Backpack-based inertial navigation and LiDAR mapping in forest environments

Mattias Tjernqvist
Department of Physics
Umeå University

A master’s thesis submitted for the degree of

Master of Science in Engineering Physics

June 2017
Title:
Backpack-based inertial navigation and LiDAR mapping in forest environments

Author:
Mattias Tjernqvist, matj0016@student.umu.se

Examiner:
Ola Ringdahl, Umeå University

Supervisor:
Johan Holmgren, Swedish University of Agricultural Sciences

Assistant Supervisor 1:
Kenneth Olofsson, Swedish University of Agricultural Sciences

Assistant Supervisor 2:
Michael Tulldahl, Swedish Defence Research Agency

Day of the defense: 2017-06-09

©2017 Mattias Tjernqvist
Abstract

Creating 3D models of our surrounding world has seen a rapid increase in research and development over the last few years. A common method is to use laser scanners. Mapping is done either by ground-based systems or airborne systems. With stationary ground-based laser scanning, or terrestrial laser scanning (TLS), it is possible to obtain high accuracy point clouds. But stationary TLS can often be a cumbersome and time-demanding task due to its lack of mobility. Because of this, much research has gone into mobilised TLS systems, referred commonly to as mobile laser scanning (MLS). Georeferencing point clouds to a world coordinate system is a difficult task in environments where global navigation satellite systems (GNSS) is unreliable. One such environment is forests, where the GNSS signal can be blocked, absorbed or reflected from the trees and canopy. Accurate georeference of points clouds for MLS systems in forests is a difficult task that can be solved by using additional measurement instruments and post-processing algorithms to reduce the accumulation of errors, also known as drift. In this thesis a backpack-based MLS system to be used in forests was tested. The MLS system was composed of a GNSS, an inertial navigation unit (INS) and a laser scanner. The collected data was post-processed and analyzed to reduce the effects of detecting multiple ground layers and multiples of the same tree due to drift. The post-processing algorithm calculated tree and ground features to be used for adjusting the point cloud in the horizontal and vertical planes.

The forest survey was done for an area roughly 40 meters in diameter. The MLS data was compared against TLS data as well as manual
caliper data — where the caliper data was only measured in an area roughly 24 meters in diameter. The results indicated that the effects of multiple ground layers and multiple tree copies were removed after post-processing. Out of the total 214 TLS trees, 185 managed to be co-registered to MLS trees. The root mean square error (RMSE) and bias of the diameter at breast height (DBH) between the MLS and TLS data were 27.00 mm and $-9.33$ mm respectively. Co-registration of the MLS and manual caliper data set gave 36 successful matches out of the total 43 manually measured DBH. The DBH RMSE and bias were 16.95 mm and $-10.58$ mm respectively. A Swedish TLS forest study obtained a DBH RMSE and bias (between TLS and caliper) of approximately 10 mm and $+0.06$ mm respectively. A Finnish backpack MLS forest study obtained a DBH RMSE and bias (between MLS and TLS) of 50.6 mm and $+11.1$ mm respectively. Evaluating the difference in radius at different heights along the tree stems between the MLS and TLS revealed a slight dependence on height, as the radius difference increased slightly closer to the stem base.

The results indicated that backpack-based MLS systems has the potential for accurate laser mapping in forests, and future development is of great interest to improve this system further.
Acknowledgements

I would like to thank my supervisor, Johan Holmgren, for his support and great interest in this project. I would also like to thank my assistant supervisors, Michael Tulldahl, for helpful discussions, and Kenneth Olofsson for stepping in as an additional assistant supervisor; Jonas Nordlöf for traveling many kilometers to assist; and Bryan Leedham and David MacDonald at NovAtel support for great discussions and advice. Lastly, I would also like to thank Axel Andersson and Konrad Steinvall for fun conversations during both "the good and the bad" times.
| 4.4 Survey Plot Co-registration | 24 |
| 4.5 Dynamic Calibration | 27 |
| 4.6 Inertial Explorer | 28 |
| 4.7 Snow, Walking Pattern & Pitch Angle | 29 |

5 Conclusions 30

References 32

A User Guidelines for Processing in Inertial Explorer 36
   A.1 Processing Settings 36
      A.1.1 Advanced IMU Settings 37
      A.1.2 Advanced GNSS Settings 40

B User Guide for the Dynamic Calibration Tools 43
   B.1 K12 Tool 43
   B.2 3DUAV Tool 45
   B.3 3DUAV Analyzer Tool 46
   B.4 Cumulative Tree Analysis Tool 48
# List of abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALS</td>
<td>Airborne Laser Scanning</td>
</tr>
<tr>
<td>DBH</td>
<td>Diameter At Breast Height</td>
</tr>
<tr>
<td>GNSS</td>
<td>Global Navigation Satellite System</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>IMU</td>
<td>Inertial Measurement Unit</td>
</tr>
<tr>
<td>INS</td>
<td>Inertial Navigation System</td>
</tr>
<tr>
<td>LiDAR</td>
<td>Light Detection And Ranging</td>
</tr>
<tr>
<td>MLS</td>
<td>Mobile Laser Scanning</td>
</tr>
<tr>
<td>RMSE</td>
<td>Root Mean Square Error</td>
</tr>
<tr>
<td>ROI</td>
<td>Region Of Interest</td>
</tr>
<tr>
<td>TLS</td>
<td>Terrestrial Laser Scanning</td>
</tr>
<tr>
<td>ZUPT</td>
<td>Zero Velocity Update</td>
</tr>
</tbody>
</table>
1

Introduction

It is becoming ever more attractive to create 3D models of our surrounding environments for civil, commercial and military purposes. A frequently used method of 3D modelling the environment is known as LiDAR (Light Detection And Ranging). A common LiDAR method for modelling an environment, or mapping, is through the usage of laser scanners. The 3D models obtained from LiDAR are usually referred to as point clouds. A great advantage of laser mapping — compared to for example, mapping by photogrammetry (by photographs) — is the capability of penetrating different objects (such as vegetation) and thus reveal any obscured objects [1]. Mapping is done either by airborne or ground-based systems — both suffering from their respective pros and cons. Laser mapping by air, commonly known as Airborne Laser Scanning (ALS), has seen a rapid increase in development and usage of small UAV:s (Unmanned Aerial Vehicles) due to their ease of operation, obtain-ability and cost-effectiveness, compared to manned aerial vehicles. The main advantage with ALS is the ability to quickly map large areas [1–4]. For example, a large part of Sweden has been mapped by ALS using aircraft [5]. Ground-based laser mapping, commonly known as terrestrial laser scanning (TLS), can either be mobile or stationary. Stationary TLS is commonly referred to as simply ”TLS” (that is, ”stationary” or ”static” is not mentioned and it is understood that simply ”TLS” refers to the stationary case). TLS have shown much promise in generating high accuracy point clouds for a wide range of different environments and purposes on ground — such as estimating tree shape and stem diameter [6–8], detecting rockfall and landslide events [9,11] as
1. INTRODUCTION

well as mapping caves and other indoor environments [12, 13]. The largest issue with TLS is it often being a cumbersome and time-demanding task, increasingly hindered by increasing size of the sample area to be mapped. Thus, mobilised TLS, commonly known as mobile laser scanning (MLS), is a research area with great opportunities for increasing the mapping area and cost-effectiveness of TLS [14, 15].

