1-1-2003

Post-mission misalignment angle calibration for airborne laser scanners

Wendy Anne Dillane
Ryerson University

POST-MISSION MISALIGNMENT ANGLE CALIBRATION FOR AIRBORNE LASER SCANNERS

by

Wendy Anne Dillane, B. Eng
Ryerson Polytechnic University, 2001

A thesis

presented to Ryerson University

in partial fulfillment of the

requirement for the degree of

Masters of Applied Science

in the Program of

Civil Engineering

Toronto, Ontario, Canada, 2003

Ryerson University requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

Abstract

An Airborne Laser Scanning (ALS) system operates by locating returned laser pulses independently from all others. Locating the returned laser pulses requires knowing precisely for each laser pulse, the aircraft position (e.g. GPS), the attitude of the aircraft (e.g. IMU), the scanner angle when the laser pulse left the sensor, and the slant range to the terrain surface for that pulse. One of the most critical errors in ALS systems is the angular misalignment between the scanner and the IMU, which is called the misalignment or boresight error. This error must be addressed before an ALS system can accurately produce data. The purpose of this thesis was to develop and test a method of estimating the small misalignment angles between the laser scanner and the combined GPS/IMU solution for position and attitude. This method is semi-automated, requires no ground control and does not re-sample the ALS data in order to match the overlapping strips of data. A computer program called Misalignment Estimator was developed to estimate the misalignment angles using a least squares adjustment. The method was tested using a data set located at the Oshawa airport and provided by Optech. The misalignment angles were estimated to be -0.0178 degrees, -0.0829 degrees and 0.0320 degrees, for roll, pitch and heading respectively. The estimation of the misalignment angles was considered to be successful. Further research into automated point matching is recommended.

Acknowledgment

I would like to express much gratitude towards my advisor, Dr. M. Chapman, P. Eng., OLS. His patience and guidance were extremely beneficial throughout my research. Also, I want to thank Jim Green and Patrick Spencer from Optech, Toronto, for their time and efforts in acquiring data for my project. In addition, I would like to thank Michael Leslar and many other graduate students who kept me on my toes throughout the classes and thesis work. Lastly, I would like to thank Dad, Mom, James and Rob for their undying love and support. Your faith in me is the root of my faith in myself. Thank-you.

Table of Contents

List of Tables

List of Figures

1.0 Introduction

Since the late 1970's, Airborne Laser Scanners (ALS) have been used for accurate determination of terrain elevations. These early systems were complex and could obtain accurate elevation data. However, without the technology to determine the position and orientation of the sensor, ALS was not a cost-effective method for terrain elevation data acquisition over large areas.

Over the last few years, ALS technology has been improved with the developments of Global Positioning System (GPS) and airborne Inertial Measurement Units (IMU). ALS has now become a cost-effective and efficient method for terrain data acquisition. An alignment calibration process for the GPS position of the sensor and the orientation parameters between the IMU and the laser scanner is required to assure reliable accurate data sets from the ALS system.

1.1 Purpose

The objective of this project was to develop a method of estimating the small misalignment angles between the laser scanner and the Inertial Measurement Unit (IMU) using overlapping data strips. This method can be used to calibrate an ALS system after it has been installed in an aircraft. The concept of calibrating the alignment angles is to model and estimate the small misalignment angles between the sensors and adjust the data to compensate for the misalignment.

Optech Ltd., develops, manufactures and sells advanced ALS systems. It has an accurate method of calibrating the misalignment angles between the laser scanner and IMU in the lab prior to the installation of the system in an aircraft, but does not offer a standard method of post-mission calibration. There are established methods of misalignment estimation used by the industry; most are manual trial and error approaches that are known to be tedious and time consuming. Other methods involve semi-automated estimation, but these require ground control and/or re-sampling of the ALS data. Overall, the purpose of this thesis is to develop and test a method of estimating the small misalignment angles between the laser scanner and the IMU. The method is semi-automated, requires no ground control and does not re-sample the ALS data in order to match the overlapping strips. As such, Optech could offer their customers a post-installation calibration possibility.

1.2 Thesis Outline

In Chapter 2, the major concepts of Airborne Laser Scanning are discussed. The components of an ALS system, the laser scanner, the GPS and IMU, are described. Additionally, ALS system errors and data applications are reviewed.

In Chapter 3, the misalignment calibration problem is described. The underlying theory and mathematical models are discussed. Also, the method of estimating the misalignment angles is presented.

In Chapter 4, test data and the treatment and fusion of the ALS system data is presented. The experimentation with the proposed method is described. In addition, the resulting estimated misalignment angles and their accuracy are discussed.

In Chapter 5, the thesis is summarized and conclusions are drawn from the research work. Also, recommendations are made concerning potential future research.

2.0 Airborne Laser Scanners

Airborne Laser Scanning (ALS) systems are used to measure, by some means, the distance between the sensor and the illuminated spot on the ground. Laser scanning involves the act of deflecting a ranging beam off an object surface, in a certain pattern, so that the object surface is sampled with a high point density (Wehr and Lohr, 1999). Lasers are used as the ranging beam because high-energy pulses can be realized in short intervals and their comparatively short wavelengths can be highly collimated using small apertures. When lasers with high pulse repetition rates were available, laser-scanning systems could then obtain a range image, which is known as laser radar. There are two commonly used acronyms used to identify laser radar: LIDAR (*LI*ght *D*etection *A*nd *R*anging) and LADAR (*LA*ser *D*etection *A*nd *R*anging) (Wehr and Lohr, 1999).

2.1 Fundamentals of Airborne Laser Scanners

Laser scanners utilize opto-mechanical scanning assemblies and are active sensing systems using a laser beam as the sensing carrier. There are two optical beams that must be considered when designing an ALS; the emitted laser beam and the received portion of that beam (Wehr and Lohr, 1999). The laser scanner only measures the line of site vector from the laser scanner aperture to a point on the earth surface.

There are two major ranging principles for laser range measurement. In order to obtain a range measurement from a laser, the transmission must be modulated (Wehr and Lohr, 1999). Currently there are two methods of modulating a laser beam. The first one is pulsed ranging which is most popular and direct. Units that employ this method are referred to as Pulse Lasers. With pulse modulation the transmitter generates a rectangular pulse with widths of 10 ns to 15 ns (Wehr and Lohr, 1999). Measurement is determined by the time-of-flight of a light pulse. This is achieved by measuring the traveling time (t_L) between the emitted and received pulse.

$$t_L = 2R / c \qquad \text{(Eq. 2.1)}$$

where R is the distance between the ranging unit and the object surface

c is the speed of light

The second type of method of modulating a laser beam is the phase difference method, which must use a laser that continuously emits light at a set frequency. These units are called Continuous Wave (CW) lasers. For this method the phase difference between the transmitted and the received signal backscattered from the object surface is measured (Wehr and Lohr, 1999).

$$t_L = (\phi / 2\pi) * T + n * T \qquad \text{(Eq. 2.2)}$$

where ϕ is the phase difference between the transmitted and received signal

T is the period of the signal

n is the number of full wavelengths included in the distance from the laser transceiver to receiver.

Theoretically, the pulse system is able to achieve an 85 times higher accuracy than the CW system, but achieving higher accuracy requires very high technical efforts and sophisticated processing methods (Wehr and Lohr, 1999). On the other hand, by using CW systems, sub-centimeter level accuracy can be easily achieved by using a higher frequency (Wehr and Lohr, 1999). In current ranging laser systems, mostly solid-state pulsed lasers are used. This is related to the commercial availability of pulse lasers, rather than based upon any technical advantages.

The scanning process has a few key parameters; the laser footprint, the swath width and the laser point density. The laser footprint (A_L) mainly depends on the divergence of the laser beam (γ) (Wehr and Lohr, 1999).

$$A_{L\ instantaneously} = (h / \cos^2 (\phi_{inst})) * \gamma \qquad \text{(Eq. 2.3)}$$

where ϕ_{inst} is the instantaneous scan angle

h is the flying height

At a flying height of 500 metres, a typical laser footprint on the ground would be 30 cm in diameter.

The swath width (SW) mainly depends on the scan angle (ϕ).

$$SW = 2\ h \tan (\phi / 2) \qquad \text{(Eq. 2.4)}$$

The laser point density is of particular interest for survey tasks. Point spacing across flight direction mainly depends on either the pulse repetition frequency for a pulse laser, or on the measurement rate for a CW laser (Wehr and Lohr, 1999). Along the flight direction, the point spacing is determined by the ground speed of the aircraft and the period of one line scan.

The scan pattern on the ground not only depends on the laser scan pattern, but also the flying direction, speed, and the terrain topography (Wehr and Lohr, 1999). The points along a line are usually scanned in equal angle steps, but their spacing on the ground is not constant.

2.1.1 Basic Airborne Laser Scanner Relations and Formulas

Range (R) and Range Resolution (ΔR) of a Pulse laser:

$$R = c(t/2); \qquad \Delta R = c(\Delta t/2) \qquad \text{(Eq. 2.5)}$$

Time (t) is measured by a time interval counter relative to a specific point on the pulse, like on the leading edge, where the signal voltage has reached a predetermined threshold value (Baltsavias, 1999). A possible error occurs if the voltage magnitudes of the transmitted and received pulses are adjusted to the same value before they are sent to the time interval counter. An example of this occurs, if the received pulse amplitude is too low then the measured time will be too long and if the received pulse amplitude is too high then the measured time will be too short (Baltsavias, 1999). As long as the range resolution is significantly small so that the highest range accuracy can be achieved and small enough to permit accuracy investigations through repeated measurements, then it is not considered as important (Baltsavias, 1999).

Maximum unambiguous range depends on various factors, including the maximum range of the time interval counter and the pulse rate. To avoid confusion in the pulses arriving at the time interval counter, usually no pulse is transmitted until the echo of the pervious pulse has been received (Baltsavias, 1999). This does not limit the maximum achievable range. Maximum range is potentially limited by laser power, beam divergence, target reflectivity, detector sensitivity and atmospheric transmission. Best range performance is achieved when the atmosphere is cool, dry and clear and at night (Baltsavias, 1999).

Accuracy of range is dependent on the ability to select the same relative position on the transmitted and received pulse in order to measure the time interval and is limited by noise, signal strength, sensor sensitivity and reproducibility of the transmitter pulse (Baltsavias, 1999). It is also important that the accuracy by which fixed time delays in the system are known and the accuracy of the time interval counter is known. Detectors should have a large response at the wavelength to be detected. Also, very little noise should be added to the system by the detector, and should have a sufficient speed of response (Baltsavias, 1999).

2.2 Components of an Airborne Laser Scanning System

There are three major components of an Airborne Laser Scanning System. These components are the laser scanner, the Global Positioning System (GPS) and the Inertial Measurement Unit (IMU). The latter two combine to make the Positioning and Orientation System (POS), which is an essential component of an ALS system. Since the laser scanner only measures the line-of sight vector from the laser scanner aperture to a point on the earth surface and records this distance and the respective scanning angles, the 3-D position of these points can only be computed if the position and orientation of the laser system is known with respect to a coordinate system. This means that a Position and Orientation System, (POS), consisting of a differential GPS (DGPS) and IMU, must support the laser system. Geocoding of the laser scanner measurements requires an accurate synchronization of the IMU, GPS and the laser scanner data. During post-processing, the synchronization on the basis of one second time intervals enables the determination and correction of possible time errors between the highly stable GPS time and the local time of the laser's computer (Wehr and Lohr, 1999).

In order to deliver the accurate 3-D position of the laser points, three data sets are needed: the POS data; the laser distance measurements and their respective scan angles; and finally the calibrated data and mounting parameters between the POS and ALS. The calibrated data and mounting parameters include the three mounting angles of the laser scanner frame, omega, phi, kappa (ω, ϕ, κ), with respect to the platform-fixed coordinate system, usually with the origin at the IMU; the position of the laser scanner with respect to the IMU; and finally, the IMU's position and orientation with respect to the GPS antenna center (Wehr and Lohr, 1999).

2.2.1 Laser Scanner

Laser scanners have three key elements: the laser ranging unit, the opto-mechanical scanner and the control and processing unit.

2.2.1.1 Laser Ranging Unit

The laser ranging unit is comprised of the emitting laser and the electro-optical receiver. A laser uses coherent light for measurements. Laser stands for Light Amplification by Stimulated Emission of Radiation and uses a highly directional optical light beam and is highly consistent in space and time. The level of consistence is very dependent on the type of laser. Gas and solid-state lasers offer higher consistency than semiconductor lasers (Wehr and Lohr, 1999). The typical size of the transmitting and receiving apertures are 8 to 15 cm. These apertures are mounted so that the transmitting and receiving paths share the same optical path to ensure that objects' surface points illuminated by the laser are always in the field of view (FOV) of the optical receiver. The narrow divergence of the laser beam defines the instantaneous field of view (IFOV), which typically ranges from 0.3 to 2 mrad (Wehr and Lohr, 1999). The IFOV is a function of the transmitting aperture D and the wavelength of the laser light λ. Due to the very narrow IFOV of the laser, the optical beam has to be directed across the flight direction in order to obtain area coverage (Wehr and Lohr, 1999).

The opto-mechanical scanner is the mechanism that directs the emitted laser towards the ground. There are four typical scanning mechanisms: Oscillating mirror, Palmer scanner, Fiber scanner and rotating polygon. These mechanisms are shown in Figure 2.1. Figure 2.2 shows the scan patterns produced by each scanning mechanisms.

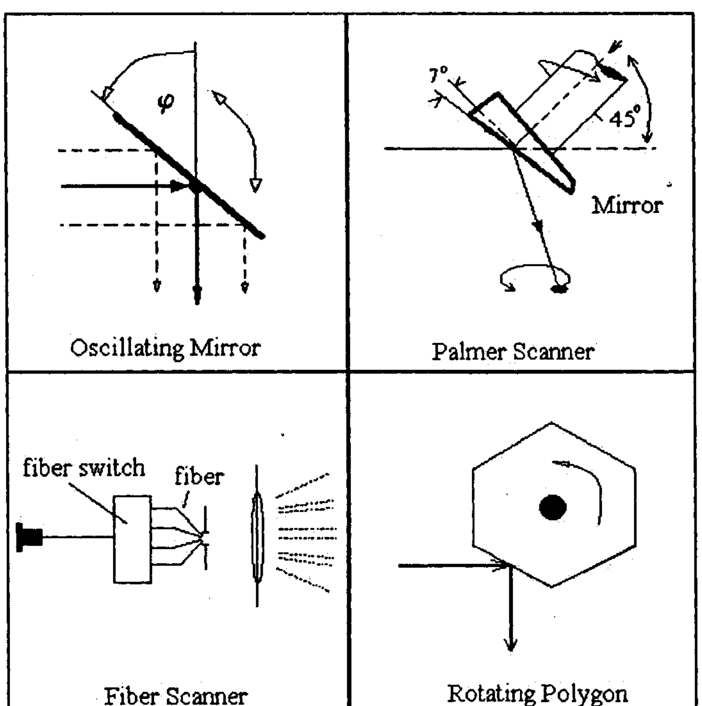

Figure 2.1 Four Typical Scanning Mechanisms

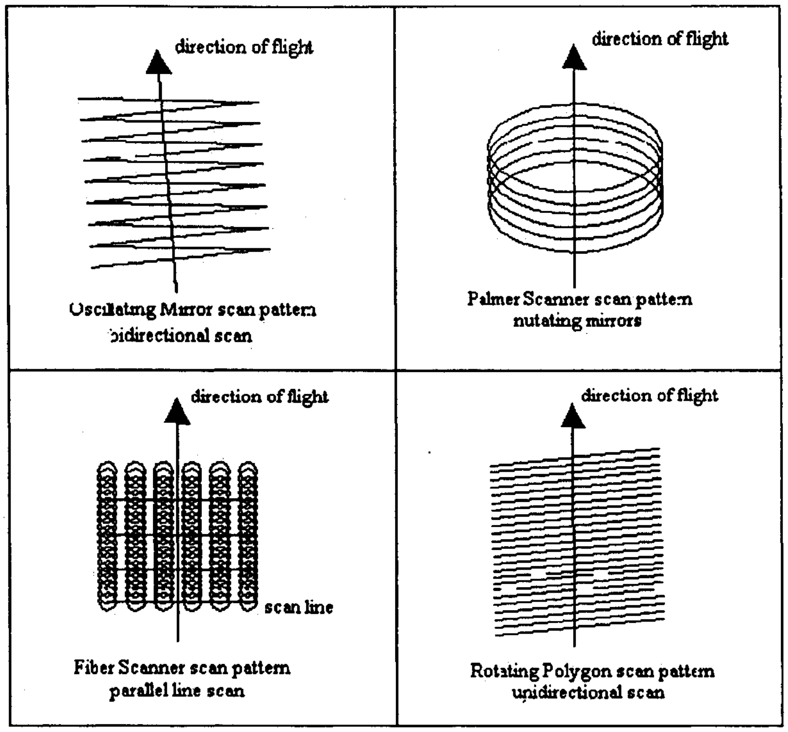

Figure 2.2 Scan Patterns for Typical Scanning Mechanisms

The Oscillating Mirror scanning mechanism incorporates a mirror that is rotated back and forth. This mechanism produces a zigzag-line (bi-directional scan) and companies such as Optech, Lieca Geosystems and TopEye have adopted this type of scanning mechanism (Morin, 2002). An advantage of this type of mechanism is that the mirror is always pointed at the ground, allowing continuous data collection. Also, the scan rate and the mirror's field of view can generally be controlled by the user. Another advantage is that oscillating mirrors can be integrated with the real-time output of the IMU, allowing the scanner to compensate for roll-type errors, which cause the edges of scans to be wavy (Morin, 2002). Disadvantages are numerous, including the changing velocity and acceleration of the mirror, which causes torsion between the mirror and the angular encoder. Corrections for such errors are required (Morin, 2002). The changing velocity causes the scan points to be unequally spaced on the ground and the point density is greater at the edges of scan lines as the mirror slows down, than at the nadir (Morin, 2002).

The second type of scanning mechanism is Palmer Scanner (nutating mirror). This scanning mechanism produces an elliptical scan pattern. The optical beam of the laser ranging unit hits the deflecting mirror, whose rotating axis is mounted so that the scanner shaft and the laser beam form an angle of 45 degrees (Wehr and Lohr, 1999). Also, the mirror is inclined by 7 degrees. This angle causes a nutation of the mirror when the scanner shaft is turning. Due to the elliptical scan, most of the measured points on the ground are scanned twice, once in the forward view; and once in the backward view (Wehr and Lohr, 1999). The redundant information is useful to calibrate the scanner and the POS in terms of the pitch angle is concerned.

A Fiber Scanner scanning mechanism produces parallel-line scans. The transmitting and receiving optics are identical, which is advantageous. An identical fiber line array is mounted in the focal plane of the receiving and transmitting lens. By means of two rotating mirrors, each fiber in the transmitting and receiving path is scanned sequentially and synchronously (Wehr and Lohr, 1999). These mirrors relay the light either from the central fiber to one of the fiber within the fiber array or vice-versa. The array transmits the pulse at a fixed angle onto the ground. The advantages of fiber scanners are, due to the small aperture of the fibers, only small moving mechanical parts are required; and high scanning speeds can be achieved, up to 630 Hz. This allows high pulse rates resulting in overlapping footprints (Wehr and Lohr, 1999). The disadvantage is that current technology only allows for a field of view that is smaller than a rotating mirror, therefore, covering less ground in a scan line (Morin, 2002).

The final type of scanning mechanism is a Rotating Polygon or constant velocity-rotating mirrors. This scanning mechanism produce parallel line (unidirectional) scans. The mirror is rotated in one direction by a motor and the angle measured either directly from the motor or from an angular encoder directly mounted to the mirror (Morin, 2002). The advantage of this scanning mechanism is that the mirror(s) is moved at a constant velocity, which means no acceleration type errors in the angle observations as with an oscillating mirror. The disadvantage of such a mechanism is that the mirror is not always pointing at the ground, therefore, not always collecting observations. Also, the field of view is less than an oscillating mirror's (Morin, 2002).

2.2.1.2 Control and Processing Unit

The control and processing unit regulates and monitors the emitted laser beam and the received portion of that beam (Wehr and Lohr, 1999). This component of a laser scanner records the line-of-sight vector and the respective scanning angles. This unit is also synchronized with and supported by a POS. Geocoding of the laser scanner measurements requires an accurate synchronization of the IMU, GPS and the laser scanner data. In order to provide this, timing is controlled by an internal clock, which is synchronized with GPS time by the use of the Pulse per Second (PPS) signal generated by the GPS receiver internal clock (Morin, 2002). Each measurement is tagged with a value from the internal clock. The tags are then referenced to the PPS and each combined range and angle measurement is assigned a GPS time tag (Morin, 2002).

2.2.2 Global Positioning System

Global Positioning System (GPS) is a one-way, all-weather positioning and timing system that provides continuous worldwide coverage (El-Rabbany, 2000). With the advent of GPS and the existence of a complete satellite constellation, GPS has made it possible to determine accurate time, position and velocity for a moving platform, in this case an aircraft. GPS is nominally composed of twenty-four satellites evenly spaced in six orbital planes. These orbits are nearly circular and have an altitude of about 20 200 km above the earth (Cosandier, 1999). The parent reference datum for GPS is WGS84.

The basic idea of positioning by using GPS is as follows. The GPS satellites send signals that are picked up by the GPS receiver. The receiver analyses the satellite signals to determine the distances to the satellites and interprets the navigation message to obtain the locations of the GPS satellites in space (El-Rabbany, 2000). The process of computing the position of the receiving GPS antenna is based on the principle of spatial trilateration, given by Equation 2.6.

$$\rho = [(x_s - x_r)^2 + (y_s - y_r)^2 + (z_s - z_r)^2]^{0.5} \qquad \text{(Eq. 2.6)}$$

where ρ is the distance between the antenna and the satellite.

x_s, y_s, z_s is the satellite position in an Earth-centered, Earth-fixed frame.

x_r, y_r, z_r is the receiver's antenna position in an Earth-centered, Earth-fixed frame.

An additional clock term is added to account for drift in quartz oscillators, which are employed in GPS receiver designs (Cosandier, 1999).

The GPS positioning mode used in the operation of the system under consideration is called Relative Positioning. Relative Positioning consists of two GPS receivers tracking the same satellites simultaneously to determine the relative coordinates. Figure 2.3 shows the use of Relative Positioning for determining the position of the remote receiver's antenna, which is mounted near the laser scanner.

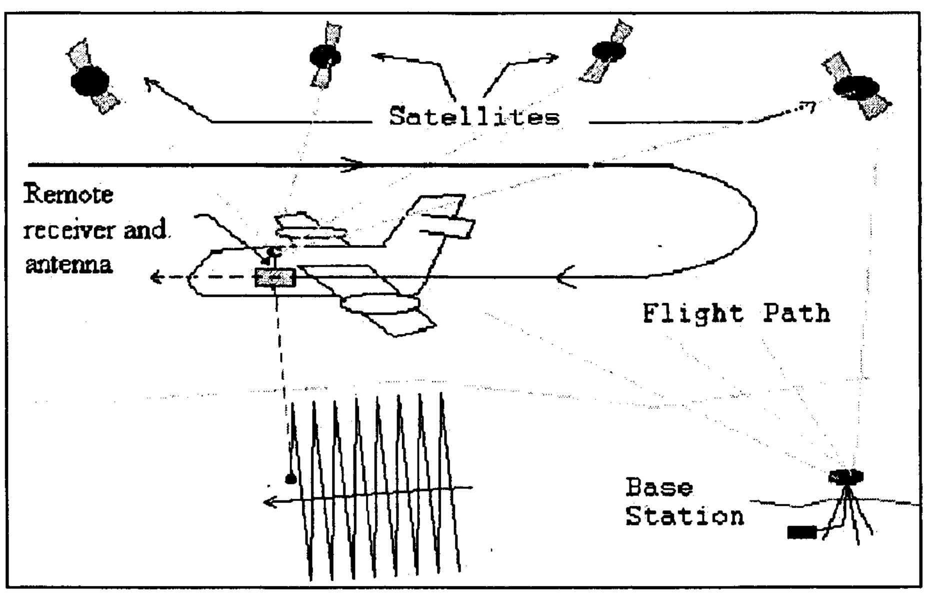

Figure 2.3 Relative Positioning Used with Aerial Remote Sensing

For Relative Positioning, the satellite coordinates X_s, Y_s, and Z_s, the distances between the antennas and the satellites and the base station coordinates X_b, Y_b, and Z_b, must be known in order to determine the relative coordinates, X_r, Y_r, and Z_r, of the onboard remote antenna.

The GPS signal structure has five components: L1, L2, C/A-code, P-code and navigation message. L1 and L2 are two carrier frequencies that all GPS satellites transmit.

The Course Acquisition - Code (C/A-code) is transmitted on the L1 frequency only and each satellite is assigned a unique code. The Precision - Code (P-code) is transmitted on both the L1 and L2 frequencies. This code is 266 days long and sent at a rate of 10.23 Mb/s. Each satellite is assigned a unique one-week segment of this code. The final component of the signal structure is the navigation message, which is transmitted on both the L1 and L2 frequencies. Within the message is orbital information called the ephemeris, clock data and almanac, which contains future orbit information, etc (El-Rabbany, 2000). GPS receivers are classified by their receiving capabilities. A Single Frequency receiver has access to the L1 frequency, while a Dual Frequency receiver has access to both L1 and L2. The Dual Frequency receivers are the most accurate receiver type and can produce geodetic quality measurements (El-Rabbany, 2000).

The distance between a receiver and a satellite can be calculated in two ways. The first method is called Pseudo-Range measurement. The satellites and the receiver generate the same code, at the same time. The receiver measures the time offset (Δt) between the code of the satellite and the code of the receiver and multiplies this by the speed of light to find the pseudorange of the satellite to the receiver (El-Rabbany, 2000). The second method is called Carrier Phase measurements. The initial phase measurement is ambiguous by an integer number of cycles. This ambiguity remains constant with time, as long as the receiver remains locked on the signal. Resolving the ambiguities is the key for centimeter level accuracy and this can be done in real time or through post-processing (El-Rabbany, 2000).

