

Graph SLAM correction for single scanner MLS forest data under boreal forest canopy

Antero Kukko ^{a,b,*}, Risto Kaijaluoto ^a, Harri Kaartinen ^a, Ville V. Lehtola ^a, Anttoni Jaakkola ^a, Juha Hyypää ^a

^a Center of Excellence in Laser Scanning Research, Finnish Geospatial Research Institute FGI, Geodeetinrinne 2, FI-02430 Masala, Finland

^b Research Institute of Measuring and Modeling for the Built Environment, Aalto University, Otakaari 4, Espoo, P.O. Box 14100, 00076 Aalto, Finland

ARTICLE INFO

Article history:

Received 6 April 2017

Received in revised form 14 July 2017

Accepted 11 September 2017

Available online 19 September 2017

Keywords:

Mobile laser scanning

GNSS-IMU

Trajectory

Mapping

Precision forestry

Graph optimization

ABSTRACT

Mobile laser scanning (MLS) provides kinematic means to collect three dimensional data from surroundings for various mapping and environmental analysis purposes. Vehicle based MLS has been used for road and urban asset surveys for about a decade. The equipment to derive the trajectory information for the point cloud generation from the laser data is almost without exception based on GNSS-IMU (Global Navigation Satellite System – Inertial Measurement Unit) technique. That is because of the GNSS ability to maintain global accuracy, and IMU to produce the attitude information needed to orientate the laser scanning and imaging sensor data. However, there are known challenges in maintaining accurate positioning when GNSS signal is weak or even absent over long periods of time. The duration of the signal loss affects the severity of degradation of the positioning solution depending on the quality/performance level of the IMU in use.

The situation could be improved to a certain extent with higher performance IMUs, but increasing system expenses make such approach unsustainable in general. Another way to tackle the problem is to attach additional sensors to the system to overcome the degrading position accuracy: such that observe features from the environment to solve for short term system movements accurately enough to prevent the IMU solution to drift. This results in more complex system integration with need for more calibration and synchronization of multiple sensors into an operational approach.

In this paper we study operation of an ATV (All -terrain vehicle) mounted, GNSS-IMU based single scanner MLS system in boreal forest conditions. The data generated by RoamerR2 system is targeted for generating 3D terrain and tree maps for optimizing harvester operations and forest inventory purposes at individual tree level. We investigate a process-flow and propose a graph optimization based method which uses data from a single scanner MLS for correcting the post-processed GNSS-IMU trajectory for positional drift under mature boreal forest canopy conditions. The result shows that we can improve the internal conformity of the data significantly from 0.7 m to 1 cm based on tree stem feature location data. When the optimization result is compared to reference at plot level we reach down to 6 cm mean error in absolute tree stem locations. The approach can be generalized to any MLS point cloud data, and provides as such a remarkable contribution to harness MLS for practical forestry and high precision terrain and structural modeling in GNSS obstructed environments.

© 2017 The Author(s). Published by Elsevier B.V. on behalf of International Society for Photogrammetry and Remote Sensing, Inc. (ISPRS). This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

1. Introduction

The greatest advantage of MLS as compared to TLS (Terrestrial Laser Scanning) is the immensely faster data collection on complex environments, for example road asset mapping and forestry (Cabo

et al., 2016; Liang et al., 2014). This is guaranteed by the kinematic approach allowing free and rapid selection of viewpoints, or trajectory of viewing/measurement to be more accurate expression, for the best visibility towards the targets of interest. The greatest challenge for MLS is that the positional accuracy achievable even with a state-of-the-art integration of Global Navigation Satellite System (GNSS) receiver and Inertial Measurement Unit (IMU) does not necessarily suffice. Even with an expensive high quality IMU, if the satellite visibility is obstructed, the localization error can quickly grow to several tens of centimeters and even to meters,

* Corresponding author at: Center of Excellence in Laser Scanning Research, Finnish Geospatial Research Institute FGI, Geodeetinrinne 2, FI-02430 Masala, Finland.