In many applications it is important to georeference the point cloud. Georeferencing a point cloud means that the local coordinates of the point cloud is referenced to world coordinates, like latitude, longitude and altitude. For clear sky environments it is often sufficient to use a Global Navigation Satellite System (GNSS) for georeferencing — where the accuracy of georeferencing is mostly limited by the positional accuracy of the GNSS [16]. But, this task becomes increasingly difficult when satellite availability decreases. Hence, for indoor environments there will be zero satellite availability, and depending solely on GNSS position estimation is impossible. For forests the availability largely depends on the amount of coverage by the forest canopy, which can block or absorb the signal causing either a poor signal or complete signal loss. Another very prominent issue that arises for GNSS usage in forests is the phenomena of multipathing, where the receiver can obtain multiple signals from the satellite, as caused by the signal bouncing on trees. Multipathening is not limited to GNSS applications in forests but prominent in any environment where there is a possibility for the signal to bounce on nearby obstructing objects reaching above the receiver — such as buildings or canyon walls [17]. These combined can give errors up to several meters in the position estimation. For MLS the problem of position estimation is increasingly difficult compared to TLS, where centimeter accuracy is achievable given enough static observational time [18]. Thus, for accurate position estimation and georeferencing to be possible in environments where GNSS availability is unreliable, then other methods have to be employed for position estimation.

The most typical solution to this problem is introducing additional instruments for estimating the position by other means. Commonly, these additional instruments are Inertial Navigation Systems (INS) or Inertial Measurement Units (IMU), cameras, and magnetometers [19]. The instruments are then fused together in what is known as sensor fusion. Sensor fusion usually refers to a
post-measurement or real time combination of the measurement data through algorithms, but may also include a direct fusion of the instrument signals (e.g. electronically). The idea behind including additional instruments and sensor fusion is to have the instruments support each other – when one instrument becomes unreliable (for example the GNSS) another instrument is given more weight (for example an INS) and vice versa. All instruments will have some form of error, but by sensor fusion the total error accumulation can be reduced. There are many different sensor fusion algorithms and techniques but usually they include Kalman filters, particle filters, lidar scan matching and bundle adjustment \cite{19,23}.

Positioning and lidar mapping in forests has seen much research in Sweden and Finland. For example, in Olofsson & Holmgren \cite{24} TLS measurements were conducted in a forest and the tree stems and heights were investigated. The forest was composed of mainly Norway spruce \textit{(Picea abies} L. Karst.) and Scots pine \textit{(Pinus sylvestris} L.). In Liang \textit{et al.} \cite{14}, a forest composed of mainly Scots pine was mapped by a robot-based MLS system. Also, in Liang \textit{et al.} \cite{25} a backpack-based MLS system mapped a forest dominated by Scots pine.

The focus of this thesis was assembly and analysis of a MLS system consisting of a GNSS, an INS and a laser scanner – mounted on a backpack. The system was tested in forest environments and the accuracy of the MLS point cloud was evaluated after post-processing with the dynamic calibration tools developed by Michael Tulldahl \textit{et al.} at Totalförsvarets forskningsinstitut (FOI). The dynamic calibration tools were modified to work with the system used in this thesis as well as expanded upon. The accuracy of the post-processed MLS point cloud was evaluated mainly by comparing tree stem diameters to TLS data obtained from a Trimble TX8 as well as manual caliper diameter measurements. A comparison was also made to other research papers on MLS and TLS in forest environments. The research questions were stated as:

- How does this MLS system compare to TLS measurements, manual caliper measurements, as well as other studies?

- Can the dynamic calibration tools be adapted to work with this MLS system, and if so, can the dynamic calibration tools be expanded/improved?
2

Experimental System & Methods

2.1 Instruments, Hardware & Software

The instruments used was a NovAtel SPAN-IGM-S1 (INS), a NovAtel GPS-702-GG (antenna) and a Velodyne VLP-16 (laser scanner), seen in Figure 2.1. Specifications of the VLP-16 laser scanner can be found in Table 2.1. Additional specifications can be found in the VLP-16 Data Sheet [26]. The VLP-16 laser scanner will henceforth be referred to as ”lidar”.

Figure 2.1: The instruments used from left to right: SPAN IGM-S1, GPS-702-GG, VLP-16. Images courtesy of NovAtel and Velodyne LiDAR.

Surveying and post-processing was mainly done on a MacBook Air (Intel i7-5650U 2.2 GHz and 8 Gb of RAM) running Windows 10 64-bit. Surveying and post-processing was also successfully conducted using a Dell laptop running
2.2 Setup

Table 2.1: Specifications of the Velodyne VLP-16.

<table>
<thead>
<tr>
<th>Specification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laser wavelength:</td>
<td>903 nm</td>
</tr>
<tr>
<td>#Channels (laser/detector pairs):</td>
<td>16</td>
</tr>
<tr>
<td>Laser points/second:</td>
<td>(\sim 300,000)</td>
</tr>
<tr>
<td>Measurement Range:</td>
<td>Up to 100 m</td>
</tr>
<tr>
<td>Field of view (Vertical):</td>
<td>(\pm 15^\circ)</td>
</tr>
<tr>
<td>Field of view (Horizontal):</td>
<td>360(^\circ)</td>
</tr>
<tr>
<td>Accuracy:</td>
<td>(\pm 3) cm (Typical)</td>
</tr>
<tr>
<td>Angular Resolution (Vertical):</td>
<td>2.0(^\circ)</td>
</tr>
<tr>
<td>Angular Resolution (Horizontal/Azimuth):</td>
<td>0.1(^\circ) − 0.4(^\circ)</td>
</tr>
<tr>
<td>Rotation rate:</td>
<td>5 Hz − 20 Hz</td>
</tr>
<tr>
<td>Dimensions:</td>
<td>103 mm Diameter × 72 mm Height</td>
</tr>
<tr>
<td>Weight:</td>
<td>830 g</td>
</tr>
</tbody>
</table>

Windows 7 as well as an ASUS laptop running Windows 8. The NovAtel Windows USB drivers were used to transfer the INS data over the USB port.