2.2.3 Inertial Measurement Unit

An Inertial Measurement Unit (IMU) is comprised of triads of accelerometers and gyros that measure incremental velocities and angular rates. Hutton et al. (circa 1997) explain that these are numerically integrated by a strapdown navigator to produce geographically referenced positions, velocities and attitudes. This system is also known as an Inertial Navigation System (INS). In order to produce accurate position and attitude, the accelerometers and gyros in the IMU must be of high quality with low noise and drift. Typically, IMUs are comprised of force rebalance or integrated silicon accelerometers and either Fiber Optic Gyros (FOG), Ring Laser Gyros (RAG) or Dry Tuned Gyros (DTI). The IMU used in the system under investigation is small and lightweight, and is mounted directly on the laser scanner, as shown in Figure 2.4. This ensures the best possible accuracy with minimized flexure between the IMU and the laser scanner. This is a key feature in airborne scanning where the highest possible accuracies are required (Hutton et al., circa 1997).

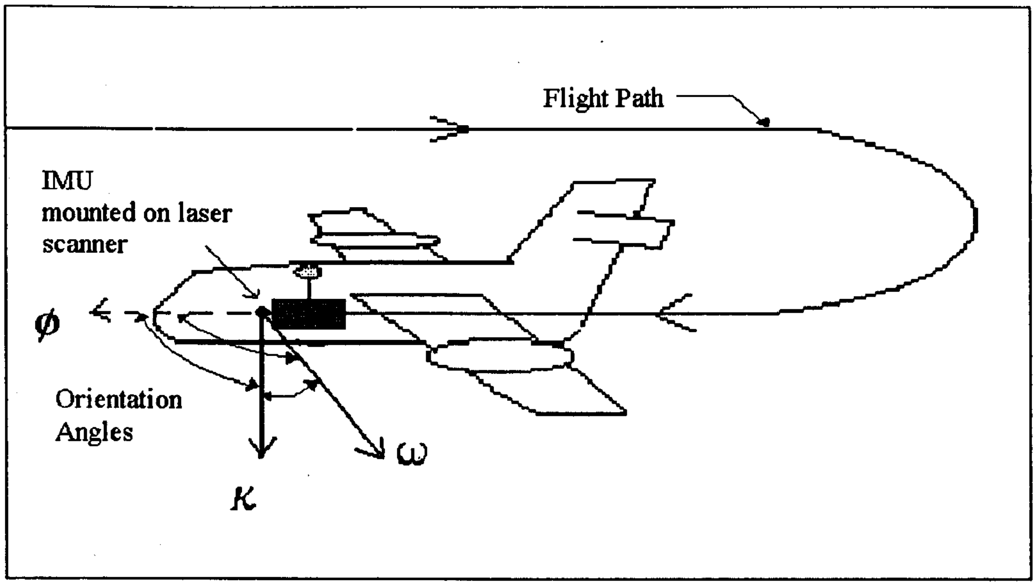

Figure 2.4 The Onboard Inertial Measurement Unit

In general, an IMU consists of three gyroscopes and three accelerometers. The gyroscopes are used to sense angular velocity ω_{ib}^{b} that describes the rotation of the b-frame with respect to the i-frame and is coordinated in the b-frame, where the b-frame is a fixed body coordinate frame and the i-frame is a properly defined inertial reference frame and can be considered as being non-accelerating and non-rotating. The accelerometers are used to sense specific force f^{b} in the b-frame. The angular velocities ω_{ib}^{b}, are integrated in time to provide orientation changes of the body relative to its initial orientation. The specific force measurements f^{b} are used to derive body acceleration which, after double integration with respect to time, give position differences relative to an initial position (El-Sheimy, 1996). The specific force measurements f^{b} and the angular velocities ω_{ib}^{b} can be used to determine all parameters required for trajectory determination by solving the following system of differential equations shown in Equation 2.7.

$$
\begin{bmatrix} r^m \\ v^m \\ R_b^m \end{bmatrix} = \begin{bmatrix} D^{-1} v^m \\ R_b^m \cdot f^b - \left(2W_{ie}^m + W_{em}^m\right) \cdot v^m + g^m \\ R_b^m \cdot \left(W_{ib}^b + W_{im}^b\right) \end{bmatrix} \qquad \text{(Eq. 2.7)}
$$

where:

D^{-1} is the scaling matrix, obtained through the integration process using the implied reference ellipsoid

g^m is the gravity vector, normally approximated by the normal gravity field

W is Earth rotation and assumed to be known with sufficient accuracy

(El-Sheimy, 1996)

In practice, due to the double integration of the IMU acceleration data, the time-dependent position errors will quickly exceed the accuracy specifications for many trajectory determination applications. In order to correct these error within the multi-sensor mapping system, a Kalman filter is used. A Kalman filter processes measurements that match the IMU solution with GPS data to obtain estimates of the significant errors in the IMU data. The Kalman filter uses the complementary nature of these error characteristics to calibrate the inertial navigation errors and aiding sensors with respect to each other (Hutton et al., 1997 circa).

2.2.4 IMU and GPS Data Integration

It is known that an IMU alone cannot achieve highly accurate position, velocity and attitude information, while GPS alone cannot provide highly accurate attitude (Sun, 1994). When IMU and GPS data are integrated, the position, roll, pitch and heading measurements can be highly accurate.

Two filters: a centralized filter or a decentralized filter, can integrate GPS and IMU. A centralized filter integrates GPS carrier phase measurements with IMU gyro and accelerometer output. A decentralized filter integrates IMU output and GPS output from their separate filters to a master filter. Generally, a centralized filter produces a tighter integration (Sun, 1994).

The integration procedure using a centralized filter is as follows (Sun, 1994): to begin with, the IMU prediction is performed at an interval given by the IMU data until the IMU time tag and GPS time match, then the GPS double difference carrier phase is computed. The double difference carrier phase is compared with the predicted values based on the IMU predicted position. If the difference is larger than a threshold, then a cycle slip has been detected and is, subsequently, corrected. After the correction of the double difference carrier phase data, the predicted values are used to update the IMU in a centralized Kalman filter. The position and attitude information output from the update are based upon the final GPS/IMU integration result (Sun, 1994). This process is repeated for the entire GPS and IMU data stream. The GPS/IMU integration scheme is shown in Figure 2.5.

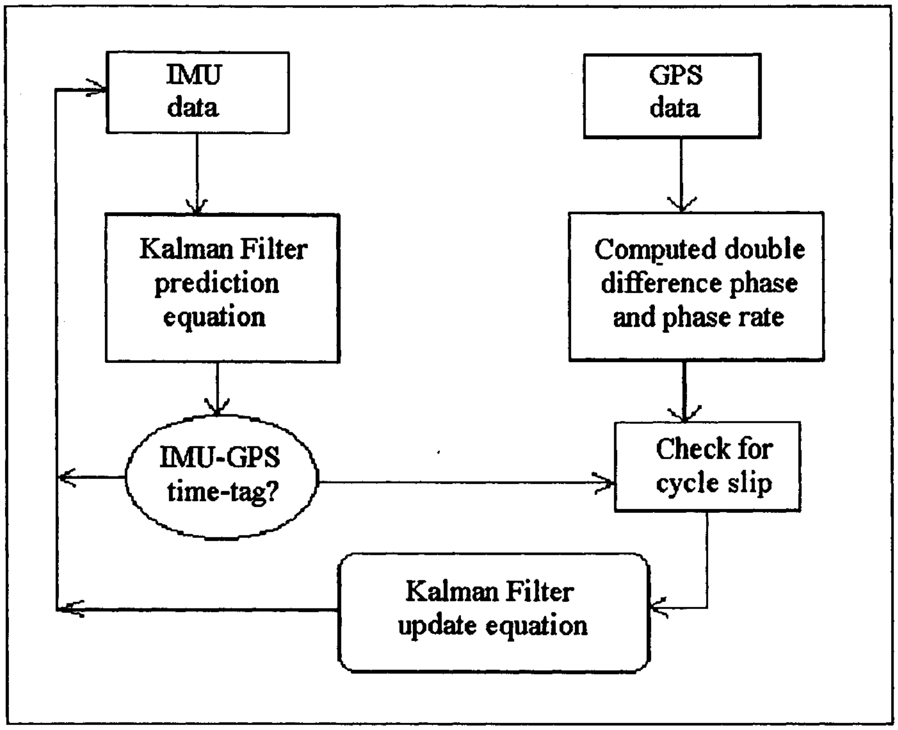

Figure 2.5 GPS/IMU Integration Scheme (Sun, 1994)

2.3 System Errors

2.3.1 Atmospheric Errors

The most important systematic atmospheric error relates to the atmospheric refraction of the laser signal. Simply stated, the laser signal is refracted, or bent, as it moves through the atmosphere. The refraction is a function of wavelength, attitude, scan angle and atmospheric conditions. Atmospheric refraction can be modeled by using a standard atmospheric model.

$$\Delta\alpha = K \cdot \tan\alpha \qquad \text{(Eq. 2.8)}$$

where α is the recorded scan angle

K is the atmospheric correction value

And $\Delta\alpha$ is the defraction angle

$$K = \left(\frac{2410H}{H^2 - 6H + 250} - \frac{2410H}{H^2 - 6H + 250}\left(\frac{h}{H}\right) \right) \times 10^{-6} \qquad \text{(Eq. 2.9)}$$

where h is the elevation of the ground about the datum

H is the elevation of the scanner above the datum

Clearly, the amount of deflection ($\Delta\alpha$) is proportional to the scan angle and the amount of defraction reaches a maximum at the edge of the scan and zero for nadir scans. The commercial airborne ALS system manufactures' atmospheric refraction models are mostly proprietary, but they are very similar to the above standard atmospheric model used in photogrammetry (Morin, 2003).

In addition to the angle correction, as seen in Figure 2.6, the distance the pulse travels along the curved ray path is different from the theoretical ray path and must be adjusted accordingly.

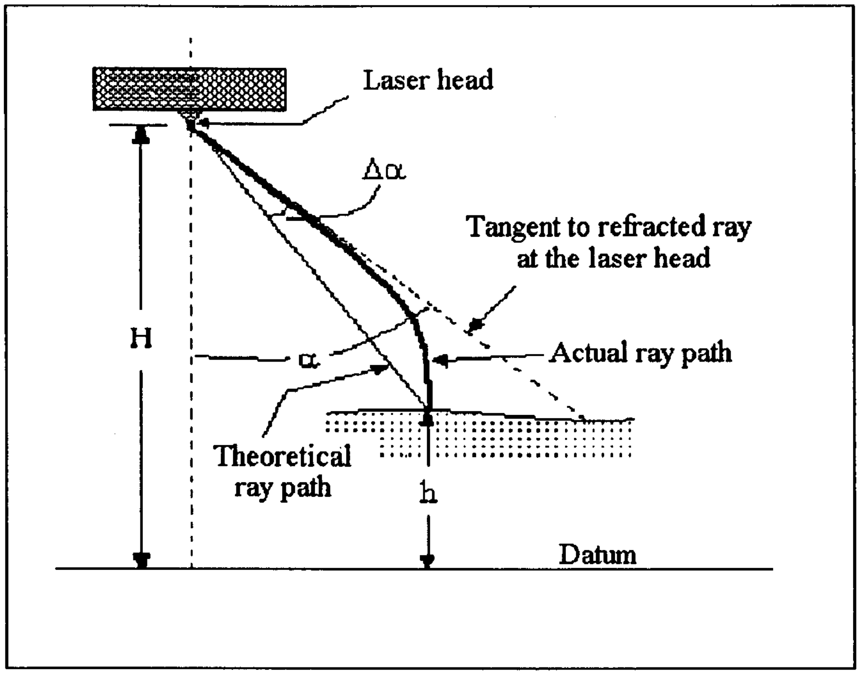

Figure 2.6 Atmospheric Refraction

Another atmospheric error that must be considered is that the received power of the range signal can be reduced by poor atmospheric conditions. Dust and smoke in the atmosphere can cause small amounts of energy to reach the receiver as backscatter and therefore complicate target identification. These random errors can mostly be reduced by proper mission planning and eliminated with filtering during post-processing (Morin, 2003).

2.3.2 Scanner Errors

One of the most noticeable errors in laser scanning is the measurement of the scan mirror rotation angle. The primary factor that influences errors in the measurement of the scan mirror rotation angle is the method of measurement. A controlling device called a galvanometer sets the mirror's position. The galvanometer itself can measure the scan mirror rotation directly, but this method of measurement does not offer the highest accuracy. Such measurement systems have shown accuracies of approximately 0.02 degrees, while another method, the use of a separate angle-encoding device, allows measurement accuracies of approximately 0.001 degrees (Morin, 2003). This can improve the laser point accuracy from decimeter to centimeter accuracy (Morin, 2003). A typical angle encoder consists of a glass disc with regular etch marks and a small laser which accurately measures the etch marks. The basic components of an angle encoder are shown in Figure 2.7.

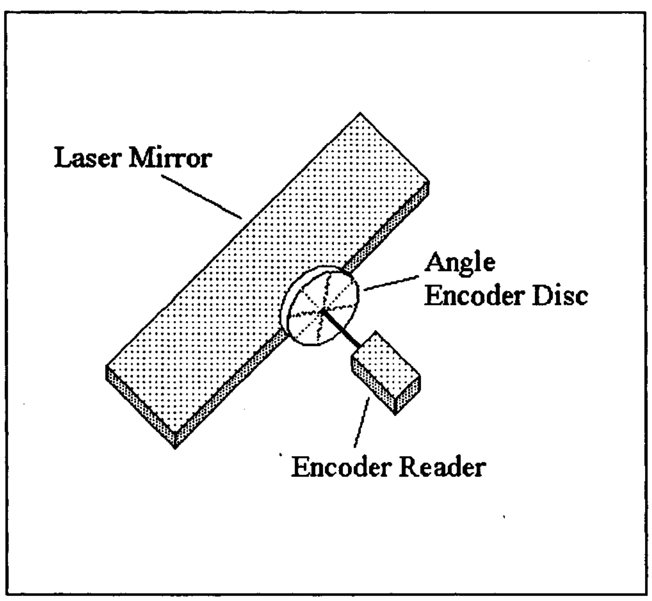

Figure 2.7 Scan Angle Encoder

The angle encoder should be precisely mounted perpendicular to the scan mirror's rotation angle to eliminate projection errors with the glass etches. If the encoder is not mounted precisely, then the projection error must be modeled or calibrated to ensure the highest accuracy from the angle encoder (Morin, 2003).

Another scanner error that affects the accuracy of the angular encoder is a timing delay in the control unit itself. The error that is created by this time delay is not a physical error, like an angle encoder error, but a systematic error that is found in the derived ground points. This timing delay error or encoder latency error occurs when the control unit does not respond quickly enough thus causing a delay between when the range measurement is recorded and when the scan angle is recorded. The delay causes the scan angle to be recorded or registered to an incorrect angle (Morin, 2003).

Even though the latency error is a systematic error, the effects of this error are visible within the derived ground points. Because an oscillating mirror comes to a stop at the edges of the FOV, the recorded scan angles are quite close in magnitude to one another and, therefore, have a minimal error. Also, as the scan approaches the nadir, slant range shortens and the visible error minimizes again. This creates a visual pattern called a "bow tie" in the derived points on the ground (Morin, 2003). Figure 2.8 illustrates how the error grows between the edge of the FOV and nadir and again from nadir to the edge of the FOV.

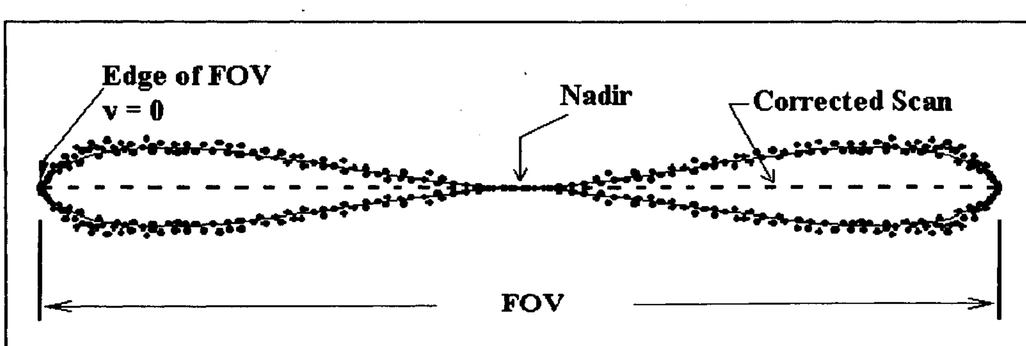

Figure 2.8 "Bow Tie", the Visible Effects of Encoder Latency on Derived Ground Points.

Encoder latency error can be as high as 15 microseconds and for a system operating at 45° FOV and a 20 Hz scan rate, 15 microseconds of latency delay could cause up to a 0.027 degree angular difference (Morin, 2003).

A latency error can be corrected by manually changing the magnitude of the latency value. When the "bow tie" pattern disappears from the derived points, the latency value has been determined. Once determined, the value can be applied to all other data sets, because it is a constant function of the control unit.

In addition, scanner torsion is another source of scanner error. Scanner torsion is caused by the changes in velocity and acceleration of the oscillating scanner mirror. Due to the nature of the oscillating scan pattern, the mirror is repeatedly accelerated and decelerated in order to reverse its direction. The scan mirror is attached to the drive motor and the angle encoder disc by a lever, as shown in Figure 2.9. The drive motor rotates the lever to move the mirror.

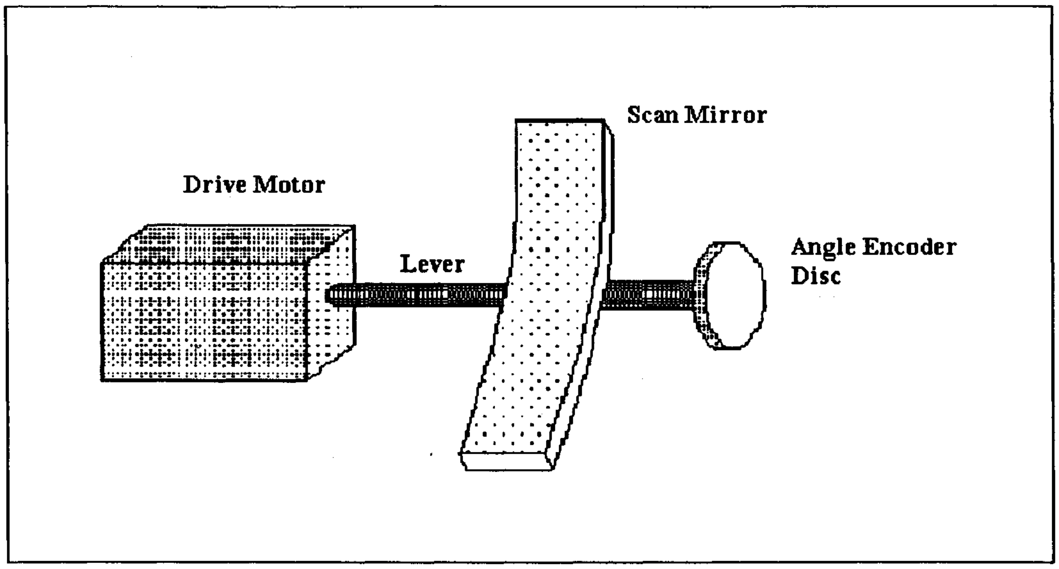

Figure 2.9 Scan Mirror Components

The mirror and the encoder have mass thus they have momentum. Figure 2.10 illustrates the velocity and acceleration of the scanner mirror and encoder at pivotal points, point A, B and C when the mirror is moving counter-clockwise. .

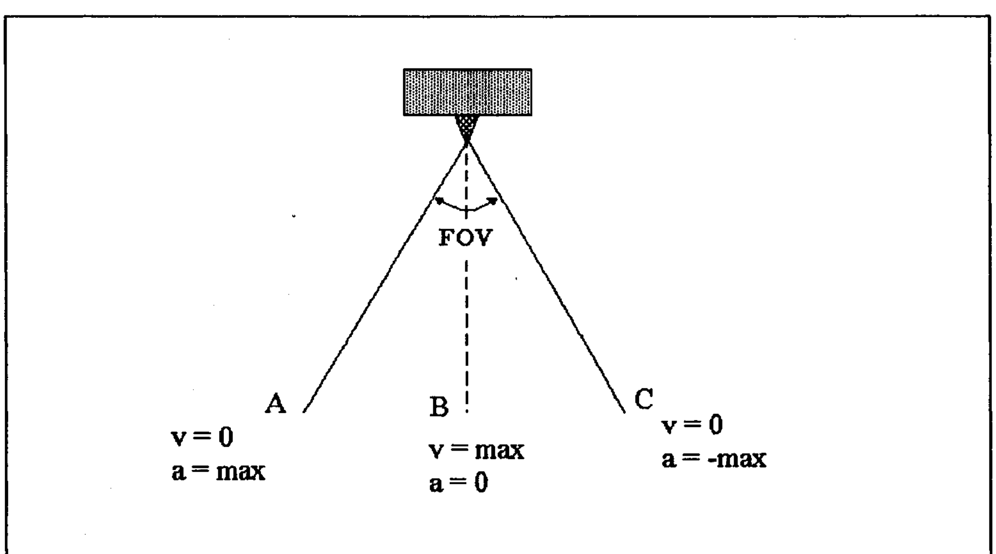

Figure 2.10 Velocity and Acceleration of the Scan Mirror

At point A, the mirror is stopped to reverse direction, so the velocity at this point is zero, but the applied force required to reverse the direction of the mirror means the acceleration is at a maximum (Morin, 2003).

At point B, halfway through the scanner's FOV, the mirror is moving at its fastest. Therefore, velocity is at its maximum, immediately before the mirror starts to slow down in order to reverse direction at the edge of the FOV. Because the mirror is about to slow down the acceleration is at zero (Morin, 2003).

At point C, the mirror has once again come to a stop in order to reverse direction, therefore, velocity equals zero. In order to stop, the mirror must have a maximum negative acceleration or maximum deceleration at this point.

Because of the large acceleration forces needed to move the mirror across the FOV between 15 to 50 times a second, the lever undergoes torsion. The torque applied by the drive motor causes torsion or a twisting of the lever. The overall effect of the torsion is that the encoder lags behind the mirror, which, itself, lags behind the position of the drive motor (Morin, 2003).

The difference in angular position between the scan mirror and the encoder causes misregistrations of the observed distances. Over flat terrain, the range measurement at the mirror position will be shorter than the range measured at the encoder. Thus, an incorrect shorter range is recorded at the encoder position. The difference between the true range and the measured distance increases proportionally with the scan angle, the FOV and the scan rate (Morin, 2003). The observable effect in the data is strip bowing, commonly known as sensor "smile" as shown in Figure 2.11.

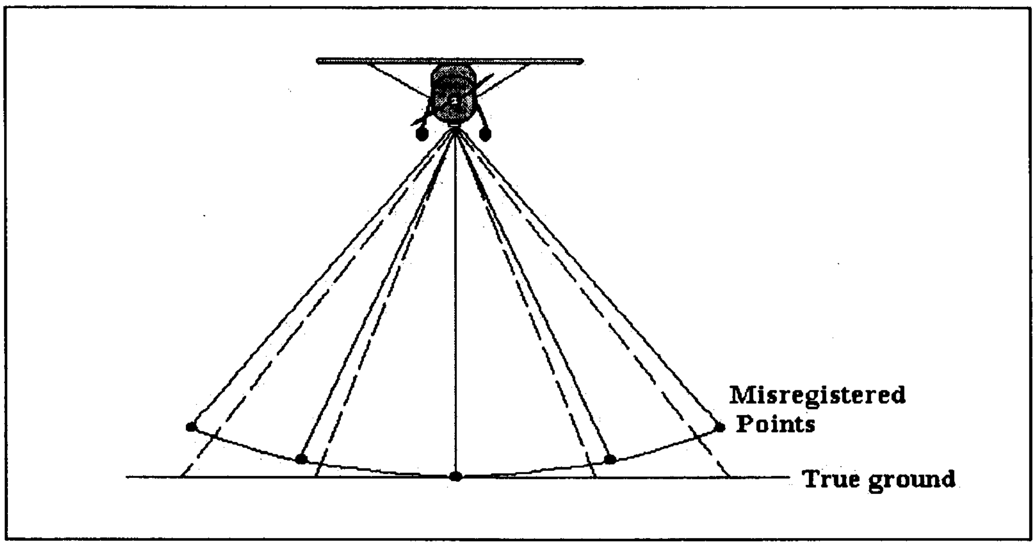

Figure 2.11 Sensor "Smile" or Strip Bowing of Scanner Torsion Error

If the encoder is located closer to the drive motor on the lever than the mirror, the "smile" becomes a "frown", with registered distances being too long (Morin, 2003).

This angular error can be modeled using the equations for momentum and torque, but a simple correction model can also be applied by modeling the parabolic smile, i.e.:

$$\beta = \beta_o + \beta_o \cdot c \qquad \text{(Eq. 2.10)}$$

where β_o is the encoder angle

c is an angle correction constant.

A negative value of c will bring in the encoder, aligning the mirror and encoder and flatten out the bowing or "smile" (Morin, 2003). Trial and error can be used to determine *c* the angle correction constant.

2.3.3 Position and Attitude Accuracy

Position accuracy depends mainly on the quality of the differential GPS (DGPS) postprocessing. Through the integration of GPS with IMU, GPS errors are smoothed and achieve typical accuracies of 5-15cm (Baltsavias, 1999). The derived accuracies also depend on GPS satellite constellation during flight, the distance of ground reference stations from the aircraft; and the accuracy of offsets and misalignment between the GPS and IMU and the IMU and laser scanner.

Attitude accuracy depends on the quality of the IMU data, the IMU interpolation error, the method of postprocessing, and the integration with the GPS. The effect of attitude errors on the 3D accuracy increases with the flying height and the scan angle. Possible time misregistration errors between the different sensors can have a ruinous effect on the position and attitude accuracy (Baltsavias, 1999). Orientation, position and range are required to be recorded at the same time. If there is a time offset between any or all of the sensors and this is not known, it will cause systematic errors. A small time offset between range and rotation angles can cause large errors in the 3D accuracy for a turbulent flight. The accuracy of the transformation from WGS84 to the local coordinate system and corrections for geoid undulations is also another consideration (Baltsavias, 1999).