E-mail address: Antero.Kukko@nls.fi (A. Kukko).

which results to multiple copies of objects appearing in the resulting point cloud for each time they are captured with the sensor. In other words, the trajectory errors propagate directly to the produced point cloud data deteriorating its quality. This problem is present in a milder form even with perfect satellite visibility as positioning errors of a couple of centimeters persist. If no further corrections, or other means of overcoming the problem, are applied to the data, object ambiguity may prevent modeling of object dimensions accurately and reliably with automated processing.

Forestry is one of the fundamental areas for the MLS research at the FGI. In this study we aim at having a practical method for operational 3D forestry capable enough for accurate tree data extraction for individual tree modeling and forest inventory purposes. Also, the method to be universal enough for possible adaptation and use in diverse MLS applications in urban and industrial contexts is of great interest. The purpose of this study is to investigate whether the data provided by a single scanner MLS system suffices for feature detection and matching, and whether we can discover such constraints that the 6D pose graph optimization is suitable to be used to correct the MLS data for GNSS-IMU trajectory drift. We pursue to improve the internal and absolute accuracy of the point clouds produced with MLS closer to the level attainable with TLS (cf. Olofsson et al., 2014; Liang et al., 2016) so that the most important forest (tree map) and tree parameters (e.g. DBH, height, stem curve) can reliably be extracted with improved spatial output efficiency. Though tested only in forestry data, we foresee possible applications for the method in improving the data integrity in urban and road environment as well, as such data often suffer from similar GNSS signal loss scenarios. Urban and industrial modeling for buildings, structures and road assets, even road surface data could greatly benefit from an automated data improvement method.

2. Trajectory optimization problem

To correct the trajectory for the geometric inconsistencies induced by errors in the GNSS-IMU observations we turn to robotics, where Simultaneous Localization and Mapping (SLAM) algorithms are widely used to provide localization information in various environments without reference coordinates (Frese and Hirzinger, 2001; Thrun, 2002; Durrant-Whyte and Bailey, 2006; Bailey and Durrant-Whyte, 2006). The procedure is roughly divided into two steps. First, an initial positioning with uncertainty is obtained. For that purpose Liu et al. (2010) and Elseberg et al. (2013) use vertically aligned 2D laser scanners, whereas Alismail et al. (2014) and Zhang and Singh (2014) employ external angular measures with a 2D laser scanner to perform the localization. Lehtola et al. (2016) use data from a single rotating scanner, and in contrast Bosse et al. (2012) use an IMU. Second, the positioning uncertainty is corrected by using data overlap from the sensors observing the environment, e.g. cameras or laser scanners (Liu et al., 2010; Elseberg et al., 2013; Alismail et al., 2014; Zhang and Singh, 2014; Lehtola et al., 2016; Bosse et al., 2012). Our previous works experimenting with data from an ATV platform used an additional scanner operated for horizontal scanning. Those works investigated use of 2D SLAM (Tang et al., 2015), and proposed IMLE (Improved Maximum Likelihood Estimation) with experiments fusing with GNSS-IMU data (Qian et al., 2017), and scan matching using a priori tree map (Chen et al., 2016) for planimetric trajectory improvement in boreal forest. In the approach introduced and reported in this paper, the initial trajectory is obtained solely from a GNSS-IMU system, and we investigate whether a merely GNSS-IMU positioned MLS system provides suitable data overlap and robustness for successful trajectory optimization without a need for additional sensors to the MLS system. This would keep future systems aimed at forest data collections simpler to use and less costly to acquire and adopt into practice.

In the context of forestry the representation of interest is a tree map (i.e. a 3D map with spatial location of detected trees), which consists of sparse features ideal to be dealt with the graph-based SLAM. The nodes of a graph represent the system poses and the detected features and edges connecting the nodes represent constraints measured by odometry or laser scanner (Bailey and Durrant-Whyte, 2006). Computationally, the graph SLAM (Thrun and Montemerlo, 2006; Carbone et al., 2014) uses sparse matrices to represent a graph of observation interdependencies, i.e., as an extended incidence matrix. An extension of the graph-based SLAM to 3D scans and poses with six degree of freedom was introduced by Borrmann et al. (2008) and Nüchter et al. (2010). However, this was implemented using the Iterative Closest Point-based (ICP) technique (Besl and McKay, 1992), which may not be ideal way to optimize a tree map representation.

