-Point Distance Used (m)	0.100	0.080	0.104
Maximum Cluster Size Used (m)	5.000	5.000	1.000
No. of Outliers Identified	4842	717	17
No. of Non-Outliers Identified	155389	15846	14711
No. of Outliers Missed	6193	155	14
% of Outliers Identified	43.88	82.22	54.84
% of Point Cloud Identified	15.15	5.66	6.20
% of Point Cloud Identified Incorrectly	14.14	5.37	6.19

5.4.2 Case Study 2 – Pontarlier, France

In late November 2009 MTL data was collected in a lumber yard outside the village of Pontarlier France. The lumber yard featured an unpaved, rough finished, mostly native clay driving area that had been pitted and grooved by the heavy vehicles using it. The weather conditions were dry, as was the clay. The ambient temperature during the collect was about 15ºC. Several strips of data were collected in rapid succession. The two strips selected were collected from successive drive passes of the lumber yard. From the two selected strips a sample area of fixed dimension was isolated and the data was extracted. The content of these two strips is summarized in Table 5.7 and the three strips are shown in Figure 5.10.

Table 5.7: Specifications for point cloud sections collected in a lumber yard in Pontarlier, France and used in algorithm testing.

Point Cloud	A2	C2
Total No. of Points	495345	257730
Total No. of Outliers	280	220
% of Point Cloud Outliers	0.06	0.09

Due to the rough condition of the road surface in the lumber yard, the vehicle was driven exceedingly slowly (about 2 km/h) during the collection of point cloud A2. As is shown in Table 5.7, the identified outliers make up about 0.06% of the total point cloud. Since

the surface conditions were relatively dry, the outliers observed in this section of the point cloud could be attributed to the interaction of the laser beam with the rough conditions of the road surface and the reflectance properties of the soil.

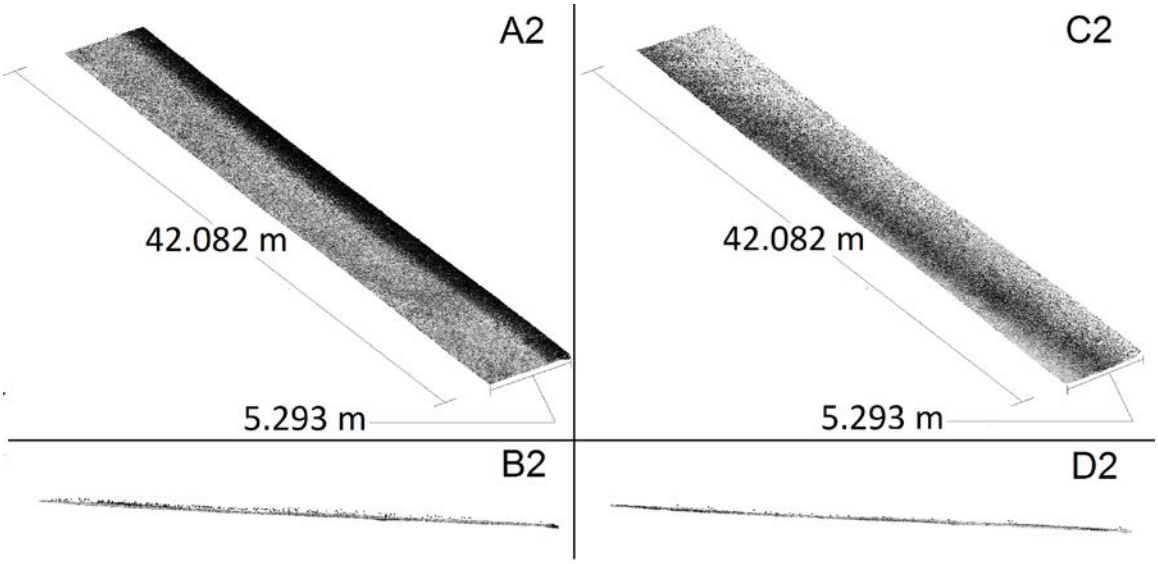

Figure 5.10: Point cloud sections collected in a lumber yard in Pontarlier, France and used in algorithm testing. Point cloud A2 contains numerous outliers spread out above the surface of the lumber yard. Point cloud B2 shows the previous point cloud from ground level, were the cluster of outliers above the road surface can be observed. Point cloud C2 also has outliers spread out above the surface of the road. Point cloud D2 shows the previous point cloud from ground level, were the cluster of outliers above the road surface can be observed.

For the collection of point cloud C2, it was decided to try and drive faster in the lumber yard (approximately 4 km/h). Table 5.7 shows that the percentage of outliers in point cloud C2 is maintained equivalent to the previous point cloud. Due to the fact that this point cloud was collected immediately following the collection of the data in point cloud A2, the weather and surface condition of the road surface can be considered to be equivalent. Therefore, the cause of the outliers in this data is most likely the same as previously stated.

5.4.2.1 Test 1 – Appropriate Window Size for the Time Series Approach

As mentioned at the beginning of Section 5.4, trials to determine the optimal window size for the Time Series approach (algorithm (a)) were performed on the data collected in Pontarlier France.

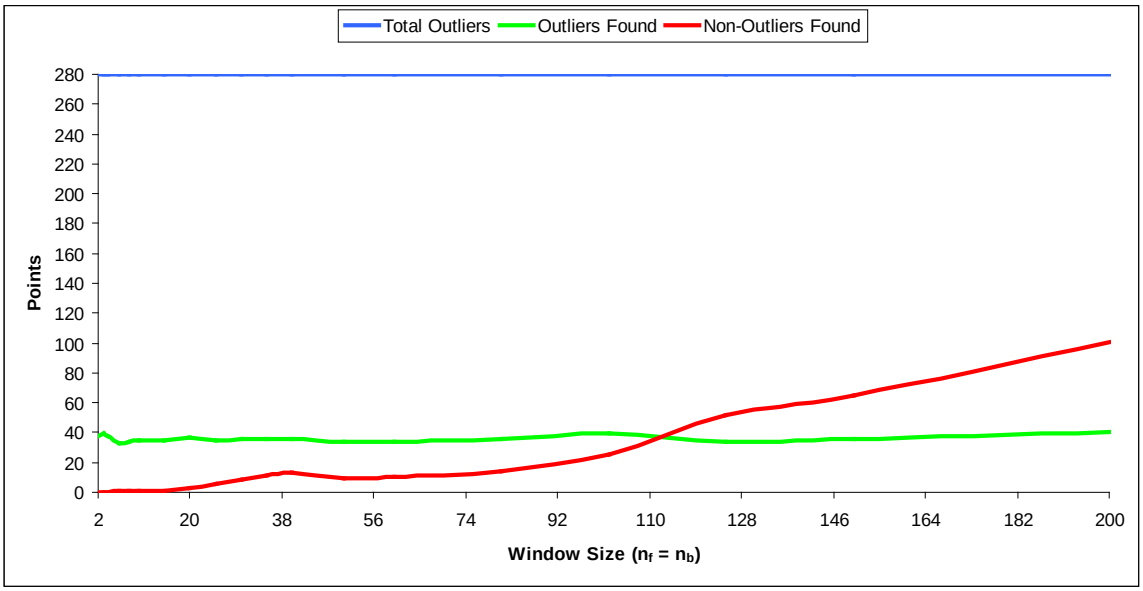

Figure 5.11: Results of varying window size in the time series approach (routine (a)) using point cloud A2.

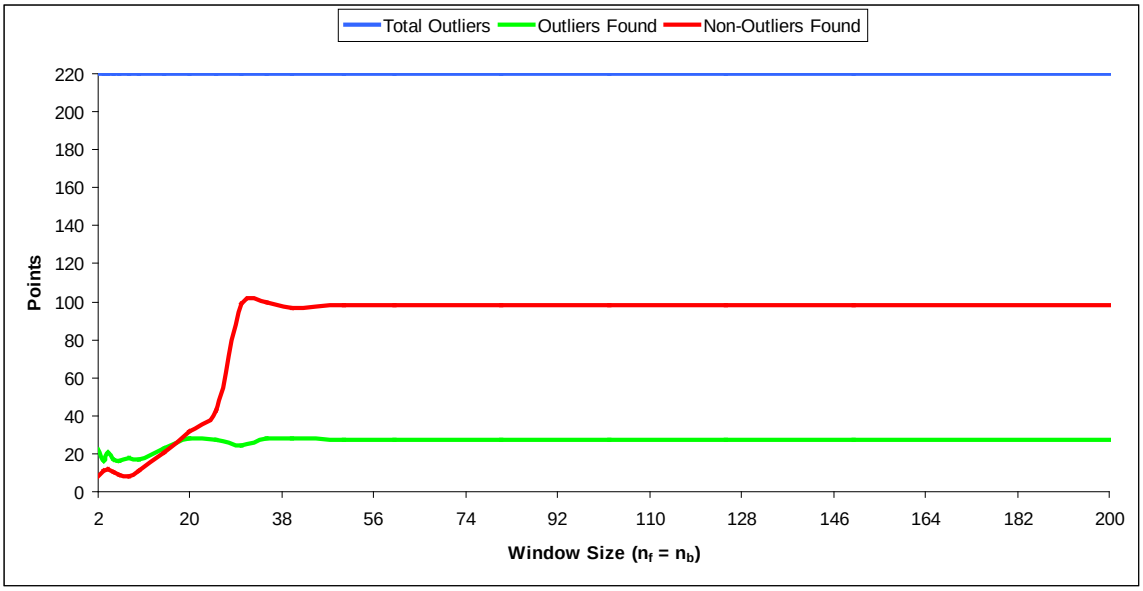

Figure 5.12: Results of varying window size in the time series approach (routine (a)) using point cloud C2.

Figure 5.11 shows the results of varying the time series window size in point cloud A2 with respect to the number of outliers and non-outliers found.

Similarly, Figure 5.12 shows the results of varying the time series window size in point cloud C2 with respect to the number of outliers and non-outliers found. These results are discussed in Section 5.5.1.

5.4.2.2 Test 2 – Appropriate Sample Size for the Spatial Series Approach

As mentioned at the beginning of Section 5.4, trials to determine the optimal data patch size for the spatial series approach (algorithm (b)) were conducted on the data collected in Pontarlier, France.

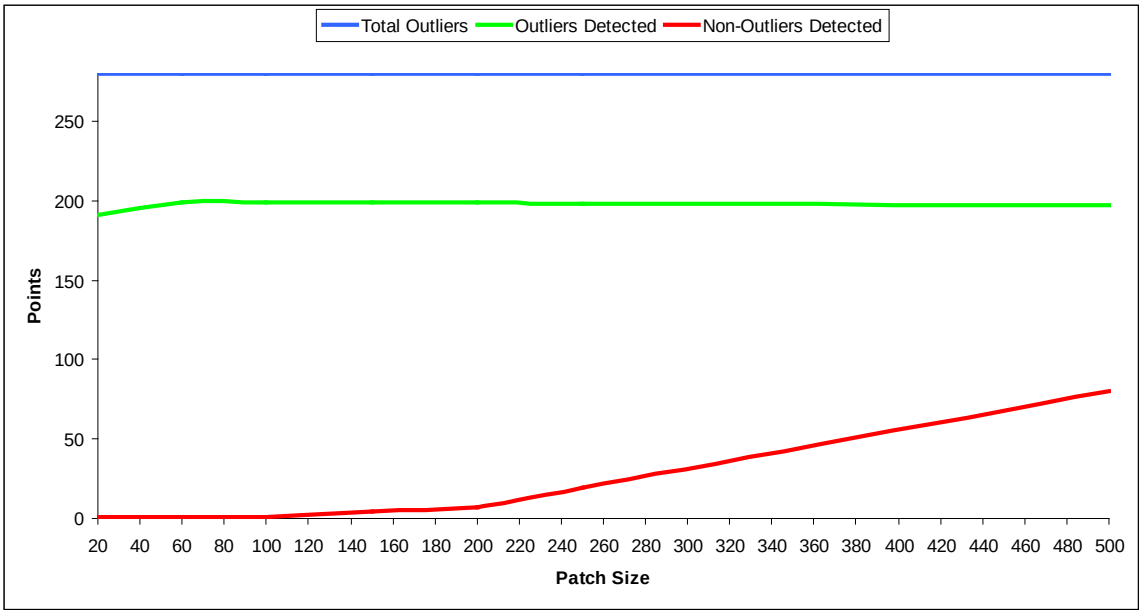

Figure 5.13: Results of varying window size in the spatial series approach (routine (b)) using point cloud A2.

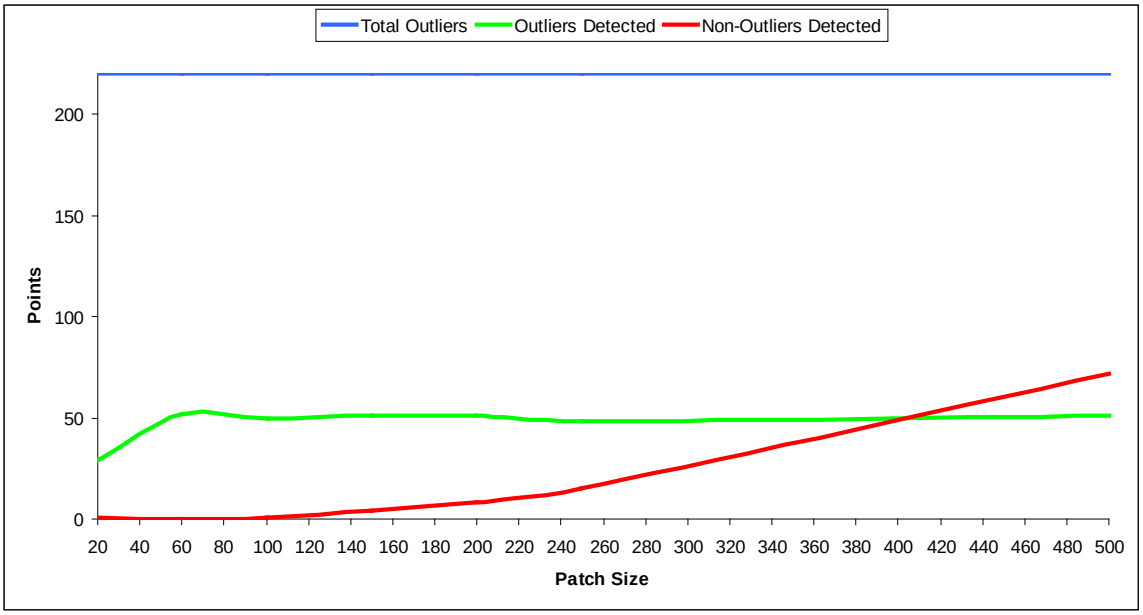

Figure 5.14: Results of varying window size in the spatial series approach (routine (b)) using point cloud C2.

Figure 5.13 shows the results for varying the spatial series patch size in point cloud A2 with respect to the number of outliers and non-outliers found. These results are discussed in Section 5.5.2. Similarly, Figure 5.14 shows the results for varying the spatial series patch size in point cloud C2 with respect to the number of outliers and non-outliers found.

5.4.2.3 Test 3 – Maximum Number of Outliers Detectable by the Routines

As mentioned at the beginning of Section 5.4, trials to determine the individual and combined effectiveness of the outlier methods presented in Section 5.2 were conducted on the data collected in Pontarlier, France. The results from tests conducted using point clouds A2 and C2 are given in Table 5.8 for algorithm (a) and in Table 5.9 for algorithm (b).

Table 5.8: Best results from trials conducted using algorithm (a) on point clouds A2 and C2.

Point Cloud	A2	C2
Window Size (points)	20	20
No. of Outliers Identified	37	28
No. of Non-Outliers Identified	3	32
No. of Outliers Missed	243	192
% of Outliers Identified	13.21	12.73
% of Point Cloud Identified	0.06	0.10
% of Point Cloud Identified Incorrectly	0.00	0.01

Table 5.9: Best results from trials conducted using algorithm (b) on point clouds A2 and C2.

Point Cloud	A2	C2
Patch Size (points)	100	60
No. of Outliers Identified	199	52
No. of Non-Outliers Identified	1	0
No. of Outliers Missed	81	168
% of Outliers Identified	71.07	23.64
% of Point Cloud Identified	0.06	0.09
% of Point Cloud Identified Incorrectly	0.00	0.00

Table 5.10: Results from trials conducted using algorithm (a) preceding algorithm (b) on point clouds A2 and C2.

Point Cloud	A2	C2
Window Size (Routine A)	20	20
Patch Size (Routine B)	100	60
No. of Outliers Identified (Routine A)	37	28
No. of Outliers Identified (Routine B)	162	29
No. of Non-Outliers Identified (Routine A)	3	32
No. of Non-Outliers Identified (Routine B)	0	0
No. of Outliers Missed	243	192
% of Outliers Identified	71.07	25.91
% of Point Cloud Identified	0.06	0.03
% of Point Cloud Identified Incorrectly	0.00	0.01

In addition, a combination of algorithms (a) and (b) was performed, where the reduced point cloud produced by algorithm (a) was input to algorithm (b). The results for this test conducted using data strips A2 and C2 are given in Table 5.10. These results are discussed in Section 5.5.3.

5.4.2.4 Test 4 –Using Commercial Software to Detect Outliers

As mentioned at the beginning of Section 5.4, the two data sets collected in the lumber yard in Pontarlier France, were loaded into Polyworks IMSurvey and the “reject outlier”

routine available from the "wrap mesh" function was used to isolate outliers in the two point clouds.

The results from these tests, conducted on the points clouds obtained in Pontarlier, France, are listed in Table 5.11 below. The parameters used in the Polyworks routine are included in Table 5.11. These results are discussed in Section 5.5.4.

Table 5.11: Results from trials conducted using Polyworks IMSurvey's reject outliers routine on point clouds A2 and C2.

Point Cloud	A2	C2
Max Spot Space Measured (m)	0.040	0.064
Min Spot Space Measured (m)	0.012	0.008
Max Point-to-Point Distance Used (m)	0.050	0.100
Maximum Cluster Size Used (m)	1.000	1.000
No. of Outliers Identified	221	84
No. of Non-Outliers Identified	30645	26841
No. of Outliers Missed	59	136
% of Outliers Identified	78.93	38.18
% of Point Cloud Identified	6.24	10.50
% of Point Cloud Identified Incorrectly	6.19	10.41

5.4.3 Case Study 3 – Washington D.C., U.S.A.

In early January 2009 MTL data was collected of downtown Washington D.C. as part of the preparation for President Obama's inauguration. Several streets in and around Pennsylvania Avenue and Constitution Avenue were collected along the proposed parade route. The weather conditions were dry. The ambient temperature during the collect was about 5ºC. Several strips of data were collected in rapid succession. The two strips selected were collected from different LiDAR sensors during the same drive segment. From the two selected strips a sample area of fixed dimension was isolated and the data was extracted (Figure 5.15). The contents of these two strips are summarized in Table 5.12.

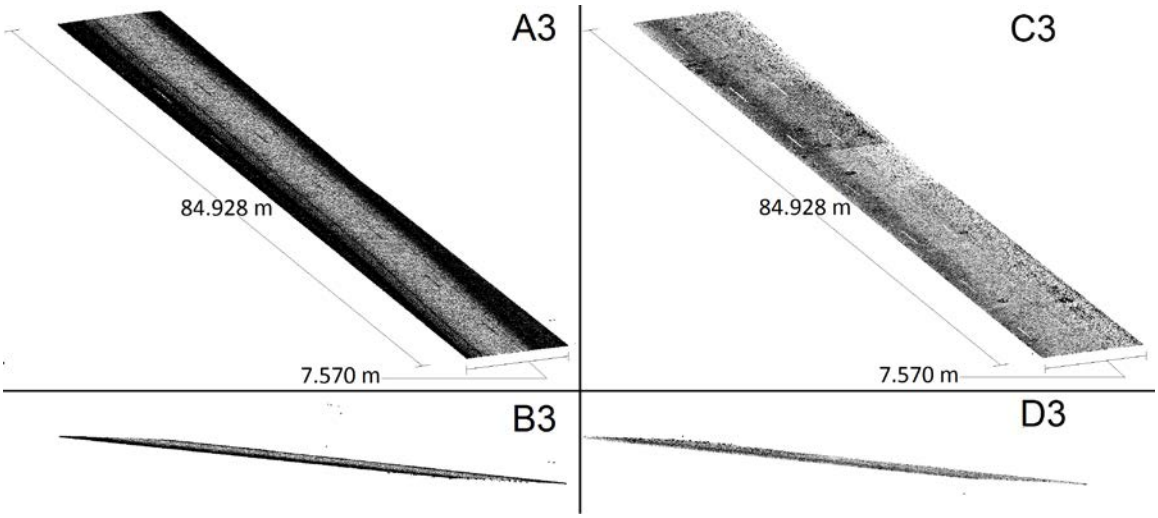

Figure 5.15: Point cloud sections collected in a street in Washington D.C., U.S.A and used in algorithm testing. Point cloud A3 contains numerous outliers spread out above the surface of the lumber yard. Point cloud B3 shows the previous point cloud from ground level, were the cluster of outliers above the road surface can be observed. Point cloud C3 also has outliers spread out above the surface of the road. Point cloud D3 shows the previous point cloud from ground level, were the cluster of outliers above the road surface can be observed.

Table 5.12: Specifications for point cloud sections collected on a street in Washington D.C., U.S.A and used in algorithm testing.

Point Cloud	A3	C3
Total No. of Points	345575	291159
Total No. of Outliers	95	54
% of Point Cloud Outliers	0.03	0.02

5.4.3.1 Test 1 – Appropriate Window Size for the Time Series Approach

As mentioned at the beginning of Section 5.4, trials to determine the optimal window size for the time series approach (algorithm (a)) were conducted on the data collected in Washington D.C.

Figure 5.16 shows the results of varying the time series window size in point cloud A3 with respect to the number of outliers and non-outliers found. Similarly, Figure 5.17 shows the results of varying the time series window size in point cloud C3 with respect to

the number of outliers and non-outliers found. These results are discussed in Section 5.5.1.

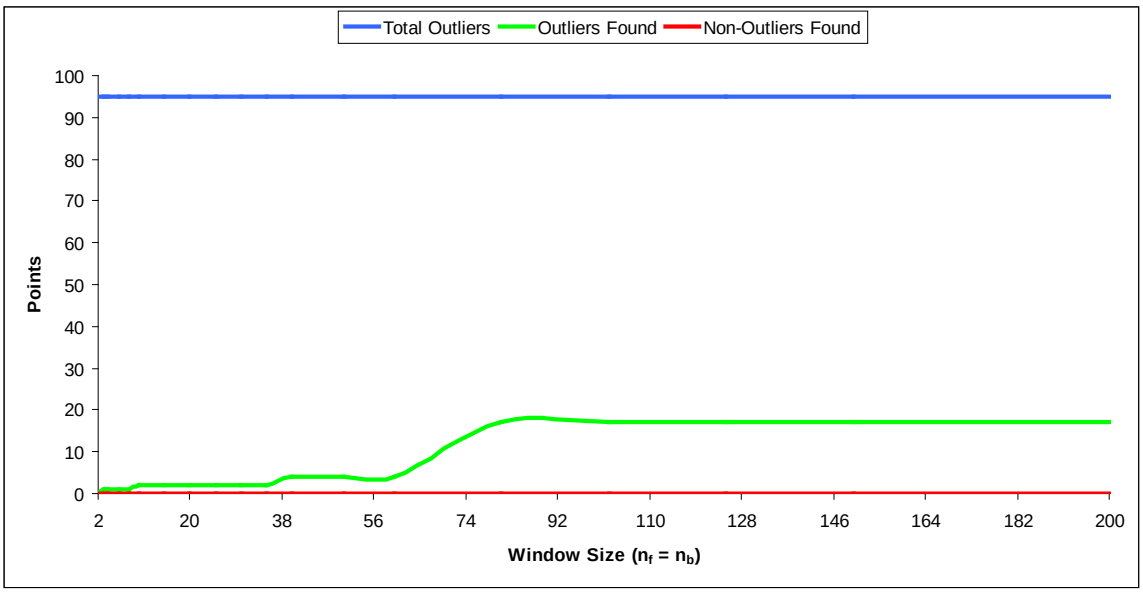

Figure 5.16: Results of varying window size in the time series approach (routine (a)) using point cloud A3.

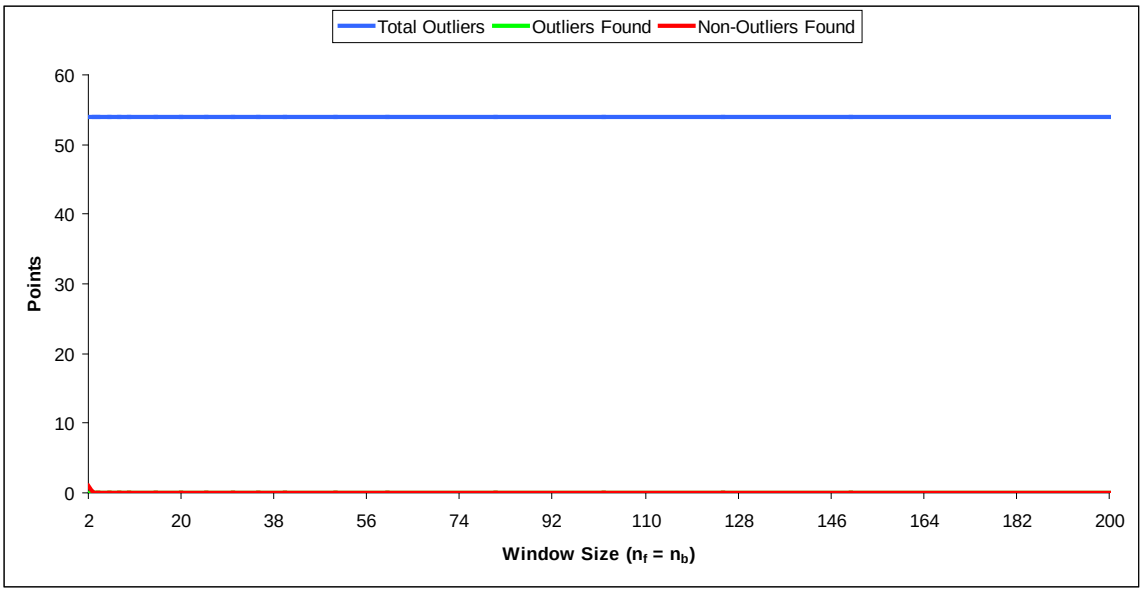

Figure 5.17: Results of varying window size in the time series approach (routine (a)) using point cloud C3.

5.4.3.2 Test 2 – Appropriate Sample Size for the Spatial Series Approach

As mentioned at the beginning of Section 5.4, trials to determine the optimal window size for the spatial series approach (algorithm (b)) were conducted on the data collected in Washington D.C.