2.2 Setup

The instruments are mounted on a backpack as seen in Figure 2.2 and an illustration of the mounting setup is seen in Figure 2.3. The lidar mounting translations and pitch angle were obtained by measurements in a photogrammetry generated point cloud. The backpack was filmed with a camera in a circular pattern and the point cloud was generated in the software Agisoft PhotoScan. Measurements in the point cloud were then performed in the software AutoDesk Recap and the lidar mounting translations and pitch angle were obtained. A detailed description of the transformation from the lidar coordinate frame to the world frame is given in Tulldahl & Larsson [1].

The lidar data was stored and sent at 10 Hz as User Datagram Protocol packets (UDP) and was logged with the software WireShark. The packets contain lidar point data — comprised of time-stamps, ranges, intensities, rotations and elevations. INS data was requested and logged at 125 Hz through a python script.
2. EXPERIMENTAL SYSTEM & METHODS

Figure 2.2: Photograph of the backpack-mounted instruments in field use.

Figure 2.3: Sketch of the mounting setup. All translations and rotations are relative the INS coordinate center. Note that the INS coordinate center is not located precisely in the middle of the physical INS enclosure.
The requested INS data logs were:

- PPSCONTROL enable positive,
- log VEHICLEBODYROTATIONB onchanged,
- log IMUTOANTOFFSETSB onchanged,
- log RAWIMUSXB onnew,
- log RAWEPHEMB onnew,
- log GLOEPHMERISB onnew,
- log TIME ontime 1,
- log TIMEB ontime 1,
- log BESTPOSB ontime 1,
- log RANGECMPB ontime 0.1.

Detailed descriptions of these can be found in the NovAtel manuals [27][29].

### 2.2.1 Time Synchronization

When disconnected from any external receiver the lidar has its own internal clock. To time synchronize the lidar with the INS the two sensors were electronically connected together, and the INS was configured to issue pulse-per-second (PPS) signals as well as once-per-second NMEA $GPRMC messages to the lidar.
2. EXPERIMENTAL SYSTEM & METHODS

2.3 INS Initialization & Kinematic Alignment

Before each survey the INS was initialized and aligned. This was done by first issuing the following commands to the INS:

- \texttt{SETIMUORIENTATION 6,}
- \texttt{VEHICLEBODYROTATION 0 0 0,}
- \texttt{APPLYVEHICLEBODYROTATION enable,}
- \texttt{SETIMUTOANTOFFSET 0.1 0 -0.1,}
- \texttt{ALIGNMENTMODE AUTOMATIC,}
- \texttt{SETALIGNMENTVEL 1.15.}

Refer to the NovAtel manuals [27–29] for a description on these commands. Following these commands kinematic alignment was performed by walking at a speed of at least 1.15 m/s, in a as constant heading as possible, until the INS status changed from "ALIGNING" to "ALIGNMENT COMPLETE". Following the acquisition of the "ALIGNMENT COMPLETE" status, walking in a figure-eight motion as well as random stops of a few seconds were performed. When the INS status changed from "ALIGNMENT COMPLETE" to "GOOD" the kinematic alignment was complete. During the entire kinematic alignment procedure attempts were made to keep the roll and pitch as close to zero as possible. The status of INS alignment was monitored through the software NovAtel Connect.

2.3.1 Survey Initialization & Methodology

The survey started and ended with at least 30 to 40 seconds of walking under clear sky with a more or less constant heading. Following this and prior to entering the forest, changes in heading was then performed (serpentine movements) in order to improve attitude convergence. When surveying inside the forest several stops were performed at predetermined points — marked by red paint. The stops were roughly 10 seconds in duration and the backpack was kept as static as possible during this time. These stops were performed for later utilization in post-processing the INS data.
2.4 Survey Area

The forest survey was conducted at 2017-03-30 in Stadslien, Umeå. The dominating tree species were Norway spruce but some Scots pine were also present. The main part of interest for the survey was an area roughly 40 meters in diameter. It is within this area that the results and evaluation will be presented, but lidar measurements were collected for a major part of the survey trajectory. The latitude and longitude for the center point of this area is at roughly 63°50′14.00″N and 20°18′38.11″E. On the inbound and return path to/from the sample area the walk pattern is somewhat arbitrary, but attempts were made to walk in the same path both ways. Around 1 to 2 decimeters of snow was present at the time of the survey.

2.5 Data Post-Processing

The INS data was post-processed in NovAtel Inertial Explorer to obtain a trajectory solution. The data was processed by tight coupling and utilized SWEPOS base station data from the station in Holmsund. The stops mentioned in Section 2.3.1 were performed in order to utilize the zero velocity update (ZUPT) feature in Inertial Explorer to obtain a better solution of the estimated trajectory. Guidelines for post-processing in Inertial Explorer can be found in Appendix A. The Inertial Explorer processed INS data and lidar data were then processed with the dynamic calibration tools. A user guide can be found in Appendix B.

Additional post-processing was conducted after this. The additional post-processing included further corrections of the trees’ xy-positions (algorithm courtesy of Holmgren, J.), as well as co-registration between the MLS, TLS and manual caliper data sets (software courtesy of Olofsson, K et al. described in [30]). The manual caliper data set was measured in an area roughly 24 meters in diameter and included 43 manually measured tree diameters at breast height (DBH), here defined at a height 1.3 meters above ground.
2. EXPERIMENTAL SYSTEM & METHODS

2.6 MLS Point Cloud Evaluation

The MLS point cloud was evaluated using the root mean square error (RMSE), bias, relative RMSE and relative bias, defined as

\[
RMSE = \sqrt{\frac{\sum_{i=1}^{n} (x_i - x_{ri})^2}{n}}, \tag{2.1}
\]

\[
Bias = \frac{1}{n} \sum_{i=1}^{n} (x_i - x_{ri}), \tag{2.2}
\]

\[
RMSE_{rel} = \frac{RMSE}{x_r} \times 100\%, \tag{2.3}
\]

\[
Bias_{rel} = \frac{Bias}{x_r} \times 100\%, \tag{2.4}
\]

where \(x_i\) is the \(i\)th estimation, \(x_{ri}\) is the \(i\)th reference value, \(x_r\) is the average reference value and \(n\) is the total number of estimations.

Further evaluation was also be done using the difference between estimation and reference.