2.3.4 GPS Errors

Cycle slips are discontinuities in GPS carrier phase observations caused by temporary signal loss. This error is caused by obstructions, radio interference, severe ionosphere and/or high receiver dynamics. Other errors and biases of GPS relate to the atmosphere, the satellite and the receiver (El-Rabbany, 2000). Table 2.1 outlines these errors, their causes and ways of correcting or preventing them.

Table 2.1 Common Errors in GPS (El-Rabbany, 2000)

Error	Cause	Correction or Prevention
Broadcast Ephemeris Errors	- result of improper prediction of satellite positions	-"Precise Ephemeris" can be applied during postprocessing
Satellite Clock Error	- the clocks are highly accurate, but not perfect	- remove through differencing between two receivers - applying satellite clock correction (in navigation message)
Receiver Clock Error	- receiver clocks less accurate then the satellite clocks	- removed through differencing between two satellites
Multipath Error	- when the satellite signal arrives at the antenna through different paths (causes problems in solving the ambiguities)	- careful site and antenna selection
Antenna Phase Centre Variations	- the point at which the signal is received is generally not the physical centre of the antenna	
Receiver Measurement Noise	- result of the limitations of the receiver's electronics	- a good system should have a minimum noise level
Ionosphere Delay	- the region between 50 km and 1000 km above the Earth speeds up the carrier phase the same amount that it slows down the pseudorange	- can be removed or reduced
Tropospheric Delay	- the region between 50 km and the Earth's surface, affect the height component more then the vertical	

Satellite geometry is another source of error in GPS. Bad satellite geometry enlarges the area of uncertainty for positioning. Such poor satellite geometry can be prevented by using the signal for satellites that are 10° above the horizon (El-Rabbany, 2000). The area of good geometry above the horizon is shown in Figure 2.12. Overall, the errors in GPS can be removed, prevented or modeled, except for cycle slips that must be dealt with by compensating for them with other GPS data or with other navigation sensor data, such as IMU data.

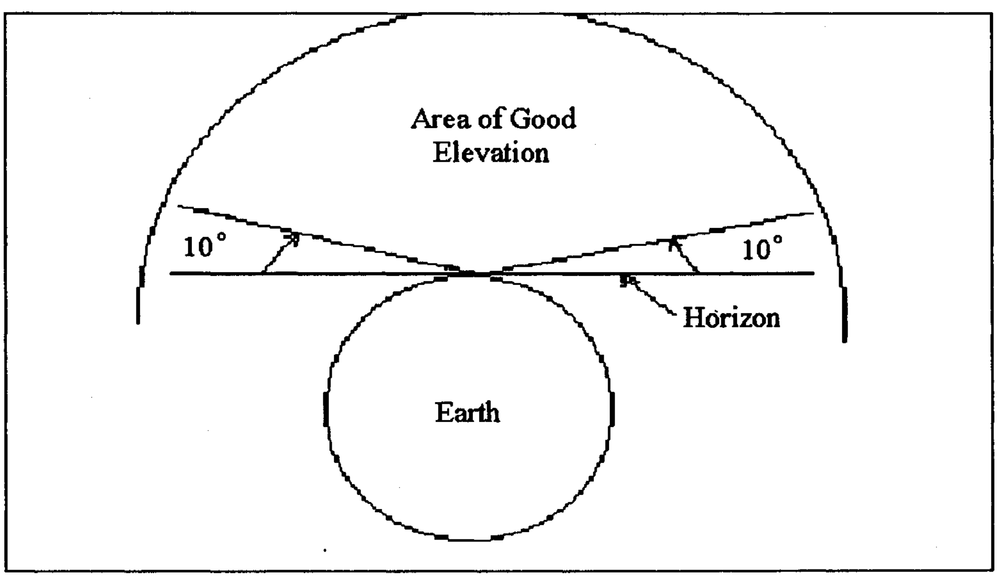

Figure 2.12 Satellite Geometry Above the Horizon

2.3.4.1 Cycle Slip Correction

The occurrence of cycle slips is an important remaining problem in GPS carrier phase positioning. A cycle slip is a change of $\Delta\nabla N$ (double difference carrier phase ambiguity) by an unknown integer number from one epoch to the next due to blockage of the signal path or signal noise (Sun, 1994). There several of ways to detect and correct cycle slips.

Since an IMU has a very good short-term stability, it can be used to detect GPS carrier phase cycle slips in a number of seconds, depending on the performance of the IMU (Sun, 1994).

When using the IMU to correct for cycle slips, the IMU is used to predict the position of the aircraft at the measurement epoch base on the last epoch. The double difference carrier phase can then be computed from this prediction. A threshold has to be given to determine the presence of a cycle slip. This threshold should be smaller than 0.5 cycles in order to reliably detect small cycle slips, but this threshold may need to be less stringent in order to avoid the incorrect detection of cycle slips (Sun, 1994).

2.3.5 Misalignment Error

A critical error in ALS systems is the angular misalignment between the scanner and the IMU, which is called the misalignment or "Boresight" error. This error must be addressed before an ALS system can accurately produce data. The magnitude and orientation of the errors induced by this misalignment are a function of several parameters, such as, flying height, scan angle, and flight direction. The components of the misalignment error (roll, pitch and heading) are illustrated in Figure 2.13.

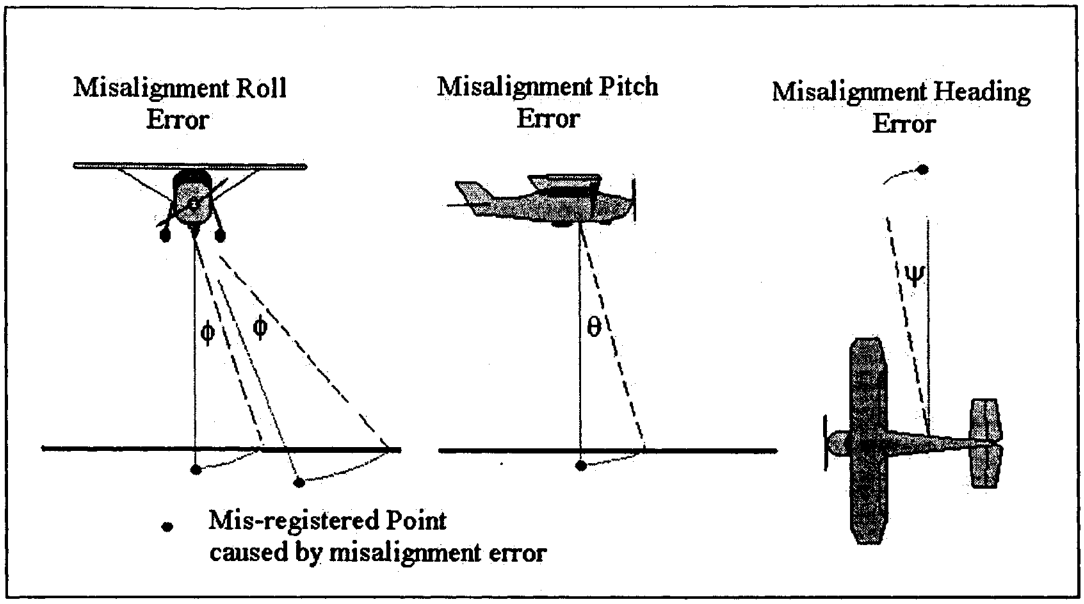

Figure 2.13 Misalignment Errors

The misalignment between the laser scanner and the IMU causes each laser point observation to be registered with an incorrect aircraft attitude, resulting in an incorrect point position. The roll error, shown in Figure 2.13, causes a slant range to be incorrectly registered compared to the orientation of the aircraft. The elevation difference tends to increase with larger scan angles. The pitch error also results in a laser slant range to be incorrectly registered compared to the orientation of the aircraft. As the slant range is longer, the entire strip tends to be pushed down.. The heading error induces a skewing in each scan line.

Unlike an image, a misalignment error affects each observation and cannot be modeled by observing the induced errors in position of the control points. These errors are correlated with flying direction. In order to correct these errors, they must be decorrelated. Observing targets taken from different flying directions accomplishes this. This is the underlying principle of the automatic misalignment estimation model.

2.4 Airborne Laser Scanner Data Applications

There are several applications for airborne laser scanner data. ALS is an accurate, fast and versatile measurement system, which can complement or even replace other existing geo-data acquisition technologies (Wehr and Lohr, 1999). The advantages of using ALS data are the ease of data acquisition, the increased ability to determine surface elevations in difficult areas and the expeditious data delivery. Also, the processed data is "digitally-ready" for GIS applications. Table 2.2 shows a variety of ALS data applications.

Table 2.2 Applications of ALS Data

Mapping using ALS Data	Corridors, such as roads, railway tracks, pipelines and waterway landscapes Electrical transmission lines and towers Flood and drainage pattern mapping Rapid mapping and damage assessment after natural or other disasters Wetlands Coastal areas and hydrographic surveys in depths up to 60 m
DTM and DEM Generation	Forested areas Urban areas Wetlands
Measurement	Coastal areas, dunes, tidal flats and determination of coastal change and erosion Volume calculations for open pit mining Snow and ice cover, i.e. glacier monitoring Derivation of vegetation parameters, i.e. tree height, crown diameter, tree density, biomass estimations
Modeling	3-D city models, i.e. animation, planning relay antenna locations for wireless communications

ALS data has attributes, such as high point collection density; registration of multiple echoes; and amplitude or intensity registration, that increase the value of the data and make ALS data uniquely proficient for the above applications (Axelsson, 2000). The high point collection density provides more accuracy in diverse landscapes. The point density is dependent on flying height, platform velocity, the sensor's field of view (FOV) and the sampling frequency of the scanner (Axelsson, 2000). The required point density varies according to the end user's application and mission planning is very important. The ability of ALS sensors to register multiple echoes of the returning pulses is how these sensors are able to collect points through vegetation and even below water surfaces. Also, this capability can be used to identify rapidly changing elevations (Axelsson, 2000). In the case of Lidar-based systems, the ability of the sensor to register the return pulse's amplitude or intensity allows the data to be viewed as an image, based on the points' x, y and intensity values. This allows for quick object identification such as separating pavement from grassland (Axelsson, 2000).

2.4.1 Filtering methods for DEM Generation

In order for the raw or unedited ALS data to be used for the applications mentioned previously, such as digital elevation models (DEM) or digital terrain models (DTM), the data must be filtered. Filtering involves the processing of the data to remove unwanted data, such as blunders and undesired objects, like vegetation or buildings. There are several filtering methods for each type of application.

2.4.1.1 Linear Prediction for Filtering

In the course of processing airborne laser scanner data to generate a digital terrain model (DTM), filtering is applied to the data in order to remove the height values of vegetation-horizons and buildings. Lohmann (2000), presents an approach to filtering laser scanner data called linear prediction and applies this method to different data sets.

Linear prediction is a statistical interpolation method and is based on the correlation of neighboring points, expressed in the covariance function. The covariance between two points, P_i and P_k depends on their spacing. If the points are close to each other then the covariance will be high. Initially, a polynomial of a very low degree or a moving plane is created for the data in order to define a trend function. The moving plane is defined by three unknown coefficients a_0, a_1, and a_2

$$z_i = a_0 + a_1 X_i + a_2 Y_i \qquad \text{(Eq. 2.11a)}$$

$$\underline{z} = z_i - (a_0 + a_1 X_i + a_2 Y_i) \qquad \text{(Eq. 2.11b)}$$

The result is the residual vector $\underline{\mathbf{z}}$, which contains the centered points of measurement z_i. These values describe the deviations of the sample points from the trend functions.

The predicted values u_i are estimated from the residuals $\underline{\mathbf{z}}$ and the described covariance information contained in $\underline{\mathbf{C}}$ using a least squares adjustment.

The interpolated surface for another point, P_j, is given by

$$u_j = \underline{\mathbf{c}}^T \, \underline{\mathbf{C}}^{-1} \, \underline{\mathbf{z}} \qquad \text{(Eq. 2.12)}$$

$$c^T = \begin{bmatrix} C(P_jP_1) & C(P_jP_2) & \cdots & C(P_jP_n) \end{bmatrix}$$

$$C = \begin{bmatrix} 1.0 & C(P_1P_2) & \cdots & C(P_1P_n) \\ C(P_2P_1) & 1.0 & \cdots & C(P_2P_n) \\ \vdots & \vdots & \ddots & \vdots \\ C(P_nP_1) & C(P_nP_2) & \cdots & 1.0 \end{bmatrix}$$

$$z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix}$$

$$P_j = a_0 + a_1X_j + a_2Y_j + u_j$$

where $\mathbf{c}^T$ the covariance between the point to be interpolated and the measurements

C the covariance matrix

z the vector of centered measurements

Note: All measurements are weighted equally.

Lohmann (2000) uses a program called DTMCOR in order to apply linear prediction to the data sets. Within the program, the user defines two tolerance factors. The first one is in respect to the moving plane. Any height values that are above the given tolerance are excluded as outliers. This is repeated iteratively until no more heights are rejected. The second tolerance factor checks the difference between the centered measured value Z and the predicated value, u_j, calculated by the software. If the residuals are bigger than the tolerance these points are also rejected.

Lohmann (2000) found that the method of linear prediction was very good for forested areas but that large buildings can cause problems. Also, the linear prediction was not well suited for undulating terrain because it results in too much smoothing.

2.4.1.2 Filtering Using Adaptive TIN models

Axelsson (2000), presents an approach to filtering laser scanner data using adaptive Triangular Irregular Network (TIN) models. This method of filtering was specifically designed to generate DEMs for complex and discontinuous ground surfaces, like dense cities and complex construction sites.

This filtering algorithm is based on a TIN. The TIN is a vector data structure for representing geographical information that is modeled as a continuous field that uses tessellated triangles of irregular shape, based on Delaunay triangulation (Burrough and McDonnell, 1998). Delaunay triangulation is a method of graphing obtained by joining pairs of points whose polyhedra are Thiessen divisions of the plane. Thiessen polygons are a tessellation of the plane such that any given location is assigned to a tile according to the minimum distance between the location and a single previously sampled point (Burrough and McDonnell, 1998). Tessellation is the process of dividing an area into smaller, contiguous tiles with no gaps in between them (Burrough and McDonnell, 1998).

Because this method is designed for complex and discontinuous ground surfaces, a high point density of greater than 1 point per m^2 is needed (Axelsson, 2000). The high point density enables structures and break lines to be accurately identified and modeled.

Before the TIN is created, statistics are collected from the entire data set. The parameters for the initial TIN densification are found using histograms of surface normal angles and elevation differences to identify median values (Axelsson, 2000). These parameters are the distance to the TIN facets and angles to the nodes. The identified median values are used to estimate the parameter thresholds (Axelsson, 2000).

The next step of the algorithm is selecting the seed points to create a sparse TIN. The seed points are selected within a user-defined grid, the size of which is based on the largest type of structure (Axelsson, 2000). Once the sparse TIN is created, its density is then increased using an iterative process. One point at a time is added to each TIN facet or tile, if it meets the criteria based on the threshold parameters. With each iteration, new threshold parameters are calculated using only the points in the current TIN (Axelsson, 2000).

The method described so far is satisfactory for simple, continuous surfaces. However, discontinuous surfaces that this algorithm was designed to deal with, have edges that could easily be cut off because the threshold values are quickly exceeded with these edges (Axelsson, 2000). So, to avoid the edge cut-offs, the TIN facet circumventing the point representing the edge and the surrounding area are examined. Mirroring the point that exceeds the threshold, to the closest node point, achieves this goal. The deviation from that TIN facet is calculated. If the mirrored point meets the criteria based on the threshold parameters, in the mirrored position, then the original point is accepted (Axelsson, 2000). Clearly, the acceptance of this point and others like it within the same iteration adjusts the threshold parameters for the next iteration, allowing more edge identifying points to be accepted by the TIN algorithm. The iterative process is stopped when no more points are added to the TIN surface (Axelsson, 2000). Figure 2.14 shows the adaptive TIN model. In Figure 2.14, box 1 shows the TIN with the edge cut off; box 2 shows an edge point being tested for the TIN model; box 3 shows a mirror of the edge point being tested; and box 4 shows the accepted edge point in the new TIN model.

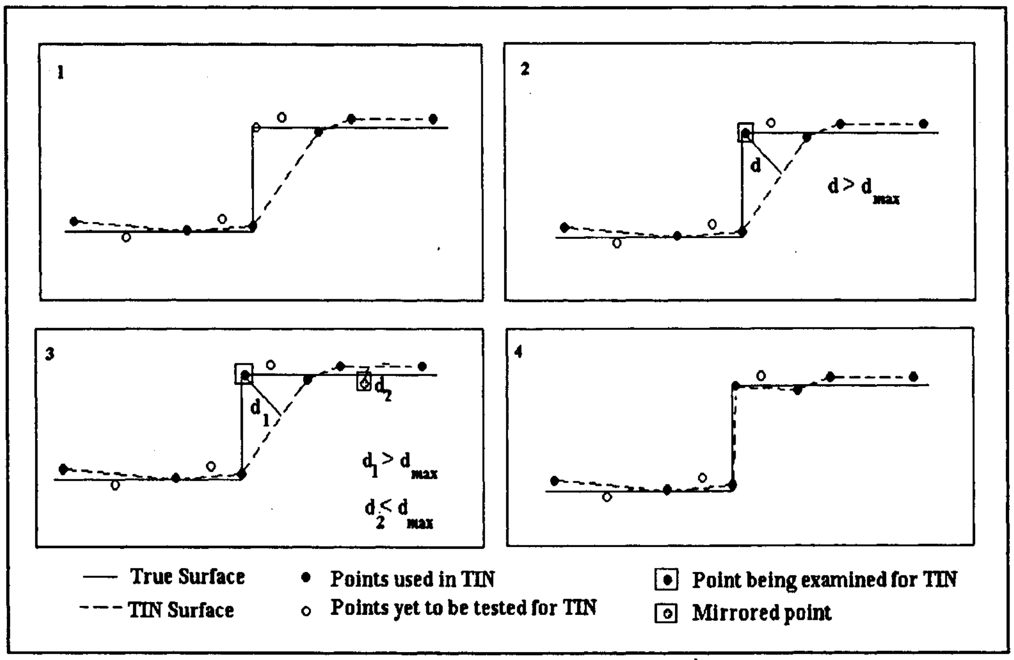

Figure 2.14 The Adaptive TIN Model (Axelsson, 2000)

2.4.2 Laser Scanning Applications for Forestry

The ability to remotely sense both the total quantity and spatial organization of forest biomass would provide a way to meet the need for forest inventory. Traditional optical sensors are only capable of providing detailed information on the horizontal distribution and limited vertical distribution of vegetation in forests (Lefsky et al., 1999). Airborne laser scanning systems are capable of providing both horizontal and vertical information with the horizontal and vertical sampling dependent on the type of ALS used and its configuration (Lima et al., 2003). Forest attributes such as canopy height can be directly retrieved from Lidar data. Direct retrieval of canopy height provides opportunities to model aboveground biomass and canopy volume. Access to the vertical nature of forest ecosystems also offers new opportunities for enhanced forest monitoring, management and planning (Lima et al., 2003).

Laser scanning systems are able to directly measure the vertical structure of a forest because the scanners employ multiple measurements of both distance and the amount of energy reflected from the many surfaces of the forest (Lefsky et al., 1999). The height and intensity returns are recorded from both the vegetation and the soil, from the top of the canopy to the ground.

Laser scanning used in forestry applications can be categorized as either 'discrete return' systems or 'full waveform' systems and differ from one another with respect to how they vertically and horizontally sample a canopy's three-dimensional structure (Lima et al., 2003). Figure 2.15 depicts both discrete return data and full waveform data.

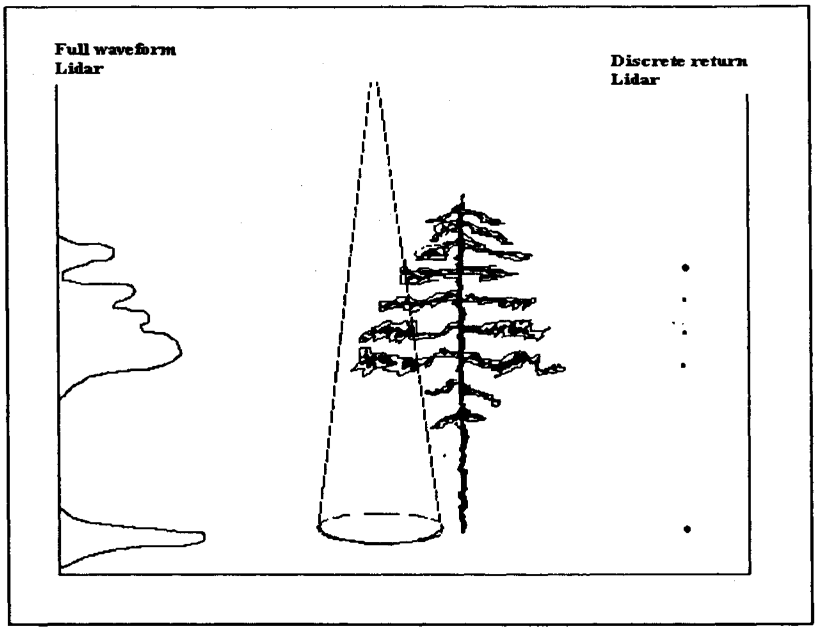

Figure 2.15 Differences between discrete return and full waveform Lidar data (Lima et al., 2003)

The vertical sampling of Lidar systems relates to the number of range samples recorded for each emitted laser pulse. The horizontal sampling is determined by the area of the footprint and the number of such footprints, or 'hits', per unit area (Lima et al., 2003).

A requirement for calculating canopy heights, using both discrete return and full waveform scanner data, is the ability to identify some ground reference level below the canopy. In the case of discrete return Lidar data, canopy height estimates are calculated by taking the difference between those Lidar returns not classified as ground and a surface representative of the terrain. This representative surface is typically interpolated from remaining returns after the point cloud has been filtered using a vegetation removal algorithm (Lima et al., 2003). For full waveform Lidar data, canopy heights can be calculated by converting the elapsed time difference between the peaks of the two most prominent modes in the amplitude waveform into range (Lima et al., 2003).

Lidar-generated canopy profile area is a significant variable in estimating gross merchantable timber volume. One approach is to use tree height measurements to accurately derived stand biomass and gross-merchantable volume by indirectly modeling (Lima et al., 2003). This principle can use discrete return data. Another approach is to transform the waveform, for full waveform Lidar data, into a canopy height profile and then, using a threshold, classify the profile into four zones (Lima et al., 2003): (i) empty, which includes no canopy or ground material; (ii) elements in the uppermost 65% of the canopy closure; (iii) area of the remaining elements not included in the uppermost 65% or empty zone; and (iv) open gap volume. Subsequently, total volume for each of the four zones for the entire area of study can be calculated for a set of waveforms in a study plot (Lima et al., 2003).

Another forest management strategy is to identify different types of tree stands within a forest. There are three different types of stands to be considered, young stands, mature stands and old-growth stands. Young stands have dense canopies and little variability in tree size. Therefore, the resulting scanner data would reflect a packed, mono-layer canopy (Lefsky et al., 1999). Old-growth stands have large live trees, high diversity of tree diameter, high diversity of species and the presence of large standing snags (dead trees) and fallen logs. The resulting scanner data would reflect high diversity of tree heights and the presence of multiple canopy layers (Lefsky et al., 1999). Matures stands have attributes of which lie between the other two types of stands.

2.4.3 Laser Scanning Mapping of Coastal Zones

Traditional hydrographic surveys of coastal zones employ acoustical techniques operated from launch-type vessels or the use of electronic distance measuring devices (Irish and Lillycrop, 1999). These traditional methods are considered slow and provide low-density measurements of the surveyed area. In the 1970's, the first generation airborne lidar bathymetry system was developed. This greatly increased the ability to accurately map coastal zones (Irish and Lillycrop, 1999).

Bathymetry is the measurement of sea depths. Airborne lidar bathymeter measures water depths by using a laser pulse that travels to the air-water interface where a portion of the energy is reflected back to the receiver (Irish and Lillycrop, 1999). These are the surface returns. The remaining energy transmits through the water and reflects off the sea bottom. These are the bottom returns. This is shown in Figure 2.16.