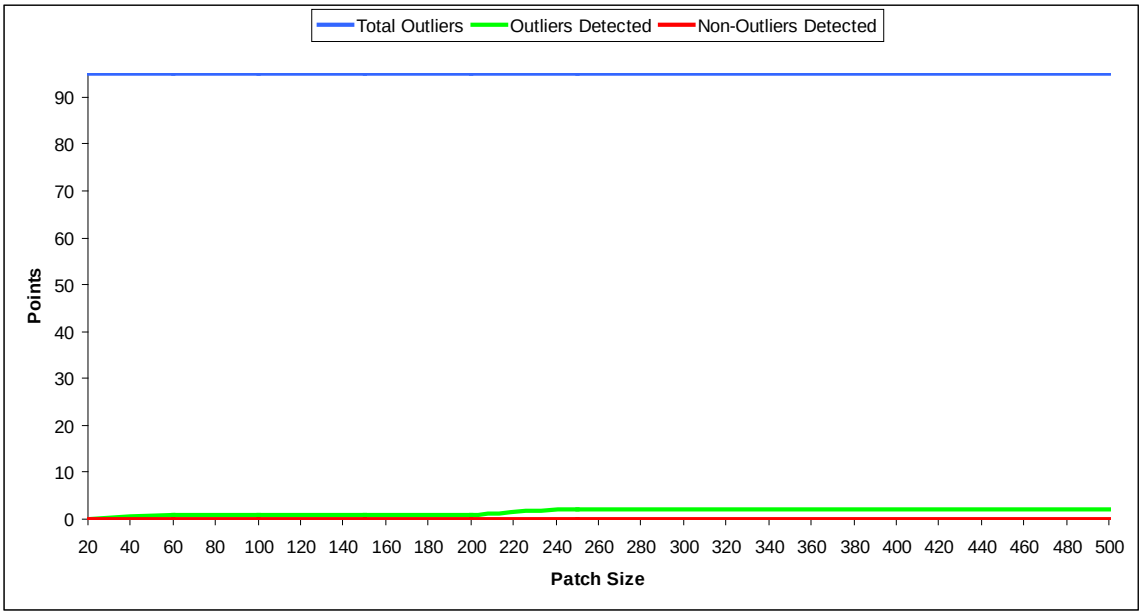

Figure 5.18: Results of varying patch size in the spatial series approach (routine (b)) using point cloud A3.

Figure 5.18 shows the results of varying the spatial series polynomial patch size in point cloud A3 with respect to the number of outliers and non-outliers found. Similarly, Figure 5.19 shows the results of varying the spatial series polynomial patch size in point cloud C3 with respect to the number of outliers and non-outliers found. These results are discussed in Section 5.5.2.

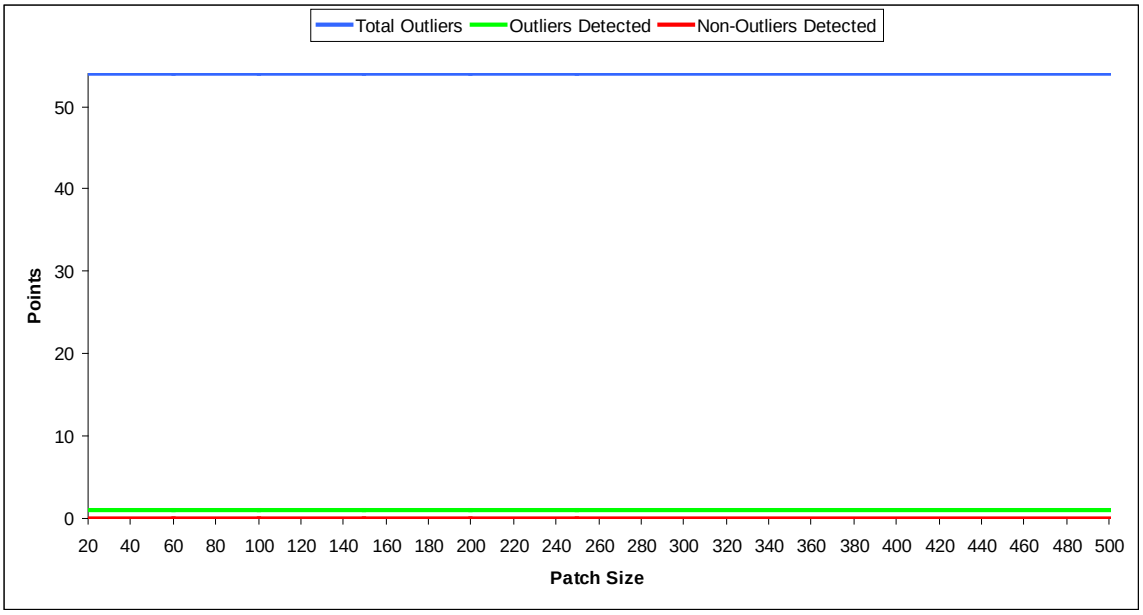

Figure 5.19: Results of varying patch size in the spatial series approach (routine (b)) using point cloud C3.

5.4.3.3 Test 3 – Maximum Number of Outliers Detectable by the Routines

As mentioned at the beginning of Section 5.4, trials to determine the individual and combined effectiveness of the outlier methods presented in Section 5.2 were conducted on the data collected in Washington, D.C.

Table 5.13: Best results from trials conducted using algorithm (a) on point clouds A3 and C3.

Point Cloud	A3	C3
Window Size (points)	200	200
No. of Outliers Identified	17	0
No. of Non-Outliers Identified	0	0
No. of Outliers Missed	78	54
% of Outliers Identified	17.89	0.00
% of Point Cloud Identified	0.03	0.02
% of Point Cloud Identified Incorrectly	0.00	0.00

The results from tests conducted using point clouds A3 and C3 are given in Table 5.13 for algorithm (a) and in Table 5.14 for algorithm (b).

In addition, a combination of algorithms (a) and (b) was performed, where the reduced point cloud produced by algorithm (a) was input to algorithm (b). The results for this test conducted using data strips A3 and C3 are given in Table 5.15. These results are discussed in Section 5.5.3.

Table 5.14: Best results from trials conducted using algorithm (b) on point clouds A3 and C3.

Point Cloud	A3	C3
Patch Size (points)	250	500
No. of Outliers Identified	2	1
No. of Non-Outliers Identified	0	0
No. of Outliers Missed	93	53
% of Outliers Identified	2.11	1.85
% of Point Cloud Identified	0.03	0.02
% of Point Cloud Identified Incorrectly	0.00	0.00

Table 5.15: Results from trials conducted using algorithm (a) preceding algorithm (b) on point clouds A3 and C3.

Point Cloud	A3	C3
Window Size (Routine A)	200	200
Patch Size (Routine B)	250	500
No. of Outliers Identified (Routine A)	17	0
No. of Outliers Identified (Routine B)	0	1
No. of Non-Outliers Identified (Routine A)	0	0
No. of Non-Outliers Identified (Routine B)	0	0
No. of Outliers Missed	78	54
% of Outliers Identified	17.89	1.85
% of Point Cloud Identified	0.00	0.00
% of Point Cloud Identified Incorrectly	0.00	0.00

5.4.3.4 Test 4 – Using Commercial Software to Detect Outliers

As mentioned at the beginning of Section 5.4, the two data sets collected in the streets of Washington D.C. were loaded into Polyworks IMSurvey and the “reject outlier” routine available from the “wrap mesh” function was used to isolate outliers in the two point clouds.

The results from these tests, conducted on the point clouds obtained in Washington D.C., are listed in Table 5.16 below. The parameters used in the Polyworks routine are also included in Table 5.16. These results are discussed in Section 5.5.4.

Table 5.16: Results from trials conducted using Polyworks IMSurvey's reject outliers routine on point clouds A3 and C3.

Point Cloud	A3	C3
Max Spot Space Measured (m)	0.100	0.102
Min Spot Space Measured (m)	0.014	0.022
Max Point-to-Point Distance Used (m)	0.110	0.150
Maximum Cluster Size Used (m)	1.000	1.000
No. of Outliers Identified	93	34
No. of Non-Outliers Identified	44627	40023
No. of Outliers Missed	2	20
% of Outliers Identified	97.89	62.96
% of Point Cloud Identified	12.94	13.76
% of Point Cloud Identified Incorrectly	12.91	13.75

5.5 Discussion

The overall results delivered by the tested routines are varied. The ability to get all of the identified outliers while eliminating the minimum number of non-outlying points is not an easy task to perform. One interesting note that was observed about all the data sets used to test these algorithms is the prevalence of the number of outliers detected in the plane of the road. Looking at the scan pattern of each road segment, the existence of disjointed line segments and discontinuous pattern distortions are observed. Partly, these observed changes in pattern can be explained by the road surface variation and terrain feature changes. Partly, these pattern changes can be explained by vehicle acceleration and deceleration.

However, the pattern discontinuities in the point clouds cannot be full explained by either of these explanations. An example of a pattern discontinuity is given in Figure 5.20.

Inquiry into why this type of discontinuity occurred revealed that the manufacturer has built into the Lynx system, an automated cycle sync between the two LiDAR scanners. If one LiDAR scanner is deemed to be moving appreciably faster or slower than the other scanner, than the speed of the offending scanner will be adjusted. Therefore the values enter by the user into the Lynx system at the time data is being collected are only nominal speed values for the mirror motion. These inconsistencies within the pattern of the point cloud had to be addressed for the temporal series by limiting the focus of the detection routine to vertical estimation. When horizontal estimation is left in place the result is much the same as the commercial software.

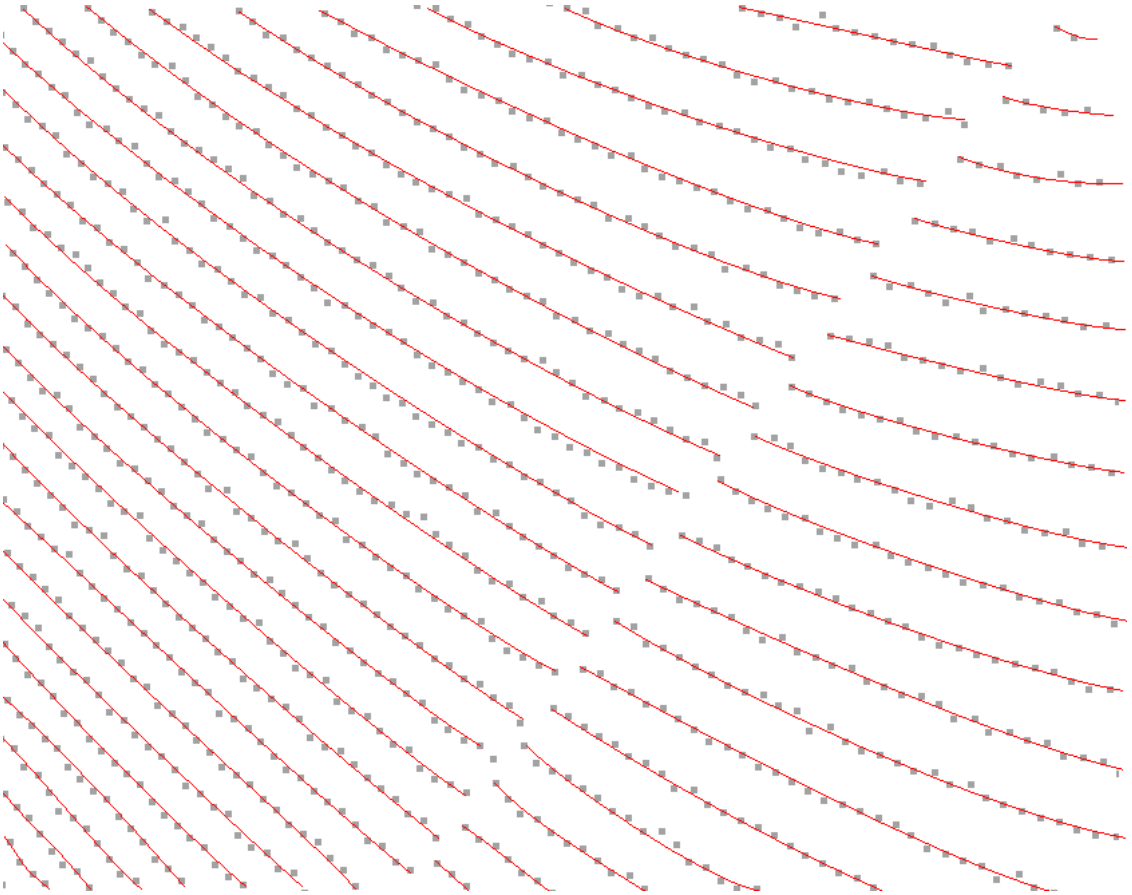

Figure 5.20: Close up image of scan pattern from point cloud B1 showing scan pattern discontinuity. The scan pattern has been enhanced with red lines tracing scan lines.

Another important detail involves the estimated standard deviation of the points in the LiDAR point cloud. Using estimates for the overall position of the DG system, the error estimates used in both routines vary from 1cm to up to 14cm. Figures 5.21, 5.22 and 5.23 show the point error estimates used during testing for each point cloud.

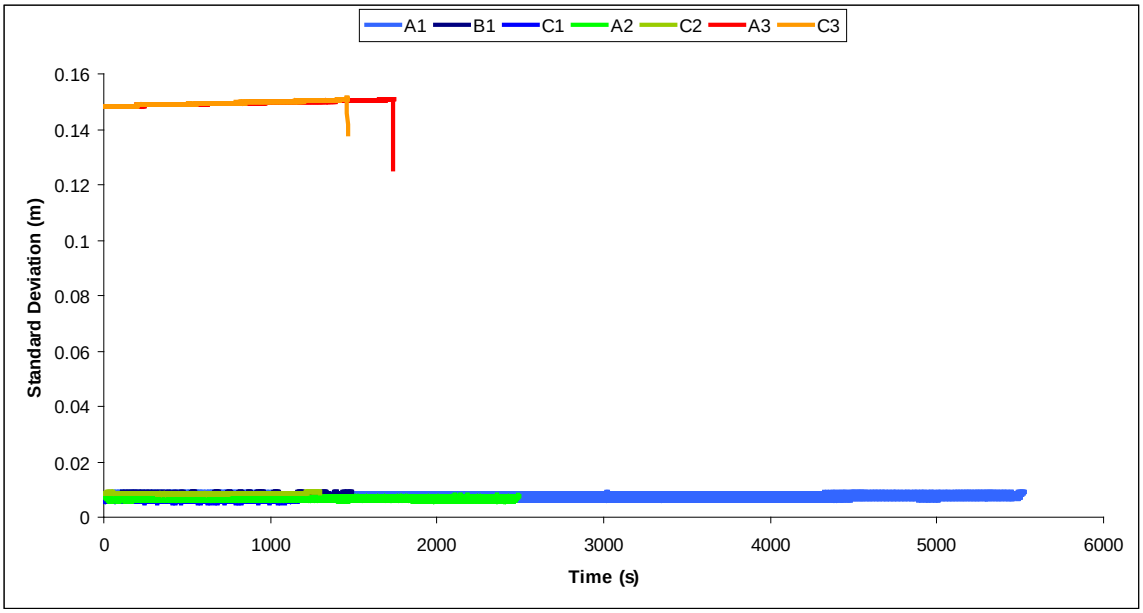

Figure 5.21: Easting error estimates for point clouds used during algorithm testing.

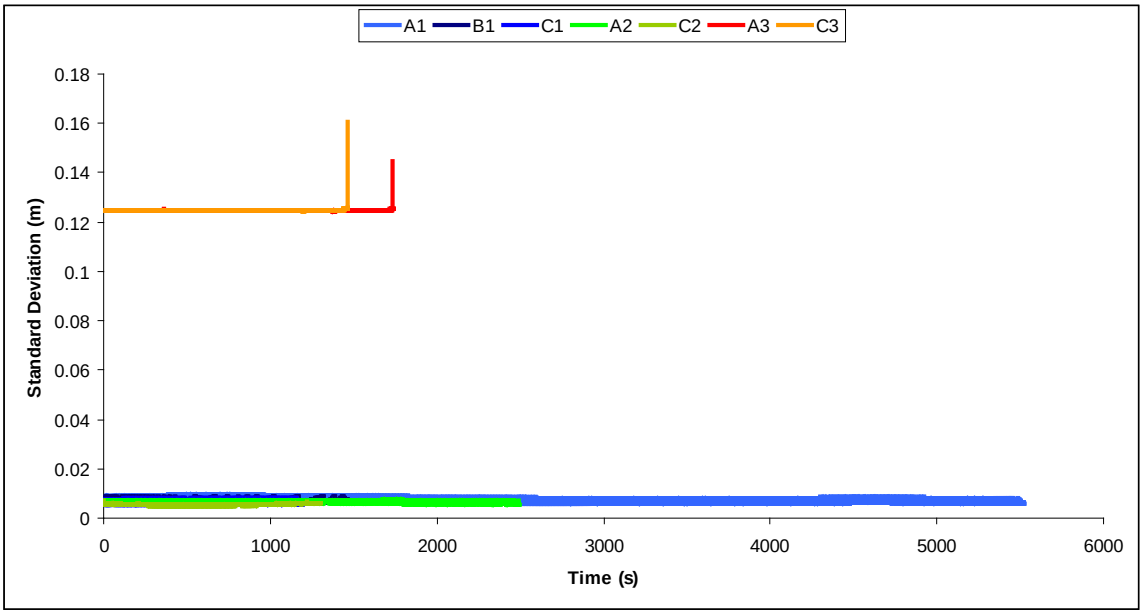

Figure 5.22: Northing error estimates for point clouds used during algorithm testing.

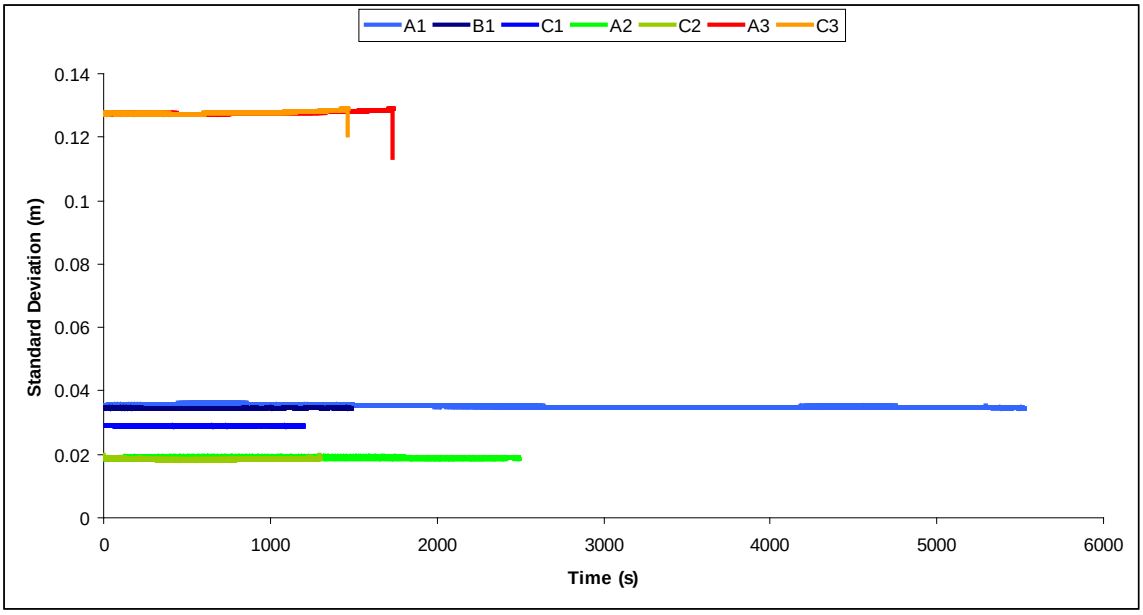

Figure 5.23: Up error estimates for point clouds used during algorithm testing.

These estimates are largely concerned with the accuracy of the GPS derived position of the vehicle at the time the points in the point cloud were collected. The LiDAR error estimates currently being used are very small, between 0.1mm and 3mm. With no boresight error estimates available, the only alternative would be to input these error estimates into the models, however, this will result in a host of non-outlier points to be selected for removal, since this value is much smaller than the observable noise in the data set.

5.5.1 Results of Window Size Determination in the Time Series Approach to Outlier Detection

The graphs for the three trial areas show a wide variation in the growth of detected outliers as window size increases. The intersection at which the number of points incorrectly identified as outliers crosses the number of true outliers detected by the algorithm also varies widely. The graphs from Sections 5.4.1.1, 5.4.2.1 and 5.4.3.1 show

that the time series approach was most effective on point cloud B1, with the majority of outliers found for all window sizes tested. These same graphs also show that this approach was least effective on point cloud C3, since no outliers were detected in this point cloud at all. If the image of point cloud B1 from Figure 5.4 is compared to the image of point cloud C3 in Figure 5.15, it can be immediately noticed that many of the outliers in point cloud B1 are obvious, while those in point cloud C3 are not.

Point cloud B1 has both individual outliers and small clumps of outliers standing at significant distances from the road surface. Most of outliers identified by the time series approach in point cloud B1 are the individual outliers, while the clumps of outliers had only some of their number detected. The outlier clumps in point cloud B1 differ from the outlier clumps in point cloud A1, in that the average point to point difference in point cloud B1 is greater than the error estimates used in the algorithm. The opposite is true for point cloud A1. The median point to point difference for the outlier clumps is about 4cm, while Figures 5.21, 5.22 and 5.23 show that the error estimates used in the model vary between about 1cm and 3cm. Where the point to point distance is less than the 3cm error estimate used, outlier points in the clumps were missed.

The outliers in point cloud C3 are all close to the road surface itself. The failure of the routine to detect any outliers here is easily explained once the error estimates for the point clouds are examined. Figures 5.21, 5.22 and 5.23, which show the error estimate used by the time series routine, indicate that error estimates in the 12cm to 15cm range, for each coordinate component (x, y, z), were used by the time series routine. These error

estimates were completely derived from the DG system error estimates, since the downtown core of Washington D.C. provided poor GPS coverage. This poor GPS coverage greatly affected the DG system solution. The fact that all of the outliers in point cloud C3, fall much closer to the road surface than 12cm explains why no outliers were detected.

Similar to point cloud C3, point clouds A2 and C2, taken in France have their outlier points close to the road surface. Most of these points fall within less than 2cm from the road surface and therefore were not identified due to the fact that the largest error estimate used in the algorithm was about 2cm.

5.5.2 Results of Patch Size Determination in the Spatial Series Approach to Outlier Detection

The graphs plotting the result of the spatial algorithm as patch size was increased also show a wide variation of performance as polynomial patch size is increased. The number of erroneously selected outliers was, compared to the previous algorithm, significantly lower. The graphs from Sections 5.4.1.2, 5.4.2.2 and 5.4.3.2 show that the spatial series approach was most effective on point cloud A2. Similar to the time series performance on point cloud B1, the spatial series found the majority of outliers for all window sizes tested on point cloud A2. These same graphs also show that this approach was least effective on point cloud C3, since almost no outliers were detected in this point cloud at all. Comparing the image of point cloud A2 (specifically the view depicted in B2) from Figure 5.10 to the image of point cloud C3 in Figure 5.15, it can be seen that the outliers

in A2 are close to the surface but noticeable, while the outliers in C3 are scattered and hard to see.

Point cloud A2, for which the time series gave an underwhelming performance and the spatial series gave a good performance, consists of relatively well spaced outliers close to the road surface. To understand why this point cloud worked well using the spatial series algorithm, the chi squared fit statistic must be examined for the calculated surfaces. The chi squared fit statistic indicates that for point cloud A2, up to 0.16% of all calculated surfaces did not fit the data well. The same statistic indicates that up to 0.42% of all calculated surfaces in point cloud B1 provided for poor fits to the data. This is nearly 3 times poorer surface fits in point cloud B1 than A2. For both models, the actual number of surfaces which failed to be calculated, due to the formation of singular matrices, were very low (< 10). Furthermore, the number of poor fit surfaces which coincided with detected outliers remains very low (less than 2%) for most patch sizes in point cloud A2. Point cloud B1, on the other hand, has a much higher number (between 3% and 26%) of outliers detected with poorly fitting surfaces. It makes sense that where outliers cluster or clump together, the quality of a surface fit around any one point would suffer from the other outliers, which lie in close proximity.

The spatial series algorithm preformed just as poorly as the time series algorithm on point clouds A3 and C3. Again the high standard deviations provided by the DG system position proved the downfall of the routine. Most likely due to the large standard deviations feed into the routine from the DG system, the number of surfaces which failed

due to ill conditioned matrices was excessive. For small patch sizes (~20 points) the number of failed surfaces was as high as 69% of the total number of surfaces computed. For larger patch sizes (> 60 points) the number of failed surfaces averaged out at about 2% of the total number tried. Except for the high error estimates on the individual point positions, point clouds A3 and C3 are ideally suited for the spatial series routine. Each of these point clouds is flat with few outliers, well-spaced apart. What this failure dramatically demonstrates is that when poor DG system position errors are included with the individual point error estimates, the success or failure of both the temporal and spatial routines is completely dependent on the quality of the trajectory solution.

5.5.3 Results of Maximum Number of Outliers Detectable by the Routines

Using the graphs generated by the last test, the optimal window (time series) and patch (spatial series) sizes were determined. Using these empirically optimized results the statistics listed in Sections 5.4.1.3, 5.4.2.3 and 5.4.3.3 were generated. These results indicate that between the two algorithms over 70% of the outliers were found in two of the test cases. Point clouds B1 and A2 each had over two thirds of their outliers identified or removed by one of the two routines. Each of these point clouds contain individually spaced outliers separated from the surface, and other outliers, by a distance greater than the input error estimates. It appears that algorithm (a) does better when the outlier points are farther from the legitimate point cloud points. It also appears that algorithm (a) is not bothered too much by clumps of outliers. Being that the algorithm is examining the point cloud scan line by scan line, this is not surprising. Only outliers lying in the same scan line as a current test point will affect the quality of the comparison.

Algorithm (b) is not affected as much by points that lie at distances from the surface, which are just outside the error estimate limit. It does seem to have a problem with clumps of outliers however. Outliers in close proximity to each other tend to disrupt the fit of the polynomial surface causing inaccurate models for comparison with outlier points.

By comparison, point clouds C1 and C2 also contain outliers spaced at an appreciable distance from the point cloud and each other, however, this distance happens, more often than not, to be less than the input error estimates. The result of this is that the two routines find only about a quarter of the identified outliers.

Point clouds A3 and C3, because of the extremely high error estimates from the DG system, give poor results from both routines. Algorithm (a) gives better results in point cloud A3, with nearly a fifth of the outliers detected, otherwise these point clouds show almost no outliers detected by either routine.

Finally, running the two algorithms in sequence, algorithm (a) followed by algorithm (b) showed that many of the points detected by one routine, had also been found by the other routine when the algorithms had been run separately. This is shown for each test case since the results of running the algorithms in series is only marginally better than running them individually. In each case, the results of running the algorithms together was about 3% better than the best outlier detection result obtained by running the routines separately.

5.5.4 Results of Using Commercial Software to Detect Outliers

As a control test the outlier removal tool, which is incorporated with the "warp mesh" function in Polyworks IMSurvey, was used on the same point clouds discussed in the previous sections. The statistics listed in Sections 5.4.1.4, 5.4.2.4 and 5.4.3.4 clearly show that the commercially available routine detected approximately the same number of outliers as algorithm's (a) and (b). Indeed, the commercial routine accomplished this while also removing vastly more non-outlier points. In fact, in the case of point cloud A1, the routine removed over 15% of the total point cloud. The manually identified outliers in this point cloud only constitute about 1% of the point cloud. This means that over 14% of the total point cloud has been unnecessarily removed. The same large removal of non-outliers can be said for all the other point clouds.