Another reason for using the graph-based SLAM (Kümmerle et al., 2011) is that pose graph optimization enables combining of various constraints to the trajectory, e.g. tree models. The errors in the GNSS-IMU based initial trajectory drift slowly enough (see, e.g. Kaartinen et al., 2015) so that it can be used for georeferencing the scanner measurements. Then, the measured trees give us easily discernable features, which can be combined with the 6D pose graph optimization algorithm embedded with a suitable model to correct errors in the initial trajectory solution.

3. Test site, MLS system and data preparations

3.1. Evo research forests

The study area is located in Evo, southern Finland (Lat 61° 11' 48" Lon 25° 6' 27"), and representing Boreal Forest Zone forests dominated by Scots pine (*Pinus sylvestris*), Norway spruce (*Picea abies*) and birch (*Betula L. sp.*). The area in Evo is mainly managed forests, but includes recreational and protected areas as well. The study area includes about 120 square sample plots of 32 m in size scattered over 25 km² of area. For this study we selected data from three test areas comprising larger plot entities of 4 (2 by 2) standard plots adjacent to each other (see Fig. 2). The forest in area A (plots 1001–1004) consists of 342 trees, area B (1048–1051) 319 and area C (1108–1111) 362 trees. All of the areas have terrain suitable for an all-terrain vehicle (ATV) movement.

3.2. RoamerR2 mobile laser scanning system

RoamerR2 is a single scanner mobile laser scanning (MLS) system developed at Finnish Geospatial Research Institute FGI for collecting 3D point cloud data for object reconstruction and environmental modeling. In this study we operated it on top of an ATV for mobility on the forest floor, and to collect tree map and individual tree level inventory data from the ground. The MLS concept is based on the idea of detailed, but fast data collection and flexible mobility around objects of interest to get full coverage stem and canopy data for reliable inventory and complete modeling in structurally complex environments. GNSS-IMU positioning also enables direct georeferencing of the obtained data without the necessity of external targets.

RoamerR2 MLS system consists of a tactical grade GNSS-IMU system providing 200 Hz system position and orientation data and a single phase shift ranging laser scanner collecting 3D point cloud data. For forestry application the scanner is mounted for vertical cross-track profiling (Helical scan mode), and Fig. 1 shows a typical ATV installation for the RoamerR2 system. The GNSS antenna (NovAtel Pinwheel GPS-702-GG) is seen on the top with the scanner (FARO Focus3D 120S) in upright vertical position for canopy and stem mapping. The tactical grade IMU (NovAtel SPAN



Fig. 1. RoamerR2 mounted on the rear rack of the ATV. The scanner provides cross-track data collection up to 100 m range and the GNSS-IMU system observes the system trajectory and orientation for kinematic mapping of the surroundings into a 3D point cloud.

UIMU-LCI) is seen below the scanner and is connected to the GNSS receiver (NovAtel Flexpak6) observing Navstar GPS and GLONASS constellation satellites.

The sensor head is elevated to about 2 m above the ground using a GlobalTruss segment mounted on the rear rack of the ATV. The scanner has a 50 degree blind angle beneath (See Fig. 4). The collected laser measurements are stored on 32 GB SDHC (10) card on the scanner and positioning data is stored using a tablet computer (Panasonic Toughpad FZ-G1, Windows 8, 8 GB RAM, 1 TB SSD). The laser scanner sends TTL level timing signal to the SPAN system event logger for mutual synchronization (3.3 V TTL pulse for start of each scan). The scanner measures precise ranges up to 100 m by measuring the phase difference of the emitted amplitude modulated beam and the scattered returning signal. Such ranging principle has its shortcomings in vegetation and as a result the point observations from the tree stems through dense canopy is somewhat reduced. Also partial hits on thin branches and no-return sky generate measurement noise that needs to be removed prior to further data processing.