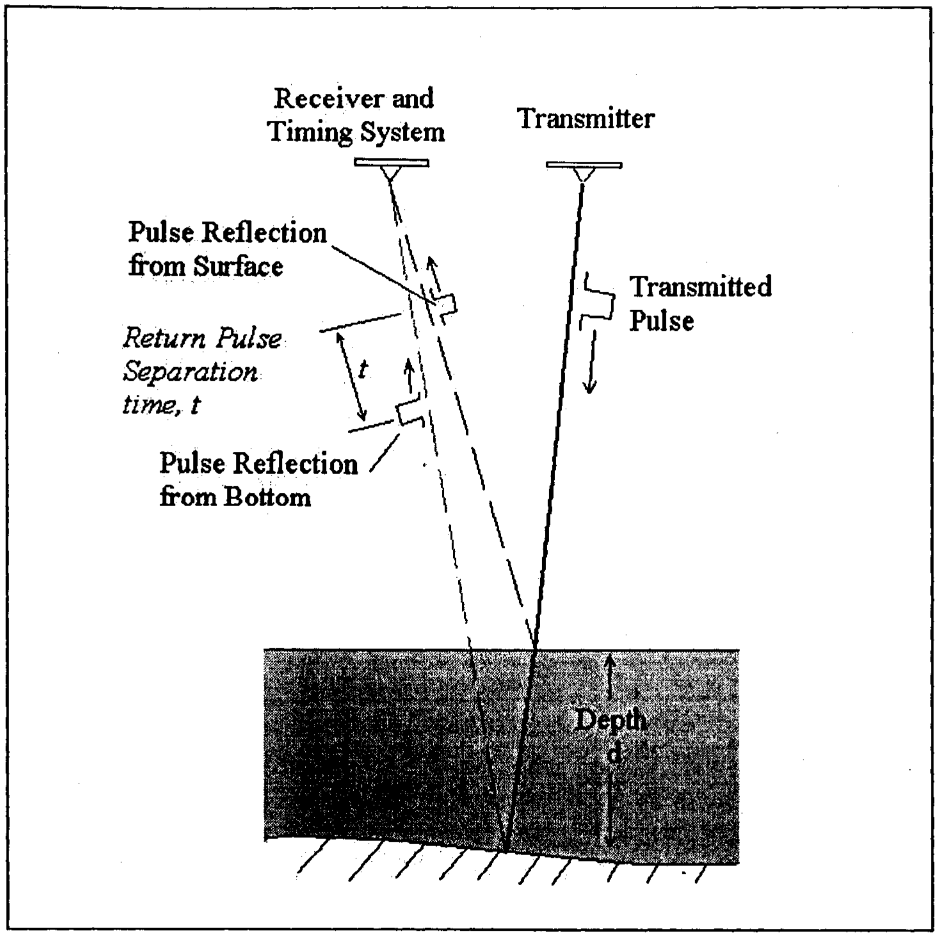

Figure 2.16 Laser Scanning Bathymetry Principle

The depth of the water is calculated using the time difference between the surface return and the bottom return (Irish and Lillycrop, 1999). The speed of light through water is taken into account during this calculation. The maximum detectable depth is limited by refraction, scattering and absorption, which reduce the strength of the bottom return. Water clarity and bottom type are also major limiting factors for depth detection (Irish and Lillycrop, 1999). Laser scanner bathymeters can collect through depths up to 60 m, which is 3 times the visible depth (Irish and Lillycrop, 1999).

In 1994, the US Army Corps of Engineers developed an airborne lidar bathymetry system called SHOALS (Scanning Hydrographic Operational Airborne Lidar System). This system employs laser scanner technology to remotely collect accurate, high-density measurements of both bathymetry and topography in coastal regions (Irish and Lillycrop, 1999). The SHOALS system uses a scanning, pulsed infrared (1064 nm) and blue-green (532 nm) laser transmitter. The infrared frequency was selected to optimize air-water interface detection. The blue-green frequency was selected to optimize water penetration, up to 60 m (Irish and Lillycrop, 1999). The blue-green frequency is also employed to measure topographic elevations. This is unlike most topographic laser scanner systems that use an infrared frequency. The system typically operates at a very low altitude of 200 m and the speed of 60 m/s, resulting in a swath width of 110m. The survey rate is approximately 16 km^2/h, which is several times faster then conventional fathometer survey rates (Irish and Lillycrop, 1999). The SHOALS system also employs a geo-referenced down-look video camera to obtain approximate positions of coastal structures, navigation aids and other objects of interest. The types of projects that the SHOALS system has been used for include mapping maintained channels and harbours, coastal structures, the creation of nautical charts and monitoring on- and off-shore coastal erosion (Irish and Lillycrop, 1999).

3.0 The Misalignment Calibration Problem

An ALS works by locating laser hits independently from all others. Locating the laser hits requires knowing precisely for each laser shot, the aircraft position from GPS, the attitude of the aircraft from IMU, the scanner angle when the laser pulse left the sensor, and the slant range to the ground for that pulse (Smith, 2002). A frequent problem with ALS data is that it shows systematic shifts in elevation and horizontal position when compared to other data sources and systematic differences between overlapping data from the same sensor (Morin, 2003). These types of systematic errors are largely due to calibration of the sensors. The most offending parameters are the alignment angles, omega, phi, kappa, between the positioning and orientation sensors and the laser-scanning sensor (Morin, 2003). The concept of calibrating the alignment angles is to identify the small misalignment angles between the sensors and to adjust the data to compensate for the misalignment.

3.1 Pre-Flight Calibration

An important step in collecting accurate ALS data is ensuring proper preflight calibration and initialization of the integrated sensors. The laser range finder and the offsets between scanner and the navigational sensors must be calibrated. Also, the GPS and IMU need to be initialized.

The laser range finder can be calibrated on the ground for its zero-bias and the variable receiver bias. Receiver bias is commonly known as "range walk" and is caused by changes in the intensity versus pulse detection (Morin and El-Sheimy, circa 2002). In order to calibrate these biases while the platform is stationary, the laser is directed to a fixed target on the ground. The distance to the target is measured with either a steel tape or an independent electronic distance measurement (EDM) system. By varying the power of the emitted laser, from the detector threshold to saturation, a curve can be established to model the apparent range based on receiver power. This curve can then be used in post processing to compensate for the receiver biases (Morin and El-Sheimy, circa 2002).

The offset between the systems can also be measured with a steel tape or through a total station survey. The measurement should be made to within 1-3 cm to minimize the offset error impact on the data (Morin and El-Sheimy, circa 2002). Any uncorrected errors in the offsets will be propagated as a bias onto the target coordinates. It should be noted that the effect of errors in the offset are minimal compared to errors in the sensors' alignment angles.

The navigation sensors must be initialized as per manufacture's specifications. For DGPS, this involves setting up near a reference base station and determining the integer ambiguities (Morin and El-Sheimy, circa 2002). It should be noted that in-flight precautions must be taken so the integer ambiguities are not lost. Avoiding large flight dynamics, like sharp turns and sudden dives, does this. As for the IMU, it needs to be initialized in order to detect and establish the local level frame (Morin and El-Sheimy, circa 2002).

3.2 Methods of Misalignment Calibration

There are different methods of calibration of the alignment between the positioning and orientation sensors and the laser-scanning sensor. The most common method is a simple manual adjustment of the misalignment angles. This method is practical, but also time consuming, biased prone, and can require ground control (Smith, 2002). Another method of calibrating the misalignment angles is called semi-automatic boresight misalignment determination (Toth, 2002). This method can be less time consuming and requires no ground control.

3.2.1 Manual Misalignment Calibration

Manual misalignment calibrations are generally a simple trial and error approach, where the operator interactively changes the misalignment angle to reach some best-fit with respect to some known surface. There are two types of manual calibration techniques, relative and absolute. The relative calibration is performed without ground control and looks for self-consistency within the data. This technique matches data with data from adjacent swaths and matches data from crossing flight lines. The absolute calibration is performed using ground control and compares the absolute location of laser points or features identified by laser points to accurate reference data (e.g. ground control) (Smith, 2002).

In order to match and compare laser reference points for accuracy evaluations for horizontal positions building edges are used due to the sudden change in elevation. For accurate vertical positions, control points on a flat surface, such as a parking lot, are used to compare the elevation of all the laser points in the area (Smith, 2002).

The methodology used by Optech for roll and pitch calibration, for post installation of an ALS system, is to use scanning passes over building in opposite directions. The roll and pitch error shows up as displacement in the position of the edge of building that is direction dependent (Smith, 2002). By flying over the building in each direction four times, the average results are the roll and pitch offsets. Horizontal positioning accuracy calculations are performed by using building edge points to determine x, y position accuracy, applying roll, pitch, and scanner scale error corrections and then uses the building to evaluate the accuracy of positions of building edges (Smith, 2002).

3.2.2 Semi-automated Misalignment Calibration

The basis of this method is to estimate the unknown misalignment angles between the positioning and orientation sensors and the laser-scanning sensor. This is done using the horizontal and vertical discrepancies between overlapping strips, possibly without the use of ground control and without manually manipulating the data (Toth, 2002).

The effectiveness of this method depends heavily on the point density of the ALS points and the overall terrain characteristics of the overlapping area. Smoothly rolling terrain is ideal for test areas because it exhibits limited surface undulations while the horizontal and vertical discrepancies between overlapping strips can still be identified (Toth, 2002).

A frequently used technique for matching the data is to interpolate the ALS points into a regular grid. The discrepancies then can be determined by surface matching of the selected regions within the overlap or by profile matching of man-made objects. Once the surface differences are known, a least squares adjustment can be performed for the unknown misalignment angles (Toth, 2002).

3.3 Semi-automated Misalignment Estimation Methodology

For this research, a semi-automated misalignment estimation technique is developed so that semi-automated misalignment calibration can be achieved after an ALS system is installed in an aircraft. Optech currently provides an in-lab calibration of the misalignment angles, but only uses a manual misalignment estimation method for post installation flight calibration.

This method does not require any ground control and is based on using two overlapping strips of ALS data that have been flown in different directions. The point differences for the separate strips over the same area are considered as observations and an adjustment is formulated to determine the misalignment angles.

Based on the observed differences in matched laser ground points, the misalignment angles are iteratively adjusted to reduce the surface discrepancies in object space using the least squares adjustment method of estimation. The coordinates of a laser point are a function of the exterior orientation of the laser sensor and the laser range vector (Toth, 2002). The mathematical model that represents the relationships between the observations, which are the exterior orientation of the laser sensor and the laser range vector, and the derived quantities, which are the misalignment angles (Anderson and Mikhail, 1998). Equation 3.1describes these relationships:

$$r_{M,k} = r_{M,INS} + R_{INS}^{M}(R_{L}^{INS} \cdot r_{L} + b_{INS}) \qquad \text{(Eq. 3.1)}$$

where $r_{M,k}$ is the 3D coordinates of point K in the mapping frame

$r_{M,INS}$ is the 3D INS coordinates in the mapping frame

R_{INS}^{M} is the rotation matrix between the INS frame and the mapping frame, measured by the GPS/INS

R_{L}^{INS} is the boresight matrix between the laser frame and the INS frame

r_{L} is the 3D object coordinates in the laser frame

b_{INS} is the boresight offset component (Toth, 2002)

The objective is to eliminate the surface differences by estimating the correct rotation angles between the INS and the laser scanner. Since the boresight-offset components are mostly negligibly small and also are not amplified by object distance, they are ignored in this investigation.

By rotating the range vector by the correct boresight angles (R_{L}^{INS}) the correct ground coordinates can be calculated using Equation 3.2.

$$\begin{bmatrix} X_g \\ Y_g \\ Z_g \end{bmatrix}^{corr} = R_{INS}^{M} R_{L}^{INS} \begin{bmatrix} X_g \\ Y_g \\ Z_g \end{bmatrix}^{L} + \begin{bmatrix} X_a \\ Y_a \\ Z_a \end{bmatrix}^{M} \qquad \text{(Eq. 3.2)}$$

Where

$\begin{bmatrix} X_g \\ Y_g \\ Z_g \end{bmatrix}^{corr}$ is the corrected ground coordinates in the mapping frame

R_{INS}^{M} is the rotation matrix between the INS frame and the mapping frame, measured by the GPS/IMU

$\begin{bmatrix} X_g \\ Y_g \\ Z_g \end{bmatrix}^{L}$ is the range vector in the laser system

$\begin{bmatrix} X_a \\ Y_a \\ Z_a \end{bmatrix}^{M}$ is the laser frame coordinates in the mapping frame at the time of measuring the ground point

The rotation matrix between the INS frame and the mapping frame is given by:

$$R_{INS}^{M} = \begin{bmatrix} \cos\phi \cos\kappa & -\cos\phi \sin\kappa & \sin\phi \\ \cos\omega \sin\kappa + \sin\omega \sin\phi \cos\kappa & \cos\omega \cos\kappa - \sin\omega \sin\phi \sin\kappa & -\sin\omega \cos\phi \\ \sin\omega \sin\kappa - \cos \sin\phi \cos\kappa & \sin\omega \cos\kappa + \cos\omega \sin\phi \sin\kappa & \cos\omega \cos\phi \end{bmatrix}$$

The unknown boresight misalignment angles can be found using least squares adjustment with the condition that the squared sum of the differences between the corrected coordinates of the matched virtual points in the different strips is a minimum.

For two overlapping ALS strips, the boresight angles can be found using the fact that the matched virtual points in the two strips should have the same coordinates. Equation 3.3, expresses the differences between the corrected coordinates (Toth, 2002).

$$\begin{bmatrix} X_g \\ Y_g \\ Z_g \end{bmatrix}_1^{corr} - \begin{bmatrix} X_g \\ Y_g \\ Z_g \end{bmatrix}_2^{corr} = R_{INS1}^{M} R_{L}^{INS} \begin{bmatrix} X_g \\ Y_g \\ Z_g \end{bmatrix}_1^{L} + \begin{bmatrix} X_a \\ Y_a \\ Z_a \end{bmatrix}_1^{M} - R_{INS2}^{M} R_{L}^{INS} \begin{bmatrix} X_g \\ Y_g \\ Z_g \end{bmatrix}_2^{L} - \begin{bmatrix} X_a \\ Y_a \\ Z_a \end{bmatrix}_2^{M} \quad \text{(Eq. 3.3)}$$

Clearly, this equation is non-linear. It is rather difficult and often impractical to seek a least squares solution of non-linear equations (Mikhail, 1976). This implies that some type of linearization must be used to achieve acceptable equations for processing. A Taylor's Series expansion is often used as a means of linearization, where only the zero and first order terms are used and all other higher-order terms are neglected. When applying a series of expansions, a set of approximate values for the unknowns, in this case, for the misalignment angles, must be chosen. Close approximations of the unknown values are important because they allow for rapid and accurate convergences (Mikhail, 1976). In this case, it is acceptable to employ initial estimates of 0 degrees because it is commonly accepted that the misalignment angles are generally less than 3 degrees in value (Toth, 2002).

With this type of least squares adjustment, the solution delivers an updated approximation vector, which is then applied to the current estimation vector. The updated vector of approximations is then used again to formulate the zero and first order terms of the linearized equations. The least squares adjustment is used to obtain another updated approximation vector and so on. This iterative process continues until the last values of the update approximation vector are insignificantly small (Mikhail, 1976). Figure 3.1 is a flowchart of the misalignment estimation adjustment process.

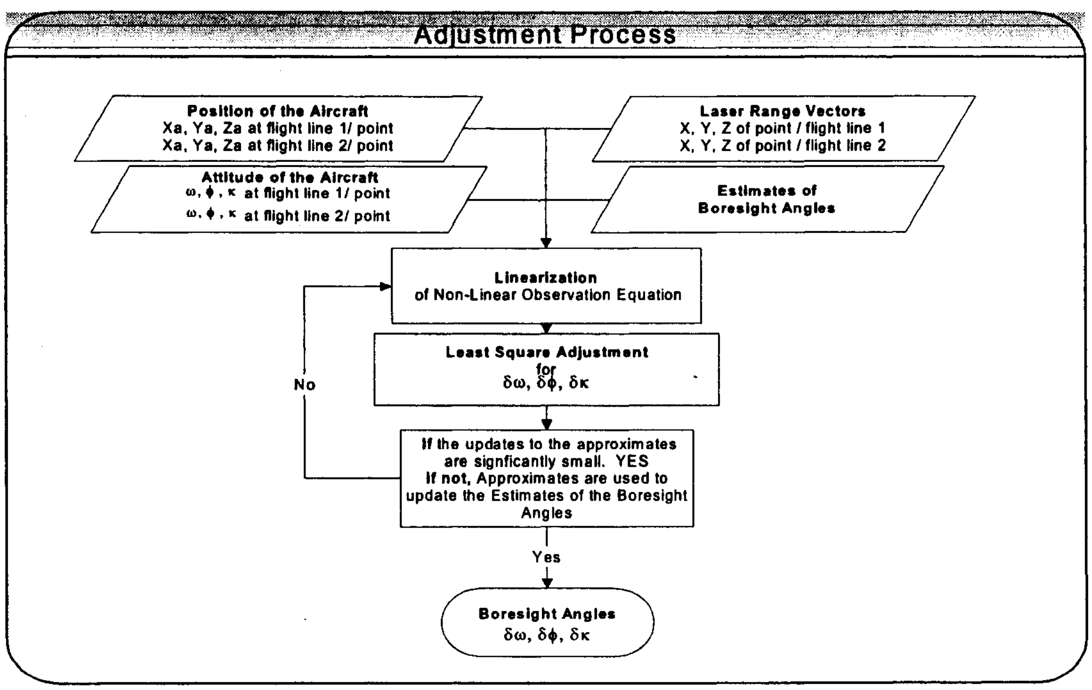

Figure 3.1 Flowchart of Misalignment Estimation Adjustment Process

3.3.1 The Misalignment Estimator Program

The Misalignment Estimator program created in this investigation is a program that calculates estimates of the misalignment angles between the IMU and the laser scanner. The input required for this program is the range vectors of matched points of two overlapping ALS point clouds, the sensor platform orientation data, position and attitude of each point, and estimates of the boresight angles. The range vectors of the matched points are derived as raw data from the ALS. This data has been processed in order to find matching points within the two data sets and remove points that are considered unmatched (i.e. have no corresponding point in the other data set). The position data of the aircraft when each of the points was collected is determined by the GPS output and is interpolated during post-flight processing. Also, the attitude data of the aircraft when each point was collected is determined by the IMU output and is interpolated during post-flight processing. The user can provide initial estimates of the misalignment angles between the ALS and the IMU, but the program has default initial estimates of zero. Estimates of zero can be assumed because these angles are differentially small.

The program was written using Microsoft Visual Basic. The program allows the user to select the input files to be used by the program. The input files have the user-defined estimates of the boresight misalignment angles contained in the first record, then the matched observed point data follow, one record per line. There are no headers for the point data. A sample input file is found in Appendix B. The records of the observed points must be in the same order in both input files, since this is how the program matches the corresponding points as matched points.

The observed point data is formatted as follows: point number, ground Easting, ground Northing, ground elevation, air Easting, air Northing, altitude, omega, phi, kappa, x-range vector, y-range vector, z-range vector, and intensity. The program can also run using roll, pitch, and heading, if needed, but available code changes are required.

The output of the program is the calculated approximates of the boresight angles for each iteration of the program and the final boresight misalignment angles calculated by the program. Also, the residuals and other statistics can be outputted. The source code of the Misalignment Estimator is given in Appendix A.

3.4 Simulations

In order to help develop the Misalignment Estimator program and test the model of the misalignment estimation method, a pseudo data set was created. This pseudo data set was used for all development and testing. This type of data set was used because of the lack of "system noise" and the ability to induce known misalignment errors within the test data for controlled testing.

3.4.1 Pseudo Data Set

The pseudo data set was created by collecting the image coordinates of 50 tie points between two overlapping digital photographs. The two overlapping photographs are shown in Figure 3.2.

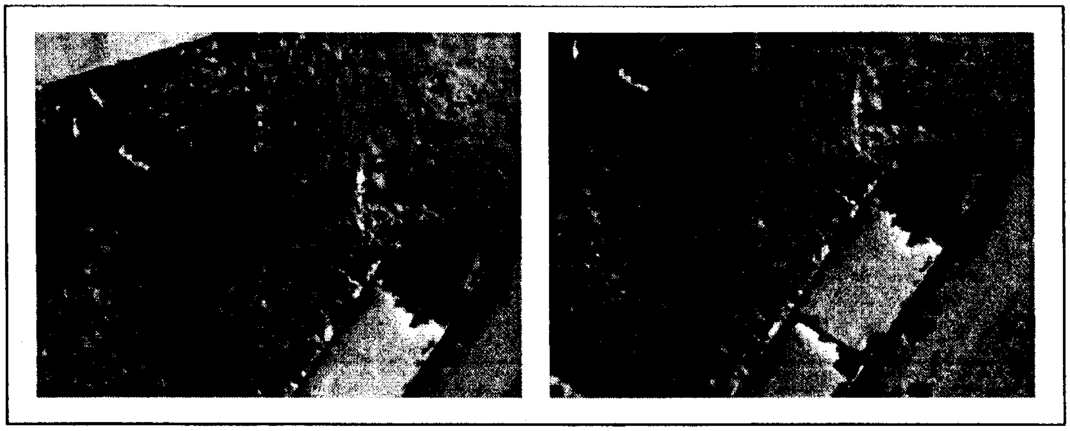

Figure 3.2 The Digital Photographs used to create the Pseudo Data set

The interior orientation parameters, exterior orientation and the GPS exposure station coordinates were known for the photographs so ground coordinates could be calculated for the tie points. Using the tie points' ground coordinates to represent laser scanner hits on the ground, two strips of matching pseudo laser scanner points were created. The layout of the pseudo laser scanner data is shown in Figure 3.3.

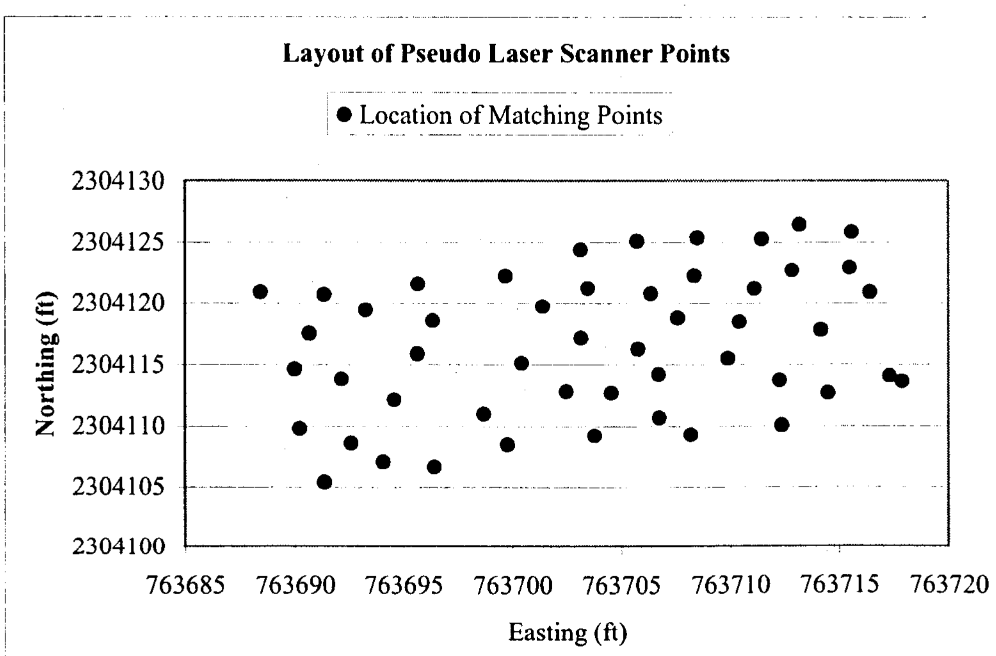

Figure 3.3 Layout of Pseudo Laser Scanner Points

Each point had ground positions (Easting, Northing, and elevation), scanner positions in the air (Easting, Northing, and elevation) and scanner orientation (omega, phi and kappa). Using the ground and air positions and orientation data, range vectors in the x, y and z direction were calculated for each point in both strips. Applying a rotation to the calculated range vectors for each point induced the misalignment error. The induced misalignment errors were +3.00 degrees for omega, phi and kappa.

The misalignment program was able to converge within eight iterations. The results of the program are seen in Table 3.1.

Table 3.1 Outputs and Results for Pseudo Data Set

Misalignment error (Degrees)		Estimated Standard Deviations	Residuals of Ground Coordinates (metres)					
			Iteration 1	Average	RMS	Iteration 8	Average	RMS
Omega	3.0000	0.2517	dx	0.32	0.01	dx	0.00	0.00
Phi	3.0000	0.7031	dy	0.09	0.02	dy	0.00	0.00
Kappa	3.0000	0.5717	dz	-0.38	0.01	dz	0.00	0.00

Clearly, the misalignment program successfully estimated the induced misalignment errors in the orientation angles, omega, phi and kappa. The estimated standard deviations of the misalignment angles show the estimated measure of dispersion. The residuals of the ground coordinates of the pseudo laser hits show that the program did adjust and indeed improve the ground coordinates residuals from an average 0.32 m in the X direction, 0.09 m in the Y direction, and -0.38 m in the Z direction in the first iteration, to no significant residuals in the eighth iteration. Root-mean-square (RMS) is the square root of the arithmetic mean of the squares of the deviation of observed values from their arithmetic mean. These values can be used as a means to check the precision of the mean values. The smaller the root-mean-square value the more precise the mean is stated. The RMS errors for X, Y and Z after the eighth iteration are reported to be insignificant also, which indicates a successful estimation of the misalignment error.

With the success of the estimation of the misalignment errors, using the Misalignment Estimator program in the controlled assessment situation, the model and methodology are considered to be sound and prepared for testing using actual ALS data.

4.0 Experimentation and Results

The previous chapter reviewed the theoretical considerations of performing the misalignment estimation calibration of the ALS system and described testing the method and results of controlled testing. This chapter describes the use of the Misalignment program with the actual ALS data, located at the Oshawa airport data set. The associated data manipulation and interpolation is discussed. Angles of misalignment, between the IMU frame and the laser scanner frame, are produced and examined. Also, the accuracy and results of the misalignment estimation will be discussed.

4.1 Oshawa Airport Data Set

Data for this thesis was collected over the Oshawa Airport, Ontario, Canada. The scanned area includes two runways, adjoining taxiways and aircraft hangers. For this airport the terrain is generally flat, with minor undulations. The ALS data was collected in two strips, the first being flown from North-West to South-East at an altitude of approximately 1980 metres and the second being flown from the North-West to the South- West, at an altitude of approximately 1120 metres. The layout of the strips, the overlapping area and matched scanner points are illustrated in Figure 4.1.