2.7 The Dynamic Calibration Algorithm

Due to error accumulation, commonly referred to as drift, in the position estimation in forests, georeferencing a point cloud to this coordinate system will give rise to effects such as multiple grounds and multiple tree copies. More specifically, these effects arise when the same part of the point cloud is seen/visited at different times during measurements, since the positional error differs at different times. Thus, multiples of the same trees at different locations as well as multiple grounds will be apparent (depending on the magnitude of the drift) in the georeferenced point cloud. The dynamic calibration algorithm aims to correct as well as adjust the point cloud for these effects.

The algorithm is a small part of a series of tools, or graphical user interfaces (GUI:s), as seen in Appendix B. In this thesis these tools are referred to as the dynamic calibration tools. In summary, the algorithm makes use of previously measured data to correct the point cloud and trajectory in two steps. In the first
2.7 The Dynamic Calibration Algorithm

step, so-called trim-values \((dx, dy, dz, \text{ roll, pitch, yaw})\) are calculated in short time periods. The trim-values adjust (by translation and rotation) the point cloud in these short time periods. In this step the algorithm also calculates so-called features to be used in the second step. In the second step, the algorithm searches and matches previously detected features with newer features across the entire time sequence. The features of greatest importance are referred to as stem sections and flats. The stem section features are fitted circles of the tree stem and the flats are estimated planar regions of the ground. These are determined by using the fact that the lidar sends out several laser beams each sweep (16 in one sweep). The feature matching is done to adjust for multiple tree copies and multiple ground layers. For example, a tree is first scanned by the lidar at time \(t_1\) and then again at a later time \(t_2\). Thus, there are copies of the same tree at these two times, located at different positions as determined by the amount of drift. The stem sections of this tree at time \(t_1\) and \(t_2\) are then matched and the point cloud data corrected as to remove the effect of multiple tree copies. The stem section features are used to correct the point cloud in the horizontal plane \((xy)\) while the flat features are used to correct it in the vertical plane \((z)\). This is described in more detail in Section 2.7.1.

Note that, technically, the dynamic calibration algorithm is only the first step described above (calculation of trim-values). The second step (where the features are used) is not a part of the dynamic calibration algorithm itself, but actually a separate post-processing step using a different dynamic calibration tool. But, in this thesis both these steps have for simplicity’s sake been defined as the dynamic calibration algorithm.

2.7.1 Main Additions

The main additions to the dynamic calibration tools is, first, having the option of not requiring post-time synchronization between lidar and INS (or IMU) using pitch information. This option is available given that the instruments are time synchronized as in Section 2.2.1. Previously, the user was required to perform forward and backward leaning motions (i.e. forward and backward pitch) at the start of a data set. This motion is required if the INS and lidar are not
electronically time synchronized, but unnecessary given that they are. Thus, the
dynamic calibration tools were expanded to also work for this case.

Second, correction of multiple grounds, which was achieved by using the flat
features. This correction is performed after the $xy$-correction. In summary, it
works by first searching for flats in a one meter radius at time $t_n$ from the lidar,
referred to as $Flats_{current}$. If one or more flat feature is detected within this
radius and time instant, they are compared against all other flats detected at
earlier times $t_n - t$ (where $t < t_n$), referred to as $Flats_{reference}$. The compari-
son works by weighting $Flats_{reference}$ based on their age (older age is weighted
higher), RMSE (flat $xyz$-coordinate RMSE) and the horizontal distance from
$Flats_{current}$ (a smaller horizontal distance is weighted higher). The total weight
is assembled and a weighted average of the position is calculated. Finally, the
difference between the centroid $z$ for $Flats_{reference}$ and $Flats_{current}$ is taken to be
the translation in $z$. 

Survey Results

The solution from Inertial Explorer and an illustration of the walking pattern is shown in Figure 3.1. The main survey area of interest is seen in the lower left corner of Figure 3.1 (the area inside the circular pattern). The total survey time was just below 12 minutes.

3.1 Time-Synchronization

The estimated INS and lidar pitch angles during the entire survey are seen in Figure 3.2. The constant lidar pitch angle at the right side of Figure 3.2 is when lidar measurements were seized (INS measurements were continued as part of the INS post-procedure following measurements in a difficult GNSS environment, as mentioned in Section 2.3.1).
Figure 3.1: The solution from Inertial Explorer of the survey trajectory plotted in Google Earth (left) and an illustration of the walking pattern (right). The Google Earth figure was generated through the Export to Google Earth functionality in Inertial Explorer. The numbering in the illustration refers to the walking order.
3.2 Motion Trajectory

The trajectory along with the ground flat sections and stem sections can be seen in Figure 3.3. The data was limited to only include data inside the forest.

Prior to the dynamic calibration the absolute difference between the mean positions (defined as $|\overline{(X, Y, Z)}_{ZUPT1} - \overline{(X, Y, Z)}_{ZUPT2}|$ in units of meters) of the northern (N1 and N2) and southern ZUPT:s (S1 and S2) are (0.27, 1.98, 0.07) and (0.57, 0.73, 1.41) respectively. After the dynamic calibration the absolute differences are (0.23, 0.33, 0.62) between the northern (N1* and N2*) ZUPT:s and (0.49, 0.15, 0.74) between the southern (S1* and S2*) ZUPT:s.

Figure 3.2: Estimated pitch angles of the INS and lidar respectively. Zoom of the dashed rectangles is seen in the figures to the right.
3. SURVEY RESULTS

Figure 3.3: Part of the survey before ((a)) and after ((b)) the dynamic calibration. The trajectory is represented by the red line. The turquoise markers represents the ground (flat sections) and the brown markers represents trees (stem sections).

Figure 3.4: The southern ((a)) and northern ((b)) ZUPT:s (marked by the rectangles) before the dynamic calibration.
3.3 Tree Stem Sections & Ground Flat Sections

The stem sections before and after the dynamic calibration in the region marked by the blue rectangle in Figure 3.3 (b) can be seen in Figures 3.6 and 3.7 respectively. Before the dynamic calibration, multiple tree copies can be observed as revealed in Figure 3.6, and after the dynamic calibration this effect has been noticeably reduced, as seen in Figure 3.7.

Similarly, the ground flat sections (for the entire data set range as in Figure 3.3) before and after the dynamic calibration is seen in Figures 3.8 and 3.9 respectively. Figure 3.8 reveals a duplicate ground layer before the dynamic calibration, which, after the dynamic calibration has been adjusted for, as seen in Figure 3.9.
3. SURVEY RESULTS

Figure 3.6: The stem sections in a ROI before the dynamic calibration. Sub-Figure (a) is a top-side view and (b) is a side-view.