3.3. Point cloud data acquisition

The experimental data comprises three boreal forest test areas (A, B, and C) that were captured with the RoamerR2 system in October 2014 as a part of an extensive data collection campaign to cover all the forest study plots in the Evo research forests. The MLS trajectory snakes through the plots to cover all the trees from multiple directions and to collect complete terrain data (as seen in Fig. 7). Table 1 gives some plot and data collection characteristics showing the average data collection speed to be about 1 m/s. The scanner scan frequency was set to 95 Hz and point measurement speed to

488 kHz, while 5 Hz satellite and 200 Hz IMU observations were recorded with the positioning system. Thus the profile spacing in the point cloud data was ca. 1 cm and the angular resolution 1.2 mrad (0.07°). The scan data was split into file blocks of 3000 profiles in each at the data collection to permit batch processing for the data preparation phase. Fig. 2 represents an illustration of the point cloud data for dataset B colored by the point elevation. The yellow circles in the figure are 50 m in diameter and are used as data buffer around the actual 32 m by 32 m field plots shown in white squares.

4. A method for trajectory optimization

4.1. Outline of the method

The proposed processing pipeline is shown in full in Fig. 3. The steps are from data collection, georeferencing, ground classification and other preparations for the raw point cloud data to trunk detection, trunk feature association, generating the graph representation of the features and the trajectory, optimization and finally georeferencing of the point cloud using the optimized trajectory. The following sections describe in detail the steps taken to reach the final optimization result.

4.2. Pre-processing of the MLS data

The initial trajectories (position and attitude) of the MLS system at each site were calculated from the GNSS-IMU data with Inertial Explorer (NovAtel Inc., Canada) post-processing software. Base station data for differential correction of the GNSS-IMU data was downloaded from virtual data service (Trimnet, Geotrim, Finland). The recorded laser measurements were then georeferenced based on the time synchronization and trajectory data using proprietary Matlab tool.

Next the georeferenced point clouds were processed with TerraScan (TerraSolid, Finland) to remove measurement noise and to extract the ground points. Points with intensity value less than 1000 (full range 11 bits, 0–2047) were removed from the further processing as experience has shown that these data points are mostly noise. Points from dark objects are also typically worse in ranging quality compared to higher reflectance surfaces. Next, isolated points, a result of failed ranging, were removed using two-step filtering requiring each point to have at least one neighbor closer than 10 cm to remove single noise points, and after that clustered noise patterns were removed by demanding at least 30 points to be found within a 1 m point neighborhood.

Ground classification was based on triangulation method implemented into TerraScan (Axelsson, 2000) with parameters set to 65° maximum terrain slope, 20° iteration angle, and 20 cm iteration distance. The seed points for the ground classification were the lowest points within 5 m radius from the original noise filtered point clouds.

To enable easy finding of tree stems, a horizontal slice of the point cloud following the terrain elevation with a constant offset from the ground is extracted. The assumption here was to have slices from the trees to constrain the horizontal drift, but also to provide ability to tie the elevation uncertainty as the slices are relative to the instantaneous local ground eliminating the effect of drift. Fig. 4 shows an example of the outcome of the pre-processing step for a single data block.

4.3. Stem extraction

The horizontal slice of the point cloud at a constant offset from the ground is used as an input for the graph optimization pipeline, which is built with the help of the Point Cloud Library (PCL) (Rusu