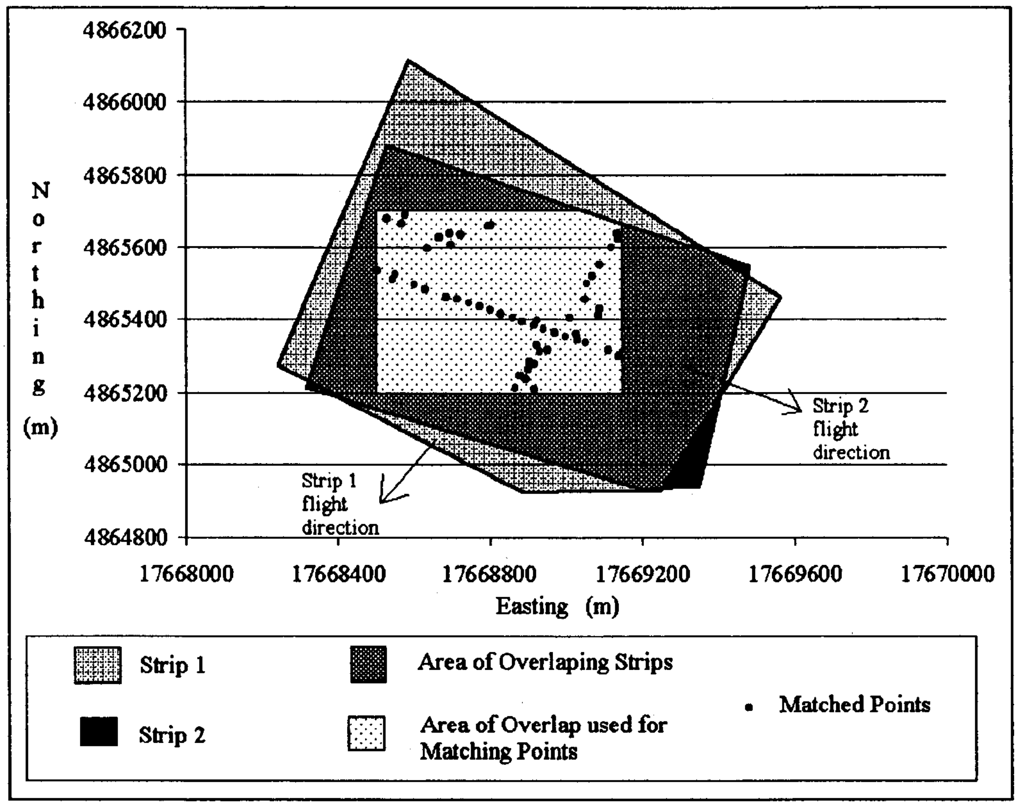

Figure 4.1 Layout of Laser Scanner Strips of Oshawa Airport

Each strip recorded approximately 600,000 laser returns. The scanner that was used is an Optech ALTM 3025 Airborne Laser Terrain Mapper. The scanner is supported by a Novatel Millennium GPS receiver and an Applanix – Optech custom position orientation system. The specifications of the ALTM 3025 are described in Table 4.1.

Table 4.1 ATLM 3025 Specifications (Copyright 2002, Optech Incorporated)

ALTM 3025 Specifications	
Operating altitude	185 - 3,000 m nominal
Range resolution	1cm
Scan angle	Variable from 0 to +/- 20 degrees
Swath width	Variable from 0 to 0.72 x altitude
Scan frequency	0 - 70 Hz, depending on scan angle (e.g., 40 Hz at +/- 20 degrees)
Horizontal accuracy	Better than 1/2,000 x altitude
Elevation accuracy	15 cm at 1,200 m 35 cm at 3,000 m
Position Orientation System	Applanix - Optech custom
GPS receiver	Novatel Millennium

The navigation solution for the ATLM 3025 uses differential GPS data collected from a base station. The lever arm between the GPS and IMU would have been entered as a parameter in the processing software. The offsets between the IMU and the laser scanner have very minor effect on accuracy of the laser scanner data and, therefore, are not considered in the calibration.

The data from the test flight was delivered as three separate files, all in ASCII format. The first data file is the ALS point file that contains GPS time, the last pulse: Easting, Northing, elevation, intensity, the first pulse: Easting, Northing, elevation, intensity, and scanner angle (degrees) values of each observed point. The second data file is the GPS aircraft position file that contains GPS time, Easting, Northing, and elevation. The third data file is the IMU aircraft orientation file that contains the record counter, GPS time, Roll, Pitch and Heading.

4.2 Data Fusion

The data set that was used to test the method purposed, was attainted in three separate files. One was the INS file that contained the attitude and heading data and the GPS time of each measurement record. Another one was the GPS file that contained the position data for the test data and the GPS time of each record. The final one was the ALS file that contained the ALS pulse records and the GPS time of each pulse. Because GPS data had previously been interpolated to the INS sampling rate by the POS software, the two data sets are easily matched by GPS time records. The ALS file, on the other hand, has a much higher sampling rate than the position and orientation data and interpolations of both the GPS positioning data and the INS data need to be completed for each ALS pulse record. It is very important to note that each ALS, INS, and GPS record includes a GPS time-tag. This GPS time-tag is the only reference that allows for the separate data sets to be interpolated into one data set.

4.2.1 Data Interpolation

In order to interpolate the position of the sensor from the GPS positioning data, linear interpolation is used. This method is applied because the GPS positioning data proved to have a linear character over short (1 sec) time intervals, when the double difference of consecutive records was checked. Equation 4.1 is solved using the two GPS position records, which are acquired immediately before the ALS pulse time and immediately afterward.

$$X_{ALS} = (X_2 - X_1) * (t_i / \Delta t) + X_1 \quad \text{(Eq. 4.1a)}$$

$$Y_{ALS} = (Y_2 - Y_1) * (t_i / \Delta t) + Y_1 \quad \text{(Eq. 4.1b)}$$

$$Z_{ALS} = (Z_2 - Z_1) * (t_i / \Delta t) + Z_1 \quad \text{(Eq. 4.1c)}$$

where X_{ALS}, Y_{ALS} and Z_{ALS} are the interpolated easting, northing and elevation position of the scanner when the ALS pulse was recorded.

X_1, Y_1 and Z_1 are the positions measured immediately before the ALS pulse time.

X_2, Y_2 and Z_2 are the positions measured immediately after the ALS pulse time.

t_i is the time of the ALS pulse minus the time of X_1.

Δt is the difference in time between X_1 and X_2.

In order to interpolate the orientation angles of the sensor from the INS orientation angles, a cubic spline interpolation is applied to the data set. Cubic spline interpolating is used because the angle data proved to have a non-linear trend, when the double differences of consecutive records were checked. The approach of a cubic spline interpolator is to divide the data set into intervals and construct a different approximating polynomial for each interval. Figure 4.2 show an example of the POS data interval and an applied cubic spline interpolation.

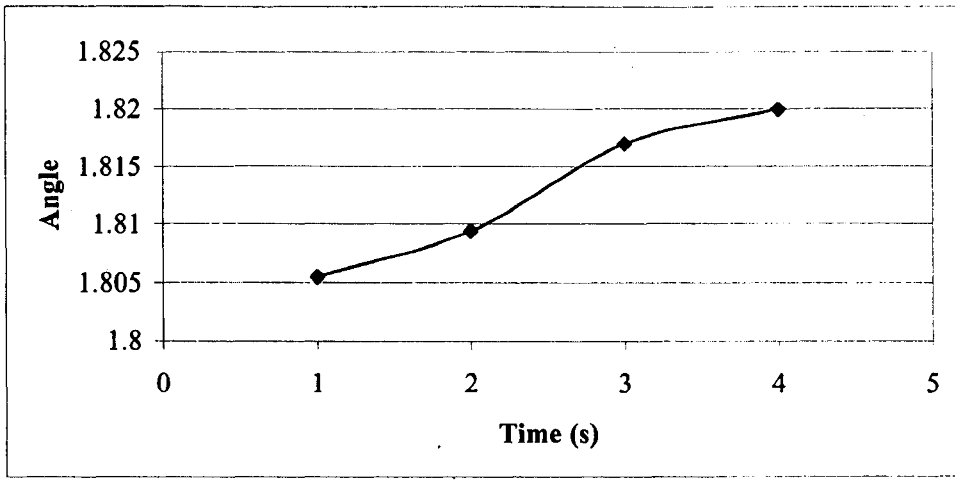

Figure 4.2 Example of POS Data Interval and Cubic Spline

The interval of four records are chosen by using the two INS records whose times are directly before the ALS pulse time and the two records directly after. Generally four constants are used to ensure sufficient flexibility in the cubic spline. Equations 4.2 to 4.5 are used to interpolate the orientation angles at the time of the Lidar pulses.

$$\omega_i = a_o + a_1 t_i + a_2 t_i^2 + a_3 t_i^3 \qquad \text{(Eq. 4.2)}$$

$$\omega_{i+1} = a_o + a_1 t_{i+1} + a_2 t_{i+1}^2 + a_3 t_{i+1}^3 \qquad \text{(Eq. 4.3)}$$

$$\omega_{i+2} = a_o + a_1 t_{i+2} + a_2 t_{i+2}^2 + a_3 t_{i+2}^3 \qquad \text{(Eq. 4.4)}$$

$$\omega_{i+3} = a_o + a_1 t_{i+3} + a_2 t_{i+3}^2 + a_3 t_{i+3}^3 \qquad \text{(Eq. 4.5)}$$

where ω is the orientation angle

a_o a_1 a_2 and a_3 are the variables of the polynomial

t is the time recorded of the angle measurement

i denotes the record number within the chosen interval

This set of equations is solved in terms of the variables a_o a_1 a_2 a_3. Once the variables of the cubic polynomial are known, then the time record of the ALS pulse is substituted into the polynomial and the interpolated orientation angle is calculated. This is completed for all three of the orientation angles for every ALS pulse record used within the data set.

4.2.2 Angle Conversion

Direct georeferencing of scanner data is accomplished by means of GPS assisted INS. Special attention must be focused on the angular data delivered by the INS and on the transformations that must be applied to convert the data into a photogrammetry compatible coordinate system.

Today's inertial referencing systems are fixed with respect to a body coordinate frame that coincides with the principal axes of the aircraft. An INS determines the heading, roll and pitch of an aircraft in the navigational coordinate frame. This coordinate frame and the angles, heading, roll and pitch (ψ ϕ θ) do not comply with coordinate frame and angles used in photogrammetry. Therefore, in order to use the INS data, it is necessary to apply transformations to bring the INS data into the correct coordinate frame and angle format.

4.2.2.1 Coordinate Systems and Angles Used in Inertial Navigation

Inertial navigation is based on the continuous integration of the accelerations measured by the INS accelerometers. These accelerations are measured in a fixed body coordinate system (b-system) as shown in Figure 4.3. The axes of this orthogonal frame are the INS sensor axes.

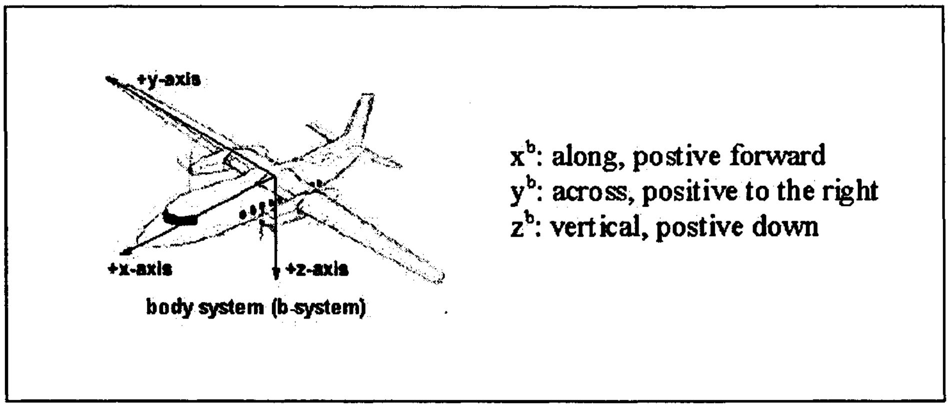

Figure 4.3 The Fixed Body Coordinate System used in Navigation

The INS provides the roll, pitch and heading angles of the INS fixed body frame with respect to the local level navigation coordinate frame (Hutton and Savina, 2000). The navigation coordinate system (n-system) differs from the fixed body coordinate system, where x^n: northward, y^n: eastward and z^n: vertical in the direction of the plumb line (Baumker and Heimes, 2001). This frame is the local level frame that is tangent to the geoid over which the sensor is navigating (Hutton and Savina, 2000). In order for the INS accelerations to be integrated into the navigation coordinate frame, a transformation is performed. This transformation is performed by a rotation matrix, which includes three rotations of the Euler angles (ψ ϕ θ).

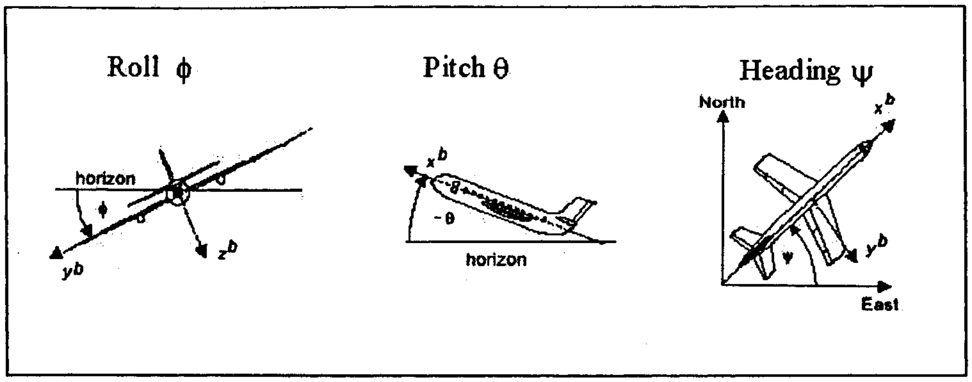

Figure 4.4 The Navigation Coordinate System and Euler Angles (ψ, ϕ, θ)

These angles and the rotation matrix are continuously updated by means of gyro measurements recorded by the INS. The roll, pitch, and heading angles are used to transform a vector from the body coordinate frame into the navigation frame. The transformation matrix itself is calculated by three consecutive rotation matrices as seen in Equations 4.6a and 4.6b, and results in an orthogonal transformation matrix (Baumker and Heimes, 2001).

$$C_b^n = R_z(\psi)\cdot R_y(\theta)\cdot R_x(\phi) = \begin{bmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}\cdot\begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}\cdot\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{bmatrix}$$

(Eq. 4.6a)

$$C_b^n = \begin{bmatrix} \cos\psi\cdot\cos\theta & \cos\psi\cdot\sin\theta\cdot\sin\phi - \sin\psi\cdot\cos\phi & \cos\psi\cdot\sin\theta\cdot\cos\phi + \sin\psi\cdot\sin\phi \\ \sin\psi\cdot\cos\theta & \sin\psi\cdot\sin\theta\cdot\sin\phi + \cos\psi\cdot\cos\phi & \sin\psi\cdot\sin\theta\cdot\cos\phi - \cos\psi\cdot\sin\phi \\ -\sin\theta & \cos\theta\cdot\sin\phi & \cos\theta\cdot\cos\phi \end{bmatrix}$$

(Eq. 4.6b)

The matrix of Equation 4.6 describes the Euler sequence of clockwise rotations required to bring the navigation frame into alignment with the body frame (Hutton and Savina, 2000).

4.2.2.2 Coordinate Systems and Angles Used in Photogrammetry

In comparison, the body coordinate system used in inertial navigation is similar to the image coordinate system (i-system) used in photogrammetry, which is referenced to the scanner sensor (Baumker and Heimes, 2001). Its x-axis points towards the direction of flight, y-axis points towards the left and the z-axis points upwards as seen in Figure 4.5.

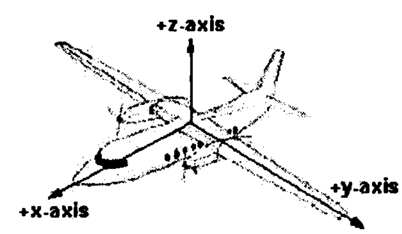

Figure 4.5 Image Coordinate System

The photogrammetric angles; omega, phi, kappa (ω,ϕ,κ) are defined as the sequence of rotations that relate this frame to the object coordinate system. The Earth centered coordinate system (E-system) is fixed to the reference ellipsoid. Its origin is at the centre of the ellipsoid, with it x-axis pointing through the equator and the Greenwich meridian, its y-axis along the equator and the +90 degree meridian, and z-axis through the North Pole (Hutton and Savina, 2000) as seen in Figure 4.6.

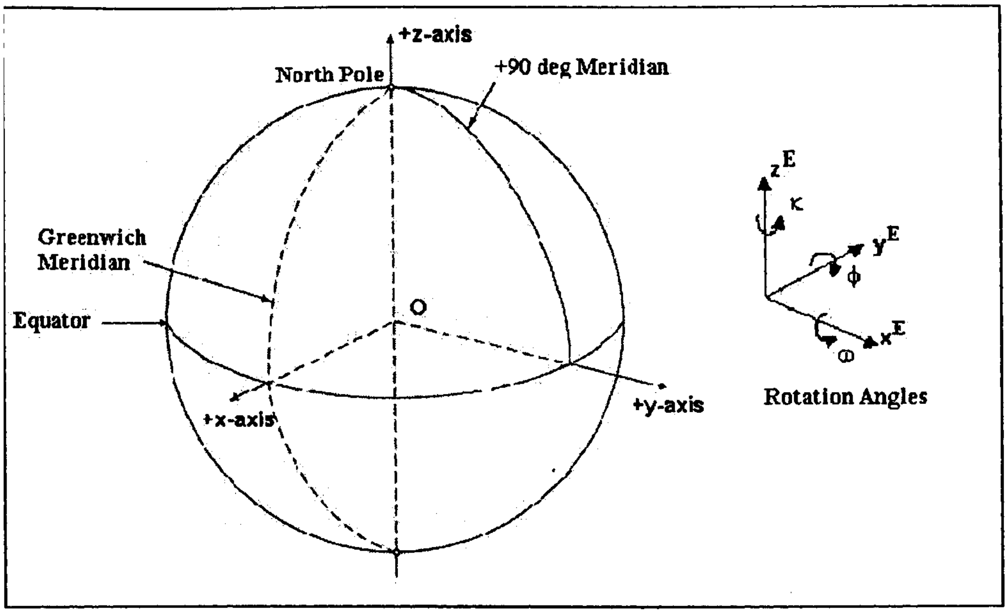

Figure 4.6 Earth Centered Coordinate System and Rotation Angles (ω, φ, κ)

The matrix in Equation 4.7, describes the sequence of clockwise rotations required to bring the image frame into alignment with the Earth centered frame.

$$C_E^i = \begin{bmatrix} \cos\phi \cos\kappa & -\cos\phi \sin\kappa & \sin\phi \\ \cos\omega \sin\kappa + \sin\omega \sin\phi \cos\kappa & \cos\omega \cos\kappa - \sin\omega \sin\phi \sin\kappa & -\sin\omega \cos\phi \\ \sin\omega \sin\kappa - \cos\omega \sin\phi \cos\kappa & \sin\omega \cos\kappa + \cos\omega \sin\phi \sin\kappa & \cos\omega \cos\phi \end{bmatrix} \text{(Eq. 4.7)}$$

If the transformation matrix above is known, then the orientation angles: omega, phi, and kappa can be computed as follows (Baumker and Heimes, 2001):

$$\begin{aligned} \phi &= \arctan\frac{C_{13}}{\sqrt{C_{23}^2 + C_{33}^2}} \\ \omega &= \arctan\frac{-C_{23}}{C_{33}} \\ \kappa &= \arctan\frac{-C_{12}}{C_{11}} \end{aligned} \qquad \text{(Eq. 4.8)}$$

4.2.2.3 Conversion from Attitude and Heading Angles to Photogrammetric Angles

Previously, the focus has been on the individual treatment of the rotation and transformation matrices and the corresponding rotation angles used in navigation and photogrammetry. Now the conversion from attitude and heading angles to photogrammetric angles will be examined.

In order to directly solve for omega, phi, and kappa from roll, pitch, and heading the following equation (4.9) must be applied (Hutton and Savina, 2000):

$$C_i^m(\omega,\phi,\kappa) = C_E^m C_n^E C_b^n(\phi,\theta,\psi)C_i^b \qquad \text{(Eq. 4.9)}$$

where $C_b^n(\phi,\theta,\psi)$ is the direction cosine matrix defining the Euler sequence of clockwise rotations required to bring the navigation frame into alignment with the body frame, as seen in Equation 4.6.

where C_n^E is the direction cosine matrix defining the relative orientation of the local level navigation coordinate frame with respect to the Earth centered coordinate frame and is defined in terms of the latitude (λ) and longitude (l) as (Hutton and Savina, 2000):

$$C_n^E = \begin{bmatrix} -\sin\lambda\cos l & -\sin l & -\cos\lambda\cos l \\ -\sin\lambda\sin l & \cos l & -\cos\lambda\sin l \\ \cos\lambda & 0 & -\sin\lambda \end{bmatrix} \qquad \text{(Eq. 4.10)}$$

where C_i^b is the direction cosine matrix defining the relative orientation of the scanner frame with respect to the INS fixed body frame (Hutton and Savina, 2000). C_i^b is defined in terms of the fixed mounting angles between the INS and the laser scanner (Θ_x, Θ_y, Θ_z).

$$C_i^b = \begin{bmatrix} \cos\Theta_y \cos\Theta_z & \cos\Theta_y \sin\Theta_z & -\sin\Theta_y \\ \sin\Theta_x \sin\Theta_y \cos\Theta_z - \cos\Theta_x \sin\Theta_z & \sin\Theta_x \sin\Theta_y \sin\Theta_z + \cos\Theta_x \cos\Theta_z & \sin\Theta_x \cos\Theta_y \\ \sin\Theta_x \sin\Theta_y \cos\Theta_z + \sin\Theta_x \cos\Theta_z & \cos\Theta_x \sin\Theta_y \sin\Theta_z - \sin\Theta_x \cos\Theta_z & \cos\Theta_x \cos\Theta_y \end{bmatrix} \quad \text{(Eq. 4.11)}$$

The mounting angles are constant because the INS is rigidly mounted to the scanner. These angles are determined once during installation (Hutton and Savina, 2000). The mounting angles bring the scanner frame into alignment with the INS fixed body frame. The succession of rotations is described as a clockwise rotation about the scanner frame z-axis by angle Θ_z, then a clockwise rotation about the once rotated scanner frame y-axis by angle Θ_y, and finally, a clockwise rotation about the twice rotated scanner frame x-axis by angle Θ_x (Hutton and Savina, 2000).

The final transformation matrix is C_E^m, which is dependent upon the local mapping frame defined by the project (Hutton and Savina, 2000).

4.2.3 Data Matching

In order to use this method of misalignment angles estimation, points from one of the overlapping strips of laser scanner data must be matched with points corresponding to the same area in the second strip of laser scanner data.

In other methods of misalignment estimation, the laser scanner is re-sampled over a regular grid or a TIN is created to produce a surface, which eases the point matching process. For this project, the data was not resampled in order to maintain the integrity of the data. Instead, the easting, northing and intensity data of the laser points were used to create an intensity image. A program called Point Cloud Data 3-D Adjuster and Analyzer (Fidera, 2003) was used to display the intensity images of the overlapping scanned areas. This program also had the ability to deliver the Easting and Northing position of selected data points.

The overlap area of the laser scanner data was divided into 16 blocks, each measuring 125 x 162.5 metres on the ground. Matching points were collected at the corners of runways and taxiways intersections and at the ends of runways painted markers. These locations were chosen because the vast difference in the reflectivity of grassland, reflective paint and asphalt made identification very easy. Figure 4.7 shows matching intensity images used to identify matching observed laser scanner points. Figure 4.8 gives an aerial photograph of the Oshawa airport, specifically the overlap area. These two figures clearly shows the detail available in the laser scanner intensity images.

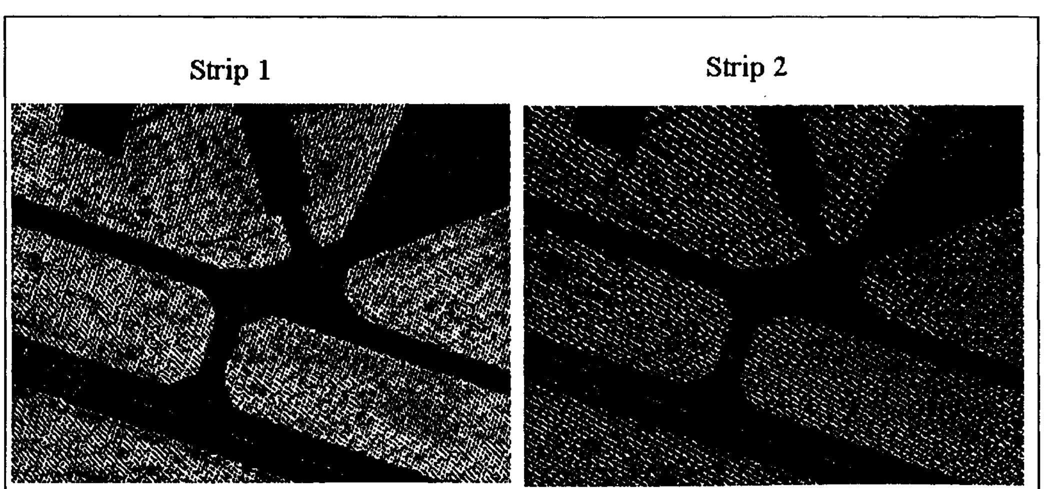

Figure 4.7 Matching Intensity Images

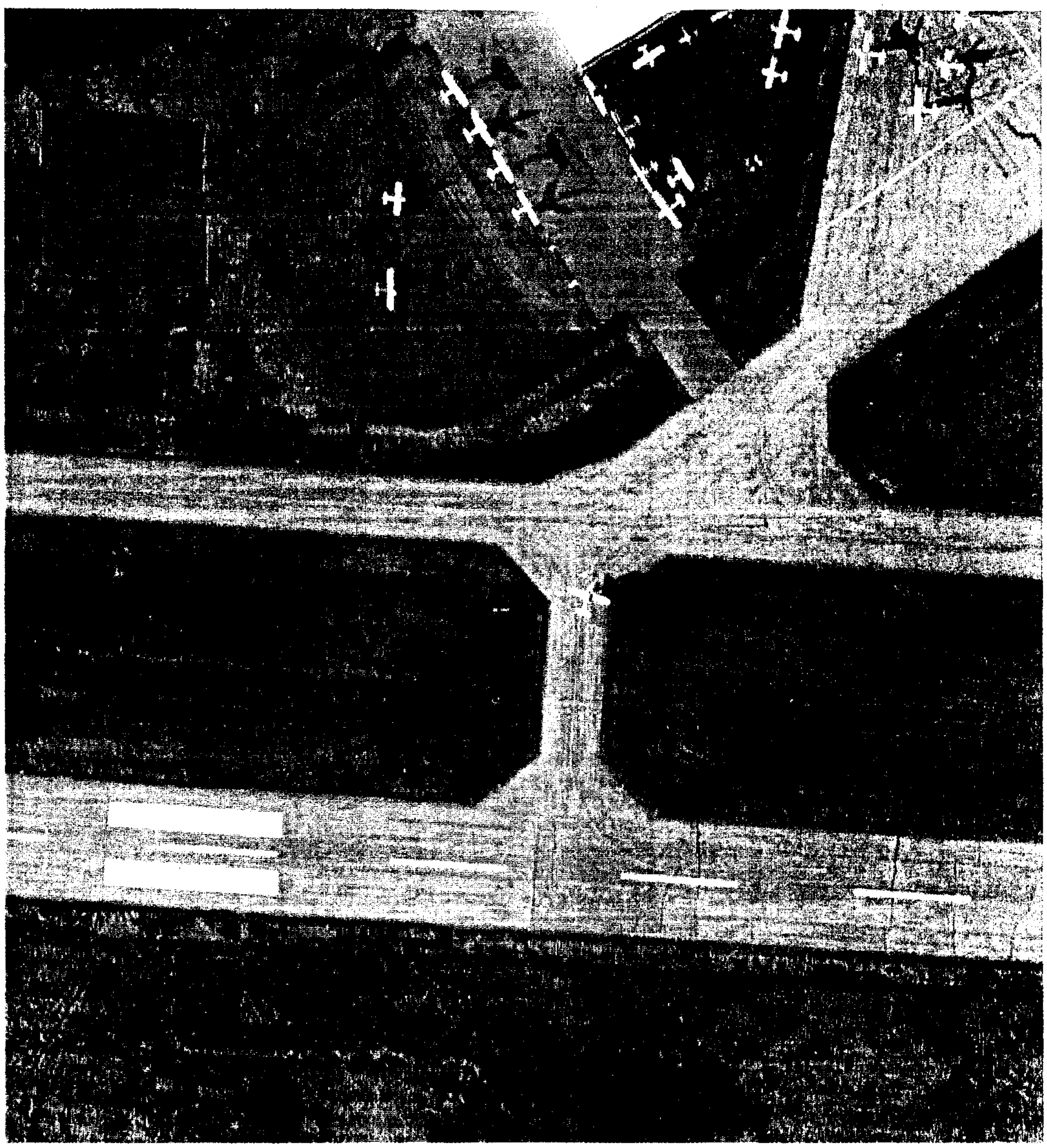

Figure 4.8 Aerial Photography of the Oshawa Airport

In all, 56 matching points were identified. The points are spread over the area of overlap and chosen for the distinct intensity changes clearly seen in both images. Figure 4.9 depicts the horizontal layout of the 56 points.