Figure 3.7: The stem sections in a ROI after the dynamic calibration. Sub-Figure (a) is a top-side view and (b) is a side-view.
3.3 Tree Stem Sections & Ground Flat Sections

**Figure 3.8:** The flat sections before the dynamic calibration. Sub-Figure (a) is a side-view and (b) is a top-side view. The red circles and arrows illustrate the corresponding regions between the two viewpoints.

**Figure 3.9:** The flat sections after the dynamic calibration. Sub-Figure (a) is a side-view and (b) is a top-side view. The red circles and arrows illustrate the corresponding regions between the two viewpoints.
3. SURVEY RESULTS

3.4 Survey Plot Co-registration

The DBH of the co-registered TLS and MLS trees is seen in Sub-Figure 3.10 (a) and the difference in diameter is seen in (b). Out of 214 TLS trees, 185 succeeded in being matched to MLS trees. The DBH estimation RMSE and bias between the TLS and MLS was 27.00 mm (relative 10.81%) and −9.33 mm (relative −3.73%) respectively. The bias indicates that $DBH_{MLS}$ was systematically lower than $DBH_{TLS}$. The number of tree pairs with $|DBH_{MLS} - DBH_{TLS}| < 50$ mm was 179 out of 185 total.

![Figure 3.10](image)

**Figure 3.10:** The DBH of the MLS, $DBH_{MLS}$, versus the DBH of the TLS, $DBH_{TLS}$ ((a)), as well as the difference in DBH, $DBH_{MLS} - DBH_{TLS}$ ((b)). The black circles represents the data points and the red line corresponds to the case where $DBH_{MLS} = DBH_{TLS}$ in (a) and the case $DBH_{MLS} - DBH_{TLS} = 0$ in (b).

Similarly, the DBH between the MLS and manual measurements using a caliper is seen in Figure 3.11. Out of the 43 manually measured tree diameters, 36 were matched to the MLS trees. The estimated DBH RMSE and bias were 16.95 mm (relative 6.65%) and −10.58 mm (relative −4.15%) respectively. Thus, the $DBH_{MLS}$ was systematically lower than $DBH_{Caliper}$. The number of tree pairs with $|DBH_{MLS} - DBH_{TLS}| < 20$ mm was 30 out of 36.
Figure 3.11: The DBH of the MLS, $DBH_{MLS}$, versus the manual caliper DBH, $DBH_{Caliper}$ ((a)), as well as the difference in DBH, $DBH_{MLS} - DBH_{Caliper}$ ((b)). The black circles represents the data points and the red line corresponds to the case where $DBH_{MLS} = DBH_{Caliper}$ in (a) and the case $DBH_{MLS} - DBH_{Caliper} = 0$ in (b).
3. SURVEY RESULTS

The difference in the radius evaluated at different heights along the tree stems between the co-registered MLS and TLS trees is seen in Figure 3.12. The radius difference increases as $z_{Stem}$ decreases.

![Figure 3.12](image)

**Figure 3.12:** Plot of the difference in radius between the MLS and TLS trees $|R_{TLS} - R_{MLS}|$ evaluated at different stem heights $z_{Stem}$. 
Discussion

4.1 Time-Synchronization

By looking at Figure 3.2 it is clear that the time-synchronization described in Section 2.2.1 worked as desired during the entire duration of the lidar measurements.

4.2 Motion Trajectory

It is difficult to give a definitive answer on the improvement of the trajectory estimate by analysis of the ZUPT:s before and after the dynamic calibration. Since it is impossible to return to the exact same positions previously visited there will always exist a difference when comparing previous to current positions.

The dynamic calibration improved the position estimate in all cases except in the z-direction for the northern ZUPT:s — where the uncorrected difference was 0.07 and the corrected was 0.62, implying a relative percentage change of 786%. The reason for this increase in z for the northern ZUPT:s after the dynamic calibration is unknown.
4.3 Tree Stem Sections & Ground Flat Sections

Analyzing Figures 3.6 and 3.7 reveals multiple tree copies as a result of drift. This is clear by looking in Figure 3.6 and noting that there are three copies of the same tree, seen at different locations in space at different times (as represented by their color). The first time a tree was seen is represented by the blue color, the second time by green, and the third by yellow. If there was zero drift then these would be positioned at exactly the same locations. The dynamic calibration solves this and the implemented $z$-correction seems to be working properly. This is verified by looking in Figure 3.7 (b), where it can be observed that the newer (green-yellow) stem sections have been translated down to the older (blue) stem sections (i.e. the copies have been translated).

Looking in Figure 3.8, a double ground layer is clearly visible as marked by the red circles. Noise is also visible well above ground level. After the dynamic calibration the double ground has been adjusted for as well as being less noisy, as verified in Figure 3.9. How well the MLS ground model compares with the TLS ground model is unknown and will need to be investigated in future studies.

4.4 Survey Plot Co-registration

Analysis of Figure 3.10 and 3.11 shows that the estimated DBH of the MLS compared to the TLS and caliper DBH is in the range of a few centimeters to millimeters. For both cases (MLS/TLS and MLS/Caliper) there is a systematic underestimation of the MLS DBH, since the bias is negative. For MLS/TLS the number of tree pairs where $DBH_{MLS} - DBH_{TLS} > 0$ (i.e. $DBH_{MLS} > DBH_{TLS}$) was 54. There is thus more than three times as many underestimated MLS DBH values. For MLS/Caliper there were 7 tree pairs where $DBH_{MLS} - DBH_{Caliper} > 0$, implying that around four times as many MLS trees had an underestimated DBH as compared to the caliper DBH. The reason for this underestimation is not fully understood. One reason could be due to the algorithm that estimates the stems (i.e. the algorithm that fits circles on the stems). The algorithm removes points on the stem that are outside a specified range offset. More specifically, if points on the stem has a root mean square (RMS) greater or equal to 15 mm,
4.4 Survey Plot Co-registration

the outer points will be removed by trimming each side of the stem (this trim is currently set to remove 13%).