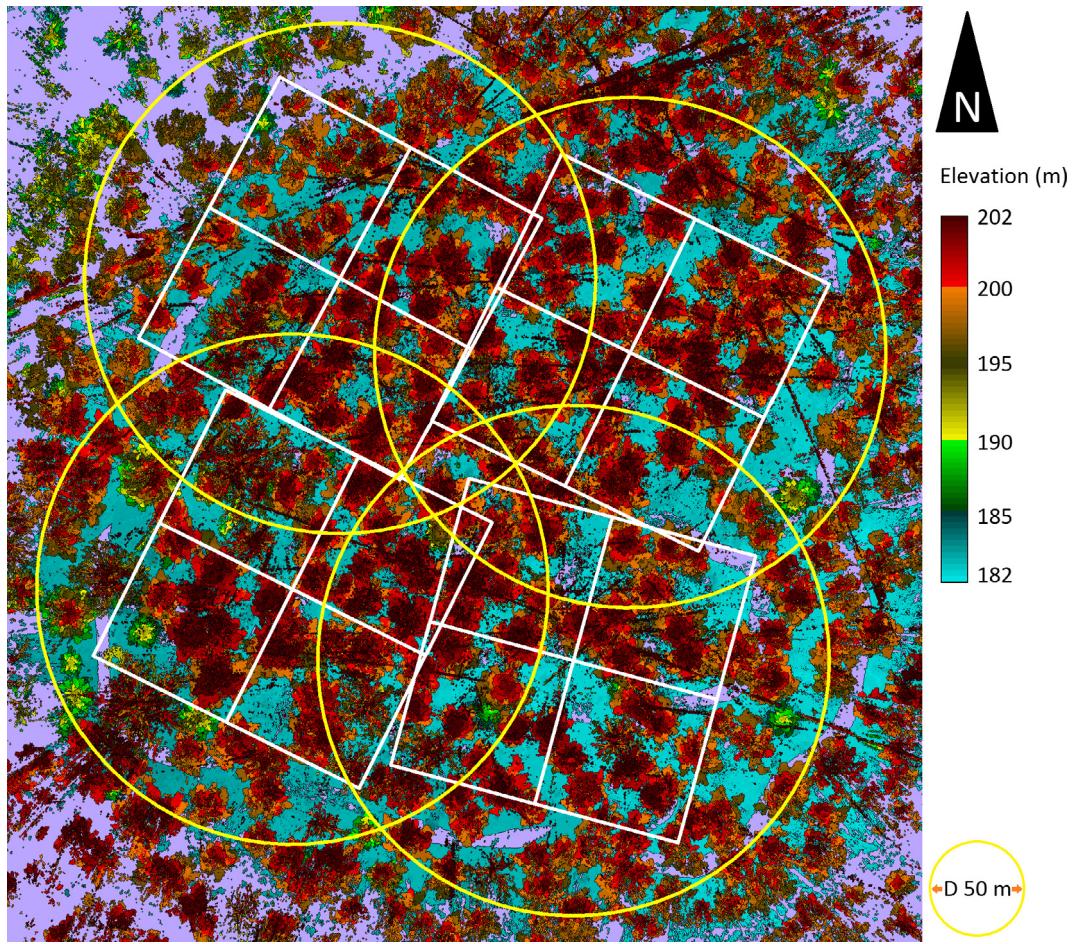


Fig. 2. Top view of the dataset B (plots 1048–1051 are delineated by white squares). Data was collected driving within and around the plot area as much as the terrain permits. This dataset contains about 385 million 3D points. There is a steep slope in the north-west corner of the plot.

Table 1
Test area and data collection characteristics.

Dataset	Trees	Trajectory length (m)	Trajectory length (s)	Date
A (1001–1004)	342	1898.7	1580	30.10.2014
B (1048–1051)	319	915.8	1247	22.10.2014
C (1108–1111)	362	1014.1	1195	15.10.2014

and Cousins, 2011). Following steps are taken to find the stem features:

1. To get rid of points not from trees each scan line is processed separately and all points not part of a sufficiently large group of spatially close points are removed.
2. Spatially close points from neighboring scanner profiles (scans) are combined into groups and points not belonging to a sufficiently large (in both number of points and in volume of bounding box surrounding them) group are removed
3. Each group of points is then processed separately to check if they represent a tree object. If enough points are closer than a preset threshold to an arc of a circle and the arc of points is long enough it is selected to represent a tree trunk and a feature object located at the determined circle center is created.