Point clouds A3 and C3, which produced the worst results from algorithms (a) and (b), were much better behaved for the commercial routine. The Polyworks "remove outlier" routine was able to remove 98% of the manually identified outliers from point cloud A3 and 63% of the manually identified outliers from point cloud C3. Of course this was accomplished at a cost of removing around 14% of the point cloud in both scans. Most of the removed points were not outliers and were removed unnecessarily. Since the commercial routine is relying on user input for spot spacing and cluster size and is not dependent on the point error estimates generated from the system itself, the commercial routine appears to have had an advantage over algorithms (a) and (b) in outlier determination in point clouds A3 and C3.

5.6 Summary

In this chapter, methods for performing outlier detection based on both the temporal and spatial information available from most MTL systems, were proposed and tested. The new method of temporal outlier detection involved modifying a α-β-γ Kalman smoother to predict the location of a point based on the location of other points in the time series and then performed a statistic test to evaluate if the difference between the predicted point and the actual point qualified the point as an outlier. The main benefits of using such a method of outlier detection is that its fast and can, if desired, be executed in real-time, while the data is being collected. Tests on three different MTL data sets showed that this algorithm could be up to 71% effective at finding outliers if the time series interval was maintained. Compared to commercial software used on the same data sets, the temporal outlier algorithm proposed and tested in this chapter, could have comparable effectiveness with far less false detections, under the right circumstances.

The new spatial methods for detecting outliers are based on a 10 parameter polynomial surface model. Two methods based on this polynomial surface were proposed and tested. The first method computed a small polynomial patch in the immediate vicinity of a test point and used a statistic test to compare the candidate point with the surface. This method assumes all the points used in the surface fitting are good points, not outliers. The second method involved using many more point cloud points to fit a surface model to a larger section of the point cloud. The residuals of the surface fitting would then be tested and any outlier removed. The surface fitting would then be iteratively run until all points whose residuals registered as outliers were removed. By its nature, this method

assumed multiple outliers would likely be present in one surface fitting. Tests on three different MTL data sets showed that the small polynomial patch method could be up to 70% effective under the right circumstances and that the large polynomial patch method could be over 80% effective under the right circumstances. In both cases the polynomial surface fitting found a comparable number of outliers to the commercial software used while returning less false detections. Combining the temporal and spatial algorithms by first running the data through the temporal algorithm and then running the reduced data set through the small patch spatial algorithm produced slightly better results than those achieved by running the data through only one algorithm.

Having developed and tested methods for detecting the outlying observations from the MTL point cloud data and having developed and tested methods for calibrating the MTL hardware in Chapter 4, it is time to apply all these techniques to real MTL data. To this end, the next chapter discusses the results of two trials conducted on real MTL data. These trials were conducted on data collected in real situations, one around a commercial office building and the other along a typical street scene.

6. Improving the Accuracy of MTL Point Cloud Data

In the previous chapters, several techniques have been presented to improve the accuracy of MTL point clouds. Chapter 4 dealt with methods for improving the measurement accuracies of the range finder within the LiDAR sensor itself and the calibration of the LiDAR sensor to the vehicle's onboard DG system. Chapter 5 dealt with identifying and removing inaccurate point cloud elements. A method for evaluating the individual point cloud elements was presented and tested in Chapter 3. By combining the techniques described in these chapters, clean accurate point clouds may be produced.

In this chapter, a procedure for improving the quality of the LiDAR point clouds from MTL systems is outlined, and then the techniques developed in Chapters 3 to 5 are demonstrated on two MTL data sets. The point clouds are first evaluated before processing by direct comparison to control, and then using the conditional variance analysis in Chapter 3. The zero error and temperature correction developed in Chapter 4 are then used to adjust the measured ranges reported by the LiDAR sensors. Next, the boresight and lever arm calibration from Chapter 4 is performed. Finally, the α-β-γ smoother from Chapter 5 is applied to the data to identify and remove outliers. Once complete, the point clouds are again evaluated against control and through the use of conditional variance analysis, to gauge the effectiveness of this procedure on real MTL data.

6.1 Procedure for Improving the Accuracy of MTL Point Clouds

The first step to improving a LiDAR point cloud is to understand how accurate the current point cloud is and which variables are producing the greatest sources of error.

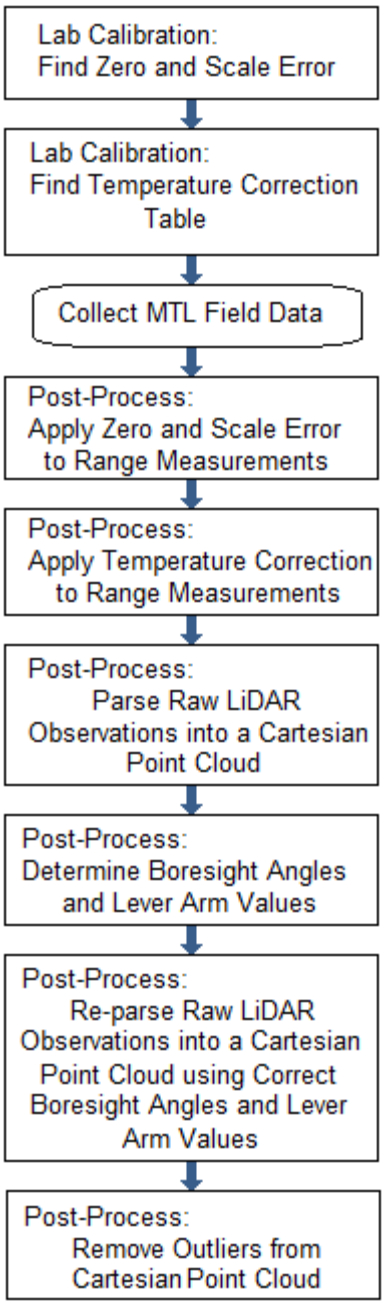

Figure 6.1: Procedure for MTL accuracy improvement.

Using residual analysis combined with the conditional variance analysis approach outlined in Chapter 3, the current state of the MTL calibration can be evaluated. After that evaluation is complete, the procedure outlined in Figure 6.1 can be followed to calibrate and process the MTL data. Once the procedure is complete, residual and conditional analysis can be performed again to evaluate and compare the data to its previous state.

An initial processing of the data needs to be performed, using estimates for the boresight and lever arm values. The better these estimates the faster the boresight and lever arm calibration from Chapter 4 will operate, however, generic values which are several degrees and up to a meter offset from their true values can be used. Once processed, extraction from the point clouds of common features between each sensor pair and any available control points is performed. These common and control points, along with the initial estimates for the boresight and lever arm values, are introduced to the calibration from Chapter 4 and updated estimates for the boresight and lever arm values are produced. These updated estimates are used to re-process the raw MTL data into new point clouds.

These new point clouds are then subjected to the α-β-γ outlier detector discussed in Chapter 5. Points are identified by this outlier detector and removed. Once the outliers have been removed, residual analysis and the variance component analysis of Chapter 3 can be re-run on this data so that it may be compared to its initial state.

6.2 Application to Real MTL Data

6.2.1 Data of a Commercial Office Building

Figure 4.13 in Chapter 4 shows a commercial office building. Table 4.16 in Chapter 4 lists the control points that were surveyed on prominent features of that building. MTL data of this building was collected again using the same LiDAR sensors calibrated for zero error and temperature drift in Chapter 4. Zero error and temperature calibration is required to ensure that the LiDAR sensors are reporting the most accurate ranges possible. To make use of the boresight and lever arm calibration method introduced in Chapter 4, it was important that at least one pair of LiDAR sensors were used and control data is available.

Figure 6.2 shows the data collected around the office building. Intensity contrast in the point clouds for both Sensor 1 and Sensor 2 was verified before testing was performed to ensure that control points located at the edges of parking lines were usable.

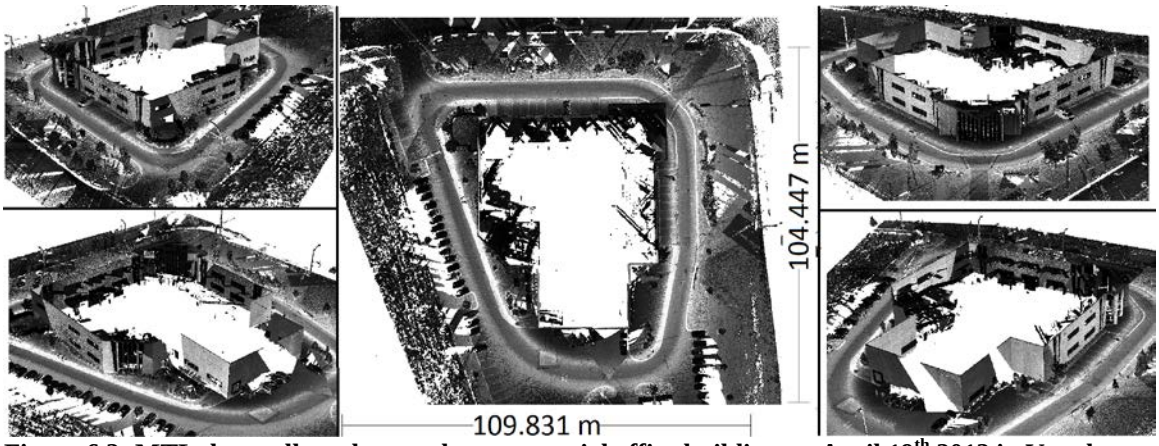

Figure 6.2: MTL data collected around a commercial office building on April 19th 2012 in Vaughan Ontario.

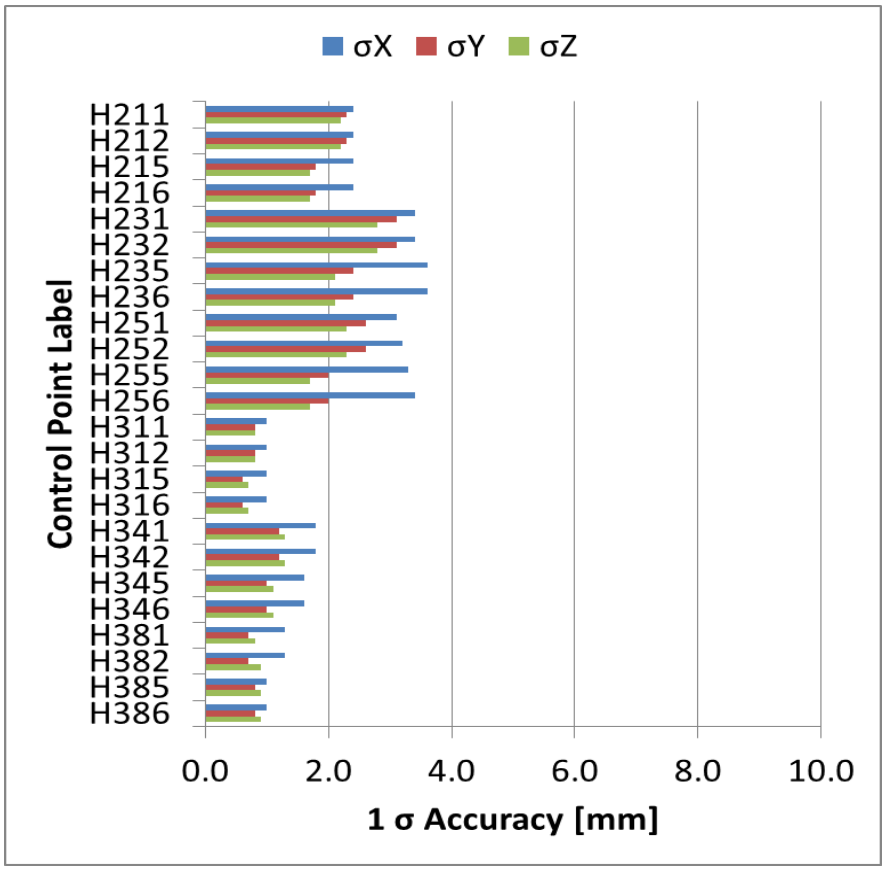

Figure 6.3: One-sigma accuracy estimates for control points surveyed on the horizontal surfaces of the building.

The control field in Table 4.16 was expanded to 74 control points for the purposes of estimating the absolute accuracy of the point cloud. The process is similar to that described by Figures 3.1 and 3.2 in Section 3.3 of Chapter 3. The points in the expanded control field were surveyed at the same time as those in Table 4.16 which were surveyed by traversing around the target site with a Leica TC1800 (1", 1mm + 2ppm) total station. ECEF coordinates were established by post processing static GPS observations of the control traverse and referencing them to the same base station as the MTL data. The static GPS was collected using a Leica 1200 GPS receiver (5mm + 0.5ppm (horizontal), 10mm + 0.5ppm (vertical)).

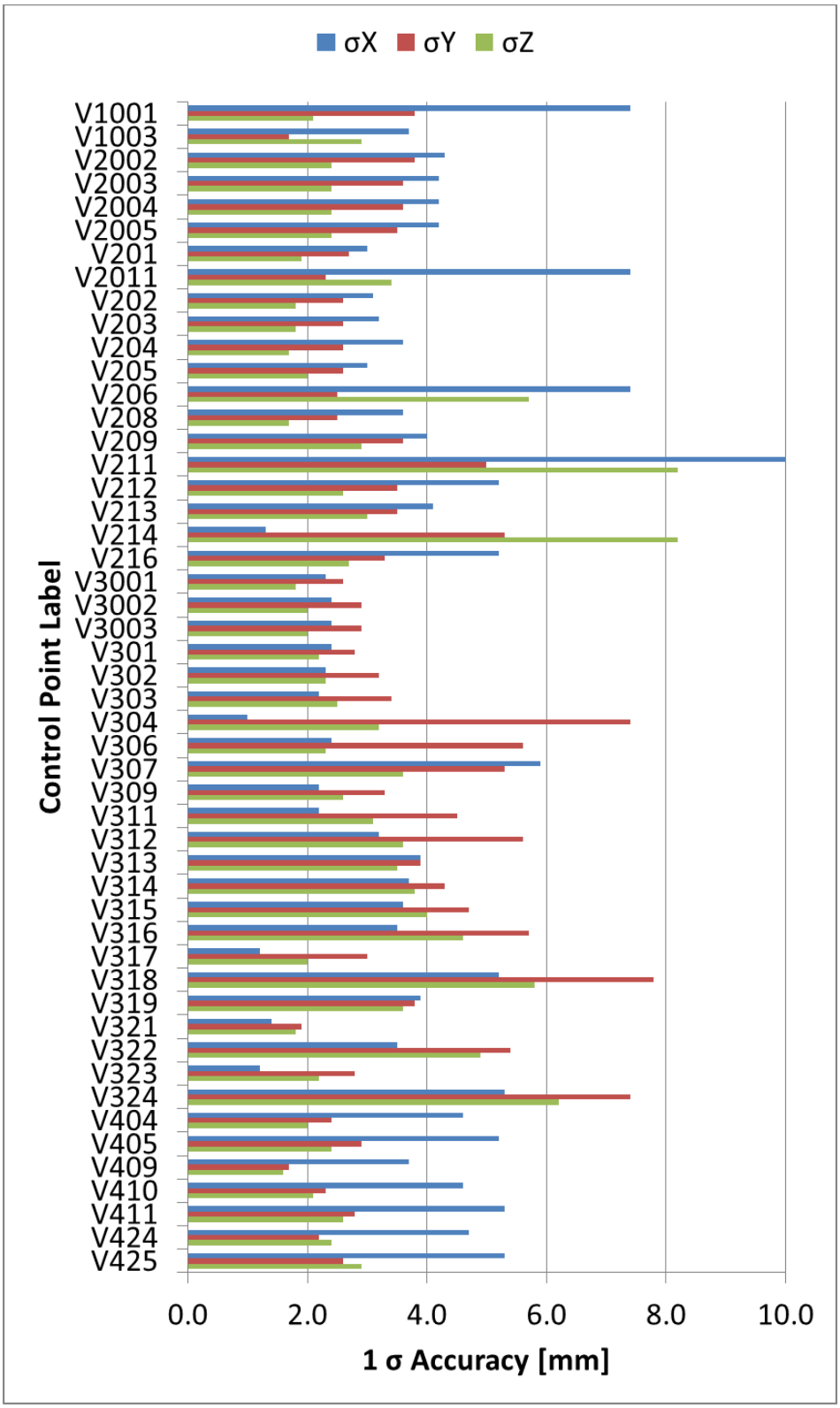

Figure 6.4: One-sigma accuracy estimates for control points surveyed on the vertical surfaces of the building.

Figure 6.3 lists all the control points used on the ground surfaces in the parking lot of the commercial office building and shows the one sigma standard deviation estimates for these points. Similarly, Figure 6.4 lists all of the control points used on the vertical surfaces of the commercial office building and graphically displays the calculated one sigma standard deviation estimates for these points. For the purposes of calibrating the two LiDAR sensors to the DG system, two control points on vertical surfaces and two control points on parking lot lines were extracted from the East, North and West sides of the building. These control points were randomly selected from the control field and apportioned equally to the individual sensors.

6.2.1.1 Assessing the Quality of the Data before Processing

Point primitives were used to extract 50 building features and 24 parking lot lines to which control points have been associated. These extracted LiDAR points were then compared to the associated control points to compute check point residual values for the point cloud. Figure 6.5 shows the results of computing the check point residuals from the point clouds of both LiDAR Sensor 1 and LiDAR Sensor 2 and the control in the parking lot. The check point residuals in Figure 6.5 are broken down into their horizontal and vertical components. It was found that the horizontal component of the check point residuals in the parking lot for Sensor 1 had a mean average of 0.061m and a standard deviation of 0.055m. For Sensor 2, the horizontal component of the check point residuals in the parking lot had a mean average of 0.058m and a standard deviation of 0.056m. The vertical component of the check point residuals in Figure 6.5 had a mean average of 0.031m and a standard deviation of 0.031m, for Sensor 1. The vertical component of the

check point residuals for Sensor 2 had a mean average of 0.020m and a standard deviation of 0.027m.

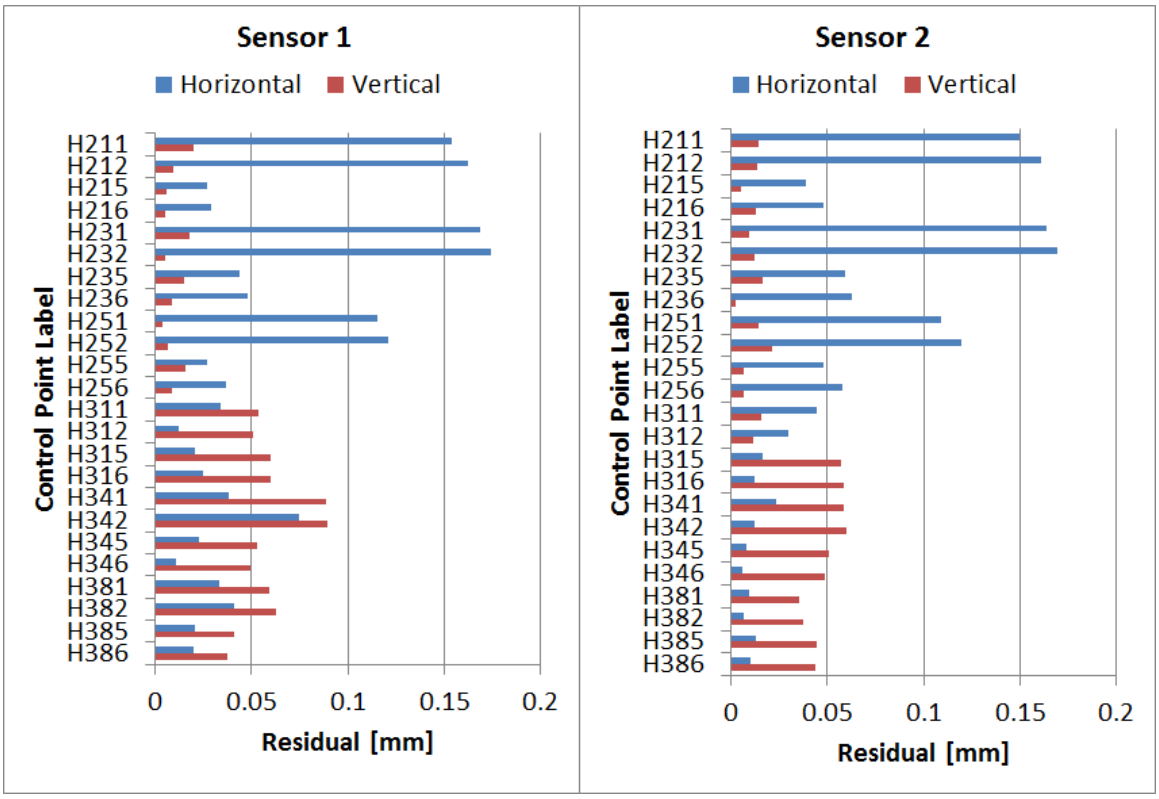

Figure 6.5: Check point residuals computed from point primitives extracted from both sensor's LiDAR point clouds and compared to the ground control points associated with parking lot lines.

A similar analysis was conducted on the 50 building features that were extracted using point primitives. Check point residuals were created by comparing the point primitives extracted from the LiDAR point clouds to the control points for both LiDAR sensors in the MTL system (Figure 6.6). As was done with the check point residuals from the parking lot lines, the check point residuals from the building features were segmented into their horizontal and vertical components.

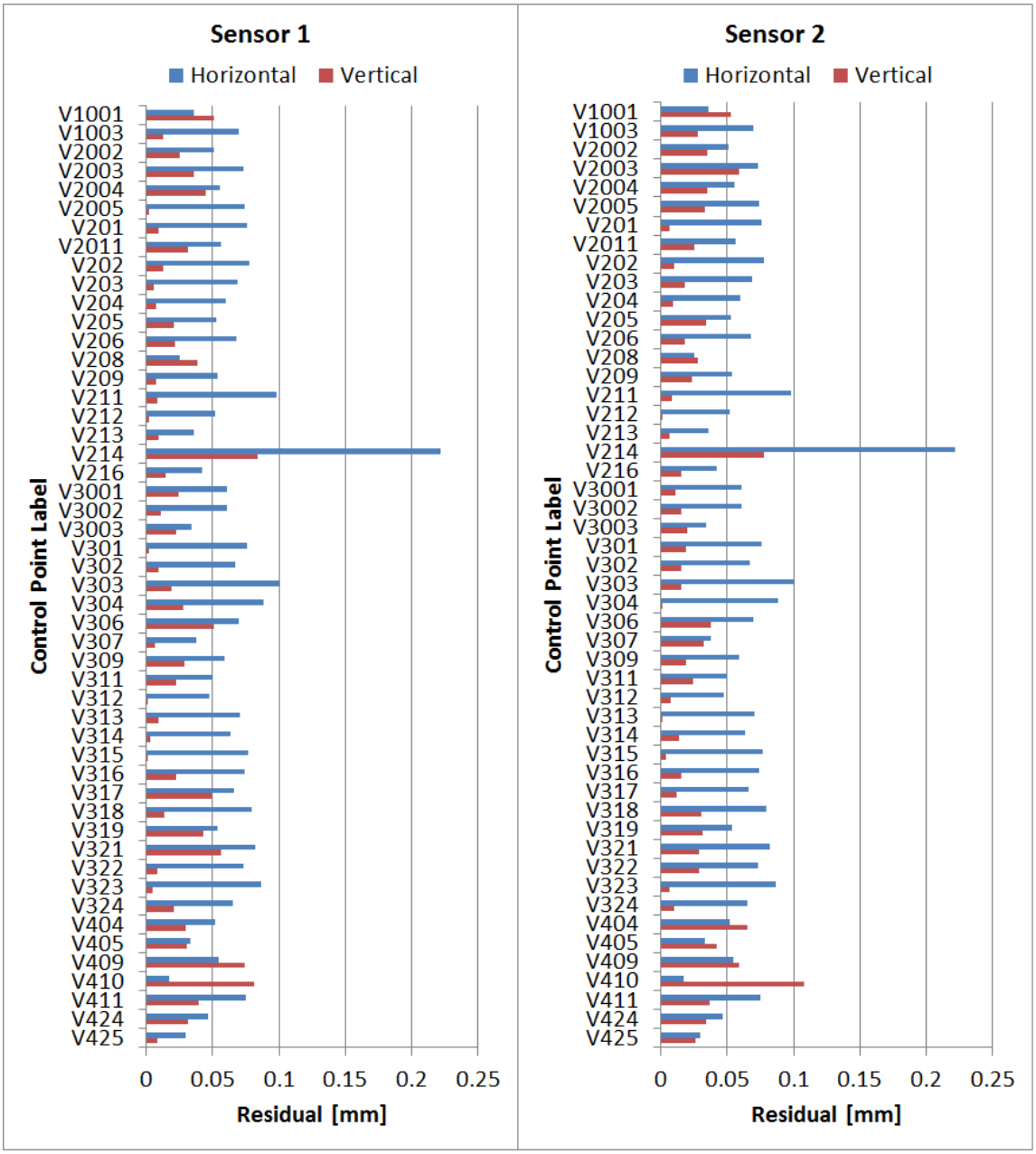

Figure 6.6: Check point residuals computed from point primitives extracted from both sensor's LiDAR point clouds and compared to the control points associated with vertical building features.

The horizontal component of the check point residuals shown in Figure 6.6 produced a mean average of 0.064m and a standard deviation of 0.029m for Sensor 1 and a mean average of 0.065m and a standard deviation of 0.036m for Sensor 2. The vertical component of the check point residuals shown in Figure 6.6 produced a mean average of

-0.002m and a standard deviation of 0.031m for Sensor 1 and a mean average of -0.010m and a standard deviation of 0.032m for Sensor 2.

Chapter 3 also presented the method of variance component analysis as a means of evaluating this data. Chapter 3 described the measurements used in Equation 3.1 to calculate points from LiDAR and DG system measurements. These measurements were grouped based on their dependencies. Group 1 consisted of the DG system position in Cartesian ECEF coordinates (X_{INS}^{ECEF} , Y_{INS}^{ECEF} , Z_{INS}^{ECEF}). The Group 1 parameters are converted from the geodetic latitude (*B*) longitude (*L*) and ellipsoidal height (*ht*) measured by the DG system. Group 2 consisted of the roll, pitch and heading (r, p, h) of the DG system with respect to the local geodetic coordinate frame. Group 3 consisted of the components of the lever arm vector (l_X, l_Y, l_Z) between the DG system and the LiDAR. Group 4 consisted of the x, y and z Euler rotations ($\theta_X, \theta_Y, \theta_Z$) of the LiDAR coordinate frame in the DG system frame. Group 5 consisted of the horizontal angle, vertical angle, range and zero error (α, β, d, K) measured by the LiDAR in the local LiDAR coordinate frame.