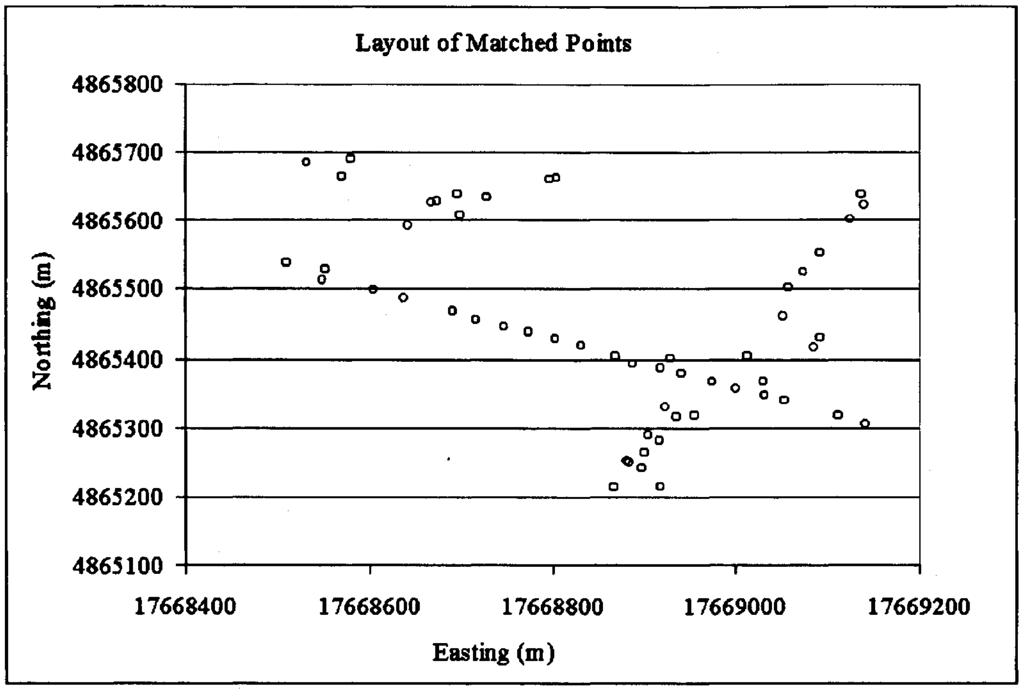

Figure 4.9 Layout of Matched Points

Once the ALS points were matched and the INS and GPS data were interpolated to fit the matched points, the range vectors needed to be calculated. Each point had ground positions (Easting, Northing, and elevation), scanner positions in the air (Easting, Northing, and elevation) and scanner orientation (omega, phi and kappa). Using the ground and air positions and orientation data, range vectors in the x, y and z direction were calculated for each point in both strips.

A simple flow chart of the process matching the ALS points and of fusing the data sets into input files for the program Misalignment Estimator is shown in Figure 4.10.

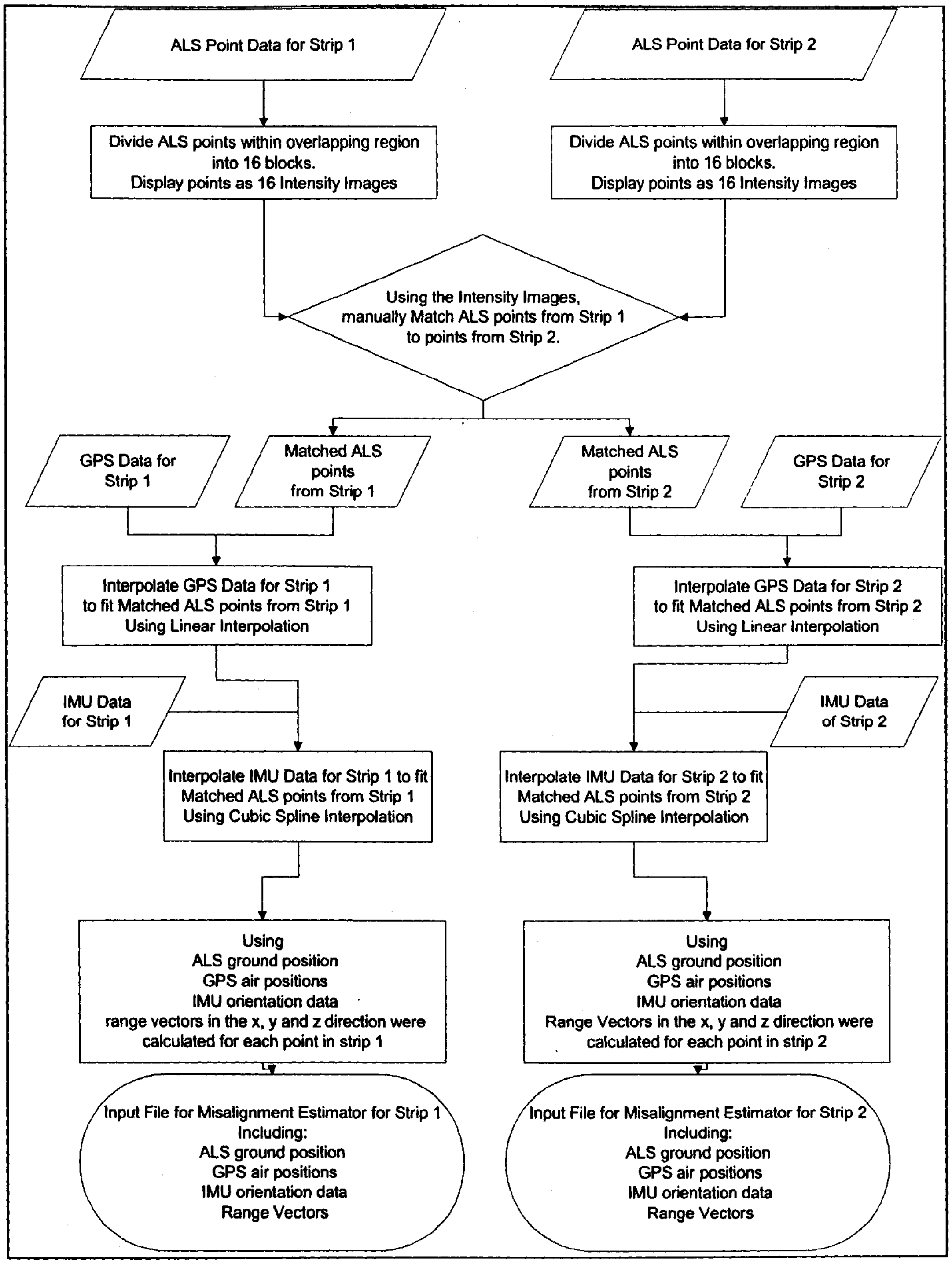

Figure 4.10 Flowchart for Matching the ALS Points and Fusing the Data Sets Process

4.3 Experimentation

In order to explore the dynamics of the overlapping strips of ALS data from the Oshawa airport data set and the collected matched points, the program, Misalignment Estimator, was run with several different groupings of matched points. Input files are created for each grouping of matched points and a priori observation standard deviations of 5.78 cm are applied to the least squares adjustment. A standard deviation of 5.78 cm was chosen because it has been shown that the precision of manually measuring discrete points in a raster image is 0.289 pixels or laser footprint units in this case (Chapman, 2003). 20 cm is the size of the instantaneous laser footprint of laser scan strip 2, multiplied by 0.289 equals 5.78 cm. The adjustment quickly converged within 6 iterations for all of the groupings of matched points. The four most significant points groupings are be reviewed. A brief description of the four significant groupings, summary of the outputted estimated misalignment angles, their accuracy, and the residuals and then the final results of the testing follows.

Point Grouping 1

Point grouping 1 includes all 56 matched points and its layout was shown in Figure 4.9. The points are spread across the overlap area to provide a thorough representation of points within the overlap. Table 4.2 shows the resulting misalignment angles and statistical information for running Misalignment Estimator with point grouping 1.

Table 4.2 Outputs and Results for Point Grouping 1

Misalignment error (Degrees)		Estimated Standard Deviation	Residuals of Ground Coordinates (metres)					
			Iteration 1	Average	RMS	Iteration 6	Average	RMS
Roll	-0.0178	0.0136	dx	2.44	2.90	dx	0.03	1.56
Pitch	-0.0829	0.0136	dy	2.61	3.04	dy	-0.00	1.52
Heading	0.0320	0.0435	dz	0.00	0.18	dz	0.10	0.34

The estimated standard deviations of the misalignment angles show the estimated measure of dispersion and are of an acceptable size. The residuals of the laser hits' ground coordinates show that the program did adjust and, indeed, improved the ground coordinates residuals from an average 2.44 m in the X direction, 2.61 m in the Y direction in the first iteration to an average 0.03 m in the X direction, 0.00 m in the Y direction in the sixth iteration. The ground coordinates residuals in the Z direction were adjusted from 0.00 m to 0.10 m. Even though the resulting residual in the Z direction increases with the adjustment, this is an excepted result of the adjustment because of the larger correction and reduction of the X and Y residuals and is due to the minimizing effect of the least square adjustment. Furthermore, due to the geometry of matched laser points, changes in the X and Y positions of points have strong influences on the Z components. Figure 4.11 depicts the adjustment effect on the Z position of matched points. The RMS errors for X and Y direction after the sixth iteration are smaller than the RMS errors after the first iteration, which shows that the adjustment improved the residuals.

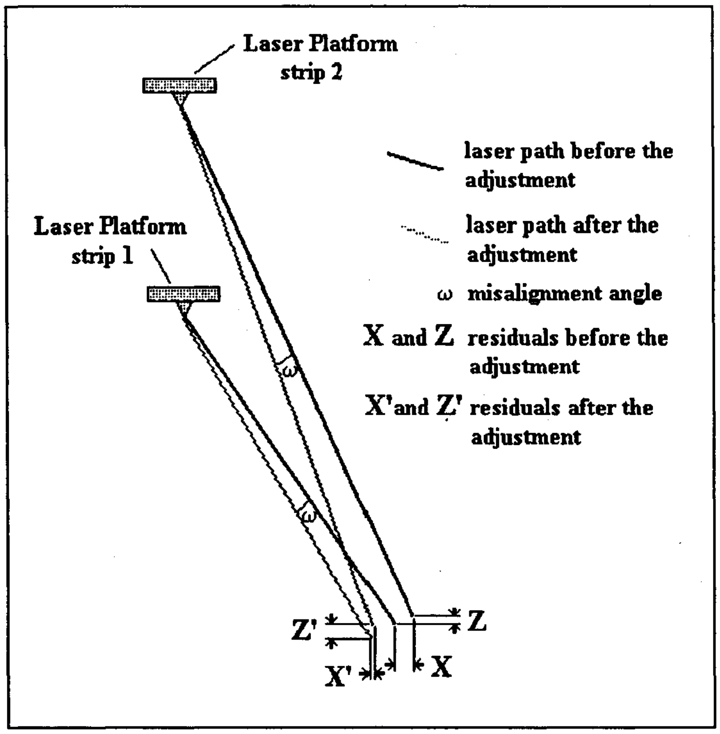

Figure 4.11 Adjustment Affect on Z Residuals

Point Grouping 2

Point grouping 2 includes 50 of the original 56 points. The six points that were removed were found have gross outlying residuals due to the manual measurement/identification errors . This grouping is similar to point grouping 1 and the points are spread across the overlap area to provide a thorough representation of points within the overlap. Figure 4.12 shows the layout of the points of grouping 2.

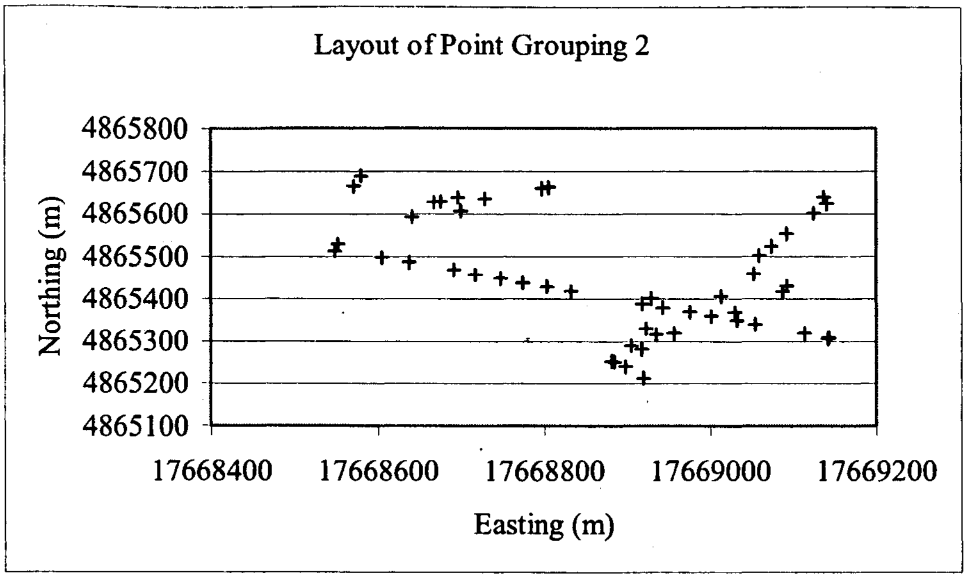

Figure 4.12 Layout of Point Grouping 2

Table 4.3 shows the resulting misalignment angles and statistical information of running Misalignment Estimator with grouping 2.

Table 4.3 Outputs and Results for Point Grouping 2

Misalignment error (Degrees)		Estimated Standard Deviation	Residuals of Ground Coordinates (metres)					
			Iteration 1	Average	RMS	Iteration 6	Average	RMS
Roll	-0.0075	0.0140	dx	2.29	1.15	dx	0.03	1.16
Pitch	-0.0870	0.0140	dy	2.72	1.12	dy	-0.00	1.08
Heading	-0.0531	0.0451	dz	-0.01	0.16	dz	0.08	0.28

The estimated standard deviations of the misalignment angles are of acceptable size. The residuals of the laser hits' ground coordinates show that the program did adjust and, indeed, improved the ground coordinates residuals from an average 2.29 m in the X direction, and 2.72 m in the Y direction in the first iteration to an average 0.03 m in the X direction and -0.00 m in the Y direction in the sixth iteration. The ground coordinates residuals in the Z direction were adjusted from 0.01 m to 0.08 m. As stated for point grouping 1, this is an excepted result of the adjustment. The RMS errors for X and Y direction after the sixth iteration are similar or smaller than the RMS errors after the first iteration, which shows that the adjustment improved the residuals and that the ground point locations of matching points are closer than in the first point grouping. Overall, the most evident change in the results of this grouping is found in the estimated roll misalignment error. This could be a result of removing points that heavily affect the roll solution.

Point Grouping 3

Point grouping 3 includes 35 of the original 56 points. The 21 points that were removed were found to have outlying residuals. This grouping is similar to point grouping 1 in that it has points that are spreading across the overlap area to provide a thorough representation of points within the overlap region. Figure 4.13 shows the layout of the points of grouping 3.

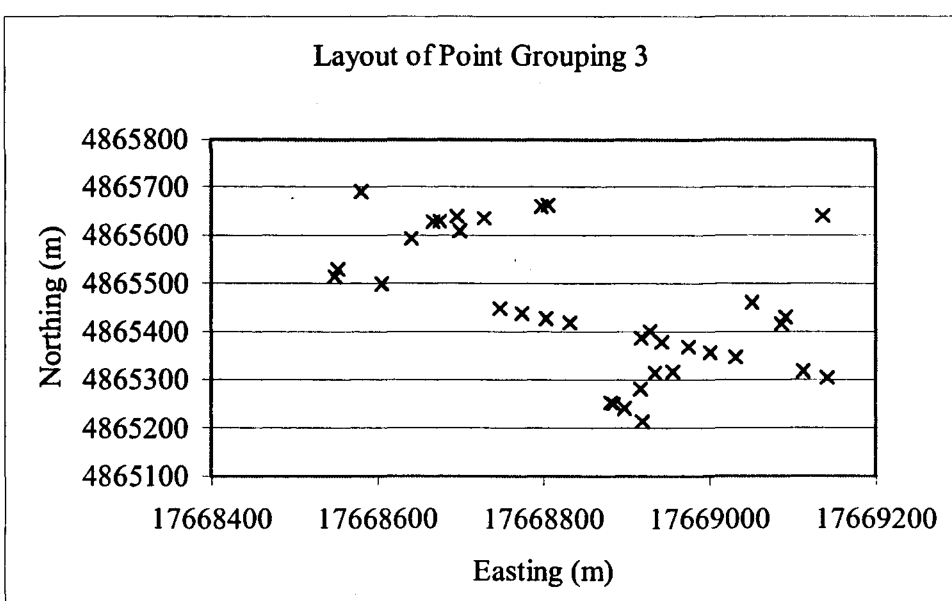

Figure 4.13 Layout of Point Grouping 3

Table 4.4 shows the resulting misalignment angles and statistical information of running Misalignment Estimator with grouping 3.

Table 4.4 Outputs and Results for Point Grouping 3

Misalignment error (Degrees)		Estimated Standard Deviation	Residuals of Ground Coordinates (metres)					
			Iteration 1	Average	RMS	Iteration 6	Average	RMS
Roll	-0.0068	0.0154	dx	2.26	1.010	dx	0.03	1.03
Pitch	-0.0883	0.0154	dy	2.79	0.908	dy	-0.01	0.86
Heading	-0.0404	0.0503	dz	0.00	0.171	dz	0.12	0.27

The estimated standard deviations of the misalignment angles are of acceptable size. The residuals of the laser hits' ground coordinates show that the program did adjust and, indeed, improved the ground coordinates residuals from an average 2.26 m in the X direction, and 2.79 m in the Y direction in the first iteration to an average 0.03 m in the X direction, and -0.01 m in the Y direction in the sixth iteration. The ground coordinates residuals in the Z direction were adjusted from 0.00m to 0.12 m. As stated for point grouping 1, this is an excepted result of the adjustment. The RMS errors for X and Y direction after the sixth iteration are similar or smaller than the RMS errors after the first iteration, which shows that the adjustment improved the residuals and that the ground point locations of matching points are closer than in the first point grouping. Overall, the most evident change in the results of this grouping is again found in the estimated roll misalignment error. This could be a result of removing points that heavily affect the roll solution.

Point Grouping 4

Point Grouping 4 is made of 20 matched points that are found in the centre of the overlapping area. This grouping is chosen because of the high density of matched points within a relatively small area, 100 x 500 metres. The points were not filtered for outliers. Figure 4.14 show the layout of point grouping 4.

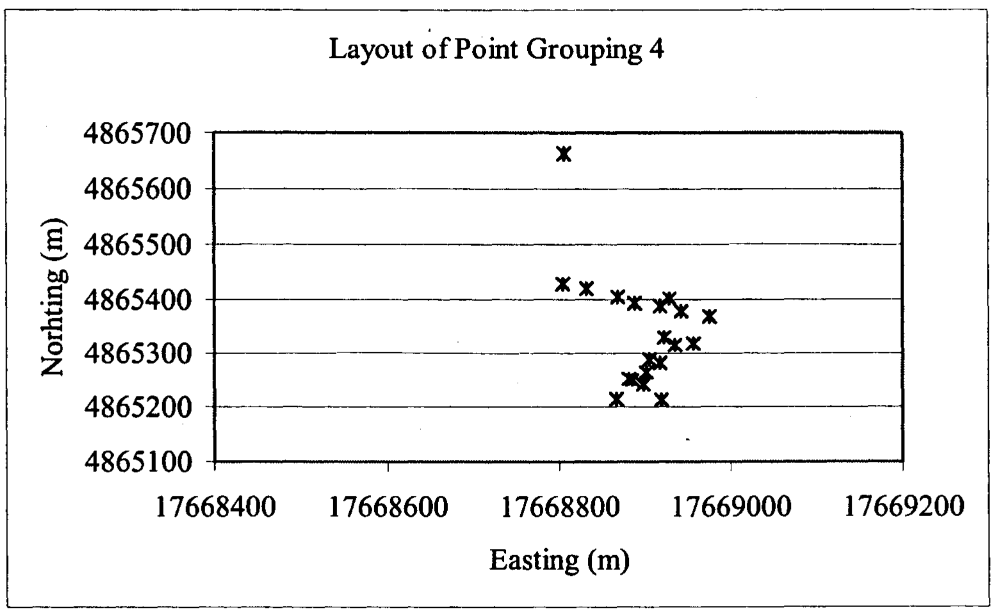

Figure 4.14 Layout of Point Grouping 4

Table 4.5 shows the resulting misalignment angles and statistical information of running Misalignment Estimator with grouping 4.

Table 4.5 Outputs and Results for Point Grouping 4

Misalignment error (Degrees)		Estimated Standard Deviation	Residuals of Ground Coordinates (metres)					
			Iteration 1	Average	RMS	Iteration 6	Average	RMS
Roll	-0.0202	0.0177	dx	8.26	1.010	dx	0.00	1.72
Pitch	-0.0755	0.0177	dy	10.03	0.941	dy	-0.00	1.45
Heading	-0.0622	0.0859	dz	0.49	0.467	dz	0.04	0.09

The estimated standard deviations of the misalignment angles are larger than for the other groupings but still of acceptable size. The residuals of the laser hits' ground coordinates show that the program did adjust and, indeed, improved the ground coordinates residuals from an average 8.26 m in the X direction, 10.03 m in the Y direction, and 0.49m in the Z direction in the first iteration to an average 0.00 m in the X direction, -0.00 m in the Y direction and 0.04 m in the Z direction in the sixth iteration. These are clearly large improvements but it probably says more abut the errors in the original ground coordinates of the laser hits. The RMS errors for X and Y direction after the sixth iteration are larger than the RMS errors after the first iteration. This shows that the adjustment did not actually improve the residuals. While the estimated misalignment angles are very similar to the angles of the first grouping, the accuracy of the residuals stand out as very poor and do not support the solution of the adjustment. This could be the result of having points only from the centre of the overlap.

4.3.1 Estimation and Accuracy of Misalignment Angles

In order to determine which grouping is the best representation of the Oshawa airport data set, the outputs and the statistical information must be examined. To begin with, the laser hit ground coordinate residuals for the four groupings are examined. Compared to the other groupings, Point Grouping 1 has the most consistent improvement in the RMS errors of the ground coordinate residuals. This means that the precision of the ground coordinate residuals was greatly increased by the adjustment. The improvement of the residuals of Point Grouping 1 can be seen in Figures 4.15 to 4.18. By comparing Figures 4.15 and 4.16, the improvement of the X and Y residuals is very evident. By comparing

Figures 4.17 and 4.18 it is evident that even though the average of the Z residuals was increased, the overall effect on the Z residuals is minimal.