In this work, a horizontal slice of data at 3–3.5 m height from the detected ground was used as an input for the tree stem detection, and is shown in purple in Fig. 4. This offset was selected to

maximize the number of observations of tree stems and to minimize the detrimental effect of the low vegetation on the point measurements and of the effect of wind on the location of the instantaneous tree trunks. The height component for the observation is selected to be the middle point of the slice of points used for circle fitting. This ensures that as long as the ground detection is reliable, the observations of a specific tree should have comparable height values. As parameter settings for this phase, the stem observations with smaller than 10 cm diameter were discarded from the optimization, and 30% arc length of a full circle was required for an acceptable feature detection.

4.4. Matching of the stem observations

To find stem observations corresponding to a particular tree the features are clustered together with euclidean nearest neighbor search with additional conditions from temporal distance and estimated stem radius. A stem observation is added to a cluster if it passes the following tests:

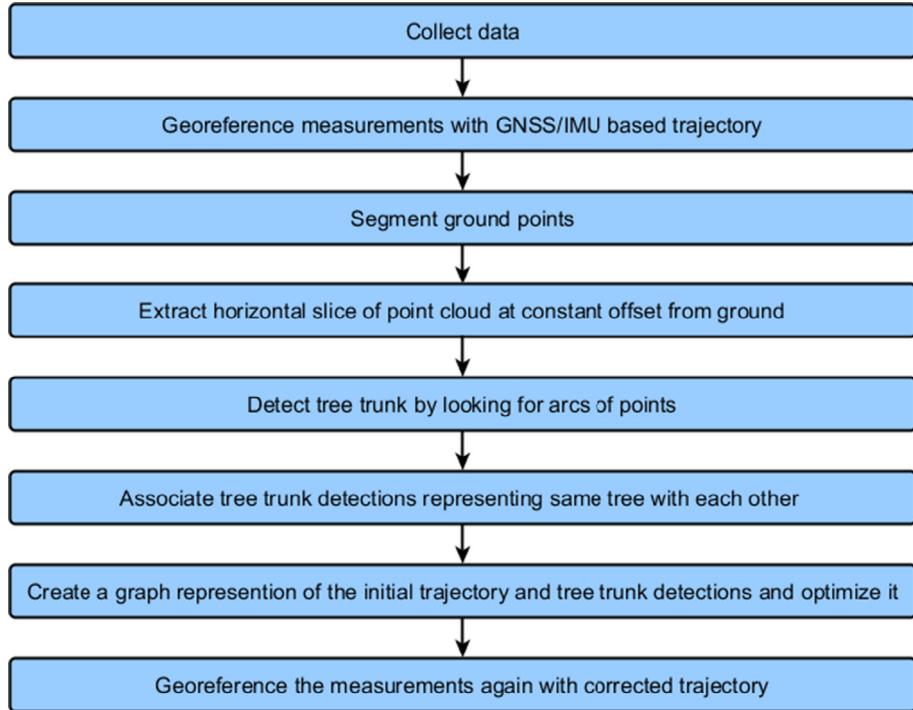


Fig. 3. Proposed processing pipeline for MLS data improvement in boreal forest inventory operations.