Using the same group categories for the LiDAR data presented here and running the data through the conditional variance analysis of Chapter 3, a breakdown of the contribution to the total error from each group of variables is obtained. This breakdown of error is presented here as the colourized LiDAR data shown in Figure 6.7 for Sensor 1 and Figure 6.8 for Sensor 2.

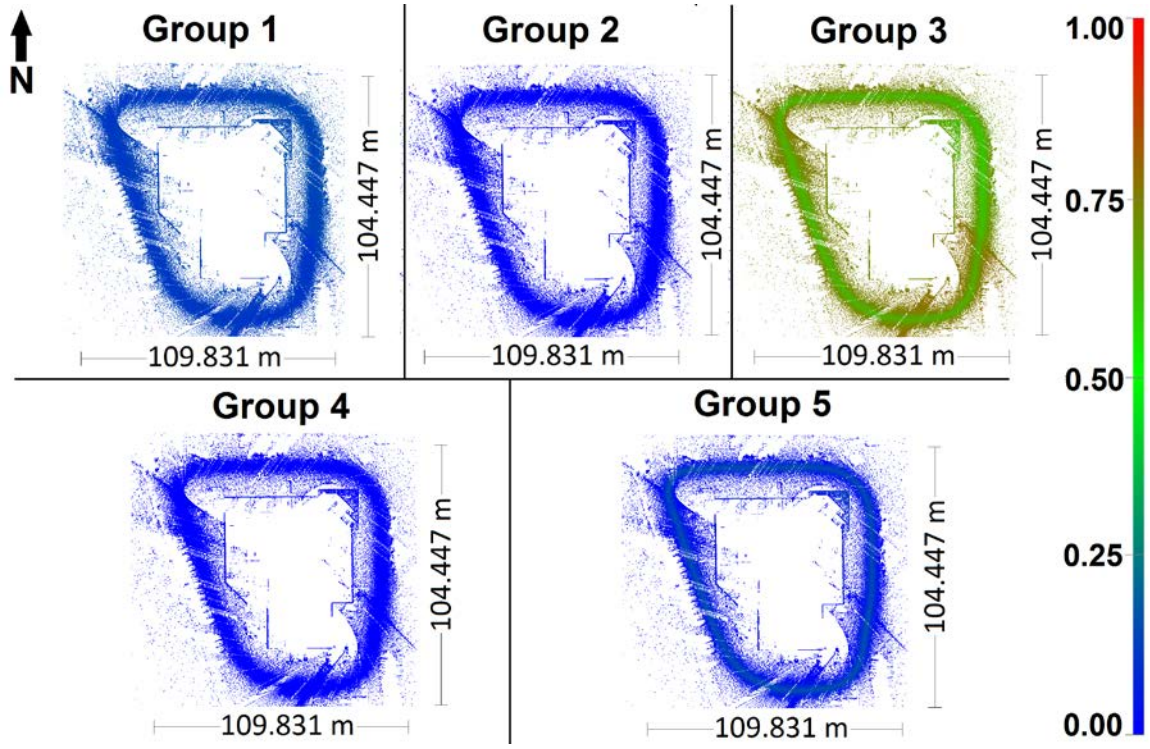

Figure 6.7: MTL data from Sensor 1 of a commercial office building before the quality of the MTL data was improved. The data has been coloured by using an artificial colour scale derived from the total effect indexes from conditional variance analysis.

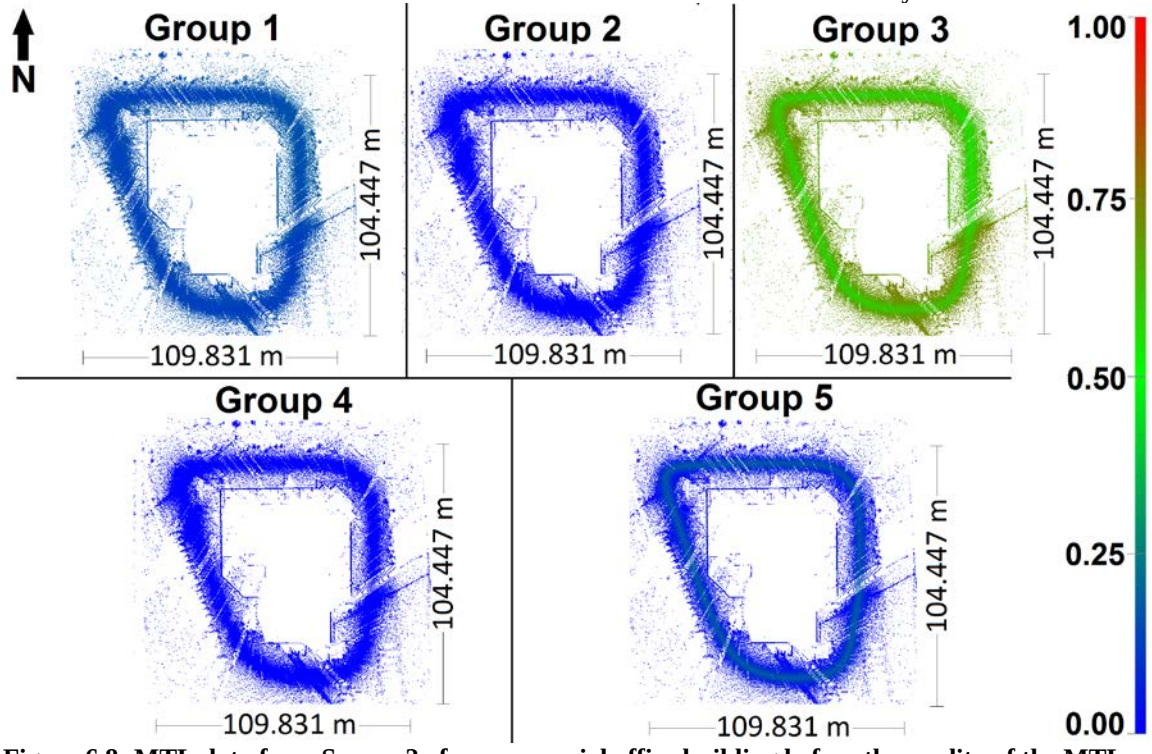

Figure 6.8: MTL data from Sensor 2 of a commercial office building before the quality of the MTL data was improved. The data has been coloured by using an artificial colour scale derived from the total effect indexes from conditional variance analysis.

In Figure 6.7, it is clear that more than 50% of the error in the Sensor 1 point cloud comes from Group 3 which represents the lever arm values. In fact, the areas of the point cloud that show orange in Group 3 represent places were lever arm errors are causing up to 75% of the error. Examining this further, the majority of the error assigned to Group 3 comes specifically from the Z lever arm. Group 1, which represents the DG system positional errors accounts for another 10% of the error in the point cloud. In this case, it is mainly the Z component of the DG system positional errors that account for this result. Group 5 shows a band of increased error significance beneath the vehicle trajectory. This increase is due solely to range error caused by the short ranges measured to the asphalt directly under the vehicle.

Figure 6.8, shows almost identical results for the point cloud from Sensor 2. The most significant contributor to the error in Sensor 2 is Group 3 and more specifically the Z lever arm. The amount of error contributed by this lever arm to the point cloud is between 60% and 75%. Group 1 in Sensor 2 is the second largest contributor to the error in the point cloud (around 10% of the error). Specifically it is the Z component of the DG system position that contributes 10% of the error to the point cloud. The third largest source of error in the close range points (ranges less than 1.5m from the sensor) is the range component of Group 5.

By applying the correction techniques in order the distribution of errors in the point cloud should become more balanced. Currently the vast majority of error is located in the Z lever arm. Correction of the data should see the percentage of error caused by the Z lever

arm reduce, while other variables, such as the DG system positional errors should increase in significance. The first step is to correct the sensor correction values for zero error and temperature drift.

6.2.1.2 Zero Error and Temperature Correction

The sensors used to produce the data shown in Figure 6.2 are the same sensors used in Chapter 4. Figure 4.10 and Figure 4.11 from Section 4.2 show the temperature drift for the two LiDAR sensors. Checking the data collection logs from Sensor 1 it was found that the sensor maintained an internal temperature between 56.2°C and 57.0°C. From Figure 4.10, this translates to a range correction of +0.0004m to 0.0000m. Similarly, the data collection logs for Sensor 2 show that the sensor maintained an internal temperature between 56.1°C and 57.2°C. From Figure 4.11, this translates to a range correction of -0.0002m to +0.0001m. These temperature corrections were applied to the ranges measured by each LiDAR sensor, however, they are vanishingly small therefore likely of little consequence.

The calibration file associated with the MTL lists the zero error for each sensor as shown in Table 6.1. Using the data collected for each sensor as shown in Figure 4.1, the one sigma standard deviations for these zero error values were estimated from the residuals between known target separations and LiDAR measured ranges. These estimated standard deviation values are also shown in Table 6.1.

Table 6.1: Zero error as calculated by the manufacturer.

	Zero Error (m)	STDev (m)
Sensor 1	-0.8715	0.011
Sensor 2	-1.2685	0.005

Adjusting the data according to the method described in Section 4.1, new zero error values and their associated one sigma standard deviation errors were estimated (Table 6.2). The values in Table 6.2 were then applied to the data before moving on to the next step, calibrating the MTL to its DG system.

Table 6.2: Zero error calculated from the method in Chapter 5.

	Zero Error (m)	STDev (m)
Sensor 1	-0.8615	0.002
Sensor 2	-1.2666	0.0013

6.2.1.3 Calibrating the LiDAR to the DG System

Calibration of the MTL system was done using a subset of the control data in Table 4.16 from Chapter 4 and some new control points. Using a set of initial approximations for the boresight and lever arm values, point clouds for each sensor were produced. Based on the results discussed in Chapter 4, Section 4.3.3.3, at least 7 control points will be needed to calibrate the sensor's lever arms. With this in mind, 9 control points from Sensor 1 (Table 6.3), 9 control points from Sensor 2 (Table 6.4) and 22 common points were extracted from these point clouds. Inputting these points into the adjustment will yield 28 degrees of freedom (22 + 9 + 9 – 12). These control points and common points were located on the West, North and East sides of the building and include both horizontal and vertical surfaces.

Running the adjustment for a single iteration, the one sigma standard deviations for the initial approximates to the boresight and lever arms were produced. The initial approximates and the calculated one sigma standard deviations are listed in Table 6.5 for both sensors. The values in Table 6.5 were used as part of the inputs to estimate the total Sobol indexes in Section 6.2.1.1.

Table 6.3: Control data used with Sensor 1.

Target ID	X [m]	Y [m]	Z [m]	σ_X [mm]	σ_Y [mm]	σ_Z [mm]	σ_{3D} [mm]
H215	838508.279	-4534994.819	4391296.377	2.4	1.8	1.7	3.5
H235	838507.915	-4534991.132	4391300.274	3.6	2.4	2.1	4.8
H255	838507.582	-4534987.429	4391304.223	3.3	2.0	1.7	4.2
H315	838477.939	-4534967.735	4391330.728	1.0	0.6	0.7	1.4
H345	838469.350	-4534968.832	4391331.064	1.6	1.0	1.1	2.2
H385	838458.643	-4534970.169	4391331.471	1.0	0.8	0.9	1.5
V204	838500.793	-4534989.516	4391314.848	3.6	2.6	1.7	4.8
V311	838461.206	-4534979.451	4391330.065	2.2	4.5	3.1	6.0
V403	838445.664	-4534997.083	4391317.459	3.7	1.8	1.5	4.4

Table 6.4: Control data used with Sensor 2.

Target ID	X [m]	Y [m]	Z [m]	σ_X [mm]	σ_Y [mm]	σ_Z [mm]	σ_{3D} [mm]
H216	838508.269	-4534994.740	4391296.463	2.4	1.8	1.7	3.5
H236	838507.904	-4534991.060	4391300.357	3.6	2.4	2.1	4.8
H256	838507.568	-4534987.349	4391304.307	3.4	2.0	1.7	4.2
H316	838477.825	-4534967.749	4391330.731	1.0	0.6	0.7	1.4
H346	838469.243	-4534968.848	4391331.071	1.6	1.0	1.1	2.2
H386	838458.531	-4534970.185	4391331.479	1.0	0.8	0.9	1.5
V213	838500.944	-4534995.744	4391299.689	4.1	3.5	3.0	6.1
V313	838478.990	-4534975.214	4391327.488	3.9	3.9	3.5	6.5
V425	838445.626	-4535000.375	4391307.945	5.3	2.6	2.9	6.6

Table 6.5: Boresight and lever arm values and one sigma standard deviations for both sensors before correction.

		Roll (°)	Pitch (°)	Heading (°)	X (m)	Y (m)	Z (m)
Sensor 1	Value	-0.0300	-22.8000	35.9600	0.000	-0.720	-0.164
	STDev	0.0014	0.0017	0.0013	0.011	0.009	0.015
Sensor 2	Value	-0.1600	-24.3000	-37.8600	0.020	0.680	-0.168
	STDev	0.0014	0.0026	0.0015	0.011	0.010	0.014

Table 6.6: Boresight and lever arm values and one sigma standard deviations for both sensors after correction.

		Roll (°)	Pitch (°)	Heading (°)	X (m)	Y (m)	Z (m)
Sensor 1	Value	0.0929	-22.6929	35.7443	-0.007	-0.699	-0.178
	STDev	0.0006	0.0007	0.0005	0.007	0.005	0.008
Sensor 2	Value	0.0344	-24.2598	-38.0934	0.027	0.719	-0.173
	STDev	0.0007	0.0009	0.0007	0.006	0.006	0.008

Allowing the adjustment to run until the solution converged; new boresight and lever arm values were estimated with their associated one sigma standard deviations. These values are listed in Table 6.6. The standard deviations in Table 6.6 indicate the boresight has changed little compared to the boresight in Table 6.5, but there is a significant improvement in the lever arm values between Tables 6.5 and 6.6.

6.2.1.4 Removing Outliers

Outlier removal was implemented to eliminate any stray points with inaccurate ranges due to target properties or angle of incidence of the laser beam. The time series approach from Chapter 5 was used to filter the dual sensor point cloud in Figure 6.2. The time series approach employed the α-β-γ Kalman smoother to predict the position of each point in the time sequence based on a sample of data points both before and after the point occurred in time. Based on the testing done in Chapter 5, shown in such Figures as 5.5, 5.6 and 5.7, it was decided to use a point sample size of 35 points both before and after each test point. Table 6.7 shows the results of this filtering, out of the approximately 25 million points in each sensor; the outlier filter removed around 4 million points as outliers. This represents 16.7% of the Sensor 1 point cloud and 14.6% of the Sensor 2 point cloud.

Table 6.7: Points removed from each sensor's point clouds by the outlier filter based on the α-β-γ Kalman smoother.

	Total No. of Points	No. of Outliers	% of Points Removed
Sensor 1	25,974,313	4,331,211	16.7
Sensor 2	24,945,314	3,648,481	14.6

The filter also failed to return a correct prediction on some points in the point cloud. Most of these points occurred where significant gaps were present in the time series.

Table 6.8 shows a breakdown of these points. They represent 0.5% of the Sensor 1 point cloud and 0.6% of the Sensor 2 point cloud. Being that these points are unchecked, they are assumed to be good points and remain in the filtered point cloud.

Table 6.8: Points that failed to return a result from the outlier filter based on the α-β-γ Kalman smoother and were therefore indeterminate.

	Total No. of Points	No. of Points Not Checked	% of Points Not Checked
Sensor 1	25,974,313	127,610	0.5
Sensor 2	24,945,314	155,361	0.6

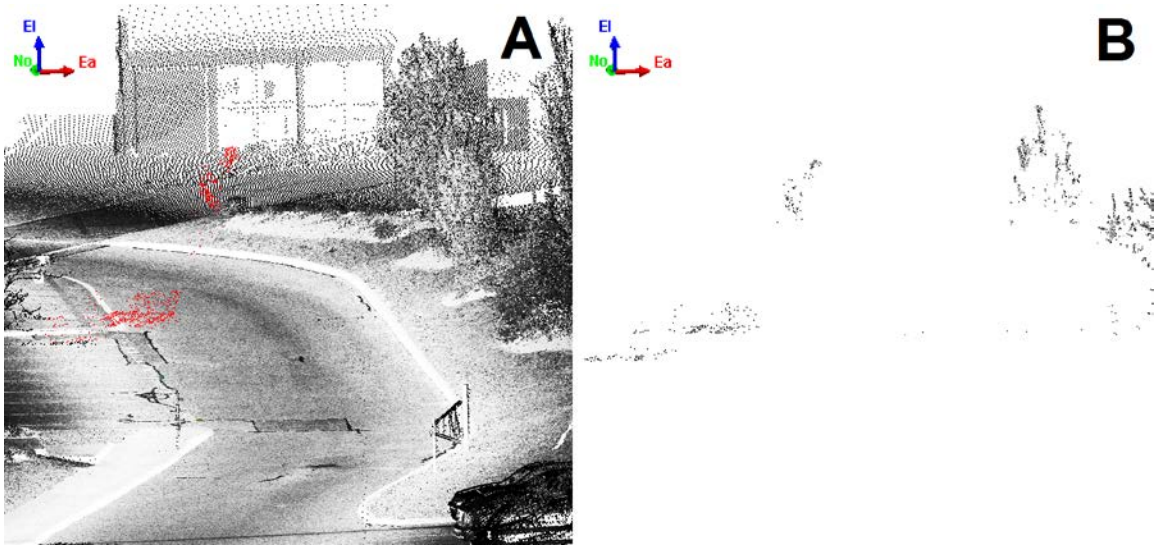

Figure 6.9: Points removed from the MTL data set. Frame A) shows vehicle exhaust highlighted red in the context of the wider point cloud, Frame B) shows the points identified by the α-β-γ Kalman smoother routine and removed from the data set.

Examining the points that were removed from the point cloud, it was found that many of them fall within the leafy parts of the shrubbery surrounding the building or stray points on the lawn. Technically, these points fit the definition of outliers as established in Chapter 5, though most people would consider them good data points. There are a few places around the building where the LiDAR has taken ranges on the vehicle exhaust and produced a cloud above the parking lot. The red points shown in Figure 6.9 indicate vehicle exhaust points correctly identified by the outlier filter and removed. The left side

of Figure 6.9 shows the points in context of the actual scan data, while the right side shows the points identified and removed from the scan in this same area.

6.2.1.5 Assessing the Quality of the Data after Processing

As was done in Section 6.2.1.1, point primitives were used to extract the same 50 building features and 24 parking lot lines. Again, the extracted LiDAR points were compared to the associated control points and check point residual values were computed for each point cloud. Figure 6.10 shows the results of computing the check point residuals in the parking lot and Figure 6.11 shows the results of computing the check point residuals on the extracted building features.

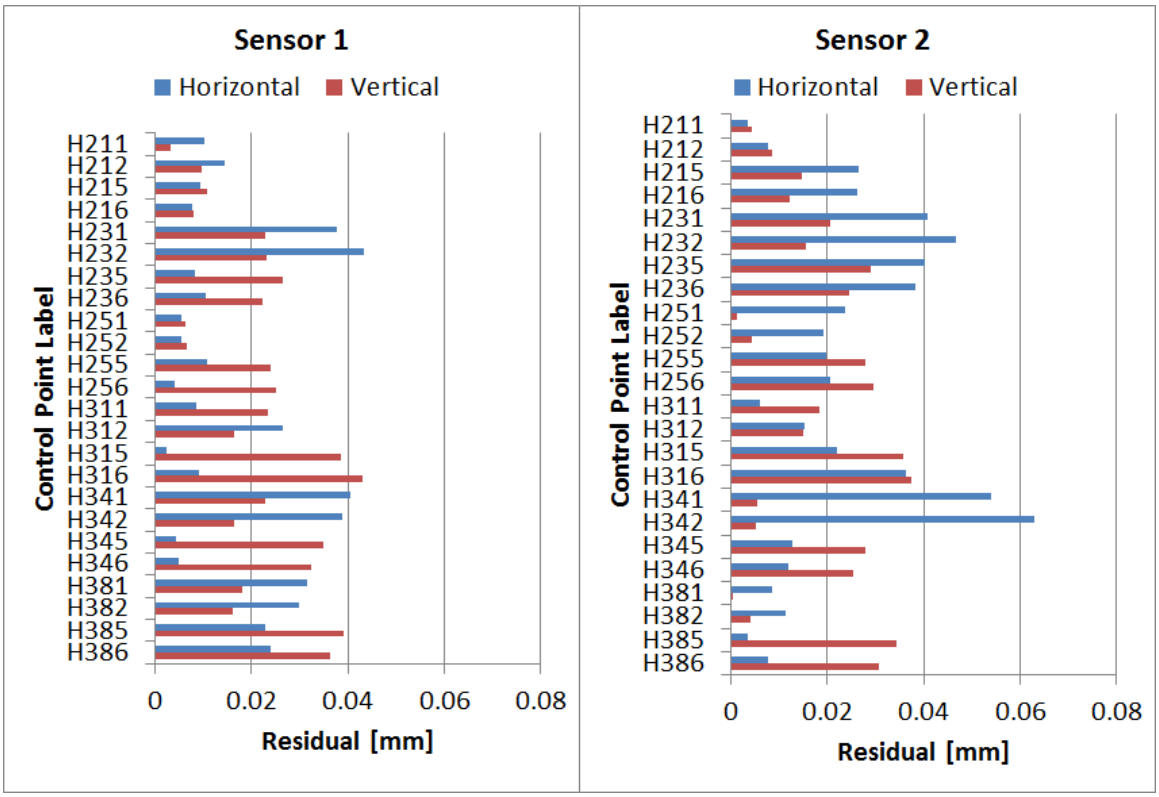

Figure 6.10: Check point residuals computed from point primitives extracted from both sensor's LiDAR point clouds and compared to the ground control points associated with parking lot lines. Note that points H215, H216, H235, H236, H255, H256, H315, H316, H345, H346, H385 and H386 were used in the adjustment process leaving the other 14 points as independent check points.

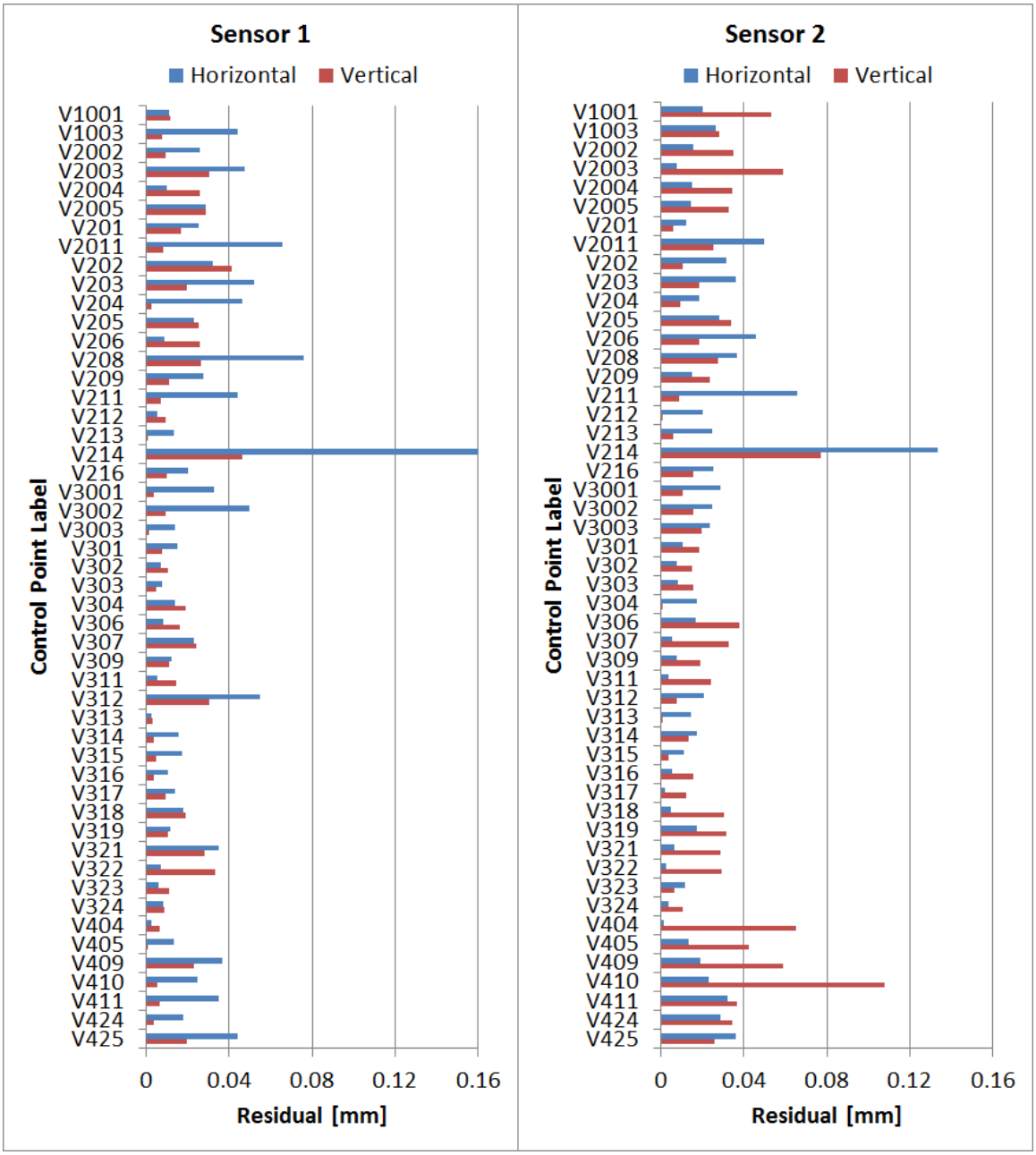

Figure 6.11: Check point residuals computed from point primitives extracted from both sensor's LiDAR point clouds and compared to the control points associated with vertical building features. Note that points V204, V213, V311, V313, V403 and V425 were used in the adjustment process leaving the other 44 points as independent check points.

The control points from Table 6.3 (H215, H235, H255, H315, H345, H385, V204, V311 and V403) and the control points from Table 6.4 (H216, H236, H256, H316, H346, H386, V213, V313 and V425) which were used to constrain the adjustment, are included

in Figures 6.10 and 6.11 to quantify the final size of the minimized residuals and so they may be contrasted with the pre-adjustment residuals (Figures 6.5 and 6.6). The number of independent check points depicted in Figures 6.10 and 6.11 are therefore reduced to the other 56 check point residuals.

From the parking lot features (Figure 6.10) it was found that the horizontal component of the check point residuals for Sensor 1 had a mean average of 0.017m and a standard deviation of 0.013m. For Sensor 2, the horizontal component of the residuals in the parking lot had a mean average of 0.024m and a standard deviation of 0.017m. The vertical component of the residuals in Figure 6.10 had a mean average of 0.016m and a standard deviation of 0.019m for Sensor 1 and a mean average of 0.012m and a standard deviation of 0.017m for Sensor 2. Similarly, from the building features (Figure 6.11) it was found that the horizontal component of the check point residuals produced a mean average of 0.027m and a standard deviation of 0.026m for Sensor 1 and a mean average of 0.021m and a standard deviation of 0.021m for Sensor 2. The vertical component of the check point residuals shown in Figure 6.11 produced a mean average of -0.010m and a standard deviation of 0.015m for Sensor 1 and a mean average of -0.015m and a standard deviation of 0.018m for Sensor 2.