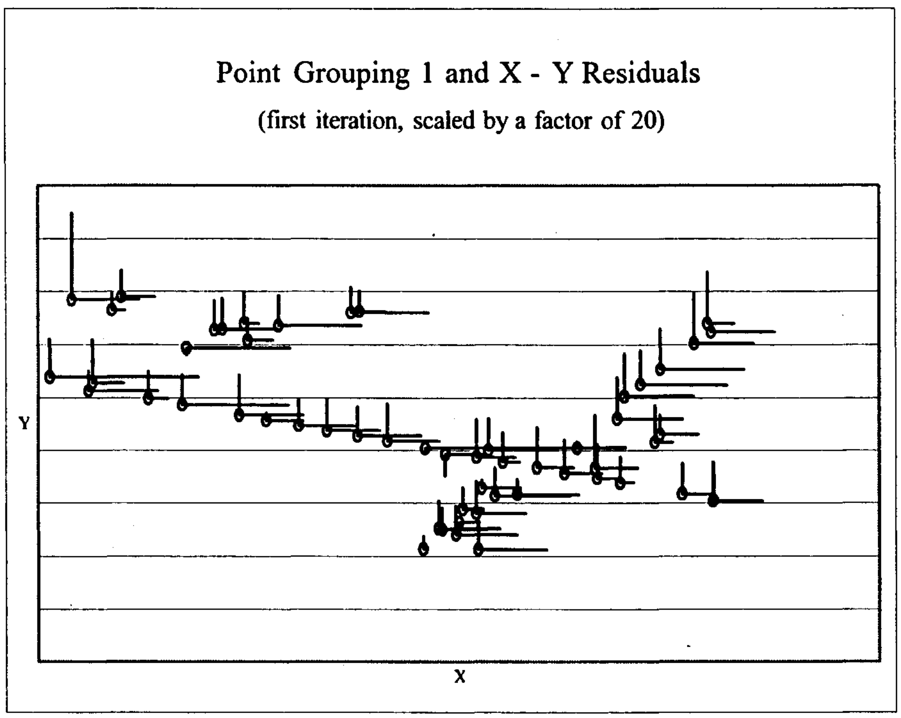

Figure 4.15 Point Grouping 1 and X - Y Residuals of the First Iteration

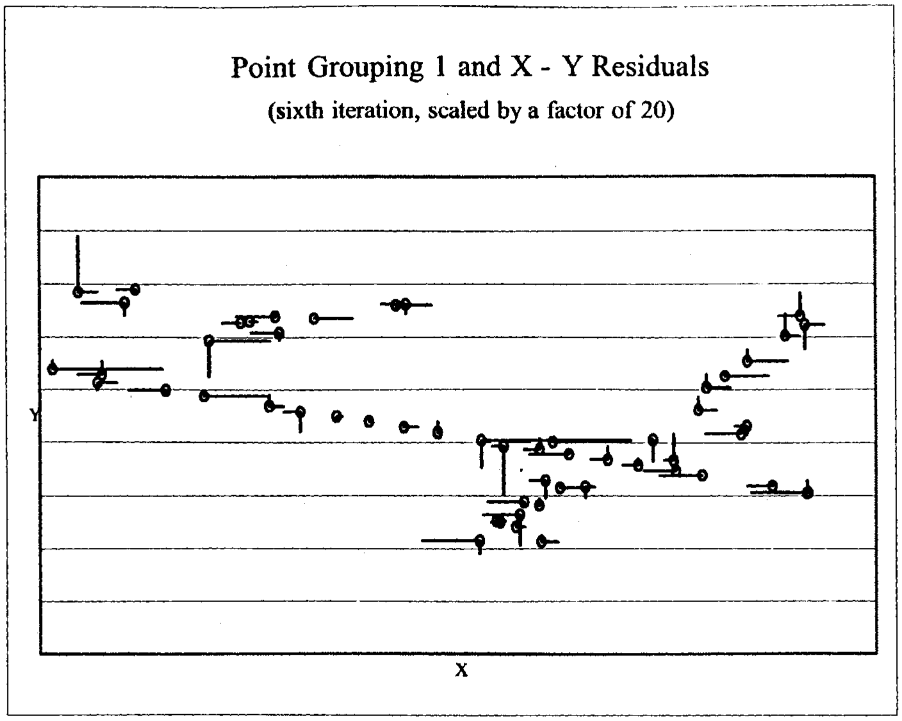

Figure 4.16 Point Grouping 1 and X - Y Residuals of the Sixth Iteration

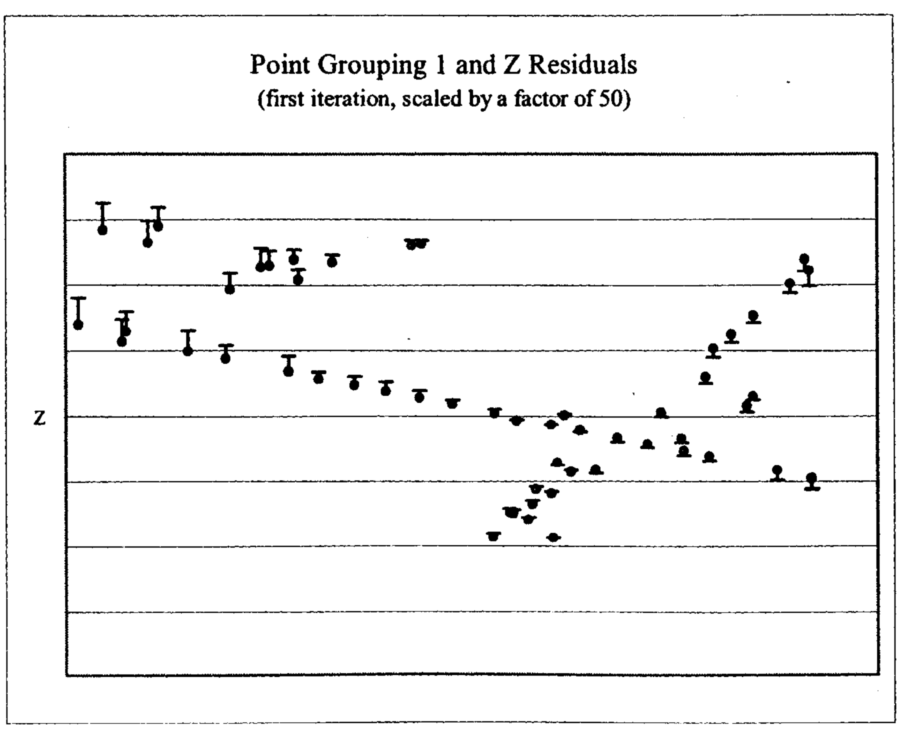

Figure 4.17 Point Grouping 1 and Z Residuals of the First Iteration

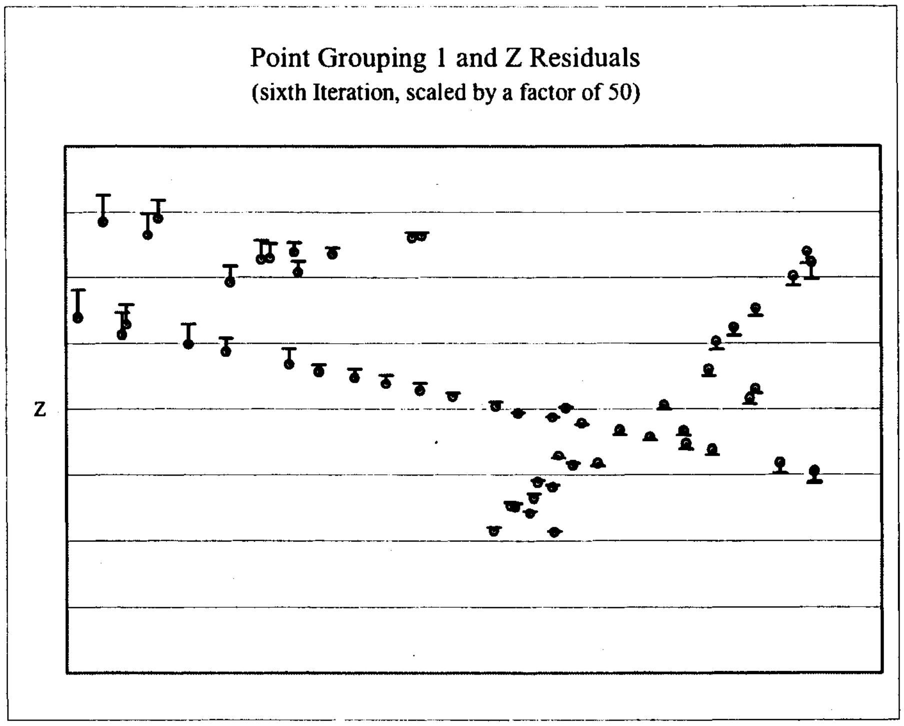

Figure 4.18 Point Grouping 1 and Z Residuals of the Sixth Iteration

Of the different point groupings, Point Grouping 1 has the smallest estimated standard deviations for the estimated misalignment angles. Because Point Grouping 1's estimated standard deviations are the smallest, it shows that the dispersion of the data for the misalignment angles is the least of the point groupings.

With the data set provided by Optech, the in-lab calibrated misalignment angles were provided for roll and pitch. The magnitude to theses angles are roll: 0.0280 degrees, and pitch: 0.0810 degrees. The magnitude of the estimated angles for Point Grouping 1 in roll and pitch are 0.0178 and 0.0829 degrees respectively. These are closer to the given calibrated angles than any of the other groupings. The in-lab calibrated misalignment angles were tested to find the estimated standard deviations for roll and pitch and to examine the residuals of the ground coordinates as a result of these angles. The results of this testing is shown in Table 4.6

Table 4.6 Estimated Standard Deviations and Residuals for In-lab Calibration Angles

Estimated Standard Deviation			Residuals (m)	
			Average	RMS
Roll	0.0136	dx	0.03	1.51
Pitch	0.0136	dy	-0.00	1.52

Comparing the average residuals of the ground coordinates of the matched laser points and the RMS of the residuals of both the in-lab and post-mission estimated misalignment errors for roll and pitch, shows that both sets of angles have the same overall effect on the point grouping. The estimated standard deviation of the in-lab calibrated misalignment errors was found to be very similar to those of the post-mission estimated misalignment errors. This comparison reveals that the post-mission estimated misalignment error angles are just as reliable as the in-lab calibration angles.

Overall, Point Grouping 1 proves to provide the best representation of the Oshawa airport data set. This point grouping performed well because of the large area of coverage and large number of points. The estimated misalignment errors of the Oshawa airport data set are shown in Table 4. 7

Table 4.7 Estimated Misalignment Error

	Degrees	Minutes of Arc
Roll	-0.0178	-1.07
Pitch	-0.0829	-4.98
Heading	0.0320	1.92

Summary, Conclusions and Recommendations

4.4 Summary

An Airborne Laser Scanning system operates by locating laser hits separately from all others. Locating the laser hits involves knowing precisely for each laser shot, the aircraft position (GPS), the attitude of the aircraft (IMU), the scanner angle when the laser pulse left the sensor, and the slant range to the ground for that pulse. One of the most crucial errors in ALS systems is the angular misalignment between the scanner and the IMU, which is called the misalignment error. This error must be dealt with before an ALS system can produce accurate data.

The purpose of this research has been to make available and test a method of estimating the small misalignment angles between the laser scanner and the IMU. This method is semi-automated, requires no ground control and does not re-sample the ALS data in order to match the overlapping strips of data.

A computer program called Misalignment Estimator was developed to estimate the misalignment angles using a least squares adjustment. The method was tested using a data set of the Oshawa airport provided by Optech. The adjustment's outputs provide a means to evaluate the grouping of matched ALS points and the precision of the estimated misalignment angles.

4.5 Conclusions

Based on the investigation completed and the results obtained from the Misalignment Estimator program, the following conclusions can be made:

- The developed method of estimating the misalignment angles is based on the differences observed between two overlapping ALS data strips. The results of applying this method, both in the simulation and real data sets, are convincing.
- The proposed method for integrating the GPS data set and ALS data set and the purposed method for integrating the IMU data set and ALS data set, in order to be used as inputs for the Misalignment Estimator program, have been confirmed to be excellent methods of data integration.
- The program, Misalignment Estimator developed to provide a means of estimating the misalignment error of an ALS system after the system has been installed in an aircraft, does succeed and provides acceptable estimates of misalignment errors.
- Of the point groupings used as input for the Misalignment Estimator program, Point Grouping 1 provided the best representation of the Oshawa airport ALS data set.
- For the Oshawa airport data set, the estimated roll misalignment error was found to be -0.0178 degrees, the estimated pitch misalignment error was found to be -0.0829 degrees, and the estimated heading misalignment error was found to 0.0320 degrees. Furthermore, the roll and pitch error estimates were found to be as reliable as the in-lab misalignment calibration error angles provided by Optech.

4.6 Recommendations

The following recommendations are proposed for future studies:

- The residuals of the measured laser hits' ground coordinates suggest that matching of the laser points of the overlapping strips could be improved. Furthermore, the proposed method of estimating the misalignment angles of ALS systems might be improved upon if a better point matching procedure was developed.
- Of the point groupings, Point Grouping 1, the group with most points over the largest area, was found to represent the ALS data the best. The estimation of the misalignment angles might be improved if a larger area of overlap and more matching points are used.
- The scope of this thesis was to only test the Misalignment Estimator program using a single simulation and a single actual ALS data set. Further testing with other data sets would confirm the ability of the program and the validity of the methodology.
- The program, Misalignment Estimator is designed based on the differences observed between two overlapping ALS data strips. The program may be able to provide better estimations of the misalignment angles if the program was able to utilize three or more overlapping strips of ALS data.
- The program, Misalignment Estimator, is designed only to model the misalignment angle errors, the program could be expanded to include modeling of other ALS system errors such as scan angle errors or encoder latency errors in order to provide a more robust calibration of the ALS system.

References

Anderson, J.M, and Mikhail, E.M., 1998, *Surveying: Theory and Practice*. WCB/McGraw Hill.

Axelsson, P. 2000. *DEM Generation from Laser Scanner Data using Adaptive TIN Models*. International Archives of Photogrammetry and Remote Sensing. Vol. XXXIII, Part B4, Amsterdam. pp. 110 - 117.

Baltsavias, E. P., 1999. *Airborne Laser Scanning:Basic Relations and Formulas*. ISPRS Journal of Photogrammetry & Remote Sensing. Vol. 54, pp. 199 - 214.

Baumker, M. and Heimes, F.J., 2001. *New Calibration and Computing Method for Direct Georeferencing of Image and Scanner Data Using the Position and Angular Data of a Hybrid Inertial Navigation System*. FH Bochum University of Applied Science.

Burrough, P., McDonnell, R., 1998. Principles of Geographical Information Systems. Oxford University Press, New York.

Chapman, M.A., 2003 *Softcopy versus Analytical Photogrammetry: A Coming of Age*, manuscript under preparation.

Cosandier, D. 1999. *Generating a Digital Elevation Model and Orthomosaic from Pushbroom Imagery*. Ph.D. Thesis, University of Calgary, Department of Geomatics Engineering. Calgary, Alberta.

El-Rabbany, A. 2000. *CVL650 Physical and Satellite Geodesy Course Notes*. Ryerson Polytechnic University, Department of Civil Engineering. Toronto, Ontario.

El-Sheimy, N. 1996. *The Development of VISTAT - A Mobile Survey System For GIS Applications*. Ph.D. Thesis, UCGE Reports. No. 20101. Calgary, Alberta.

Fidera, A., 2003. *Point Cloud Data 3-D Adjuster and Analyzer*. Graduate Project. Ryerson University, Toronto, Ontario.

Hutton, J. and Savina, T. 2000. *Computation of Phi Omega Kappa from Roll Pitch and Heading*. Technical Note, Applanix Corporation, Markham, Ontario.

Hutton, J., Savina, T., and Lithopoulos, E. circa 1997. *Photogrammetric Applications of Applanix's Position and Orientation System (POS)*. Applanix Corporation, Markham, Ontario.

Irish, J. and Lillycrop, W.J. 1999. *Scanning Laser Mapping of the Coastal Zone: The SHOALS system*. ISPRS Journal of Photogrammetry & Remote Sensing. Vol. 54, pp. 123 - 129.

Lefsky, M., Cohen, W., Acker, S., Parker, G., Spies, T., and Harding, D. 1999. *Lidar Remote Sensing of the Canopy Structure and Biophysical Properties of Douglas-Fir Western Hemlock Forests*. Remote Sensing Environment. Vol. 70. pp. 339 – 361.

Lima, K., Treitza, P., Wulderb, M., St-Ongec, B., and Floodd, M. 2003. *Lidar Remote Sensing of Forest Structure*. Progress in Physical Geography. Vol. 27, 1. pp. 88 – 106.

Lohmann, P. 2000. *Approaches to the Filtering of Laser Scanner Data*. International Archives of Photogrammetry and Remote Sensing. Vol. XXXIII, Part B, Amsterdam. pp. 540 - 547.

Mikhail, E.M. 1976. *Observations and Least Squares*. Dun-Donnelley. pp. 101-120.

Morin, K. 2003. *Calibration of Airborne Laser Scanners*. M. Sc. Thesis, UCGE Report No. 20179.

Morin, K., El-Sheimy, N. circa 2002. *A Comparison of Airborne Laser Scanning Data Adjustment Methods*. Department of Geomatics Engineering, The University of Calgary Canada.

Smith, R. 2002. *Calibration Procedures for the Optech ALTM.* Presented to the Digital Camera and LIDAR Product Characterization Workshop. Stennis Space Flight Center

Sun, H. 1994. *GPS/INS Integration for Airborne Applications*. Ph.D. Thesis, UCGE Reports. No. 20069. Calgary, Alberta.

Toth, C.K. 2002. *Calibrating Airborne Lidar Systems*. Proceeding of ISPRS Commission II Symposium, Integrated Systems for Spatial Data Production, Custodian and Decision Support. Vol. 35, part 2, pp. 475-480.

Wehr, A., Lohr, U. 1999. *Airborne Laser Scanning – An Introduction and Overview*. ISPRS Journal of Photogrammetry & Remote Sensing. Vol. 54, pp. 68 - 82.

Appendix A

Source code for Misalignment Estimator program.

Form

```
Sub command1_click()

Dim filenumin1 As Integer
Dim filenumin2 As Integer
Dim filenumout As Integer
Dim filepath As String, inputfile As String
Dim outputfile As String

Dim counter As Integer
Dim counter1 As Integer
Dim counter2 As Integer
Dim ii As Integer
Dim l As Integer
Dim iter As Integer

Dim pointnum As Integer

Dim intensity As Integer

Dim ygrd As Double
Dim xgrd As Double
Dim zgrd As Double
Dim yair As Double
Dim xair As Double
Dim zair As Double

Dim y1grd As Double
Dim x1grd As Double
Dim z1grd As Double
Dim y1air As Double
Dim x1air As Double
Dim z1air As Double

Dim y2grd As Double
Dim x2grd As Double
Dim z2grd As Double
Dim y2air As Double
Dim x2air As Double
Dim z2air As Double
```

```
Dim omega As Double
Dim phi As Double
Dim kappa As Double

Dim omega1 As Double
Dim phi1 As Double
Dim kappa1 As Double

Dim omega2 As Double
Dim phi2 As Double
Dim kappa2 As Double

Dim pi As Double

Dim strip1arr() As Double
Dim strip2arr() As Double

Dim xrange As Double
Dim yrange As Double
Dim zrange As Double

Dim xrange1 As Double
Dim yrange1 As Double
Dim zrange1 As Double

Dim xrange2 As Double
Dim yrange2 As Double
Dim zrange2 As Double


Dim a As Double  'constants of INS Rotation Matrix
Dim b As Double
Dim c As Double
Dim d As Double
Dim e As Double
Dim f As Double
Dim g As Double
Dim h As Double
Dim i As Double

Dim A1 As Double
Dim b1 As Double
Dim c1 As Double
Dim d1 As Double
Dim e1 As Double
Dim f1 As Double
```

```
Dim g1 As Double
Dim h1 As Double
Dim i1 As Double

Dim A2 As Double
Dim b2 As Double
Dim c2 As Double
Dim d2 As Double
Dim e2 As Double
Dim f2 As Double
Dim g2 As Double
Dim h2 As Double
Dim i2 As Double

Dim r00, r01, r02, r10, r11, r12, r20, r21, r22 As Double


Dim xcomb As Double  'reps [xgrd - (a*xrange + b*yrange + c*zrange) - xair]
Dim ycomb As Double
Dim zcomb As Double

Dim xcomb1 As Double
Dim ycomb1 As Double
Dim zcomb1 As Double

Dim xcomb2 As Double
Dim ycomb2 As Double
Dim zcomb2 As Double

Dim domega As Double 'reps delta omega, delta phi and delta kappa
Dim dphi As Double
Dim dkappa As Double

Dim dw As Double 'omega for estimation and iteration
Dim dp As Double
Dim dk As Double

Dim w_arr() As Double
Dim a_arr() As Double  'the design matrix
Dim n_arr(3, 3) As Double 'from TransposeATA_Arrays!
Dim u_arr(3) As Double
Dim x_arr(3) As Double

Dim dumw, dump, dumk As Double
Dim dumw2, dump2, dumk2 As Double
```

```
Dim sumk, angk As Double

'open strip #1 input file

   filenumin1 = FreeFile
   filepath = App.Path
   inputfile = Text1.Text

   Open filepath + "\" + inputfile For Input As filenumin1

   Input #filenumin1, dumw, dump, dumk


   counter1 = 0

      Do
         Input #filenumin1, pointnum, xgrd, ygrd, zgrd, xair, yair, zair, omega, phi, kappa,
                  xrange, yrange, zrange, intensity

         counter1 = counter1 + 1

      Loop While Not EOF(filenumin1)

   Close #filenumin1

ReDim strip1arr(counter1 - 1, 12)

   Open filepath + "\" + inputfile For Input As filenumin1
      Input #filenumin1, dumw, dump, dumk
      For i = 0 To counter1 - 1
         Input #filenumin1, pointnum, xgrd, ygrd, zgrd, xair, yair, zair, omega, phi, kappa,
                  xrange, yrange, zrange, intensity

         strip1arr(i, 0) = pointnum
         strip1arr(i, 1) = xgrd
         strip1arr(i, 2) = ygrd
         strip1arr(i, 3) = zgrd
         strip1arr(i, 4) = xair
         strip1arr(i, 5) = yair
         strip1arr(i, 6) = zair
         strip1arr(i, 7) = omega
         strip1arr(i, 8) = phi
         strip1arr(i, 9) = kappa
         strip1arr(i, 10) = xrange
         strip1arr(i, 11) = yrange
```

```
        strip1arr(i, 12) = zrange
    Next i

  Close #filenumin1

 'open strip #2 input file

  filenumin2 = FreeFile
  filepath = App.Path
  inputfile = Text2.Text

  Open filepath + "\" + inputfile For Input As filenumin2

  Input #filenumin2, dumw2, dump2, dumk2

  counter2 = 0

    Do
        Input #filenumin2, pointnum, xgrd, ygrd, zgrd, xair, yair, zair, omega, phi, kappa,
                xrange, yrange, zrange, intensity

        counter2 = counter2 + 1

    Loop While Not EOF(filenumin2)

  Close #filenumin2

ReDim strip2arr(counter2 - 1, 12)

  Open filepath + "\" + inputfile For Input As filenumin2
    Input #filenumin2, dumw2, dump2, dumk2

    For i = 0 To counter2 - 1
        Input #filenumin2, pointnum, xgrd, ygrd, zgrd, xair, yair, zair, omega, phi, kappa,
                xrange, yrange, zrange, intensity

        strip2arr(i, 0) = pointnum
        strip2arr(i, 1) = xgrd
        strip2arr(i, 2) = ygrd
        strip2arr(i, 3) = zgrd
        strip2arr(i, 4) = xair
        strip2arr(i, 5) = yair
        strip2arr(i, 6) = zair
        strip2arr(i, 7) = omega
        strip2arr(i, 8) = phi
        strip2arr(i, 9) = kappa
```

```
        strip2arr(i, 10) = xrange
        strip2arr(i, 11) = yrange
        strip2arr(i, 12) = zrange

      Next i


    Close #filenumin2


    filenumout = FreeFile
    filepath = App.Path
    outputfile = Text3.Text

    Open filepath + "\" + outputfile For Output As filenumout
'--------------------------------------------
    pi = Atn(1) * 4
'--------------------------------------------

    dw = 0# * pi / 180
    dp = 0# * pi / 180
    dk = 0# * pi / 180

    ReDim a_arr(3 * (counter1), 3) As Double
    ReDim w_arr(3 * (counter1)) As Double

    For iter = 0 To 5

    Print #filenumout, iter

'Calculating Boresight Angles

'****** ****** ****** ****** ******
' below is within iteration
 'sumk = 0

      'rotation matrix for misalignment estimates
        r00 = Cos(dp) * Cos(dk)
        r01 = -Cos(dp) * Sin(dk)
        r02 = Sin(dp)
        r10 = (Cos(dw) * Sin(dk)) + (Sin(dw) * Sin(dp) * Cos(dk))
        r11 = (Cos(dw) * Cos(dk)) - (Sin(dw) * Sin(dp) * Sin(dk))
        r12 = -Sin(dw) * Cos(dp)
        r20 = (Sin(dw) * Sin(dk)) - (Cos(dw) * Sin(dp) * Cos(dk))
        r21 = (Sin(dw) * Cos(dk)) + (Cos(dw) * Sin(dp) * Sin(dk))
        r22 = Cos(dw) * Cos(dp)
```

```
For ii = 0 To counter1 - 1
   x1grd = strip1arr(ii, 1)
   y1grd = strip1arr(ii, 2)
   z1grd = strip1arr(ii, 3)
   x1air = strip1arr(ii, 4)
   y1air = strip1arr(ii, 5)
   z1air = strip1arr(ii, 6)
   omega1 = strip1arr(ii, 7) * pi / 180 'going from deg to rad!
   phi1 = strip1arr(ii, 8) * pi / 180
   kappa1 = strip1arr(ii, 9) * pi / 180
   xrange1 = strip1arr(ii, 10)
   yrange1 = strip1arr(ii, 11)
   zrange1 = strip1arr(ii, 12)

'constants of INS Rotation Matrix using Omega Phi Kappa

   'A1 = Cos(phi1) * Cos(kappa1)
   'b1 = -Cos(phi1) * Sin(kappa1)
   'c1 = Sin(phi1)
   'd1 = (Cos(omega1) * Sin(kappa1)) + (Sin(omega1) * Sin(phi1) * Cos(kappa1))
   'e1 = (Cos(omega1) * Cos(kappa1)) - (Sin(omega1) * Sin(phi1) * Sin(kappa1))
   'f1 = -Sin(omega1) * Cos(phi1)
   'g1 = (Sin(omega1) * Sin(kappa1)) - (Cos(omega1) * Sin(phi1) * Cos(kappa1))
   'h1 = (Sin(omega1) * Cos(kappa1)) + (Cos(omega1) * Sin(phi1) * Sin(kappa1))
   'i1 = Cos(omega1) * Cos(phi1)

   'constants of INS Rotation Matrix using Roll Pitch Heading

   A1 = Cos(phi1) * Cos(kappa1)
   b1 = (Sin(omega1) * Sin(phi1) * Cos(kappa1)) - (Cos(omega1) * Sin(kappa1))
   c1 = (Cos(omega1) * Sin(phi1) * Cos(kappa1)) + (Sin(omega1) * Sin(kappa1))
   d1 = Cos(phi1) * Sin(kappa1)
   e1 = (Sin(omega1) * Sin(phi1) * Sin(kappa1)) + (Cos(omega1) * Cos(kappa1))
   f1 = (Cos(omega1) * Sin(phi1) * Sin(kappa1)) - (Sin(omega1) * Cos(kappa1))
   g1 = -Sin(phi1)
   h1 = Sin(omega1) * Cos(phi1)
   i1 = Cos(omega1) * Cos(phi1)


   xrange = r00 * xrange1 + r01 * yrange1 + r02 * zrange1
   yrange = r10 * xrange1 + r11 * yrange1 + r12 * zrange1
   zrange = r20 * xrange1 + r21 * yrange1 + r22 * zrange1


   'xcomb gives the transformed coordinates for strip 1
```

```
    xcomb1 = (A1 * xrange + b1 * yrange + c1 * zrange)
    ycomb1 = (d1 * xrange + e1 * yrange + f1 * zrange)
    zcomb1 = (g1 * xrange + h1 * yrange + i1 * zrange

    xcomb1 = xcomb1 + x1air
    ycomb1 = ycomb1 + y1air
    zcomb1 = zcomb1 + z1air

    x2grd = strip2arr(ii, 1)
    y2grd = strip2arr(ii, 2)
    z2grd = strip2arr(ii, 3)
    x2air = strip2arr(ii, 4)
    y2air = strip2arr(ii, 5)
    z2air = strip2arr(ii, 6)
    omega2 = strip2arr(ii, 7) * pi / 180    'going from deg to rad!
    phi2 = strip2arr(ii, 8) * pi / 180
    kappa2 = strip2arr(ii, 9) * pi / 180
    xrange2 = strip2arr(ii, 10)
    yrange2 = strip2arr(ii, 11)
    zrange2 = strip2arr(ii, 12)

'constants of INS Rotation Matrix using Omega Phi Kappa

    'A2 = Cos(phi2) * Cos(kappa2)
    'b2 = -Cos(phi2) * Sin(kappa2)
    'c2 = Sin(phi2)
    'd2 = (Cos(omega2) * Sin(kappa2)) + (Sin(omega2) * Sin(phi2) * Cos(kappa2))
    'e2 = (Cos(omega2) * Cos(kappa2)) - (Sin(omega2) * Sin(phi2) * Sin(kappa2))
    'f2 = -Sin(omega2) * Cos(phi2)
    'g2 = (Sin(omega2) * Sin(kappa2)) - (Cos(omega2) * Sin(phi2) * Cos(kappa2))
    'h2 = (Sin(omega2) * Cos(kappa2)) + (Cos(omega2) * Sin(phi2) * Sin(kappa2))
    'i2 = Cos(omega2) * Cos(phi2)

'constants of INS Rotation Matrix using Roll Pitch Heading

    A2 = Cos(phi2) * Cos(kappa2)
    b2 = (Sin(omega2) * Sin(phi2) * Cos(kappa2)) - (Cos(omega2) * Sin(kappa2))
    c2 = (Cos(omega2) * Sin(phi2) * Cos(kappa2)) + (Sin(omega2) * Sin(kappa2))
    d2 = Cos(phi2) * Sin(kappa2)
    e2 = (Sin(omega2) * Sin(phi2) * Sin(kappa2)) + (Cos(omega2) * Cos(kappa2))
    f2 = (Cos(omega2) * Sin(phi2) * Sin(kappa2)) - (Sin(omega2) * Cos(kappa2))
    g2 = -Sin(phi2)
    h2 = Sin(omega2) * Cos(phi2)
    i2 = Cos(omega2) * Cos(phi2)
```

```
xrange3 = r00 * xrange2 + r01 * yrange2 + r02 * zrange2
yrange3 = r10 * xrange2 + r11 * yrange2 + r12 * zrange2
zrange3 = r20 * xrange2 + r21 * yrange2 + r22 * zrange2

'xcomb2 gives the transformed coordinates for strip 2
xcomb2 = (A2 * xrange3 + b2 * yrange3 + c2 * zrange3) + x2air
ycomb2 = (d2 * xrange3 + e2 * yrange3 + f2 * zrange3) + y2air
zcomb2 = (g2 * xrange3 + h2 * yrange3 + i2 * zrange3) + z2air

xcomb = xcomb1 - xcomb2
ycomb = ycomb1 - ycomb2
zcomb = zcomb1 - zcomb2

l = ii * 3

'Partial derivatives a_arr ( )

 'd(omega)

 a_arr(l, 0) = b1 * ((-Sin(dw) * Sin(dk) + Cos(dw) * Sin(dp) * Cos(dk)) * xrange1
+ (-Sin(dw) * Cos(dk) - Cos(dw) * Sin(dp) * Sin(dk)) * yrange1 - (Cos(dw) * Cos(dp)) *
zrange1) + c1 * ((Cos(dw) * Sin(dk) + Sin(dw) * Sin(dp) * Cos(dk)) * xrange1 +
(Cos(dw) * Cos(dk) - Sin(dw) * Sin(dp) * Sin(dk)) * yrange1 - (Sin(dw) * Cos(dp)) *
zrange1) - (b2 * ((-Sin(dw) * Sin(dk) + Cos(dw) * Sin(dp) * Cos(dk)) * xrange2 + (-
Sin(dw) * Cos(dk) - Cos(dw) * Sin(dp) * Sin(dk)) * yrange2 - (Cos(dw) * Cos(dp)) *
zrange2) + c2 * ((Cos(dw) * Sin(dk) + Sin(dw) * Sin(dp) * Cos(dk)) * xrange2 +
(Cos(dw) * Cos(dk) - Sin(dw) * Sin(dp) * Sin(dk)) * yrange2 - (Sin(dw) * Cos(dp)) *
zrange2))

 a_arr(l + 1, 0) = e1 * ((-Sin(dw) * Sin(dk) + Cos(dw) * Sin(dp) * Cos(dk)) *
xrange1 + (-Sin(dw) * Cos(dk) - Cos(dw) * Sin(dp) * Sin(dk)) * yrange1 - (Cos(dw) *
Cos(dp)) * zrange1) + f1 * ((Cos(dw) * Sin(dk) + Sin(dw) * Sin(dp) * Cos(dk)) *
xrange1 + (Cos(dw) * Cos(dk) - Sin(dw) * Sin(dp) * Sin(dk)) * yrange1 - (Sin(dw) *
Cos(dp)) * zrange1) - (e2 * ((-Sin(dw) * Sin(dk) + Cos(dw) * Sin(dp) * Cos(dk)) *
xrange2 + (-Sin(dw) * Cos(dk) - Cos(dw) * Sin(dp) * Sin(dk)) * yrange2 - (Cos(dw) *
Cos(dp)) * zrange2) + f2 * ((Cos(dw) * Sin(dk) + Sin(dw) * Sin(dp) * Cos(dk)) *
xrange2 + (Cos(dw) * Cos(dk) - Sin(dw) * Sin(dp) * Sin(dk)) * yrange2 - (Sin(dw) *
Cos(dp)) * zrange2))

 a_arr(l + 2, 0) = h1 * ((-Sin(dw) * Sin(dk) + Cos(dw) * Sin(dp) * Cos(dk)) *
xrange1 + (-Sin(dw) * Cos(dk) - Cos(dw) * Sin(dp) * Sin(dk)) * yrange1 - (Cos(dw) *
Cos(dp)) * zrange1) + i1 * ((Cos(dw) * Sin(dk) + Sin(dw) * Sin(dp) * Cos(dk)) *
xrange1 + (Cos(dw) * Cos(dk) - Sin(dw) * Sin(dp) * Sin(dk)) * yrange1 - (Sin(dw) *
Cos(dp)) * zrange1) - (h2 * ((-Sin(dw) * Sin(dk) + Cos(dw) * Sin(dp) * Cos(dk)) *
xrange2 + (-Sin(dw) * Cos(dk) - Cos(dw) * Sin(dp) * Sin(dk)) * yrange2 - (Cos(dw) *
```