The obtained DBH RMSE and bias show that there is much promise to use this system for forest surveying. For comparison, in Olofsson & Holmgren [24], the obtained DBH RMSE and bias were 1.175 cm and +0.688 cm respectively for the pine trees, and 1.026 cm and +0.565 cm respectively for the spruce trees. The reference data were manual caliper measurements. These results indicated that the TLS and caliper measurements were in very good agreement with each other. In Liang et al. [14], where they utilized a robot-based MLS system, the relative DBH RMSE and bias were 8.17% and −1.82% respectively. In Liang et al. [25], where instead a backpack-based MLS system was used, the DBH RMSE and bias were 5.06 cm (relative 14.63%) and +1.11 cm (relative +3.20%) respectively. In both studies by Liang et al. they used TLS measurements as reference data. We also compare this to the co-registration between caliper and TLS, as see in Figure 4.1. The DBH RMSE and bias were 10.69 mm (relative 4.56%) and −0.97 mm (relative −0.41%). In this case the number of trees that succeeded in being co-registered is 42 out of 43 caliper trees. It can be noted that the DBH RMSE and bias between TLS and caliper obtained in this thesis, are in good agreement with the DBH RMSE and bias obtained by Olofsson & Holmgren [24].
Figure 4.1: The DBH of the TLS, $DBH_{TLS}$, versus the manual caliper DBH, $DBH_{Caliper}$ ((a)), as well as the difference in DBH, $DBH_{TLS} - DBH_{Caliper}$ ((b)). The black circles represents the data points and the red line corresponds to the case where $DBH_{TLS} = DBH_{Caliper}$ in (a) and the case $DBH_{TLS} - DBH_{Caliper} = 0$ in (b).
Analyzing Figure 3.12 indicate that there seems to be a dependency on the height where the difference in radius is calculated. Figure 3.12 show that there is a larger number of values where $|R_{TLS} - R_{MLS}| > 20$ mm at lower stem heights $z_{Stem}$. Reasons for why the radius difference increases lower down on the stem could be a result of both snow being present during the MLS survey but also due to the shape of the tree stem base. Since the base of the tree stem can differ significantly in shape compared to the rest of the stem (due to roots) it will be difficult to define (for any method or algorithm) a radius near the stem base. Also, in obtaining Figure 3.12 it is assumed that the base of the MLS and TLS trees is the "actual" base of the tree. If snow covered the base during the MLS survey then the post-processed tree stems could be missing the actual base. Thus there will be a mismatch between the base of the TLS trees and MLS trees, giving an increased error at the base.

### 4.5 Dynamic Calibration

The dynamic calibration has worked well in post-processing the INS and lidar data. Given the results it shows much promise for future usage and with a lot of room for further additions and improvements.

One addition is forward and backward optimization. The dynamic calibration is dependent on being able to detect previous data (e.g. stem sections). If the survey walking pattern is such that it cannot detect previous data for large periods of time (e.g. large loops), the dynamic calibration optimization will fail. A forward/backward optimization would be a solution to this problem.

It was observed that the dynamic calibration was sensitive to the environment in which it was allowed to run/optimize in (see Appendix B.3). When the dynamic calibration was allowed to run in parts of the data where there were few trees surrounding the user, the results exhibited unstable behaviors, which included parts of the point cloud shifting noticeably in the $z$-direction (before the vertical correction) in these data parts. For this reason the dynamic calibration was only allowed to optimize well inside the forest — with trees surrounding the user in all directions. Another addition to the algorithm would thus be to develop it for more open environments as well.
4. DISCUSSION

But even so, the dynamic calibration result differed depending on the optimization range inside the forest. It was observed that different parts of the data failed the dynamic calibration optimization (seen in Figure 3.3 (b) as the missing parts of the trajectory) between runs of different optimization ranges. The reason for this has not been researched thoroughly and will need to be investigated in the future.

Finally, the dynamic calibration is dependent on forest properties such as tree diameter and density. Forests where the tree diameter and/or tree density is too small/sparse (on average), may result in a poor dynamic calibration result. How dependent it is on these forest properties is not known. Some rough experiments during the start of the project indicated that the dynamic calibration might struggle with forests where the trees has an average diameter less than or equal to 15 cm.

4.6 Inertial Explorer

Usage of Inertial Explorer has both its positives and negatives. Inertial Explorer is overall a very robust and user friendly post-processing software where getting comfortable with the software does not take long. Inertial Explorer makes use of many advanced algorithms that would take considerable time and effort to implement oneself to the same or better standard as the software. One downfall with using Inertial Explorer is that it is a black box. Meaning there is no possibility for the user to see or modify the source code after one’s desire. Also, for processing to be possible with Inertial Explorer the user is required to always calibrate and align the INS before data collection. This can be quite tedious if the INS is only capable of kinematic alignment, as is the case with the INS used in this thesis. But, changing to an INS capable of static calibration/alignment is simply a ”quality of life” upgrade that I consider to be only worth considering in the future – closer to a finished product.

Inertial Explorer gives good solution estimates of motion trajectories – especially in clean GNSS environments where the solution is very accurately estimated – but it is sensitive in more difficult GNSS environments. The input parameters
for obtaining a converging solution (where diverging implies either that the trajectory blows up in one or several regions, or that Inertial Explorer simply will not compute a solution) can change between different surveys. Solutions may also be sensitive to changes in these parameters. Therefore, it is difficult to give a general user guide for post-processing data sets and obtaining "good" solutions in difficult GNSS environments. Some guidelines is given in Appendix A.

4.7 Snow, Walking Pattern & Pitch Angle

Whether or not the snow affected the results in any way is not known and requires future investigations to determine this. It is not likely that the snow had any major impact on the results, but could very well have had minor impacts, such as the discussion in Section 4.4 regarding the reason for why the relative radius increases lower down on the stem. It could also be the case that the snow had positive impacts. For example, since snow covered the ground vegetation it might be easier to define a ground level and thus an increased number of flat features from the dynamic calibration.

Not presented in this report, due to time constraints, was another survey conducted right after the survey included in the report. The second survey was similar to the first but differed in the walking pattern at the area of interest. Future studies of walking patterns is of great interest as it is a subject of importance to optimize data collection, reduce survey time, cost and drift. It may be the case that walking patterns will differ depending on the sample area — where factors such as sample area size and tree density might play an important role — and is something that should be taken into consideration before conducting a survey.

Different pitch angles for the lidar were not investigated in this project. This is something that is of interest to investigate in the future, to see how it affects the results. The pitch angle used here was chosen mostly based on qualitative guesses. First, having the lidar angled too little would imply a risk of obtaining insufficient data of the tree trunks and the ground close to the lidar; while having a too large pitch angle would imply a risk of obtaining insufficient data higher up on the tree stem as well as seeing too few trees (trees located farther away).
Conclusions

In this thesis, a backpack-based MLS system was put together and tested in a forest environment.

There is no well established limit in the forest industry of what is considered as "good" DBH RMSE and bias values. Since no measurement instrument or method is perfect (certainly not TLS and calipers), it is difficult to say one-hundred percent of the time, which instrument/method has the most errors. For example, if only one caliper measurement is taken for each tree (i.e. measured only in one direction), it might give a much worse DBH estimate as compared to the MLS. Ideally, we would like that the comparison between all methods (MLS/TLS, MLS/caliper, TLS/caliper) to converge to each other. That is, we would like that the DBH RMSE and bias to be as small and with as little variation as possible between the methods. Of course, this would ideally not be limited to only DBH RMSE and bias. Even so, we believe the results serves to show that backpack-based MLS systems has the potential of accurate lidar mapping in forests. Developing this system further is of great interest due to its' advantages over TLS and manual caliper measuring.