In addition, the calibrated and filtered data was grouped as in Section 6.2.1.1 and run through conditional variance analysis. The results of conditional variance analysis for Sensor 1 are shown in Figure 6.12. In Figure 6.12, it is observed that the Group 3 variables now account for around 50% of the error in the data, while the Group 1

variables now account for up to 20% of the error in the data. The Group 5 variables under the vehicle trajectory also now contribute up to 20% of the error in the point cloud.

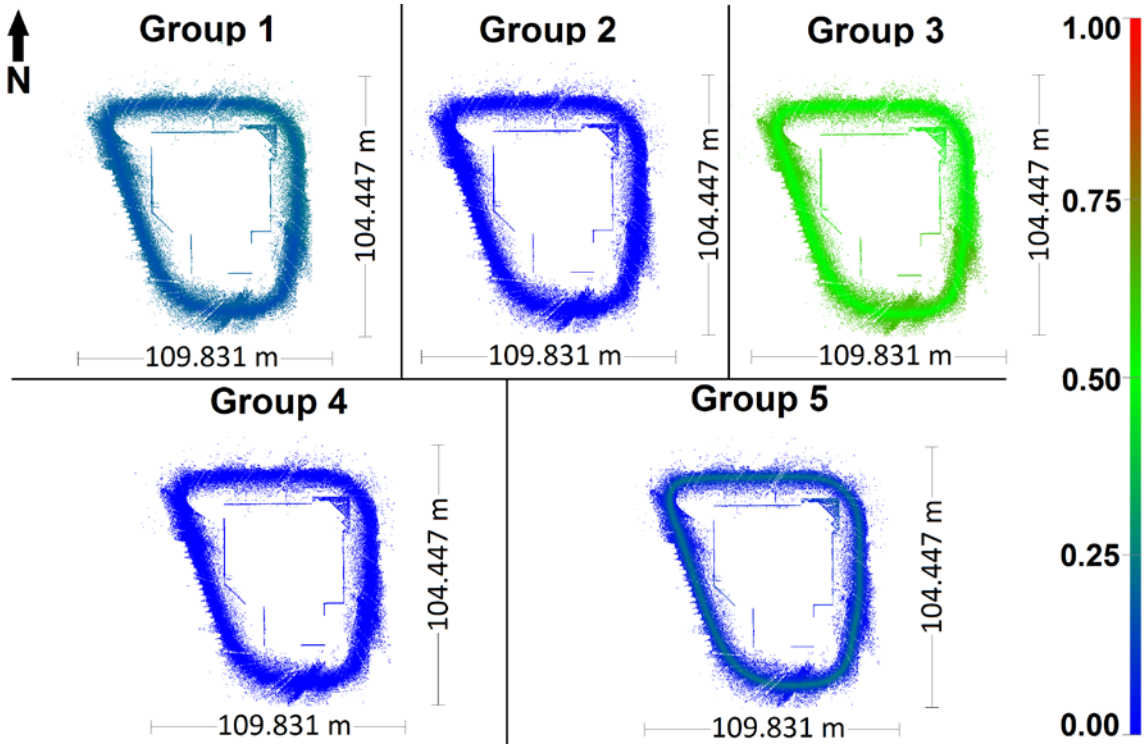

Figure 6.12: MTL data from Sensor 1 of a commercial office building after the quality of the MTL data was improved. The data has been coloured by using an artificial colour scale derived from the total effect indexes from conditional variance analysis.

A similar result is found from Sensor 2 (Figure 6.13), where Group 3 variables account for 40% to 50% of the error in the point cloud, Group 1 variables account for up to 25% of error in the point cloud and Group 5 variables account for up to 20% of error in the point cloud.

These results show that the major bias caused by the Z lever arm has been significantly reduced and that errors from other error sources have become more prominent as the Z lever arm has been reduced. It should also be noted that Figures 6.12 and 6.13 show a

great reduction of the scattered individual points. Figures 6.7 and 6.8 show that many of these scattered individual points had high error contributions from the Z lever arm.

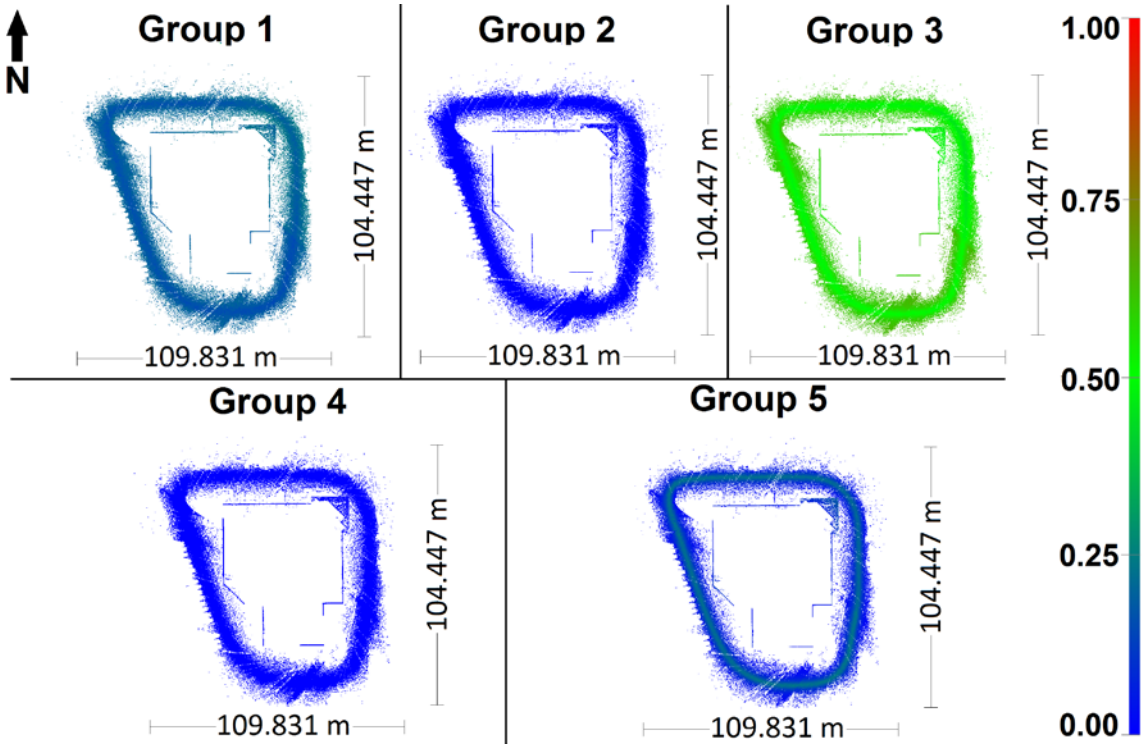

Figure 6.13: MTL data from Sensor 2 of a commercial office building after the quality of the MTL data was improved. The data has been coloured by using an artificial colour scale derived from the total effect indexes from conditional variance analysis.

6.2.2 Data of a Typical Street Scene

Data was collected with a different MTL system than used in Section 6.2.1 along a typical street scene at Oakdale road, in the city of Toronto, Canada. Figure 6.14 shows a dimensioned planimetric view of the LiDAR collected along this corridor combined with a detailed close up of the street scene. The street scene consisted of one story commercial buildings, trees, overhead wires, sidewalks, grass, road signage, manholes, catch basins and asphalt. Control data was surveyed along the sidewalks and on the corners of building along a 400m stretch of the street. To survey the control, a traverse

was established along the length of the test corridor using a Topcon GTS-235w (2", 2mm + 2ppm) total station. ECEF coordinates were established by post processing static GPS observations of the control traverse and referencing them to the same base station as the MTL data. The static GPS was collected using a Topcon Legacy E GPS receiver (3mm + 1ppm (horizontal), 5mm + 1.5ppm (vertical)).

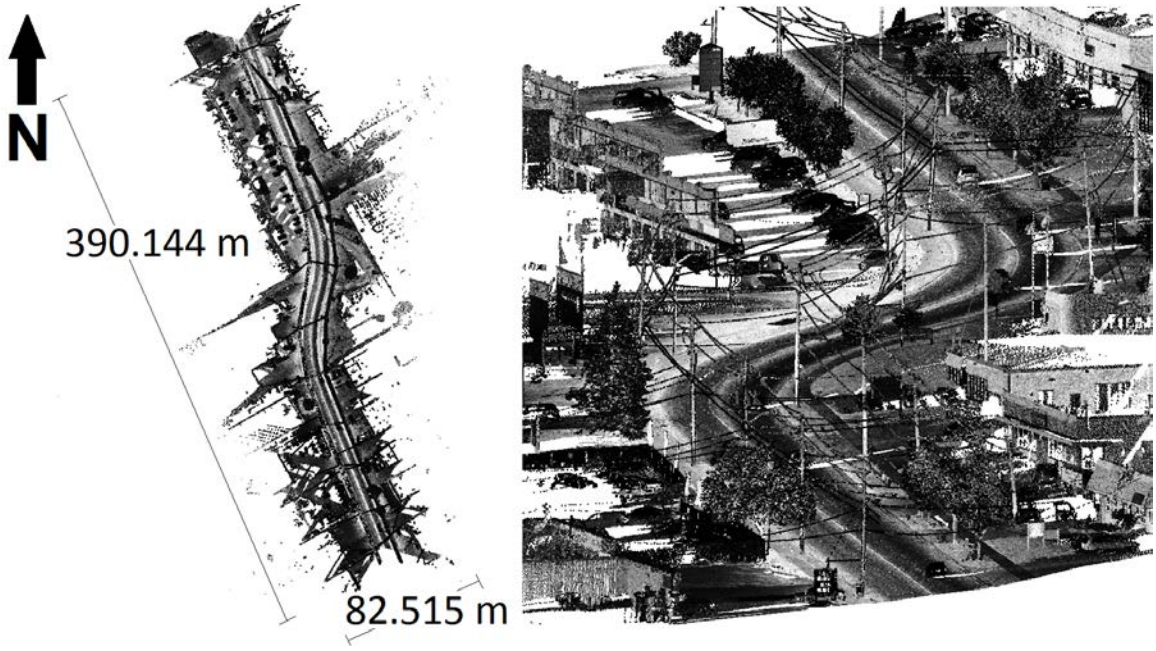

Figure 6.14: Dual sensor MTL data of a 400m stretch of Oakdale Road in Toronto, Canada.

Figure 6.15 lists all the control points used on both the ground surfaces and the vertical building walls. Figure 6.15 also shows the one sigma standard deviation estimates for these points. The quality of the control used in this test was not as good as that used in Section 6.2.1; with most of the control having 1 sigma accuracy estimates in excess of 1cm. In addition, targets on the sidewalk were not ideal as the oblique angle of incidence at which the MTL could see these points contributed to elongated laser footprints reducing the accuracy of individual LiDAR points. Using specialty made targets was

considered, but it was decided that using the natural targets available in the street better replicated the normal conditions under which users of MTL function.

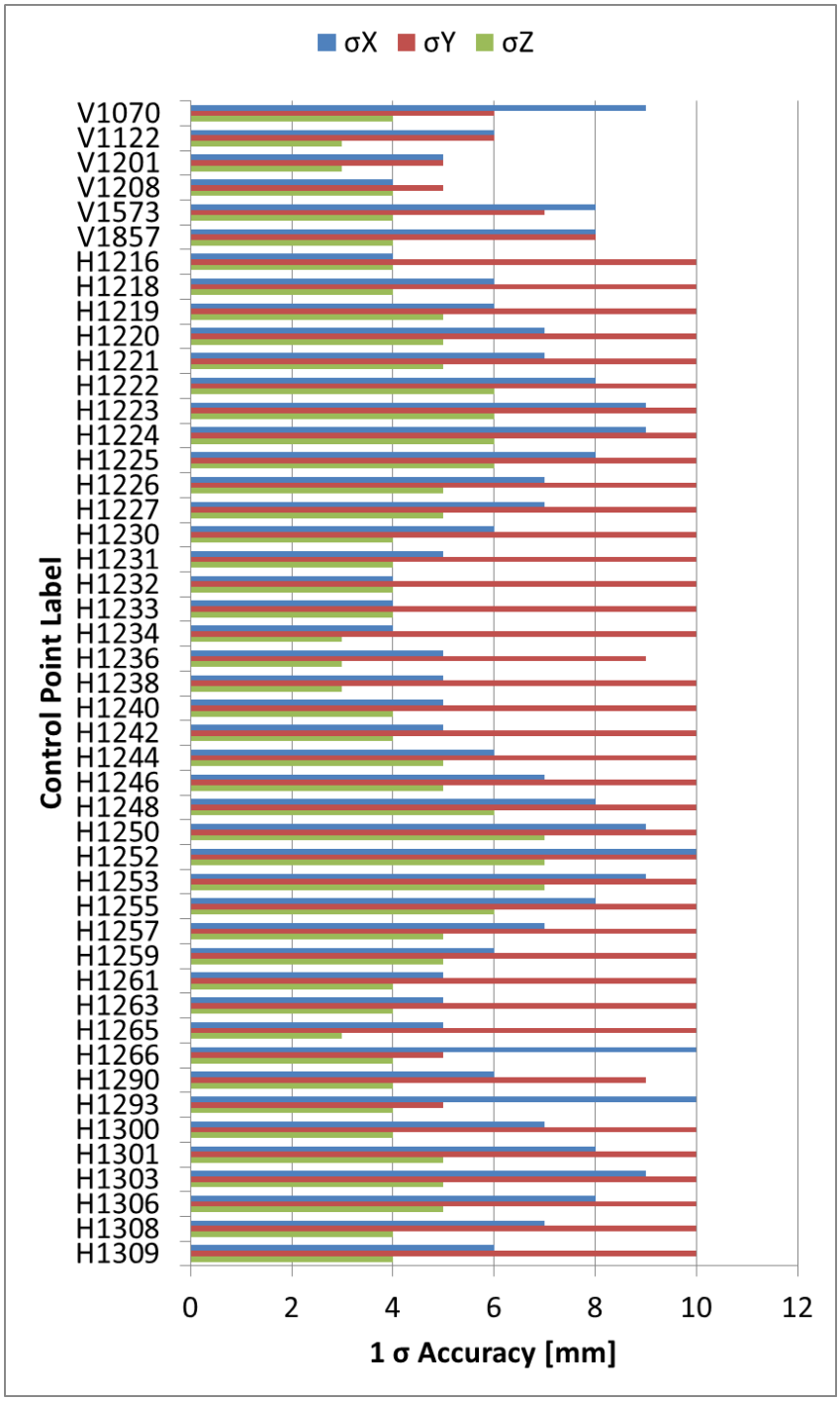

Figure 6.15: One-sigma accuracy estimates for control points surveyed on both horizontal and vertical surfaces located along a 400m section of street in Toronto, Canada.

6.2.2.1 Assessing the Quality of the Data Before Processing

Similar to the previous case in Section 6.2.1.1, primitive geometry was used to extract point features from the point cloud that corresponds to the control points listed in Figure 6.15. These extracted points from the LiDAR point clouds of each sensor, were then compared to the control points. Residuals were formed between the extracted check points and the control points, the results of which are shown in Figure 6.16. The check point residuals in Figure 6.16 are separated into their horizontal and vertical components. The designator V and H are used on residuals to indicate if the feature is a vertical (V) feature such as a building wall or a horizontal (H) feature such as a sidewalk seem.

Based on the check point residuals labeled V in Figure 6.16, it was found that the horizontal component of the V check point residuals on the buildings had a mean average of 0.062m and a standard deviation of 0.016m for Sensor 1 and a mean average of 0.041m and a standard deviation of 0.023m for Sensor 2. The vertical component of the V check point residuals on the buildings had a mean average of -0.025m and a standard deviation of 0.019m for Sensor 1 and a mean average of -0.035m and a standard deviation of 0.017m for Sensor 2.

A similar analysis was conducted on the check point residuals labeled H (ground points) in Figure 6.16. It was found that the horizontal component of the H check point residuals shown in Figure 6.16 produced a mean average of 0.034m and a standard deviation of 0.018m for Sensor 1 and a mean average of 0.037m and a standard deviation of 0.017m for Sensor 2. The vertical component of the H check point residuals shown in Figure

6.16 produced a mean average of -0.010m and a standard deviation of 0.016m for Sensor 1 and a mean average of -0.002m and a standard deviation of 0.019m for Sensor 2.

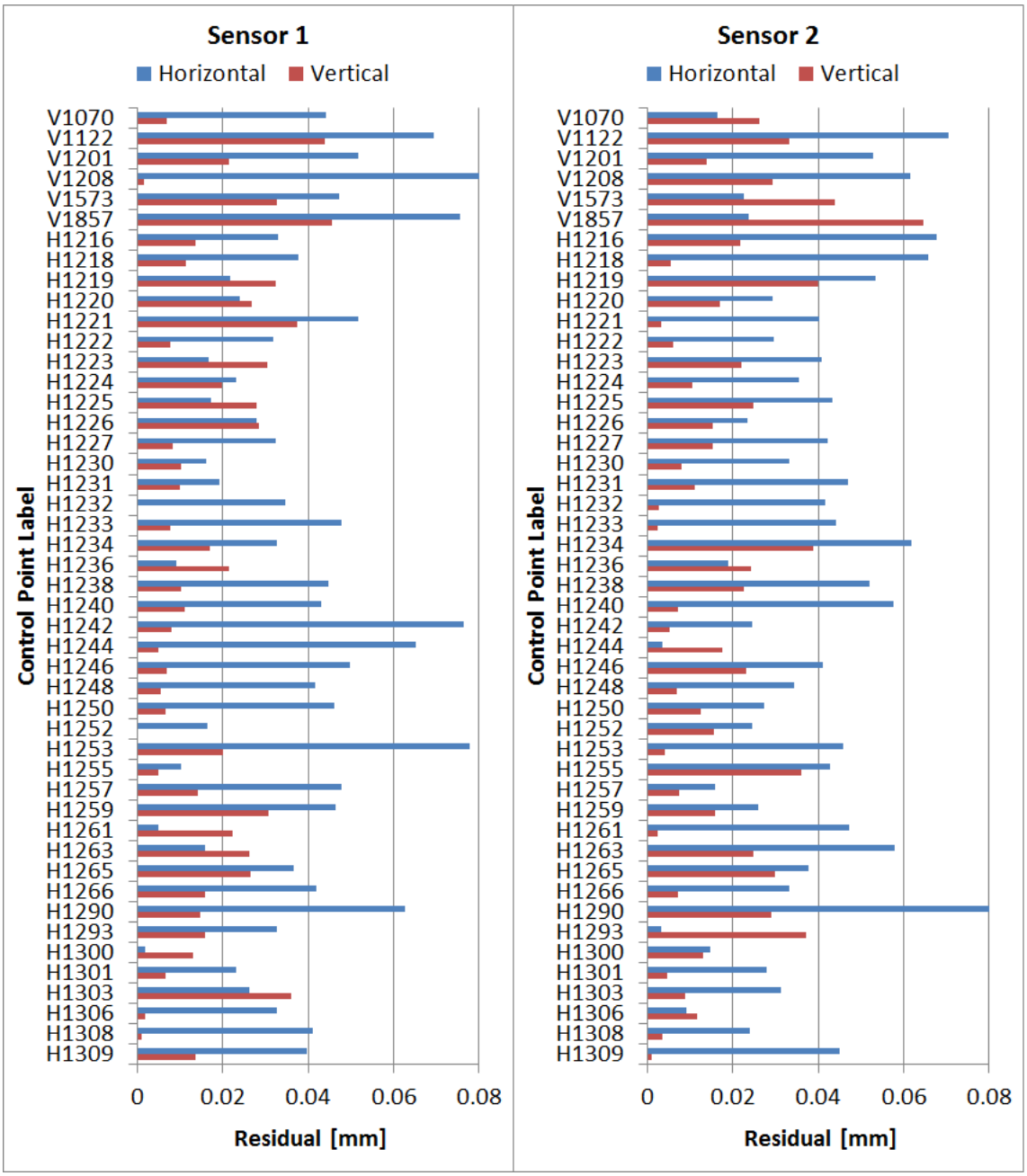

Figure 6.16: Check point residuals computed from point primitives extracted from both sensor's LiDAR point clouds and compared to the control points associated with unique street features before correction.

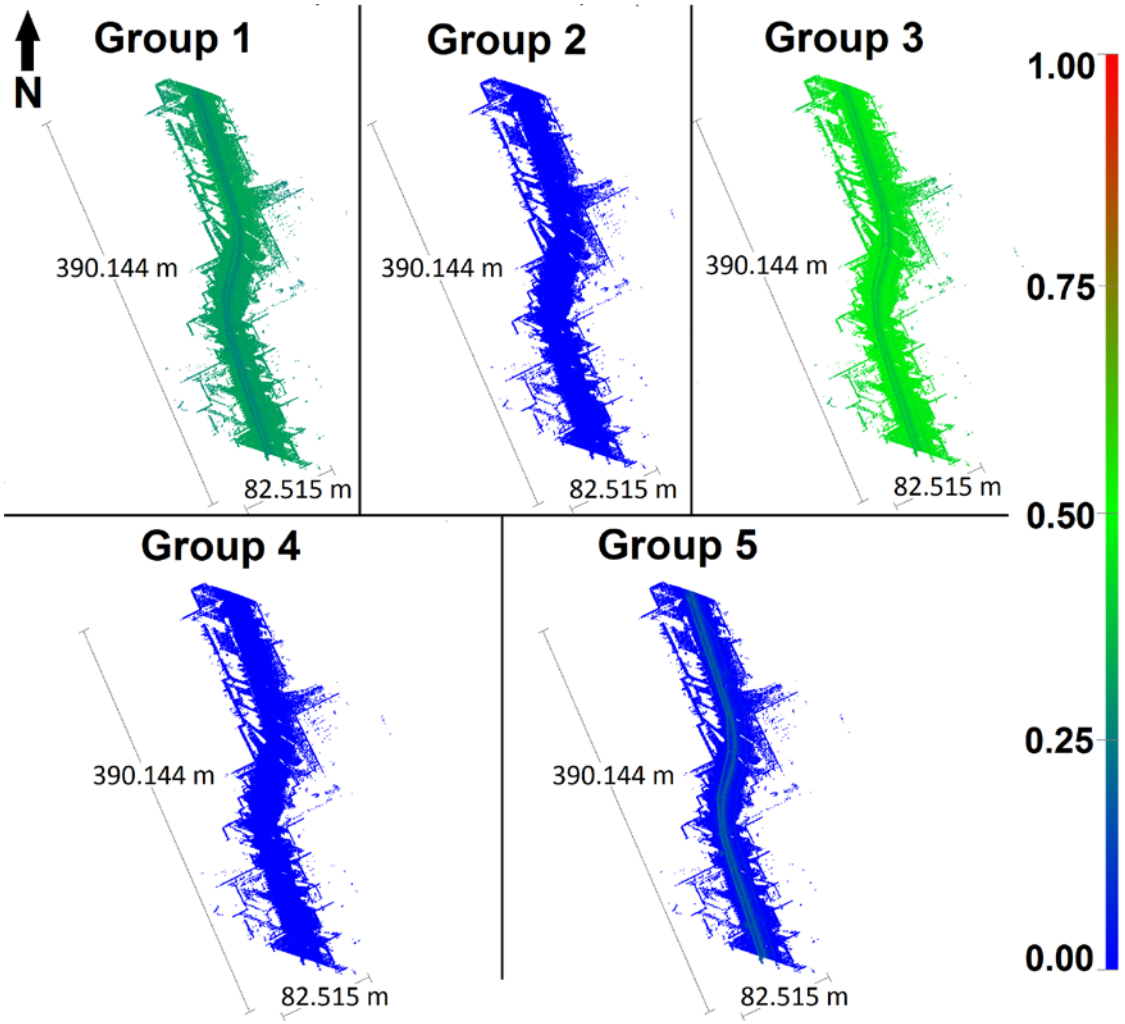

Figure 6.17: MTL data from Sensor 1 of a typical street scene before the quality of the MTL data was improved. The data has been coloured by using an artificial colour scale derived from the total effect indexes from conditional variance analysis.

Performing conditional variance analysis and using the variable groups as established in Chapter 3, it was found that the majority of the errors in the point cloud come from Group 3, specifically the Z lever arm. Figure 6.17 shows that for Sensor 1, the Z lever arm makes up for almost 50% of the error in the overall point cloud. Group 1, the position of the system from the DG system trajectory is the next major contributor. Group 5 does show error spikes directly under the vehicle; these are caused by a degradation of the laser range accuracy caused by its proximity to the road. Similarly,

Figure 6.18 shows the same pattern for Sensor 2, where the Z lever arm is contributing the most error to the point cloud, the DG system position the next most error in the point cloud and the LiDAR is only contributing a significant amount of error directly under the vehicle's trajectory.

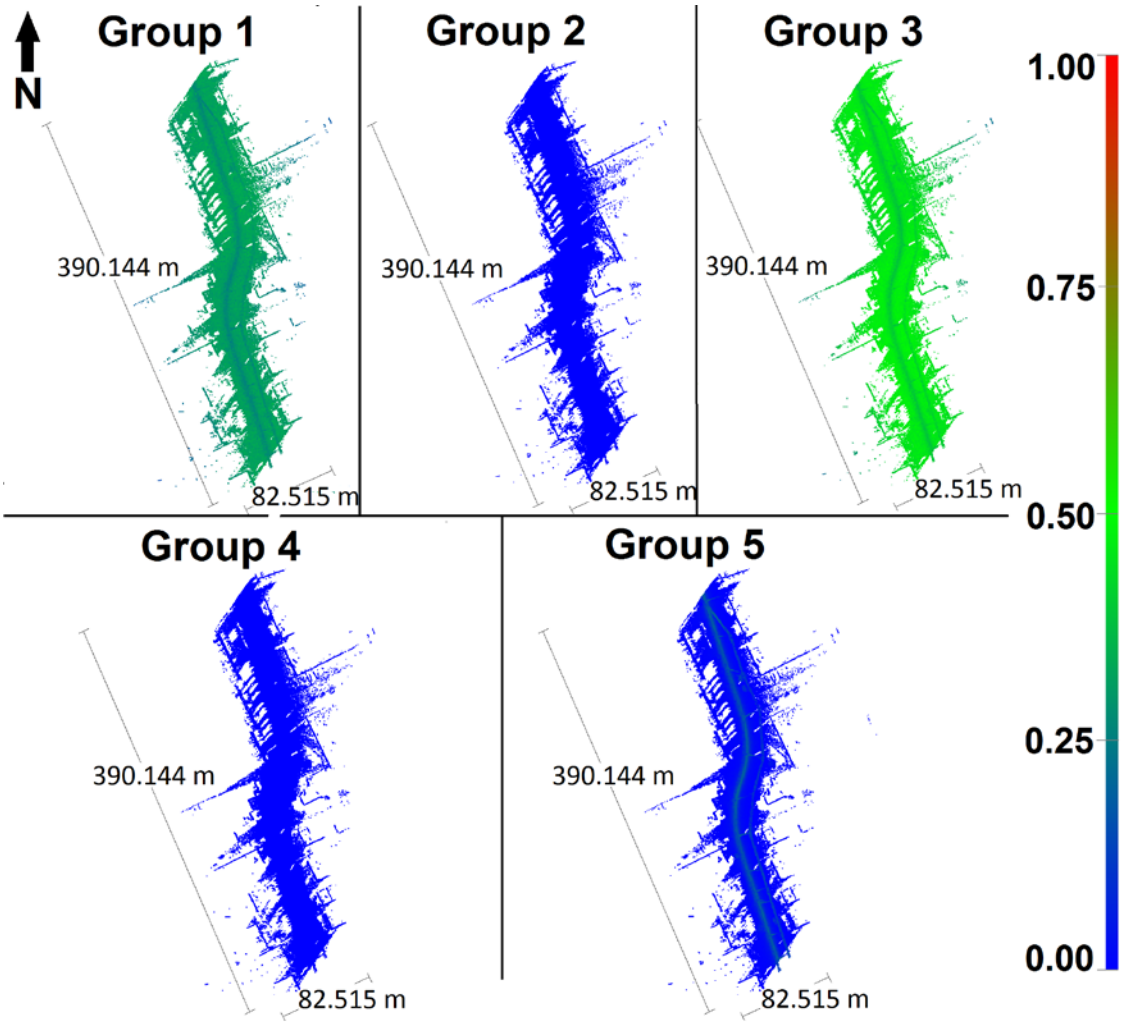

Figure 6.18: MTL data from Sensor 2 of a typical street scene before the quality of the MTL data was improved. The data has been coloured by using an artificial colour scale derived from the total effect indexes from conditional variance analysis.