Cos(dp)) * zrange2) + i2 * ((Cos(dw) * Sin(dk) + Sin(dw) * Sin(dp) * Cos(dk)) * xrange2 + (Cos(dw) * Cos(dk) - Sin(dw) * Sin(dp) * Sin(dk)) * yrange2 - (Sin(dw) * Cos(dp)) * zrange2))

'd(phi)

a_arr(l, 1) = A1 * (-Sin(dp) * Cos(dk) * xrange1 + Sin(dp) * Sin(dk) * yrange1 + Cos(dp) * zrange1) + b1 * (Sin(dw) * Cos(dp) * Cos(dk) * xrange1 - Sin(dw) * Cos(dp) * Sin(dk) * yrange1 + Sin(dw) * Sin(dp) * zrange1) + c1 * (-Cos(dw) * Cos(dp) * Cos(dk) * xrange1 + Cos(dw) * Cos(dp) * Sin(dk) * yrange1 - Cos(dw) * Sin(dp) * zrange1) - (A2 * (-Sin(dp) * Cos(dk) * xrange2 + Sin(dp) * Sin(dk) * yrange2 + Cos(dp) * zrange2) + b2 * (Sin(dw) * Cos(dp) * Cos(dk) * xrange2 - Sin(dw) * Cos(dp) * Sin(dk) * yrange2 + Sin(dw) * Sin(dp) * zrange2) + c2 * (-Cos(dw) * Cos(dp) * Cos(dk) * xrange2 + Cos(dw) * Cos(dp) * Sin(dk) * yrange2 - Cos(dw) * Sin(dp) * zrange2))

a_arr(l + 1, 1) = d1 * (-Sin (dp) * Cos(dk) * xrange1 + Sin(dp) * Sin(dk) * yrange1 + Cos(dp) * zrange1) + e1 * (Sin(dw) * Cos(dp) * Cos(dk) * xrange1 - Sin(dw) * Cos(dp) * Sin(dk) * yrange1 + Sin(dw) * Sin(dp) * zrange1) + f1 * (-Cos(dw) * Cos(dp) * Cos(dk) * xrange1 + Cos(dw) * Cos(dp) * Sin(dk) * yrange1 - Cos(dw) * Sin(dp) * zrange1) - (d2 * (-Sin(dp) * Cos(dk) * xrange2 + Sin(dp) * Sin(dk) * yrange2 + Cos(dp) * zrange2) + e2 * (Sin(dw) * Cos(dp) * Cos(dk) * xrange2 - Sin(dw) * Cos(dp) * Sin(dk) * yrange2 + Sin(dw) * Sin(dp) * zrange2) + f2 * (-Cos(dw) * Cos(dp) * Cos(dk) * xrange2 + Cos(dw) * Cos(dp) * Sin(dk) * yrange2 - Cos(dw) * Sin(dp) * zrange2))

a_arr(l + 2, 1) = g1 * (-Sin (dp) * Cos(dk) * xrange1 + Sin(dp) * Sin(dk) * yrange1 + Cos(dp) * zrange1) + h1 * (Sin(dw) * Cos(dp) * Cos(dk) * xrange1 - Sin(dw) * Cos(dp) * Sin(dk) * yrange1 + Sin(dw) * Sin(dp) * zrange1) + i1 * (-Cos(dw) * Cos(dp) * Cos(dk) * xrange1 + Cos(dw) * Cos(dp) * Sin(dk) * yrange1 - Cos(dw) * Sin(dp) * zrange1) - (g2 * (-Sin(dp) * Cos(dk) * xrange2 + Sin(dp) * Sin(dk) * yrange2 + Cos(dp) * zrange2) + h2 * (Sin(dw) * Cos(dp) * Cos(dk) * xrange2 - Sin(dw) * Cos(dp) * Sin(dk) * yrange2 + Sin(dw) * Sin(dp) * zrange2) + i2 * (-Cos(dw) * Cos(dp) * Cos(dk) * xrange2 + Cos(dw) * Cos(dp) * Sin(dk) * yrange2 - Cos(dw) * Sin(dp) * zrange2))

'd(kappa)

a_arr(l, 2) = A1 * (-Cos(dp) * Sin(dk) * xrange1 - Cos(dp) * Cos(dk) * yrange1) + b1 * ((Cos(dw) * Cos(dk) - Sin(dw) * Sin(dp) * Sin(dk)) * xrange1 + (-Cos(dw) * Sin(dk) - Sin(dw) * Sin(dp) * Cos(dk)) * yrange1) + c1 * ((Sin(dw) * Cos(dk) + Cos(dw) * Sin(dp) * Sin(dk)) * xrange1 + (-Sin(dw) * Sin(dk) + Cos(dw) * Sin(dp) * Cos(dk)) * yrange1) - (A2 * (-Cos(dp) * Sin(dk) * xrange2 - Cos(dp) * Cos(dk) * yrange2) + b2 * ((Cos(dw) * Cos(dk) - Sin(dw) * Sin(dp) * Sin(dk)) * xrange2 + (-Cos(dw) * Sin(dk) - Sin(dw) * Sin(dp) * Cos(dk)) * yrange2) + c2 * ((Sin(dw) * Cos(dk) + Cos(dw) * Sin(dp) * Sin(dk)) * xrange2 + (-Sin(dw) * Sin(dk) + Cos(dw) * Sin(dp) * Cos(dk)) * yrange2)) a_arr(l + 1, 2) = d1 * (-Cos (dp) * Sin(dk) * xrange1 - Cos(dp) * Cos(dk) * yrange1) + e1 * ((Cos(dw) * Cos(dk) - Sin(dw) * Sin(dp) * Sin(dk)) * xrange1 + (-Cos(dw) * Sin(dk) - Sin(dw) * Sin(dp) * Cos(dk)) * yrange1) + f1 * ((Sin(dw) * Cos(dk) + Cos(dw) * Sin(dp) * Sin(dk)) * xrange1 + (-Sin(dw) * Sin(dk) + Cos(dw) * Sin(dp) * Cos(dk)) * yrange1) - (d2 * (-Cos(dp) * Sin(dk) * xrange2 - Cos(dp) * Cos(dk) * yrange2) + e2 * ((Cos(dw) * Cos(dk) - Sin(dw) * Sin(dp) * Sin(dk)) * xrange2 + (-Cos(dw) * Sin(dk) - Sin(dw) * Sin(dp) * Cos(dk)) * yrange2) + f2 * ((Sin(dw) * Cos(dk) + Cos(dw) * Sin(dp) * Sin(dk)) * xrange2 + (-Sin(dw) * Sin(dk) + Cos(dw) * Sin(dp) * Cos(dk)) * yrange2))

a_arr(l + 2, 2) = g1 * (-Cos (dp) * Sin(dk) * xrange1 - Cos(dp) * Cos(dk) * yrange1) + h1 * ((Cos(dw) * Cos(dk) - Sin(dw) * Sin(dp) * Sin(dk)) * xrange1 + (-Cos(dw) * Sin(dk) - Sin(dw) * Sin(dp) * Cos(dk)) * yrange1) + i1 * ((Sin(dw) * Cos(dk) + Cos(dw) * Sin(dp) * Sin(dk)) * xrange1 + (-Sin(dw) * Sin(dk) + Cos(dw) * Sin(dp) * Cos(dk)) * yrange1) - (g2 * (-Cos(dp) * Sin(dk) * xrange2 - Cos(dp) * Cos(dk) * yrange2) + h2 * ((Cos(dw) * Cos(dk) - Sin(dw) * Sin(dp) * Sin(dk)) * xrange2 + (-Cos(dw) * Sin(dk) - Sin(dw) * Sin(dp) * Cos(dk)) * yrange2) + i2 * ((Sin(dw) * Cos(dk) + Cos(dw) * Sin(dp) * Sin(dk)) * xrange2 + (-Sin(dw) * Sin(dk) + Cos(dw) * Sin(dp) * Cos(dk)) * yrange2))

'Alternate Partial derivatives

'a_arr(l, 0) = (c1 * yrange1 - b1 * zrange1) - (c2 * yrange2 - b2 * zrange2)
'a_arr(l, 1) = (A1 * zrange1 - c1 * xrange1) - (A2 * zrange2 - c2 * xrange2)
'a_arr(l, 2) = (b1 * xrange1 - A1 * yrange1) - (b2 * xrange2 - A2 * yrange2)

'a_arr(l + 1, 0) = (f1 * yrange1 - e1 * zrange1) - (f2 * yrange2 - e2 * zrange2)
'a_arr(l + 1, 1) = (d1 * zrange 1 - f1 * xrange1) - (d2 * zrange2 - f2 * xrange2)
'a_arr(l + 1, 2) = (e1 * xrange1 - d1 * yrange1) - (e2 * xrange2 - d2 * yrange2)

'a_arr(l + 2, 0) = (i1 * yrange1 - h1 * zrange1) - (i2 * yrange2 - h2 * zrange2)
'a_arr(l + 2, 1) = (g1 * zrange1 - i1 * xrange1) - (g2 * zrange2 - i2 * xrange2)
'a_arr(l + 2, 2) = (h1 * xra nge1 - g1 * yrange1) - (h2 * xrange2 - g2 * yrange2)

'applying priori observation standard deviations

stddev = 0.0578 'assumed Laser footprint: 20cm *0.289 = 5.78cm

a_arr(l, 0) = a_arr(l, 0) / stddev
a_arr(l, 1) = a_arr(l, 1) / stddev
a_arr(l, 2) = a_arr(l, 2) / stddev

a_arr(l + 1, 0) = a_arr(l + 1, 0) / stddev
a_arr(l + 1, 1) = a_arr(l + 1, 1) / stddev
a_arr(l + 1, 2) = a_arr(l + 1, 2) / stddev

```
        a_arr(l + 2, 0) = a_arr(l + 2, 0) / stddev
        a_arr(l + 2, 1) = a_arr(l + 2, 1) / stddev
        a_arr(l + 2, 2) = a_arr(l + 2, 2) / stddev


        w_arr(l) = xcomb
        w_arr(l + 1) = ycomb
        w_arr(l + 2) = zcomb


        Print #filenumout, ii, xcomb1, ycomb1, zcomb1, xcomb2, ycomb2, zcomb2,
         w_arr(l), w_arr(l + 1), w_arr(l + 2), , a_arr(l, 0), a_arr(l, 1), a_arr(l, 2),
         a_arr(l + 1, 0), a_arr(l + 1, 1), a_arr(l + 1, 2), a_arr(l + 2, 0), a_arr(l + 2, 1),
         a_arr(l + 2, 2)

        w_arr(l) = xcomb / stddev
        w_arr(l + 1) = ycomb / stddev
        w_arr(l + 2) = zcomb / stddev

      Next ii


      Call TransposeATA_Arrays(a_arr(), counter1 * 3, 3, n_arr())

      Call TransposeATB_Arrays(a_arr(), w_arr(), counter1 * 3, 3, u_arr())

      Call Inverse_Arrays(n_arr(), u_arr(), 3, x_arr())


      dw = dw - x_arr(0) ' changed sign
      dp = dp - x_arr(1) ' changed sign
      dk = dk - x_arr(2) ' changed sign


      Print #filenumout, "angles", dw * 180 / pi, dp * 180 / pi, dk * 180 / pi


      'End If

    Next iter

Close #filenumout

End Sub
```

Modules

```
Public Sub TransposeATA_Arrays(a_matrix() As Double, maxrows As Integer, maxcols
        As Integer, n_matrix() As Double)

Dim row, col, sol As Integer


  For row = 0 To maxcols - 1
    For col = 0 To maxcols - 1
      n_matrix(row, col) = 0#
      For sol = 0 To maxrows - 1
        n_matrix(row, col) = n_matrix(row, col) + a_matrix(sol, row) * a_matrix(sol,
              col)
      Next sol
    Next col
  Next row
End Sub



Public Sub TransposeATB_Arrays(a_matrix() As Double, b() As Double, maxrows As
        Integer, maxcols As Integer, u_vector() As Double)

Dim row, col As Integer

  Dim i As Integer, j As Integer

  'v = flag
  For row = 0 To maxcols - 1
    u_vector(row) = 0#
    For col = 0 To maxrows - 1
      u_vector(row) = u_vector(row) + a_matrix(col, row) * b(col)
    Next col
    col = 1
  Next row
End Sub



Public Sub Inverse_Arrays(n_matrix() As Double, u_vector() As Double, maxcols1 As
        Integer, x() As Double)

Dim sum As Double
Dim k, i, j, n As Integer
Dim flag As Integer

 ReDim work(maxcols1) As Double, b(maxcols1) As Double
```

```
flag = 1
  n = maxcols1 - 1
  For i = 0 To n
    For j = i To n
      sum = n_matrix(i, j)
      For k = i - 1 To 0 Step -1
        sum = sum - n_matrix(i, k) * n_matrix(j, k)
      Next k
      If i = j Then
        If sum <= 0# Then
          Message = "choldc failed"
          ButtonsAndIcons = vbOKOnly + vbExclam ation
          Title = "ERROR MSG."
          MsgBox Message, ButtonsAndIcons, Title
        Else
          work(i) = Sqr(sum )
        End If
      Else
        n_m atrix(j, i) = sum / work(i)
      End If
    Next j
  Next i


  For i = 0 To n
    sum = u_vector(i)
    For k = i - 1 To 0 Step -1
      sum = sum - n_matrix(i, k) * x(k)
    Next k
    x(i) = sum / work(i)
  Next i

  For i = n To 0 Step -1
    sum = x(i)
    For k = i + 1 To n
      sum = sum - n_matrix(k, i) * x(k)
    Next k
    x(i) = sum / work(i)
  Next i

  flag = 1
  If flag = 1 Then
    For i = 0 To n
      n_m atrix(i, i) = 1# / work(i)
      For j = i + 1 To n
```

```
        sum = 0#
          For k = i To j - 1
            sum = sum - n_matrix(j, k) * n_matrix(k, i)
          Next k
        n_matrix(j, i) = sum / work(j)
      Next j
    Next i
  End If

  Debug.Print n_matrix(0, 0), n_matrix(0, 1), n_matrix(0, 2)
  Debug.Print n_matrix(1, 0), n_matrix(1, 1), n_matrix(1, 2)
  Debug.Print n_matrix(2, 0), n_matrix(2, 1), n_matrix(2, 2)

End Sub
```

Appendix B

Sample input file for Misalignment Estimator Program

Intial Approximates of the Misalignment Angles

point number	X (Ground Coordinates)	Y (Ground Coordinates)	Z (Ground Coordinates)	X (Air Coordinates)	Y (Air Coordinates)	Z (Air Coordinates)	roll or omega	pitch or phi	heading or kappa
0	0	0							
15	806	662.7	101.1	1035	551.5	1977.26	0.2408	1.2143	-143.514
21	804.3	428.3	99.44	975.4	385.1	1978.9	0.3804	1.4599	-148.388
22	832.5	418.9	99.26	977.7	391.5	1978.82	0.5003	1.4909	-148.26
23	869	404.6	99.08	980.1	398	1978.74	0.6057	1.5193	-148.105
24	888	393.4	98.84	980.1	397.9	1978.74	0.6053	1.5192	-148.106
25	929.3	401.8	98.88	990.6	427	1978.36	0.8549	1.607	-147.117
26	918.1	387.9	98.65	984.1	409.1	1978.59	0.7539	1.5617	-147.775
27	942.8	378.7	98.43	985.9	414.1	1978.53	0.8023	1.5769	-147.603
28	975.4	368.1	98.19	988.8	422.1	1978.42	0.847	1.595	-147.307
29	923	329.4	97.56	965.3	357.4	1979.24	-0.202	1.357	-148.74
30	935.2	314.7	97.63	963	350.9	1979.31	-0.334	1.3499	-148.79
31	956.6	317.5	97.4	967.7	363.9	1979.16	-0.059	1.3698	-148.681
32	919.4	213.1	95.97	928.6	255.1	1980.28	-1.929	1.2308	-149.385
33	904.9	288.7	97.27	950.1	315.3	1979.69	-1.058	1.3587	-149.103
34	917.8	281.4	97.26	950.1	315.3	1979.69	-1.058	1.3587	-149.103
35	901.5	264.2	97.2	942	292.6	1979.92	-1.459	1.3325	-149.278
36	881.7	252	96.99	934.4	271.5	1980.13	-1.749	1.2772	-149.367
37	885.3	250	97.09	934.4	271.5	1980.13	-1.749	1.2772	-149.367
38	898.2	241.1	97.02	933.9	269.9	1980.14	-1.767	1.2705	-149.367
39	867.1	214.3	97.03	920.1	230.9	1980.5	-2.145	1.2015	-149.358

X range	Y range	Z range	Intensity
158.03311	-233.67237	-1872.2563	8
170.926676	-138.9407	-1874.8329	15
158.025722	-116.02275	-1875.1419	10
140.618589	-84.129611	-1875.8029	10
130.374432	-64.583127	-1876.5277	7
117.833673	-40.136642	-1876.5277	13
118.369921	-41.947486	-1877.0259	16
107.054425	-19.52111	-1877.7739	16
92.798029	10.3608165	-1878.7393	8
95.2693777	8.58139761	-1879.9305	10
86.8142513	27.5176064	-1880.0292	11
78.566096	35.7973862	-1880.3844	9
69.8517943	94.8381921	-1881.1219	12
97.1348822	34.3564076	-1880.3327	11
89.789456	47.2317252	-1880.2792	14
93.1074202	51.6220997	-1880.3617	11
97.2522201	47.3197905	-1880.8748	14
95.1858327	50.8604261	-1880.7077	19
87.1471257	64.6282038	-1880.5557	12
93.5124775	57.6829796	-1881.0749	11