Future development and studies would be to conduct further testing of the system in forests with different and varying properties, such as tree and vegetation density, tree species and tree diameters; expand and improve the dynamic calibration based on these tests; and conduct surveys with varying walking patterns.
Perhaps in the future this technology can be used to create autonomous harvesters. Or, simply as a helping tool for harvester drivers.

Developing this system further is not only interesting in forest applications, since the technology can be applied in other areas. Especially in creating self-driving cars, where of course, the demands are high for very reliable and accurate positioning and robot vision (i.e. the ability for the car to "see" the surrounding world).
References


Appendix A

User Guidelines for Processing in Inertial Explorer

This chapter covers guidelines that may or may not work for obtaining better solutions in Inertial Explorer for difficult GNSS environments. The required input parameter values and settings for obtaining the "best" possible solution is difficult to define. The best advice giveable is experimentation with the processing settings and using your own judgment of what seems to be the "best" solution obtainable. It should also be mentioned that excellent sources for information regarding Inertial Explorer are the Inertial Explorer 8.70 User Manual, the GrafNav/GrafNet 8.70 User Guide (see the NovAtel manuals in [31, 32]) as well as contacting NovAtel Support.

A.1 Processing Settings

It is highly recommended to process by tight coupling (as opposed to the alternative of loosely coupled processing), hence the following section will only cover settings for tightly coupled processing.

The main processing window is shown in Figure A.1. It is recommended to use "Multi-pass" for the processing direction. Make sure that the profile is the "SPAN Pedestrian" under "Processing Settings". This setting should be automatically detected but it is recommended to verify this for unprocessed projects. If the pedestrian profile is loaded it is shown in the processing window by the small
A.1 Processing Settings

figure in the lower right of the window, as seen in Figure A.1. This can also be verified by looking in the .gpb file where the "Environment" should be "GNSS Pedestrian" (this file is generated after loading the raw .gps when creating a project or by using NovAtel Raw Data Converter). If the correct calibration commands have been set before surveying then the correct "IMU Installation" settings should automatically be loaded through the checkbox "Read rotations and lever arms from IMR file". If the INS was incorrectly configured at calibration the correct rotations and translations have to be input manually. This is done by unchecking "Read rotations and lever arms from IMR file" and configuring the fields under "Lever Arm Offset" and "Body to IMU Rotation".

![Figure A.1: The processing window for tightly coupled processing.](image)

**Figure A.1:** The processing window for tightly coupled processing.

A.1.1 Advanced IMU Settings

Sometimes, for processing to work it is necessary to raise the "Heading SD Tolerance" (default 45 degrees) found in "Advanced IMU" → "Alignment" → "Options", as seen in Figure A.2 (an error message will be displayed if the processing
A. USER GUIDELINES FOR PROCESSING IN INERTIAL EXPLORER

Figure A.2: The alignment options window.
A.1 Processing Settings

fails due to this issue). This parameter determines the tolerance for which the heading standard deviation must fall below in order for the processing of the alignment routine to move into navigation mode. Raising this parameter further when processing works will not affect the solution.

If the survey contains ZUPT:s, as mentioned in Sections 2.3.1 and 2.5, the solution accuracy can improve significantly by utilizing the software’s ZUPT detection feature. The settings that controls the ability to detect ZUPT:s is found under ”Advanced IMU” → ”Updates” → ”Automated ZUPT Detection Tolerances”, shown in Figure A.3. The values for these tolerances needs to be experimented with for each survey. The ”Raw Measurement” tolerance determines the threshold for raw gyro measurements, ”Velocity” determines the GPS velocity threshold, and ”Period” is the time span for which measurements are averaged. It is usually sufficient to have a period $\geq 5$ seconds.

![Figure A.3: The window for changing the ZUPT detection tolerances.](image)
A. USER GUIDELINES FOR PROCESSING IN INERTIAL EXPLORER

Another setting that may greatly improve the solution is the "VELCONSTRAINT" command that can be added by the user under "Advanced IMU" → "User Cmds", seen in Figure A.4. This command controls $x$, $y$ and $z$ velocity constraints. The input format is given by "VELCONSTRAINT = OFF/ON sdevX OFF/ON sdevY OFF/ON sdevZ ON/OFF updateInterval". Recommended input values are: ON 0.3-0.4 OFF 1.0 OFF 1.0 ON 1.0, but these can of course be experimented with.

Figure A.4: The window for adding user commands for IMU processing.

A.1.2 Advanced GNSS Settings

It was found that higher GNSS update rates resulted in a poorer solution estimate. This was tested by collecting GNSS data at a higher update rate (10 Hz) and then in post-processing subsequently decrease the update rate. The accuracy between
different update rates was then determined mainly by comparing the trajectory solutions visually against each other (the results differed significantly between update rates, making a visual comparison valid). The tested update rates were 10 Hz, 5 Hz, 2 Hz and 1 Hz — where 1 Hz gave the best results. The setting for changing the GNSS processing update rate is found under ”Advanced GNSS” → ”General” → ”Processing Interval and Time Range”. Note also that, this was only tested for two data sets conducted at Stadsliden. It is very likely that higher GNSS rates might not give poorer solution estimates in other cases. For example, the GNSS update rate might depend on properties such as the canopy coverage, where high coverage (as was the case at Stadsliden) may require a lower GNSS update rate, whereas for low coverage it might be better to have a higher update rate.

![Figure A.5: The window for changing the GNSS update interval.](image)

Lastly, a setting that greatly affect solution accuracy is ”Ionospheric processing” found under ”Advanced GNSS” → ”Measurement” → ”Ionospheric processing”, shown in Figure [A.6](image). This setting is by default turned on but it is advisable
A. USER GUIDELINES FOR PROCESSING IN INERTIAL EXPLORER

to disable this. In most cases, having this setting turned on is valid, but for difficult GNSS environments there is difficulty in tracking the L2 frequency signal. That is, there is cycle slip in the L2 data (meaning there are many more L1 measurements than L2), and ionospheric processing requires that both the L1 and L2 data to be cycle slip free.

![Figure A.6: The window for controlling ionospheric processing.](image)

To process the data press "Process" as seen in Figure [A.1]
Appendix B

User Guide for the Dynamic Calibration Tools

The dynamic calibration tools are written in MATLAB and contains four steps of data processing through four graphical user interfaces (GUI). These GUI:s will be described in detail in the Sections below. Make sure that the main directory for the dynamic calibration MATLAB files is added to path within MATLAB before continuing. The GUI:s may be modified by using the ”guide” command in MATLAB.