As stated in Section 6.2.1.2, applying the correction techniques, previously outlined in Chapters 3 to 5, to these data sets should cause the distribution of errors in the point cloud to become more balanced. Currently the vast majority of error is located in the Z lever

arm. Correction of the data should see the percentage of error caused by the Z lever arm reduce, while other variables, such as the DG system positional errors should increase in significance.

6.2.2.2 Zero Error and Temperature Correction

No temperature correction data is available for the MTL system used in this data collect. The data logs from the system indicate that Sensor 1 operated at a constant temperature of 42.9°C and that Sensor 2 operated at a constant temperature of 42.5°C for the duration of the collect. For the sensors used in Section 6.2.1 this would have indicated range corrections of -0.0068m and 0.0041m, respectively. However, since no temperature chart exists for these sensors, it is assumed that the temperature error is vanishingly close to zero.

Table 6.9: Zero error as calculated by the manufacturer.

	Zero Error (m)	STDev (m)
Sensor 1	-0.8522	0.01
Sensor 2	-0.9926	0.01

The zero error provided by the manufacturer is listed in Table 6.9. The two sensors used to collect this data were not available for zero error calibration as described in Chapter 5. Since the manufacturer does not provide information about the accuracy of their zero error calculations, it will be assumed, based on the standard deviations in Table 6.1, that the zero errors for each sensor will have a one sigma standard deviation of approximately 0.01m.

6.2.2.3 Calibrating the LiDAR to the DG System

Calibration of the LiDAR sensors was done using the dual sensor technique presented in Chapter 4. As was done in Section 6.2.1.3, initial approximates to the boresight and lever arm values were estimated and point clouds were produced for each sensor. Based on the results discussed in Chapter 4, Section 4.3.3.3, at least 7 control points will be needed to calibrate the sensor's lever arms. With this in mind, a total of 12 control points from Figure 6.15 were extracted from the point clouds of both sensors, along with 47 common points.

Table 6.10: Control data used with Sensor 1.

Target ID	X [m]	Y [m]	Z [m]	σ_X [mm]	σ_Y [mm]	σ_Z [mm]	σ_{3D} [mm]
V1070	839018.250	-4538405.588	4387631.624	9.0	6.0	4.0	11.5
V1573	839020.064	-4538312.463	4387731.492	8.0	7.0	4.0	11.4
H1236	838989.795	-4538404.521	4387630.561	5.0	9.0	3.0	10.7
H1259	838980.492	-4538286.936	4387759.012	6.0	10.0	5.0	12.7
H1261	838985.796	-4538298.300	4387745.988	5.0	10.0	4.0	11.9
H1300	838997.051	-4538366.640	4387670.448	7.0	10.0	4.0	12.8
H1308	838997.248	-4538361.571	4387675.803	7.0	10.0	4.0	12.8

Table 6.11: Control data used with Sensor 2.

Target ID	X [m]	Y [m]	Z [m]	σ_X [mm]	σ_Y [mm]	σ_Z [mm]	σ_{3D} [mm]
V1122	838981.939	-4538348.297	4387706.811	6.0	6.0	3.0	9.0
V1857	839000.080	-4538264.829	4387789.119	8.0	8.0	4.0	12.0
H1223	838994.929	-4538329.136	4387711.486	9.0	10.0	6.0	14.7
H1244	838979.050	-4538287.385	4387758.909	6.0	10.0	5.0	12.7
H1248	838968.929	-4538264.941	4387784.488	8.0	10.0	6.0	14.1
H1303	838998.774	-4538385.141	4387650.135	9.0	10.0	5.0	14.4
H1309	839000.351	-4538355.482	4387681.842	6.0	10.0	4.0	12.3

Table 6.10 lists 6 control points and their estimated standard deviations. The control points in Table 6.10 correspond to features extracted from the Sensor 1 point cloud and used in the adjustment to find the calibration parameters for the two sensors. Table 6.11

lists another 6 control points and their estimated standard deviations. The control points in Table 6.11 correspond to features extracted from the Sensor 2 point cloud and used in the adjustment to find the calibration parameters for the two sensors. Inputting the 47 common points along with 7 control points on Sensor 1 and the 7 control points on Sensor 2, means the two sensor adjustment will have 49 degrees of freedom (47 + 7 + 7 – 12).

Running the adjustment for a single iteration, the one sigma standard deviations for the initial approximates to the boresight and lever arms were produced. The initial approximates and the calculated one sigma standard deviations are listed in Table 6.12 for both sensors. The values in Table 6.12 were used as part of the inputs to estimate the total Sobol indexes in Section 6.2.2.1.

Table 6.12: Boresight and lever arm values and one sigma standard deviations for both sensors before correction.

		Roll (°)	Pitch (°)	Heading (°)	X (m)	Y (m)	Z (m)
Sensor 1	Value	-0.0700	-29.7000	37.5500	0.000	-0.717	-0.169
	STDev	0.0005	0.0016	0.0004	0.009	0.009	0.010
Sensor 2	Value	-0.1500	-29.8500	-37.0500	0.000	0.673	-0.162
	STDev	0.0005	0.0014	0.0004	0.009	0.009	0.010

Table 6.13: Boresight and lever arm values and one sigma standard deviations for both sensors after correction.

		Roll (°)	Pitch (°)	Heading (°)	X (m)	Y (m)	Z (m)
Sensor 1	Value	-0.1453	-29.3109	37.6422	-0.005	-0.724	-0.196
	STDev	0.0003	0.0012	0.0003	0.004	0.004	0.004
Sensor 2	Value	-0.2928	-29.6620	-36.9568	-0.013	0.685	-0.178
	STDev	0.0003	0.0011	0.0003	0.004	0.004	0.004

Allowing the adjustment to run until the solution converged; new boresight and lever arm values were estimated with their associated one sigma standard deviations. These values are listed in Table 6.13. The standard deviations in Table 6.13 indicate that the boresight values have undergone small incremental changes compared to the values in Table 6.12,

but there is a significant improvement in the accuracy of the lever arm values between Tables 6.11 and 6.12.

6.2.2.4 Removing Outliers

As was done in Section 6.2.1.4, outlier removal was implemented to eliminate points with inaccurate ranges due to target properties or angle of incidence of the laser beam. The time series approach from Chapter 5 was used to filter both sensors' point clouds as shown in Figure 6.14. Several attempts were made to determine the optimal window size for this data set and it was found that a point sample size of 10 points both before and after each test point gave the best results. Table 6.14 shows the results of this filtering. Out of the approximately 8 million points in each sensor; the outlier filter removed around 100,000 points as outliers. This represents 1.2% of the Sensor 1 point cloud and 1.2% of the Sensor 2 point cloud.

Table 6.14: Points removed from each sensor's point clouds by the outlier filter based on the α-β-γ Kalman smoother.

	Total No. of Points	No. of Outliers	% of Points Removed
Sensor 1	8,096,474	99899	1.2
Sensor 2	8,100,610	100,336	1.2

Table 6.15: Points that failed to return a result from the outlier filter based on the α-β-γ Kalman smoother and were therefore indeterminate.

	Total No. of Points	No. of Points Not Checked	% of Points Not Checked
Sensor 1	8,096,474	10,960	0.4
Sensor 2	8,100,610	5,560	0.1

The filter also failed to return a correct prediction on some points in the point cloud. Most of these points occurred where significant gaps were present in the time series. Table 6.15 shows a breakdown of these points. They represent 0.4% of the Sensor 1

point cloud and 0.1% of the Sensor 2 point cloud. Being that these points are unchecked, they are assumed to be good points and remain in the filtered point cloud.

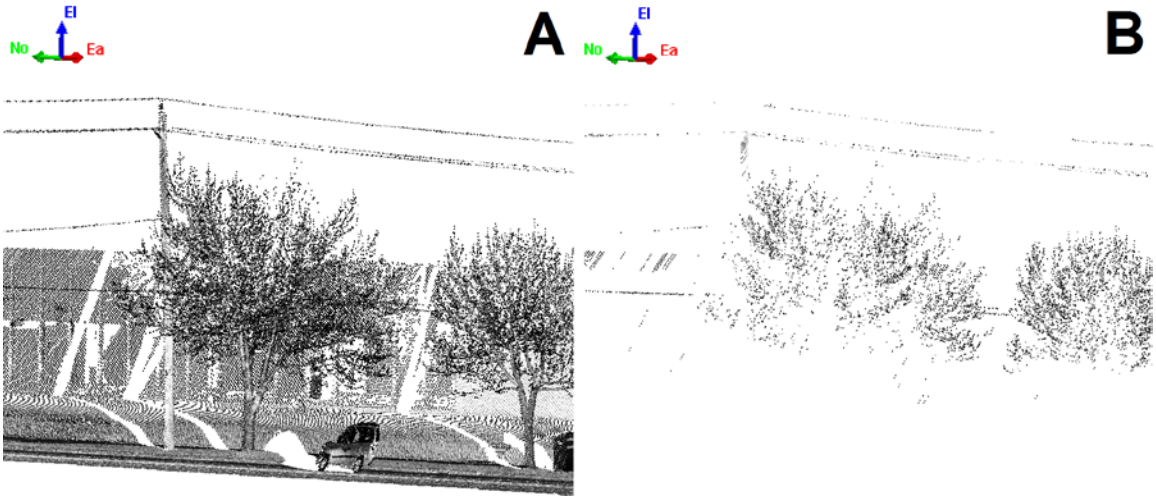

Figure 6.19: Points removed from the MTL data set. Frame A) shows calibrated MTL data before the outlier removal was performed, Frame B) shows the points identified by the α-β-γ Kalman smoother routine and removed from the data set.

Examining the points that were removed from the point cloud, it was found that there were virtually no outliers in the point clouds from the two sensors. The vast majority of the points removed from the point cloud fell within the leafy parts of the trees and shrubs lining the street. Some of the points that were removed also belonged to the overhead wires that paralleled the street. Technically, these points fit the definition of outliers as established in Chapter 5, though most people would consider them good data points. Figure 6.19 shows the complete point cloud from a section of the street and the points selected by the algorithm for removal. As is shown in Figure 6.19, the tips of the tree branches and the overhead wires form the vast majority of the data removed by the outlier algorithm.

6.2.2.5 Assessing the Quality of the Data after Processing

To assess the quality of the processed data, primitive geometry was again used to extract point features from the corrected point clouds from the two sensors. The points extracted from the post-processed point clouds correspond to the 47 control points listed in Figure 6.15. Residuals were formed between the extracted check points and the control points, the results of which are shown in Figure 6.20.

The control points from Table 6.10 (V1070, V1573, H1236, H1259, H1261, H1300 and H1308) and the control points from Table 6.11 (V1122, V1857, H1223, H1244, H1248, H1303 and H1309) which were used to constrain the adjustment, are included in Figure 6.20 to quantify the final size of the minimized residuals and so they may be contrasted with the pre-adjustment residuals (Figure 6.16). The number of independent check points depicted in Figure 6.20 is therefore reduced to the other 33 check point residuals. As was done in Section 6.2.2.1, the check point residuals in Figure 6.20 are separated into their horizontal and vertical components. The designator V and H are used on the check point residuals to indicate if the feature is a vertical (V) feature such as a building wall or a horizontal (H) feature such as a sidewalk seem.

Based on the residuals labeled V in Figure 6.20, it was found that the horizontal component of the V check point residuals on the buildings had a mean average of 0.049m and a standard deviation of 0.029m for Sensor 1 and a mean average of 0.038m and a standard deviation of 0.032m for Sensor 2. The vertical component of the V check point residuals on the buildings had a mean average of -0.006m and a standard deviation of

0.013m for Sensor 1 and a mean average of -0.005m and a standard deviation of 0.025m for Sensor 2.

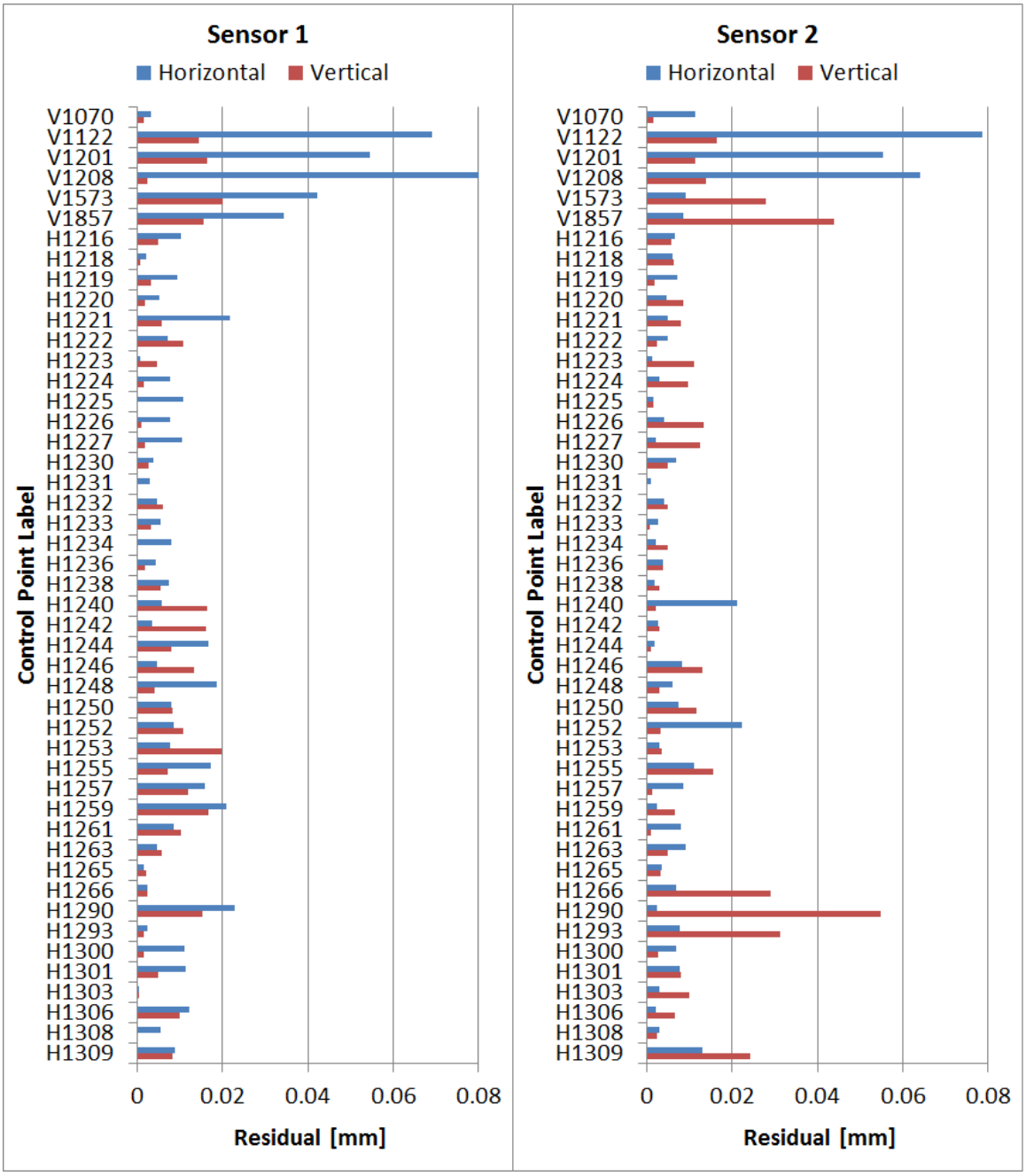

Figure 6.20: Check point residuals computed from point primitives extracted from both sensor's LiDAR point clouds and compared to the control points associated with unique street features after correction. Note that points V1070, V1122, V1573, V1857, H1236, H1223, H1259, H1244, H1261, H1248, H1300, H1303, H1308 and H1309 were used in the adjustment process leaving the other 33 points as independent check points.

Furthermore, it was found that the horizontal component of the H check point residuals shown in Figure 6.20 produced a mean average of 0.009m and a standard deviation of 0.006m for Sensor 1 and a mean average of 0.006m and a standard deviation of 0.005m for Sensor 2. The vertical component of the H check point residuals shown in Figure 6.20 produced a mean average of 0.004m and a standard deviation of 0.007m for Sensor 1 and a mean average of -0.005m and a standard deviation of 0.012m for Sensor 2.

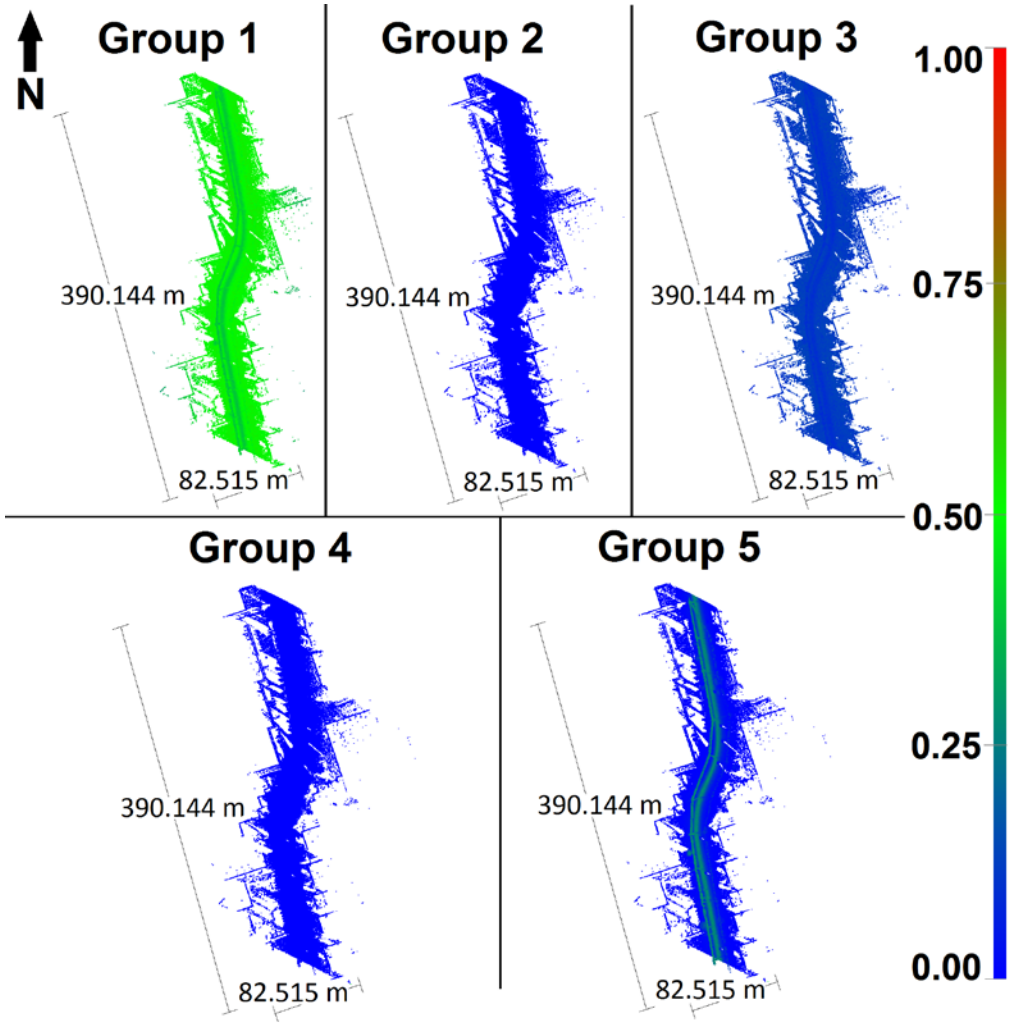

Figure 6.21: MTL data from Sensor 1 of a typical street scene after the quality of the MTL data was improved. The data has been coloured by using an artificial colour scale derived from the total effect indexes from conditional variance analysis.

Again, performing conditional variance analysis and using the variable groups as established in Chapter 3 it was found that the calibration adjustment had reduced the

significance of the Z lever arm substantially. Figure 6.21 shows that Group 3 (DG system to LiDAR sensor lever arms) contributes to less than 15% of the overall error in the Sensor 1 point cloud, compared with the 50% it contributed before correction (Figure 6.17). As expected Figure 6.21 now shows that most of the error in the point cloud is derived from Group 1 which contains the three position parameters from the DG system. In addition, the significance of the errors in Group 5 from the laser range finder has increased.

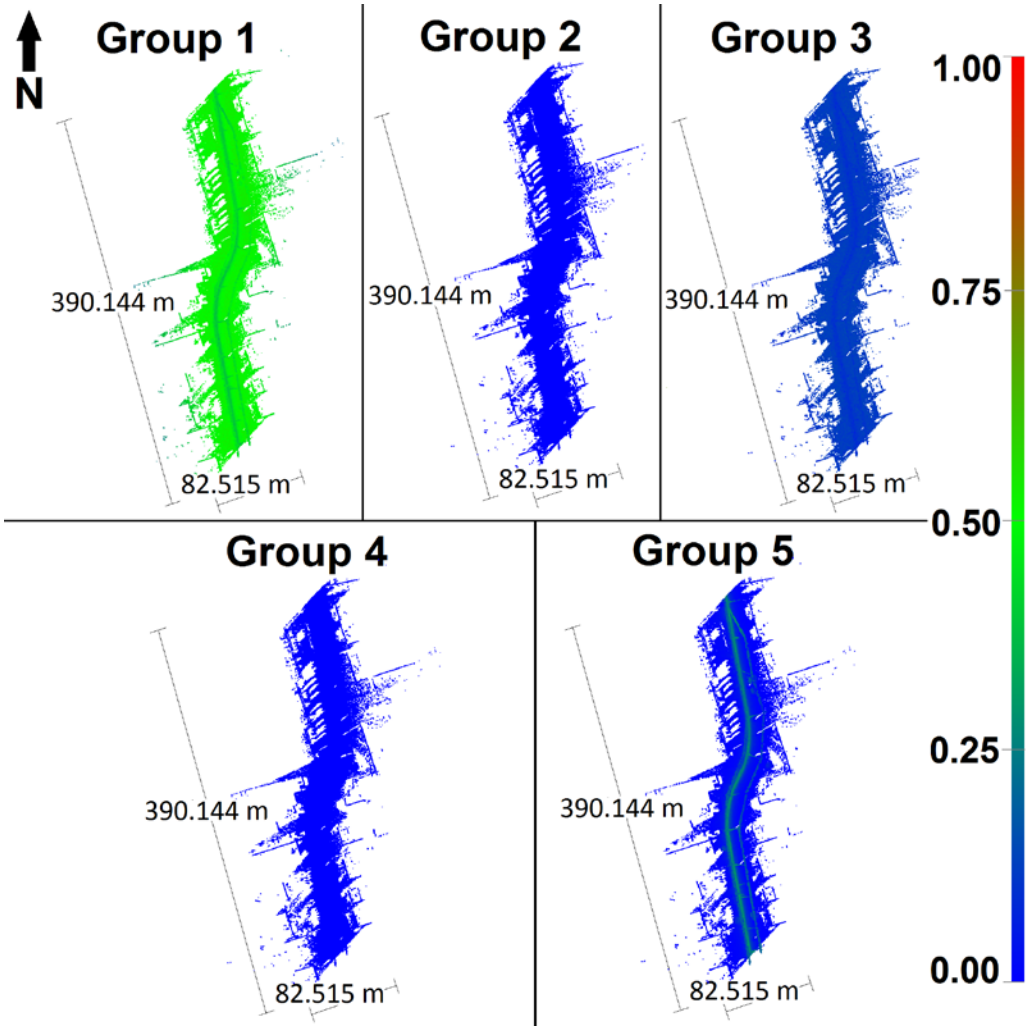

Figure 6.22: MTL data from Sensor 2 of a typical street scene after the quality of the MTL data was improved The data has been coloured by using an artificial colour scale derived from the total effect indexes from conditional variance analysis.

Similar to Figure 6.21, Figure 6.22 shows the distribution of errors in the point cloud derived from Sensor 2 after correction. Again the DG system position errors have become the most significant error source after calibration correction. Group 3 in Figure 6.22 shows that a significant improvement has been made to the amount of error contributed to the point cloud by the sensor to DG system lever arm values.

6.3 Summary

In this chapter, two sets of real data were processed using the outlier detection, zero error calibration, temperature correction, boresight angle and lever arm calibration techniques discussed in Chapters 4 and 5. In addition, the data was evaluated using the error analysis techniques outlined in Chapter 3. A procedure for performing the correction of the test case data was first presented and then the data from the two test cases were processed. Test case 1 showed that by following the outlined processing procedure for the MTL data, the residuals between a control field and the MTL point cloud were reduced by 4.4cm for points located on both horizontal and vertical target surfaces. Conditional variance analysis on test case 1 showed a 25% reduction in the errors generated by the boresight angle uncertainties and an increase in the significance of the uncertainty of the DG system position. Similarly, test case 2 showed an average post procedure reduction in the residuals between control points and MTL data of 2~3cm on horizontal surfaces and 1~2cm on vertical surfaces. Conditional variance analysis on test case 2 showed more than a 30% reduction in the significance of the uncertainty in the boresight angle parameters, causing the DG system position to become the most significant source of error within the post procedure point cloud.

7. Conclusions and Recommendations for Future Work

7.1 Conclusions

The first objective of this research, the establishment of methods for the testing and determination of the accuracy of MTL point clouds, was successfully achieved, and can definitely be applied to any other LiDAR system as a general tool to study instrumental accuracy and reliability. Through the use of residual analysis, error propagation and conditional variance analysis, real data from two MTL systems was analyzed and it was found that the actual errors exceeded the manufacturer's estimates of system accuracy by over 10mm. Conditional variance analysis on these systems has shown that the contribution by the interactions among the measured parameters to the variances of the points in MTL point clouds is insignificant. The sizes of the variances for the measurements used to produce a point are the primary source of error in the output point cloud. In particular, under a loosely controlled error condition, the Z lever arm and roll angle from the LiDAR to the DG system contribute more errors in the output point cloud than any of the other parameters, including the DG system position. Under the tightly controlled boresight and lever arm calibration conditions, short range laser measurements and DG system positioning errors become the dominant source of error in the point cloud. Therefore, MTL systems that have been mounted and boresighted using non-rigorous methodologies provide the greatest source of systematic error in the resulting point cloud. It is therefore essential that high quality rigorous methods be used to derive the integration parameters so that high quality LiDAR data may be collected with any MTL system.

The second objective, the creation of methods for the reduction of errors in all phases of data acquisition and processing, was successful achieved, in that all the test data experienced measureable accuracy improvements after processing occurred. The methods for the boresight and lever arm calibration of the LiDAR sensors to the DG system, the zero error calibration, the temperature correction and the outlier detector all succeeded in improving the accuracy and the quality of the MTL data tested. It was found that the method of simultaneous calibration of the boresight angles and lever arms for two LiDAR sensors (Chapter 4, Section 4.3) did improve the system calibration values. Comparing the two sensors to one another produced enough unique information to enable the computation of the boresight parameters without the aid of ground control points. The redundancy indexes and the variance contribution indexes from multiple MTL system measurements indicated that the lever arms, especially the Z lever arm was extremely dependent on the number, quality and distribution of the ground control points used in the adjustment. It was found that the ground control points had to be located on both of the horizontal and vertical surfaces throughout the collection area for the lever arms to be accurately calibrated. The X and Y lever arms were found to have sub centimetre accuracy estimates when the 7 control points were located on either the horizontal or vertical surfaces, however, the Z lever arm component required that the 7 ground control points be dispersed on both types of surfaces for an accurate calibration. Applying this calibration technique to real MTL data from two different systems produced an improvement in the accuracy of both the estimated boresight angles and the

estimated lever arm parameters. In both case studies, the estimated errors were reduced to less than ±4 arc seconds for all boresight angles and less than ±8mm for all lever arms.

The laboratory calibration of the zero error (Chapter 4, Section 4.1) proved to be successful. Calibration values computed for the sensors used in the trial around the commercial office building produced zero errors with the accuracy of ±1~2mm. These accuracy estimates were a distinct improvement over the accuracy estimates for the original zero error values which had accuracy estimates as low as ±11mm. The scale errors estimated from the laboratory data showed a fair amount of fluctuation and the associated accuracy estimates were comparatively high. Therefore, as the maximum range for the tested sensors is 100m and the manufacturer ignores scale errors in these sensors, the scale error is probably not an issue for this LiDAR system and can be discarded.

The temperature correction in MTL laser range finders (Chapter 4, Section 4.2) has been demonstrated to be effective. When temperature observations are available, it is possible not only to characterize the changes in the measured ranges due to temperature in MTL sensors, but also to produce a table of corrections for the measured ranges. The laboratory data presented here showed that to a fixed target the MTL sensors used to scan the commercial office building displayed a temperature related range error of up to 2.5cm. That being said, the temperature measured in the LiDAR sensors during the field trial around the commercial office building produced range corrections for both sensors that were sub-millimetre and therefore insignificant.

The outlier algorithms have proven to be effective at filtering erroneous points from MTL point clouds. The outlier filter based on the α-β-γ smoother proved capable of accurately identifying and removing up to 70% of the outliers in some point cloud trials of Chapter 5 without erroneously removing any non-outlier points. This performance could be improved by combining the outlier filter based on the α-β-γ Kalman smoother with the outlier filters based on the surface fitting routines. The main weakness of the outlier filter based on the α-β-γ Kalman smoother appears to occur whenever the pattern of the MTL scan lines is interrupted, such as at the end of each scan line. Applying the outlier filter based on the α-β-γ smoother to the MTL data collected during the two trials in Chapter 6 showed that it could identify and remove outliers caused by vehicle exhaust, while preserving the non-outlier points within these point clouds. In both trials, the outlier filter removed LiDAR points collected around vegetation. Technically, these points fit the definition of outliers as established in Chapter 5, though most people would consider them good data points.

The two test cases on real MTL data, used to evaluate the effectiveness of the error correction techniques previously mention, showed that in both cases significant improvements to the final MTL point clouds occurred. Test case 1 showed a significant improvement of the residuals between extracted LiDAR points and control points located on both horizontal and vertical surfaces. Residuals located on horizontal surfaces showed an improvement of 44mm, while the estimated standard deviations for these same points were reduced to as much as 1/5 of their previous values. Residuals on vertical surfaces also showed an improvement of 44mm; however the reduction in the estimated standard

deviations stayed largely the same. At the same time, conditional variance analysis showed that before the data was improved, the majority of error within the point cloud was emanating from the Z lever arm, which was responsible for 60%~75%.of the error. After improving the quality of this data, it was found the lever arms still contribute a significant amount of error in the point clouds, up to 50%. Conditional variance analysis, however, indicates that the error in the point clouds is now more evenly distributed with the DG system and the LiDAR playing a more significant role. The residuals between the surveyed control points and the MTL point clouds produced mean averages in the range of 12mm~27mm and one sigma standard deviations between ±13mm~±26mm, which slightly exceeds the limits set in [5].

Test case 2 showed an improvement of the residuals between extracted LiDAR points and control points located on horizontal surfaces of between 20mm~30mm, while the estimated standard deviations for these same points were cut in half. The residuals for the few points located on vertical surfaces saw less improvement, with a reduction in the residual's mean average of between 10mm~20mm, while the standard deviation for these same points increased. At the same time, conditional variance analysis showed that before the data was improved, the majority of error within the point cloud was emanating from a single variable. The Z lever arm was responsible for up to 50% of the error in the point clouds produced from these sensors. After the improvement to the quality of this data, the lever arms take on far less significance, making the DG system the greatest source of error in these point clouds. In this test case the control points on the horizontal road surfaces had mean averages in the range of 4mm~9mm and one sigma standard

deviations in the range ±5mm~±12mm. However, the control points on the vertical building surfaces had mean averages in the range 5mm~49mm and one sigma standard deviations between ±13mm~±32mm, which again exceed the limits set in [5].

7.2 Recommendations for Future Work

The work presented here has examined and proposed solutions for some of the error sources encountered by MTL users. Being that the MTL consists of many parts, each containing their own sensors and each contributing to the overall error in the system, further work on other aspects of the system could reduce system errors so as to meet the accuracy requirements in specific engineering applications. In the examination of MTL errors presented in this dissertation, no work on correcting or moderating errors generated by the DG system were addressed, as it is equipped as a standalone self-contained component to which a user normally has no access to either its internal hardware and/or software. One possible method for post collection adjustment of the DG system trajectory may involve modifying the calibration adjustment presented in Chapter 4 to include trajectory variables. It may also be possible to successfully integrate the zero error and temperature correction techniques presented in Sections 4.1 and 4.2, respectively, into the calibration model presented in Section 4.3, so that a simultaneous estimation of these values occurs.

Another possibility for future work would be to upgrade the temporal based outlier detection routine to more effectively identify and preserve power line and vegetation data. Currently, the routine treats the individual scan line from the MTL in relative isolation, causing linear features pointed in the direction of the MTL trajectory to be

falsely identified. By combining the methodology presented in Section 5.2.1 with another filter that works on identified outliers in conjugate lines, such false detections may be eliminated. This methodology may even provide the basis for some sort of automatic feature extractor.

The standardization of MTL techniques for geodetic applications in terms of technical specifications and operational procedures, similar to those used by other geodetic instrumental systems, is also needed in practice. However, a specific authority, such as the Ministry of Transportation, may be required to establish and maintain these standards. Further work on producing such standardized specifications and procedures, capable of encompassing the range of MTL applications should be under taken.

REFERENCES

1. 2009. Lynx Mobile Mapper M1 Spec Sheet. Optech Incorporated, 300 Interchange Way, Vaughan, Ontario, Canada, L4K5Z8, URL: http://www.optech.ca/pdf/Lynx_SpecSheet_110909_web.pdf [accessed February 12th 2012].

2. 2014. Lynx SG1: Summary Specification Sheet. Optech Incorporated, 300 Interchange Way, Vaughan Ontario, Canada.

3. 2014. Riegl VQ-450: Summary Specification Sheet. Riegl Laser Measurement Systems GmbH, Riedenburgstrasse 48 3580 Horn Austria. URL: http://www.riegl.com/uploads/tx_pxpriegldownloads/10_DataSheet_VQ-450_rund_2014-09-02.pdf [Accessed March 3rd 2016]

4. 2016. Leica Pegasus Two: Summary Specification Sheet. Leica Geosystems AG, Heerbrugg, Switzerland. URL: http://leica-geosystems.com/products/mobile-sensor-platforms/capture-platforms/leica-pegasus_two [Accessed July 7th 2016]

5. Francis, D., Gourley, B., Turner, J., Busby, I., Blood, K., Blackey, G., Creed, N., Doll, K. and Hawksworth, D., 2008. British Columbia Ministry of Transportation Construction Supervision Survey Guide. BC Ministry of Transportation. pp. 13 http://www.th.gov.bc.ca/publications/eng_publications/survey/construction_survey_guide.pdf [Accessed June 3rd 2016]

6. Rieger, P., N. Studnicka, M. Pfennigbauer and G. Zach. 2010. Boresight alignment method for mobile laser scanning systems. Journal of Applied Geodesy. Volume 4, Issue 1, Pages 13–21, ISSN (Online) 1862-9024, ISSN (Print) 1862-9016, DOI: 10.1515/jag.2010.002, June 2010

7. Chan, T., D. Lichti and C. Glennie. 2013. Multi-feature based boresight self-calibration of a terrestrial mobile mapping system. ISPRS Journal of Photogrammetry and Remote Sensing, Volume 82, pp. 112-124.

8. Skaloud, J. and Schaer, P., 2007. Towards Automated LiDAR Boresight Self-calibration. 5th International Symposium on Mobile Mapping Technology, Padova (Italy), May 29-31, 2007.

9. LeScouarnec, R., Touze, T., Lacambre, J.B. and Seube, N., 2013. A Positioning Free Calibration Method for Mobile Laser Scanning Applications. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume II-5/W2, 2013. ISPRS Workshop Laser Scanning 2013, 11 – 13 November 2013, Antalya, Turkey.

10. Hassan, E.H.H, 2014. Calibration of Multi-Sensor Laser Scanning Systems. Thesis Submitted to the Department of Geomatics Engineering, University of Calgary, In Partial Fulfilment of the Degree of Doctor of Philosophy. January 2014.

11. Glennie, C., 2012. Calibration and Kinematic Analysis of the Velodyne HDL-64E S2 LiDAR Sensor. Photogrammetric Engineering and Remote Sensing 78(4), 339-347.

12. LeScouarnec, R., Touze, T., Lacambre, J.B. and Seube, N., 2014. A New Reliable Boresight Calibration Method for Mobile Laser Scanning Applications. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-3/W1, 2014, EuroCOW 2014, the European Calibration and Orientation Workshop, 12-14 February 2014, Castelldefels, Spain.

13. Csanyi-May, N., 2008. A Rigorous Approach to Comprehensive Performance Analysis of State-of-the-Art Airborne Mobile Mapping Systems. A dissertation presented to the Ohio State University, in partial fulfillment of the degree of doctor of philosophy.

14. Toth, C.K., 2002. Calibrating Airborne LiDAR Systems. Proceedings of the ISPRS Symposium, Integrated System for Spatial Data Production, Custodian and Decision Support, Xi'an, China August 20-23, 2002. Commission II.

15. Reshetyuk, Y., 2009. Self-calibration and direct georeferencing in terrestrial laser scanning. A thesis presented to the Royal Institute of Technology, Department of Transport and Economics, in partial fulfillment of the degree of Doctor of Philosophy in Infrastructure Geodesy. Stockholm, Sweden.

16. Gordon S.J. and Lichti D.D., 2004. Terrestrial laser scanners with a narrow field of view: the effect on 3D resection solutions. Surv Rev 37(292): 448 – 468.

17. Kilpelä, A., 2004. Pulsed time-of-flight laser range finder techniques for fast, high precision measurement applications. Academic Dissertation, Department of Electrical and Information Engineering, University of Oulu, Finland. http://herkules.oulu.fi/isbn9514272625/isbn9514272625.pdf

18. Lichti, D.D. and Harvey, B.R., 2002. The Effects of Reflecting Surface Material Properties on Time-Of-Flight Laser Scanner Measurements. Proceedings of the ISPRS Symposium on Geospatial Theory, Processing and Applications, ISPRS Commission IV, Volume XXXIV Part 4, July 9-12, 2002, Ottawa, Canada.

19. Ingensand, H., Ryf, A. and Schulz, T., 2003. Performances and experiences in terrestrial laserscanning. In Optical 3-D Measurement Techniques VI, A. Grün / H. Kahmen (Eds.). URL: http://users.ntua.gr/mtsakiri/FIG_TaskForce615/03_optical3d_zurich.pdf [Accessed May 27, 2012].

20. Shrestha, R., Carter, W., Slatton, C. and Dietrich, W., 2007. Research-Quality Airborne Laser Swath Mapping: The Defining Factors. A White Paper issued by the National Center for Airborne Laser mapping, University of California, Berkley. Version 1.2. URL: http://ncalm.berkeley.edu/reports/NCALM_WhitePaper_v1.2.pdf [Accessed May 27, 2012]

21. 2011. Calibration of the Leica ALS50 II, Website of the Natural Environment Research Council. URL: http://arsf-dan.nerc.ac.uk/trac/wiki/Processing/LiDARCalibrationProcedure [Accessed May 27, 2012].

22. Dillane, W., 2003. Automated Misalignment Angle Calibration for Airborne Laser Scanners. A thesis presented to Ryerson University, Toronto, Canada, in partial fulfillment of the degree of Master of Applied Science.

23. Skaloud, J. and Lichti, D., 2006. Rigorous Approach to Bore-Sight Self Calibration in Airborne Laser Scanning. ISPRS Journal of Photogrammetry & Remote Sensing, Volume 61, Issue 1, pp.47-59, 2006.

24. Hebel, M. and Stilla, U., 2012. Simultaneous Calibration of ALS Systems and Alignment of Multiview LiDAR Scans of Urban Areas. IEEE Transactions on Geoscience and Remote Sensing, Vol. 50, No. 6, June 2012, pp. 2364-2379.

25. Alshawa, M., Smigiel, E., Grussenmeyer, P. and Landes, T., 2007. Integration of a Terrestrial LiDAR on a Mobile Mapping Platform: First Experiences. 5th International Symposium on Mobile Mapping Technology, Padua, Italy, 2007.

26. Glennie, C., 2007. Reign of Point Clouds: A Kinematic Terrestrial LiDAR Scanning System. Inside GNSS Magazine, Volume 2, Number 7, Fall 2007.

27. Glennie, C., 2007. Rigorous 3D error analysis of kinematic scanning LiDAR systems. Journal of Applied Geodesy, Volume 1, Issue 3 (2007), pp. 147–157. ISSN (Online) 1862-9024, ISSN (Print) 1862-9016, DOI: 10.1515/jag.2007.017, November 2007.

28. Glennie, C., 2009. A Kinematic Terrestrial LiDAR Scanning System. Proceedings of the Transportation Research Board 88th Annual Meeting. Washington D.C., January 11 to 15, 2009. Paper No. 09-0122.

29. Haala, N., Peter, M., Kremer, J. and Hunter, G., 2008. Mobile LiDAR mapping for 3D point cloud collection in urban areas: a performance test. Procceedings of the 21st International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 37, Part B5, Commission 5, pp. 1119ff, Beijing, China, July 2008.

30. Barber, D., Mills, J., and Smith-Voysey, S., 2008. Geometric validation of a ground-based mobile laser scanning system. ISPRS Journal of Photogrammetry and Remote Sensing, Volume 63, Issue 1, January 2008, Pages 128-141.

31. Shen, J., Liu, J., Zhao, R. and Lin, X., 2011. A Kd-tree-based Outlier Detection Method for Airborne LiDAR Point Clouds. IEEE, Proceedings of the International Symposium on Image and Data Fusion, 2011. ISBN: 978-1-4577-0967-8.

32. Brenner, C., 2006. Aerial Laser Scanning. ISPRS Commission VI, Special Interest Group Technology Transfer Caravan. Notes from the International Summer School, Digital Recording and 3D Modeling, Aghios Nikolaos, Greece, April 24-26th, 2006.

33. Liadsky, J., 2007. Introduction to LiDAR. Presented to the Naval Postgraduate School, LiDAR Littoral Studies Workshop, May 2007. URL: http://www.nps.edu/Academics/Centers/RSC/Workshops/LiDAR_littoral.html

34. 2014. Safety of Laser Products – Part 1: Equipment Classification and Requirements, Edition 3. International Electronic Commission. Stability Date 2019, pp. 1 – 220. https://webstore.iec.ch/publication/3587#additionalinfo [accessed November 23th 2016].

35. 2014. ILRIS Terrestrial Laser Scanner: Summary Specification Sheet. Optech Inc., 300 Interchange Way, Vaughan Ontario, L4K 5Z8 URL: http://www.teledyneoptech.com/wp-content/uploads/ILRIS-Spec-Sheet-140730-WEB.pdf [accessed March 10th 2016]

36. 2010. LPM321 Spec Sheet. RieglUSA, URL: http://products.rieglusa.com/Asset/10_DataSheet_LPM-321_18-03-2010.pdf [accessed June 29th 2012].

37. 2009. Faro Laser Scanner Photon 120/20 Summary Specification Sheet. Faro Inc., 250 Technology Park Lake Mary, FL 32746, USA. URL: http://www.dirdim.com/pdfs/DDI_FARO_Laser_Scanner_Photon.pdf [Accessed March 3rd 2016]

38. 2016. Velodyne LiDAR, HDL-64E: High Definition Real-Time 3D LiDAR. Velodyne LiDAR, Inc. 345 Digital Drive, Morgan Hill, CA 95037. URL: http://velodyneLiDAR.com/docs/datasheet/63-9194_Rev-D_HDL-64E_Data%20Sheet_Web.pdf [Accessed March 3rd 2016]

39. Sabatini, R. and Richardson M.A., 2010. Annex B - Laser Range Equation and Detection Performance. NATO Science and Technology Organization. Presented at the Systems Concepts and Integration Panel, ISBN 978-92-837-0082-1 URL: http://www.cso.nato.int/pubs/rdp.asp?RDP=RTO-AG-300-V26 [Accessed August 25th, 2012]

40. 2010. LANDMark Marine: Brochure and Technical Specifications. Applanix, A Trimble Company, 85 Leek Crescent Richmond Hill Ontario Canada L4B 3B3. URL: http://www.applanix.com/media/downloads/products/brochures/LandMark _Marine_Brochure.pdf [Accessed March 3rd 2016]

41. 2015. POS LV Specifications. Applanix, A Trimble Company, 85 Leek Crescent Richmond Hill Ontario Canada L4B 3B3. URL: http://www.applanix.com/media/downloads/products/specs/poslv_specifications12 032012.pdf [Accessed March 3rd 2016]

42. 2016. SPAN - GNSS/INS Combined Systems. Novatel Incorporated, 1120- 68th Avenue N.E., Calgary, Alberta, Canada, T2E 8S5, URL: http://www.novatel.com/products/span-gnss-inertial-systems/span-combined-systems/ [accessed November 23rd, 2016]

43. 2016. iXblue: INS - GNSS systems. 11 Erie drive, Natick, United States, 01760, URL: https://www.ixblue.com/products/ins-gnss-systems [accessed November 23rd, 2016]

44. 2012. Trimble AP50 GNSS-Inertial OEM System. Applanix: A Trimble Company, 85 Leek Crescent, Richmond Hill, Ontario, Canada. URL: http://www.applanix.com/media/downloads/products/specs/poslv_specifications12 032012.pdf [Accessed December 2013].

45. 2013. SBS Coverage. Geneq Inc. 10700 Secant Street Montreal, H1J 1S5, Canada. URL: http://www.sxbluegps.com/technology/sbas-made-easy/ [Accessed March 3rd 2016]

46. Leslar, M., 2009. Extraction of geo-spatial information from LiDAR-based mobile mapping system for crowd control planning. IEEE Xplore Special Issue, Science and Technology for Humanity (TIC-STH), 2009 IEEE Toronto International Conference. pp. 468-472. ISBN: 978-1-4244-3877-8.

47. Kostamovaara J., Määttä K. and Myllylä R.,1991. Pulsed time-of-flight laser range-finding techniques for industrial applications. In: SPIE Proceedings, Vol. 1614, pp 283 – 295.

48. Määttä K., Kostamovaara J. and Myllylä R., 1993. Profiling of hot surfaces by pulsed time-of-flight laser range finder techniques. Appl Opt 32(27): 5324 – 5347.

49. Amann M-Ch., Bosch T., Lescure M., Myllylä R. and Riox M., 2001. Laser ranging: a critical review of usual techniques for distance measurement. Opt Eng 40 (1): 10 – 19.

50. Pfeifer, N., Dorninger, P., Haring, A., Fan, H., 2007. Investigating Terrestrial Laser Scanning Intensity Data: Quality and Functional Relations. In: Gruen, A., Kahmen, H. (Eds.) International Conference on Optical 3-D Measurement Techniques VIII, Zürich, Switzerland, ISBN 3-906467-67-8, pp. 328-337.

51. Keller, F. and Sternberg H., 2013. Multi-Sensor Platform for Indoor Mobile Mapping: System Calibration and Using a Total Station for Indoor Applications. Remote Sensing, Volume 5, pp. 5805-5824, ISSN 2072-4292, doi:10.3390/rs5115805.

52. Hebert M. and Krotkov E., 1992. 3D measurements from imaging laser radars: how good are they? Image and Vision Computing 10(3): 170 – 178.

53. Kahmen, H. and Faig, W., 1988. Surveying. Walter de Gruyter, Berlin/New York

54. Rueger, J.M., 1990. Electronic distance measurement. An Introduction, 3rd totally revised edn. Springer Verlag, Berlin – Heidelberg – New York – London – Paris – Tokyo – Hong-Kong.

55. Linke, A., 2005. Calibration of Terrestrial Laser Scanners. Summary of Master of Science Thesis, School of Surveying and Spatial Information Systems, the University of New South Wales.

56. Gordon, S.J., 2005. Structural Deformation Measurement Using Terrestrial Laser Scanners. PhD thesis, Curtin University of Technology, Department of Spatial Sciences, Australia.

57. Schulz T. and Ingensand H., 2004. Influencing variables, precision and accuracy of terrestrial laser scanners. In: Proceedings of INGEO 2004 and FIG Regional Central and Eastern European Conference on Engineering Surveying, Bratislava, Slovakia, November 11 – 13. URL: http://www.geometh-data.ethz.ch/downloads/SchulzT_TS2_Bratislava_2004.pdf [Accessed July 6th 2016]

58. Rietdorf, A., 2005. Automatisierte Auswertung und Kalibrierung von scannenden Messsystemen mit tachymetrischem Messprinzip. Dissertationen, Technischen Universität Berlin

59. Lichti, D.D. and Franke, J., 2005. Self-calibration of the iQsun 880 laser scanner. In: Gruen A, Kahmen H (eds.) Optical 3-D Measurement Techniques VII, Vol. I, pp 112 – 122.

60. Boehler, W., Bordas Vicent, M. and Marbs, A., 2003. Investigating Laser Scanner Accuracy. Originally Presented at the XIXth CIPA Symposium at Antalya, Turkey, 30 September – 4 October, 2003. Updated for Web Presentation October 2003.

61. Mills, J. and Barber, D., 2006. An Addendum to the Metric Survey Specifications for English Heritage – the collection and archiving of point cloud data obtained by terrestrial laser scanning or other methods. Version 06/11/2006. URL: http://archive.cyark.org/temp/EH3DHeritageaddendum2006.pdf [accessed June 30th 2012].

62. Boehler, W. and Marbs A., 2005. Investigating Laser Scanner Accuracy. URL: http://scanning.fhmainz.de/scannertest/results300305.pdf [accessed July 1st 2012].

63. Marshall, G.F. (ed), 2004. Handbook of Optical and Laser Scanning. Marcel Dekker Inc., New York.

64. Ratcliffe, S., 2005. A digital signal processing technique for time-of-flight laser rangefinding. Gruen A, Kahmen H (eds.) Optical 3-D Measurement Techniques VII, Vol. II, pp 359 – 364.

65. Jelalian, A.V., 1992. Laser Radar Systems. Artech House, Boston – London.

66. Weichel, H., 1990. Laser beam propagation in the atmosphere. SPIE Optical Engineering Press, Bellingham, Washington USA.

67. Granthamn, J.W., Stargardt, C.D., Dungey, C. and Meidunas, E., 1997. Laser Radar in Adverse Weather. In: SPIE Proceedings, Vol. 3065, pp 84 – 93.

68. Ciddor, Ph.E., 1996. Refractive index of air: new equations for the visible and near infrared. Appl Opt 35(9): 1566 – 1573.

69. Ciddor, Ph.E. and Hill, R.J., 1999. Refractive index of air. 2. Group index. Appl Opt 38(9): 1663 – 1667.

70. Harvey, B.R., 2004. Registration and transformation of multiple site terrestrial laser scanning. Geomat Res Australas 80: 33 – 50.

71. Johansson, M., 2002. Explorations into the behaviour of three different high-resolution groundbased laser scanners in the build environment. In: Proceedings of International Workshop on Scanning for Cultural Heritage Recording – Complementing or Replacing Photogrammetry, Corfu, Greece, 1 – 2 September.

72. Jaselskis, E.J., Cacker, Th., Andrle, S.J. and Gao, Z., 2003. Pilot Study on Improving the Efficiency of Transportation Projects Using Laser Scanning. URL: http://www.ctre.iastate.edu/mtc/reports/LaserScanBody.htm [Accessed July 14th 2012]

73. Beraldin, J.A., 2004. Integration of Laser Scanning and Close-Range Photogrammetry – The Last Decade and Beyond. In: Proceedings of the XXth

ISPRS Congress, Commission VII, Istanbul, Turkey, July 12 – 23. URL: http://www.isprs.org/proceedings/XXXV/congress/comm5/papers/188.pdf [Accessed July 14th 2012].

74. Blais, F., Picard, M. and Godin, G., 2004. Accurate 3D Acquisition of Freely Moving Objects. In: Proceedings of the 2nd International Symposium on 3D Data Processing, Visualization and Transmission, Thessaloniki, Greece, September, 6 – 9.

75. Boba, M., Ussyshkin, V., Slater, M., Sitar, M. and Szameitat, W., 2008. Impact of an Optimized Position and Orientation System on the Final Accuracy of LiDAR Data. Proceedings of the ISPRS Congress, Volume XXXVII, Part B1, Commission 3, Working Group 3, Beijing, China, 2008. ISSN 1682-1750.