B.1 K12 Tool

The first post-processing step is to unpack the raw lidar data using the K12 tool GUI (loaded by the MATLAB function ”K12_TOOL.m”), as seen in Figure B.1. However, before unpacking the lidar data two settings need to be checked. The first setting is the ”Crop to forward angles +/−” input. This setting will crop the lidar data to only include data that is located ±X degrees (where $0^\circ \leq X \leq 180^\circ$) from the y-axis of the lidar (see Figure 2.3). Note that, as of now this setting demands that the lidar is mounted with the y-axis pointing either forward or backward relative the walking direction – otherwise the lidar data will be cropped in unwanted ways. The second setting is ”Select Velodyne Model & FW”, which is simply a checkbox to select the Velodyne lidar model used when collecting the data. **Note** that it is recommended to always press the
B. USER GUIDE FOR THE DYNAMIC CALIBRATION TOOLS

desired checkbox, even if it is already checked when loading the K12 tool. If it is not checked the GUI checkbox may display the correct model but actually uses the other.

After checking these two settings the raw lidar data may be unpacked using the "Read K12 (Wireshark capture)" button. The required format of the lidar input file is a K12-file, which can be saved using WireShark. Note that the K12-file is required to have the same name as well as be located in the same directory as the WireShark capture file (.pcapng-file). This will read the data packets, convert the data bytes to physical units (e.g. from bytes to meters) and then divide the converted data into chunks — saved as MATLAB .mat files. When the unpacking has finished the K12 tool may be closed. A new folder with the saved chunks has also been created and will be located in the same directory as the K12 and WireShark capture files (folder name with the format "<K12 file name>_UNPAC_<Date of unpacking>").

![Figure B.1: The K12 tool GUI. The important functions are marked by green.](image)
B.2 3DUAV Tool

The second post-processing step is loading the INS data (already post-processed with IE) and combine this with the unpacked lidar data. This is done by using the 3DUAV tool GUI (loaded by the MATLAB function "TOOL_3DUAV.m"), seen in Figure B.2. Before anything else the time zone difference between UTC and the local PC clock need to be input under "UTC Time Zone Diff.". After this has been done the INS data can be loaded under "Poselog NovAtel". The loaded INS data can be plotted with "Testplot Upsampled".

The second file that needs to be loaded is the system mount which defines the translations and rotations needed to transform the lidar coordinate frame to the INS coordinate frame. The format of this file is a MATLAB structure saved as a .mat file. An example of this file with the required format is seen in Figure B.3.

Once this has loaded the next step is to resample the lidar using "Resample Tlog & Velo". This option is only available given that the lidar and INS is synchronized as described in Section 2.2.1. This will upsample the lidar data to work with the sample rate of the INS. A plot of the pitch as a function of time.
for both instruments will appear once the resampling has finished — indicating if the instruments are time synchronized or not.

The last step is to use "Batch Process" which will iterate over every chunk and transform both instruments coordinates frames to a world frame (defined as $+y$ North, $+x$ East). The button "Test Process Selected" can be used to test the transformation to world coordinates for one or several chunks. Note that this button functions in the same way as batch process, but it is not recommended to use this button for processing the entire trajectory. The folder directory requested after pressing batch process can either be the directory where the folder generated from the K12 tool is located (that is, the folder one level above the K12 generated folder), or the K12 generated folder itself.

Once the processing has finished a new folder with the processed data will be located in the K12 folder (format "ProcTlog<Date of batch processing>"). A telemetry log index file (format "ProcTlogIndex<Date of batch processing>.mat") has also been created and is located in the same folder as the WireShark files (the K12 file and WireShark capture file). The 3DUAV tool may be closed once the batch process has finished.

### B.3 3DUAV Analyzer Tool

The third post-processing step is the dynamic calibration processing step using the 3DUAV analyzer tool GUI (MATLAB function "ANALYZER_3DUAV.m").
seen in Figure B.4. Start by loading the processed telemetry log index file using ”Load Tlog Processed” and set ”Choose Flightlines ID” to 1.

The button ”Mark ROI” can be used to mark a (rectangular) region of interest (ROI) in the 2D figure of the trajectory and point cloud. The point cloud in the marked ROI can then be visualized using any of the ”PlotROI 3D Alt & <Property>” buttons (where <Property> is ”LidInt”, ”LidRange”, ”LidElev” or ”LidTime”).

To start the dynamic calibration processing first choose ”DynCal MultiFeature FOREST” and then press ”Process”. This process will create new files in the ProcTlog-folder. The created file ”CumulRefs.mat” will contain information regarding features such as tree stem sections (estimated circles of the tree stems) and flat sections (estimated flat ground sections). Descriptions and definitions of the columns is found in ”MultiFeatureDescriptor_ver160912.xlsx” located in the directory of the dynamic calibration MATLAB files. When the processing has finished the 3DUAV analyzer may be closed.

![Figure B.4: The 3DUAV analyzer tool GUI.](image)
B. USER GUIDE FOR THE DYNAMIC CALIBRATION TOOLS

B.4 Cumulative Tree Analysis Tool

In the fourth and final step in the dynamic calibration processing chain the features (tree stem sections and flat sections) obtained from the previous step is post-processed further. The post-processing of the features involves removing the effect of duplicate trees and multiple ground due to error accumulation. This is performed using the cumulative tree analysis tool GUI (MATLAB function “CUMUL_TREE_ANALYSIS.m”), seen in Figure B.5.

First, the file ”CumulRefs.mat” has to be loaded using ”Load CUMUL_REFS (from DynCal MultiFeat Forest)”. The ”Limit data to DynCal Lock RMSE <” value controls the data allowed to be used. Setting this value above 1 m will allow the data that failed to be dynamically calibrated with the 3DUAV analyzer tool to be used. The checkbox ”Use DynCal pos.” determines whether to use the dynamically calibrated trajectory or not. Any of these changes are applied by pressing ”Apply”. The buttons ”Mark ROI” and ”ROI plot StemPts” can be used to mark a ROI and plot the corresponding tree stem points in that region.

The post-processing is started by pressing ”PostCorrect”. When it has finished a file ”CumulRefsCORR.mat” with the corrected tree stems and flat sections is located in the same folder as ”CumulRefs.mat”. ”CumulRefsCORR.mat” can then be loaded with this tool and the corrected trees can be analyzed.
Figure B.5: The cumulative tree analysis tool GUI.