76. 2008. Lynx Software Manual – Revision B. Optech Incorporated, 300 Interchange Way, Vaughan Ontario, Canada, L4K5Z8.

77. 2008. Boresight Calibration. A Lynx Mobile Mapper Presentation given by Optech Incorporated, 300 Interchange Way, Vaughan, Ontario, Canada, L4K5Z8.

78. 2007. Hysweep User's Manual. Hypack Incorporated, 56 Bradley Street, Middletown, Connecticut, USA.

79. 2010. LiDAR Calibrator. Virtual Geomatics Incorporated, 12741 Research Blvd. Suite 403 Austin Texas USA, 78759-4329.

80. 2011. PDS2000 Multibeam Office Training. PDS2000 v3.7. Reson B.V., Stuttgartstraat 42-44, 3047 AS Rotterdam, Postbus 10300, 3004 AH Rotterdam, +31-10-245-15-00, info@reson.nl

81. Jeeninga, B., 2012. Mobile Laser Calibration. A presentation given by QPS. Huis ter Heideweg 16, 3705 LZ Zeist, The Netherlands, tel : +31 (0)30 69 41 200, fax : +31 (0)30 69 23 663.

82. Dix, M., A. Abd - Elrahman, B. Dewitt and L. Nash. 2011. Accuracy Evaluation of Terrestrial LiDAR and Multibeam Sonar Systems Mounted on a Survey Vessel. *J. Surv. Eng.* doi: 10.1061/(ASCE)SU.1943-5428.0000075.

83. Picard, A., T. Touze, N. Seube, J.G. Nistad and M. Rondeau. 2012. Calibration of Vessel Mounted LiDAR. Proceedings of the Canadian Hydrographic Conference, May 15th to 17th, Niagra Falls, Ontario, Canada.

84. 2011. Lynx Survey Operation Manual – Revision C. Optech Incorporated, 300 Interchange Way, Vaughan Ontario, Canada, L4K5Z8.

85. Habib, A., Bang, K.I., Kersting, A.P. and Chow, J., 2010. Alternative Methodologies for LiDAR System Calibration. Remote Sensing (2), pp. 874-907. doi:10.3390/rs2030874.

86. 2014. ORION C300-1: Airborne LiDAR Summary Specification Sheet. Optech Incorporated, 300 Interchange Way, Vaughan Ontario, Canada.

87. He, M., Zhao, H., Davoine, F., Cui, J. and Zha, H., 2013. Pairwise LiDAR Calibration Using Multi-Type 3D Geometric Features in Natural Scene. 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) November 3-7, 2013. Tokyo, Japan.

88. Thiel, K.H. and Wehr, A., 2004. Performance capabilities of laser scanners – an overview and measurement principle analysis. In: Proceedings of the ISPRS working group VIII/2 "Laser-Scanners for Forest and Landscape Assessment", Freiburg, Germany, October 3 – 6. URL: http://www.isprs.org/proceedings/XXXVI/8-W2/THIEL.pdf [accessed July 2nd 2012].

89. Hancock, JA, 1999. Laser Intensity-Based Obstacle Detection and Tracking. Doctoral dissertation, technical report CMU-RI-TR-99-01, Robotics Institute, Carnegie Mellon University. URL: http://www.ri.cmu.edu/pub_files/pub1/hancock_john_1999_1/hancock_john_1999_1.pdf [accessed July 2nd 2012].

90. Csanyi, N. and Toth, C., 2006. LiDAR Data Accuracy: The Impact of Pulse Repetition Rate. Proceedings of the ASPRS conference "Measuring the Earth II". San Antonio, Texas. November 2006. URL: http://www.asprs.org/a/publications/proceedings/fall2006/0009.pdf [accessed July 2nd 2012].

91. Schulz, T., 2007. Calibration of a Terrestrial Laser Scanner for Engineering Geodesy. PhD thesis, ETH Zurich.

92. Sotoodeh, S., 2006. Outlier Detection in Laser Scanner Point Clouds. Proceedings of the ISPRS Commission V Symposium, Image Engineering and Vision Metrology, Commission V, WG V/3, Volume XXXVI, Part 5, Dresden 25-27 September 2006. pp.297-302.

93. Lu, C.T., Chen, D. and Kou, Y., 2003. Algorithms for Spatial Outlier Detection. Proc of the 3rd IEEE Int. Conf. on Data Mining (ICDM'03). November 19-22, 2003, Melbourne Florida, USA.

94. Zheng, Min-qi; Chong-cheng, C.; Jia-xiang, L.; Ming-hui1, F. and Tamas, J., 2008. An Algorithm for Spatial Outlier Detection Based on Delaunay Triangulation.

Proceedings of the International Workshop on Computational Intelligence in Security for Information Systems (CISIS'08). October 23-24, 2008, Genova, Italy.

95. Pang, G., 2011. Outlier Detection and Data Filtering in LiDAR Data with Multiple Attributes. A thesis submitted in partial fulfilment of the requirements for the degree of Master of Philosophy to the Hong Kong Polytechnic University, Department of land Surveying and Geo-Informatics, 2010.

96. Yamanishi, K., Takeuchi, J., Williams, G. and Milne P., 2004. On-Line Unsupervised Outlier Detection Using Finite Mixtures with Discounting Learning Algorithms. Data Mining and Knowledge Discovery, Volume 8, Issue 3, May 2004, pp. 275-300. doi:10.1023/B:DAMI.0000023676.72185.7c.

97. Tian, X., Xu, L., Li, X., Jing, L. and Zhao, Y., 2012. A Kernel-density-estimation-based Outlier Detection for Airborne LiDAR Point Clouds. IEEE Xplore Special Issue, Imaging Systems and Techniques (IST), 2012 IEEE London International Conference. pp. 263-266. ISBN: 978-1-4577-1776-5.

98. Hou, W., Wang, X., Zhang, C., Ji, Z. and Zhang, X., 2014. Minimum Spanning Tree-Based Digital Terrain Model Detection From Light Detection and Ranging Points. Inverse Problems in Science and Engineering, 2014, Volume 22, No. 6, pp. 988–1001.

99. Matkan, A.A., Hajeb, M., Mirbagheri, B., Sadeghian, S. and Ahmadi, M., 2014. Spatial Analysis for Outlier Removal from LiDAR Data. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-2/W3, 2014 The 1st ISPRS International Conference on Geospatial Information Research, 15–17 November 2014, Tehran, Iran.

100. Hodge, V. and Austin, J., 2004. A Survey of Outlier Detection Methodologies. Artificial Intelligence Review, Volume 22, No. 1, pp. 85-126.

101. Al-Zoubi, M., 2009. An Effective Clustering-Based Approach for Outlier Detection. European Journal of Scientific Research, 2009, Volume 28 Issue 2, pp. 310-316.

102. Last, M. and Kandel, A., 2001. Automated Detection of Outliers in Real-World Data. In Proceedings of the Second International Conference on Intelligent Technologies, pp. 292-301.

103. Leslar, M., Hu, B. and Wang, J.G., 2014. Error Analysis of a Mobile Terrestrial LiDAR System. Geomatica, Volume 68, No. 3, 2014, pp. 183 to 194.

104. Goulden, T. 2009. Prediction of Error Due to Terrain Slope in LiDAR Observations. A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Engineering in the Department of Geodesy and

Geomatics Engineering, University of New Brunswick. January 2009. Published as technical report No. 265, University of New Brunswick.

105. Tellinghuisen, J. 2001. Statistical Error Propagation. *Journal of Physical Chemistry*, Volume 105, pp. 3917-3921.

106. Saltelli, A., M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana and S. Tarantola. 2008. *Global Sensitivity Analysis: The Primer*. John Wiley & Sons, Ltd. pp. 20-25.

107. Glen, G. and K. Isaacs. 2012. Estimating Sobol Sensitivity Indexes Using Correlations. *Environmental Modelling & Software*, Volume 37, pp. 157-166.

108. Bowyer, P. and F.M. Danson. 2004. Sensitivity of Spectral Reflectance to Variation in Live Fuel Moisture Content at Leaf and Canopy Level. *Remote Sensing of Environment*. Volume 92, pp. 297-308.

109. Schwieger, V. 2004. Variance-based Sensitivity Analysis for Model Evaluation in Engineering Surveys. *INGEO 2004 and FIG Regional Central and Eastern European Conference on Engineering Surveying*. Bratislava, Slovakia, November 11-13, 200.

110. 2010. LiDAR Acquisition, QA/QC and Contour Mapping. A report prepared for Bastrop County Texas, March 31st 2010. Report Prepared by HALFF, 4030 West Braker Lane, Suite 450, Austin Texas 78759. URL: http://www.twdb.state.tx.us/RWPG/rpgm_rpts/0704830725_LiDAR_BastropCounty.pdf [accessed June 29th 2012].

111. Chow, J.C.K, Lichti, D.D. and Glennie, C., 2011. Point-Based Versus Plane-Based Self-Calibration of Static Terrestrial Laser Scanners. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XXXVIII-5/W12, 2011. ISPRS Calgary 2011 Workshop, 29-31 August 2011, Calgary, Canada.

112. Leslar, M., Wang, J.G. and Hu, B., 2016. Boresight and Lever Arm Calibration of a Mobile Terrestrial LiDAR System. Geomatica, Volume 70, No. 2, 2016, pp. 31 to 46.

113. Pope, A.J., 1976. The Statistics of Residuals and the Detection of Outliers. NOAA Technical Report NOS 65 NGS 1, Rockville Md. page 13.

114. Wang, J.G., 2009. Reliability analysis in Kalman filtering, Journal of Global Positioning Systems, Vol. 8, No. 1 (2009), pp. 86-96.

115. Wang, J.G., Gopaul, N. and Scherzinger, B., 2009. Simplified Algorithms of Variance Component Estimation for Static and Kinematic GPS Single Point

Positioning, Journal of Global Positioning Systems, Vol. 8, No. 1 (2009), pp. 43-51.

116. Leslar, M., Wang, J.-G., Hu, B., 2011. Comprehensive Utilization of Temporal and Spatial Domain Outlier Detection Methods for Mobile Terrestrial LiDAR Data. Remote Sensing, 2011, Vol. 3, No. 8, pp. 1724-1742.

117. Wang, J., 1997. Pre-Processing of INS Data with the Help of the α-β-γ Filter. Internal Report to the Institute of Geodesy, Uniβw Munich, Neubiberg, Germany, July 1997.

118. Jia, P. and Zhu, Z., 1984. Optimal Estimation and Applications. Science Press, Beijing, 1984.

119. Wang, J.G., 2009. Least Squares Quadric Surface Fitting with the help of Statistical Tests– A Case Study in Industrial Surveying. International Geomatics Forum, Qingdao, China, May 29-30.

120. Drixler, E., 1993. Analyse der Form und Lage von Objekten im Raum. Dissertation, Reihe C, Heft Nr. 409, Deutsche Geodätische Kommission, München 1993.

121. 2010. Polyworks/IMSurvey: Comparison and Verification Software Reference Guide Version 11.0 for Windows, July 2010. InnovMetric Software Inc. 2014 Cyrille-Duquet, Suite 310, Qubec City, Quebec, Canada, G1N 4N6.

Appendix A

In Section 3.2.1 the LiDAR to Geocentric coordinate conversion that was used to translate local MTL coordinates to global Geocentric coordinates was introduced. For the purposes of error analysis and boresight calibration, the first derivatives of P^{ECEF} needed to be determined with respect to several of the input parameters. The derivations of these first derivatives used during the course of this research are presented here.

Roll (θ_1)

$$\frac{\partial P^{ECEF}}{\partial \theta_1} = \frac{\partial P_{INS}^{ECEF}}{\partial \theta_1} + \frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial \theta_1} + \frac{\partial L_{LiDAR-p}^{ECEF}}{\partial \theta_1}$$

$$\frac{\partial P_{INS}^{ECEF}}{\partial \theta_1} = 0$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial \theta_1} = 0$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial \theta_1} = \frac{\partial [R_1(B,L) \cdot R_2(r,p,h) \cdot R_3(\theta_1,\theta_2,\theta_3) \cdot l_p^{LiDAR}(\alpha,\beta,d,K)]}{\partial \theta_1}$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial \theta_1} = R_1(B,L) \cdot R_2(r,p,h) \cdot \frac{\partial R_3(\theta_1,\theta_2,\theta_3)}{\partial \theta_1} \cdot l_p^{LiDAR}(\alpha,\beta,d,K)$$

$$\frac{\partial R_3(\theta_1,\theta_2,\theta_3)}{\partial \theta_1} = \begin{bmatrix} 0 & \omega_{12} & \omega_{13} \\ 0 & \omega_{22} & \omega_{23} \\ 0 & \omega_{32} & \omega_{33} \end{bmatrix}$$

Where

$$\omega_{12} = \cos(\theta_1) \cdot \sin(\theta_2) \cdot \cos(\theta_3) + \sin(\theta_1) \cdot \sin(\theta_3)$$
$$\omega_{13} = -\sin(\theta_1) \cdot \sin(\theta_2) \cdot \cos(\theta_3) + \cos(\theta_1) \cdot \sin(\theta_3)$$
$$\omega_{22} = \cos(\theta_1) \cdot \sin(\theta_2) \cdot \sin(\theta_3) - \sin(\theta_1) \cdot \cos(\theta_3)$$
$$\omega_{23} = -\sin(\theta_1) \cdot \sin(\theta_2) \cdot \sin(\theta_3) - \cos(\theta_1) \cdot \cos(\theta_3)$$
$$\omega_{32} = \cos(\theta_1) \cdot \cos(\theta_2)$$
$$\omega_{33} = -\sin(\theta_1) \cdot \cos(\theta_2)$$

Pitch (θ_2)

$$\frac{\partial P^{ECEF}}{\partial \theta_2} = \frac{\partial P_{INS}^{ECEF}}{\partial \theta_2} + \frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial \theta_2} + \frac{\partial L_{LiDAR-p}^{ECEF}}{\partial \theta_2}$$

$$\frac{\partial P_{INS}^{ECEF}}{\partial \theta_2} = 0$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial \theta_2} = 0$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial \theta_2} = \frac{\partial [R_1(B,L) \cdot R_2(r,p,h) \cdot R_3(\theta_1,\theta_2,\theta_3) \cdot l_p^{LiDAR}(\alpha,\beta,d,K)]}{\partial \theta_2}$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial \theta_2} = R_1(B,L) \cdot R_2(r,p,h) \cdot \frac{\partial R_3(\theta_1,\theta_2,\theta_3)}{\partial \theta_2} \cdot l_p^{LiDAR}(\alpha,\beta,d,K)$$

$$\frac{\partial R_3(\theta_1,\theta_2,\theta_3)}{\partial \theta_2} = \begin{bmatrix} -\sin(\theta_2)\cdot\cos(\theta_3) & \sin(\theta_1)\cdot\cos(\theta_2)\cdot\cos(\theta_3) & \cos(\theta_1)\cdot\cos(\theta_2)\cdot\cos(\theta_3) \\ -\sin(\theta_2)\cdot\sin(\theta_3) & \sin(\theta_1)\cdot\cos(\theta_2)\cdot\sin(\theta_3) & \cos(\theta_1)\cdot\cos(\theta_2)\cdot\sin(\theta_3) \\ -\cos(\theta_2) & -\sin(\theta_1)\cdot\sin(\theta_2) & -\cos(\theta_1)\cdot\sin(\theta_2) \end{bmatrix}$$

Heading (θ_3)

$$\frac{\partial P^{ECEF}}{\partial \theta_3} = \frac{\partial P_{INS}^{ECEF}}{\partial \theta_3} + \frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial \theta_3} + \frac{\partial L_{LiDAR-p}^{ECEF}}{\partial \theta_3}$$

$$\frac{\partial P_{INS}^{ECEF}}{\partial \theta_3} = 0$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial \theta_3} = 0$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial \theta_3} = \frac{\partial [R_1(B,L) \cdot R_2(r,p,h) \cdot R_3(\theta_1,\theta_2,\theta_3) \cdot l_{LiDAR-p}^{b}(\alpha,\beta,d,K)]}{\partial \theta_3}$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial \theta_3} = R_1(B,L) \cdot R_2(r,p,h) \cdot \frac{\partial R_3(\theta_1,\theta_2,\theta_3)}{\partial \theta_3} \cdot l_p^{LiDAR}(\alpha,\beta,d,K)$$

$$\frac{\partial R_3(\theta_1,\theta_2,\theta_3)}{\partial \theta_3} = \begin{bmatrix} \kappa_{11} & \kappa_{12} & \kappa_{13} \\ \kappa_{21} & \kappa_{22} & \kappa_{23} \\ 0 & 0 & 0 \end{bmatrix}$$

Where

$$\kappa_{11} = -\cos(\theta_2)\cdot\sin(\theta_3)$$

$$\kappa_{12} = -\sin(\theta_1)\cdot\sin(\theta_2)\cdot\sin(\theta_3) - \cos(\theta_1)\cdot\cos(\theta_3)$$

$$\kappa_{13} = -\cos(\theta_1)\cdot\sin(\theta_2)\cdot\sin(\theta_3) + \sin(\theta_1)\cdot\cos(\theta_3)$$

$$\kappa_{21} = \cos(\theta_2)\cdot\cos(\theta_3)$$

$$\kappa_{22} = \sin(\theta_1)\cdot\sin(\theta_2)\cdot\cos(\theta_3) - \cos(\theta_1)\cdot\sin(\theta_3)$$

$$\kappa_{23} = \cos(\theta_1)\cdot\sin(\theta_2)\cdot\cos(\theta_3) + \sin(\theta_1)\cdot\sin(\theta_3)$$

X Lever Arm(l_X)

$$\frac{\partial P^{ECEF}}{\partial l_X} = \frac{\partial P_{INS}^{ECEF}}{\partial l_X} + \frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial l_X} + \frac{\partial L_{LiDAR-p}^{ECEF}}{\partial l_X}$$

$$\frac{\partial P_{INS}^{ECEF}}{\partial l_X} = 0$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial l_X} = R_1(B,L) \cdot R_2(r,p,h) \cdot \frac{\partial l_{LiDAR}^{INS}}{\partial l_X}$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial l_X} = 0$$

$$\frac{\partial l_{LiDAR}^{INS}}{\partial l_X} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Y Lever Arm(l_Y)

$$\frac{\partial P^{ECEF}}{\partial l_Y} = \frac{\partial P_{INS}^{ECEF}}{\partial l_Y} + \frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial l_Y} + \frac{\partial L_{LiDAR-p}^{ECEF}}{\partial l_Y}$$

$$\frac{\partial P_{INS}^{ECEF}}{\partial l_Y} = 0$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial l_Y} = R_1(B,L) \cdot R_2(r,p,h) \cdot \frac{\partial l_{LiDAR}^{INS}}{\partial l_Y}$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial l_Y} = 0$$

$$\frac{\partial l_{LiDAR}^{INS}}{\partial l_Y} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

Z Lever Arm(l_Z)

$$\frac{\partial P^{ECEF}}{\partial l_Z} = \frac{\partial P_{INS}^{ECEF}}{\partial l_Z} + \frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial l_Z} + \frac{\partial L_{LiDAR-p}^{ECEF}}{\partial l_Z}$$

$$\frac{\partial P_{INS}^{ECEF}}{\partial l_Z} = 0$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial l_Z} = R_1(B,L) \cdot R_2(r,p,h) \cdot \frac{\partial l_{LiDAR}^{INS}}{\partial l_Z}$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial l_Z} = 0$$

$$\frac{\partial l_{LiDAR}^{INS}}{\partial l_Z} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

POS Roll (r)

$$\frac{\partial P^{ECEF}}{\partial r} = \frac{\partial P_{INS}^{ECEF}}{\partial r} + \frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial r} + \frac{\partial L_{LiDAR-p}^{ECEF}}{\partial r}$$

$$\frac{\partial P_{INS}^{ECEF}}{\partial r} = 0$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial r} = \frac{\partial \left[R_1(B,L) \cdot R_2(r,p,h) \cdot l_{LiDAR}^{INS} \right]}{\partial r}$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial r} = \frac{\partial [R_1(B,L) \cdot R_2(r,p,h) \cdot R_3(\theta_1,\theta_2,\theta_3) \cdot l_p^{LiDAR}(\alpha,\beta,d,K)]}{\partial r}$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial r} = R_1(B,L) \cdot \frac{\partial R_2(r,p,h)}{\partial r} \cdot l_{LiDAR}^{INS}$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial r} = R_1(B,L) \cdot \frac{\partial R_2(r,p,h)}{\partial r} \cdot R_3(\theta_1,\theta_2,\theta_3) \cdot l_p^{LiDAR}(\alpha,\beta,d,K)$$

$$\frac{\partial R_2(r,p,h)}{\partial r} = \begin{bmatrix} 0 & \rho_{12} & \rho_{13} \\ 0 & \rho_{22} & \rho_{23} \\ 0 & \rho_{32} & \rho_{33} \end{bmatrix}$$

Where

$$\rho_{12} = \cos(r) \cdot \sin(p) \cdot \cos(h) + \sin(r) \cdot \sin(h)$$
$$\rho_{13} = -\sin(r) \cdot \sin(p) \cdot \cos(h) + \cos(r) \cdot \sin(h)$$
$$\rho_{22} = \cos(r) \cdot \sin(p) \cdot \sin(h) - \sin(r) \cdot \cos(h)$$
$$\rho_{23} = -\sin(r) \cdot \sin(p) \cdot \sin(h) - \cos(r) \cdot \cos(h)$$
$$\rho_{32} = \cos(r) \cdot \cos(p)$$
$$\rho_{33} = -\sin(r) \cdot \cos(p)$$

POS Pitch (p)

$$\frac{\partial P^{ECEF}}{\partial p} = \frac{\partial P_{INS}^{ECEF}}{\partial p} + \frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial p} + \frac{\partial L_{LiDAR-p}^{ECEF}}{\partial p}$$

$$\frac{\partial P_{INS}^{ECEF}}{\partial p} = 0$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial p} = \frac{\partial \left[R_1(B,L) \cdot R_2(r,p,h) \cdot l_{LiDAR}^{INS} \right]}{\partial p}$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial p} = R_1(B,L) \cdot \frac{\partial R_2(r,p,h)}{\partial p} \cdot l_{LiDAR}^{INS}$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial p} = \frac{\partial [R_1(B,L)\cdot R_2(r,p,h)\cdot R_3(\theta_1,\theta_2,\theta_3)\cdot l_p^{LiDAR}(\alpha,\beta,d,K)]}{\partial p}$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial p} = R_1(B,L)\cdot \frac{\partial R_2(r,p,h)}{\partial p}\cdot R_3(\theta_1,\theta_2,\theta_3)\cdot l_p^{LiDAR}(\alpha,\beta,d,K)$$

$$\frac{\partial R_2(r,p,h)}{\partial p} = \begin{bmatrix} -\sin(p)\cdot\cos(h) & \sin(r)\cdot\cos(p)\cdot\cos(h) & \cos(r)\cdot\cos(p)\cdot\cos(h) \\ -\sin(p)\cdot\sin(h) & \sin(r)\cdot\cos(p)\cdot\sin(h) & \cos(r)\cdot\cos(p)\cdot\sin(h) \\ -\cos(p) & -\sin(r)\cdot\sin(p) & -\cos(r)\cdot\sin(p) \end{bmatrix}$$

POS Heading (h)

$$\frac{\partial P^{ECEF}}{\partial h} = \frac{\partial P_{INS}^{ECEF}}{\partial h} + \frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial h} + \frac{\partial L_{LiDAR-p}^{ECEF}}{\partial h}$$

$$\frac{\partial P_{INS}^{ECEF}}{\partial h} = 0$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial h} = \frac{\partial \left[R_1(B,L)\cdot R_2(r,p,h)\cdot l_{LiDAR}^{INS}\right]}{\partial h}$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial h} = R_1(B,L)\cdot \frac{\partial R_2(r,p,h)}{\partial h}\cdot l_{LiDAR}^{INS}$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial h} = \frac{\partial [R_1(B,L)\cdot R_2(r,p,h)\cdot R_3(\theta_1,\theta_2,\theta_3)\cdot l_p^{LiDAR}(\alpha,\beta,d,K)]}{\partial h}$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial h} = R_1(B,L)\cdot \frac{\partial R_2(r,p,h)}{\partial h}\cdot R_3(\theta_1,\theta_2,\theta_3)\cdot l_p^{LiDAR}(\alpha,\beta,d,K)$$

$$\frac{\partial R(r,p,h)}{\partial h} = \begin{bmatrix} \xi_{11} & \xi_{12} & \xi_{13} \\ \xi_{21} & \xi_{22} & \xi_{23} \\ 0 & 0 & 0 \end{bmatrix}$$

Where

$$\xi_{11} = -\cos(p)\cdot\sin(h)$$
$$\xi_{12} = -\sin(r)\cdot\sin(p)\cdot\sin(h) - \cos(r)\cdot\cos(h)$$
$$\xi_{13} = -\cos(r)\cdot\sin(p)\cdot\sin(h) + \sin(r)\cdot\cos(h)$$
$$\xi_{21} = \cos(p)\cdot\cos(h)$$
$$\xi_{22} = \sin(r)\cdot\sin(p)\cdot\cos(h) - \cos(r)\cdot\sin(h)$$
$$\xi_{23} = \cos(r)\cdot\sin(p)\cdot\cos(h) + \sin(r)\cdot\sin(h)$$

POS X (x_{pos})

$$\frac{\partial P^{ECEF}}{\partial x_{pos}} = \frac{\partial P_{INS}^{ECEF}}{\partial x_{pos}} + \frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial x_{pos}} + \frac{\partial L_{LiDAR-p}^{ECEF}}{\partial x_{pos}}$$

$$\frac{\partial P_{INS}^{ECEF}}{\partial x_{pos}} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial x_{pos}} = 0$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial x_{pos}} = 0$$

POS Y (y_{pos})

$$\frac{\partial P^{ECEF}}{\partial y_{pos}} = \frac{\partial P_{INS}^{ECEF}}{\partial y_{pos}} + \frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial y_{pos}} + \frac{\partial L_{LiDAR-p}^{ECEF}}{\partial y_{pos}}$$

$$\frac{\partial P_{INS}^{ECEF}}{\partial y_{pos}} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial y_{pos}} = 0$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial y_{pos}} = 0$$

POS Z (z_{pos})

$$\frac{\partial P^{ECEF}}{\partial z_{pos}} = \frac{\partial P_{INS}^{ECEF}}{\partial z_{pos}} + \frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial z_{pos}} + \frac{\partial L_{LiDAR-p}^{ECEF}}{\partial z_{pos}}$$

$$\frac{\partial P_{INS}^{ECEF}}{\partial z_{pos}} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$\frac{\partial L_{INS-LiDAR}^{ECEF}}{\partial z_{pos}} = 0$$

$$\frac{\partial L_{LiDAR-p}^{ECEF}}{\partial z_{pos}} = 0$$