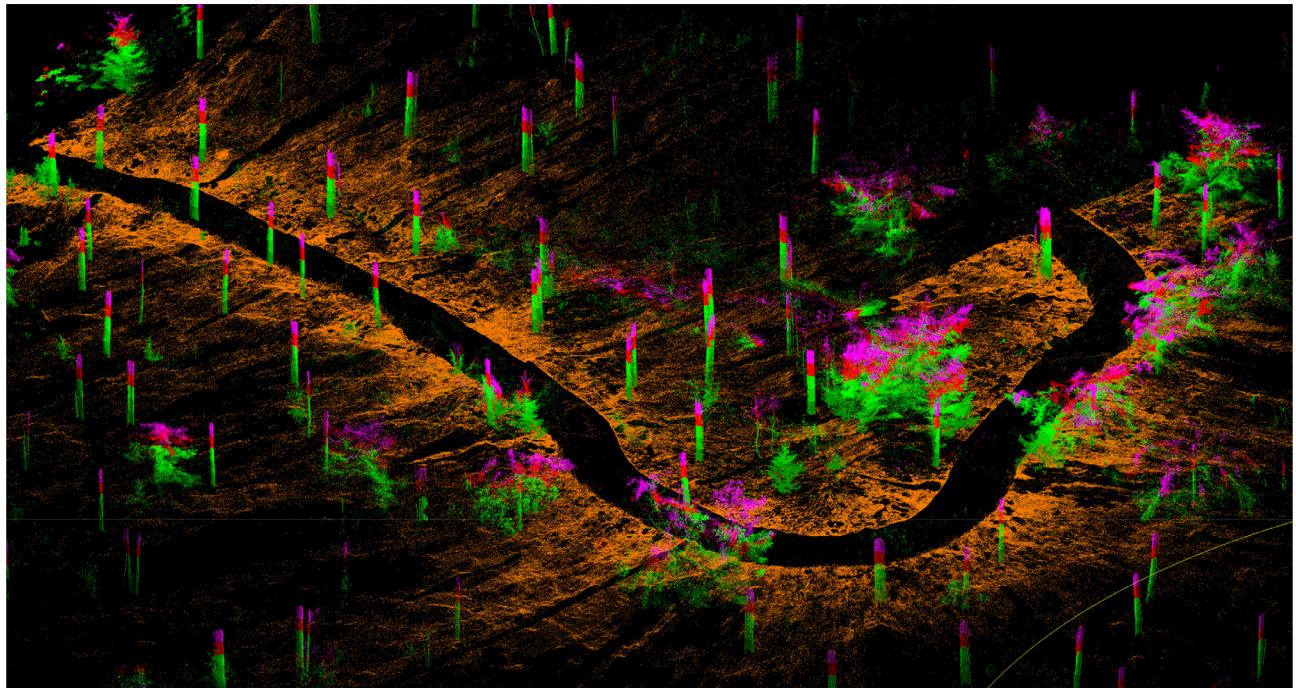


Fig. 4. Preprocessing result for a single block of ATV mounted RoamerR2 forestry data. Ground (brown) and 0.5 m thick tree trunk slices from 1 to 3.5 m height from the ground are shown in greenish and reddish colors. The slice at 3–3.5 m height from ground, shown in purple, was used in this study. The black streak on the ground is caused by the sensor blind angle, and is about 1.5 m wide.

$$\Delta d_{ij} < \Delta d_{max} \quad (1)$$

$$|r_i - r_j| < a \times \max(r_i, r_j) \quad (2)$$

with some existing member of the cluster, and passes the test

$$\Delta d_{ij} < b \times \Delta t_{ij} \quad (3)$$

with each member of the cluster. Δd_{ij} is the planar distance between the centers of feature circles representing the tree stems i and j , r denotes their circle radii and Δt_{ij} is the time difference between the observations. Constants a , b and the maximum distance allowed Δd_{max} are parameters of the observation clustering algorithm. The first two tests ensure we cluster together nearby observations of trees with similar stem radii. The last test ensures

that observations of similar tree stem observations close to each other (small Δt_{ij} as the scanner sweeps by) are not grouped together; in effect it sets the maximum for the allowed trajectory drift. In the following experiments Δd_{max} was set to 1.0 m, constant a was set to 0.1 (10% difference in the estimated stem radiiuses allowed) and b was set to 0.05 m/s (less than 0.05 m/s average drift between the observations).

4.5. Correction estimation for the trajectory

Corrections for the trajectory are estimated with the help of pose graph optimization. Pose graph optimization is commonly used in the robotics to perform simultaneous localization and mapping (SLAM) for mobile robots. For pose graph optimization the trajectory is formulated as a graph where the poses at consecutive timestamps (200 Hz in our case) and the detected features at certain time instance (captured as a mean of the feature points' timestamps) form the nodes, and the edges (or constraints) between the nodes are formed from the measured relative transformations between them. The problem can be formulated as follows:

$$F(x) = \sum_{i,j \in C} e(x_i, x_j, z_{ij})^T \Omega_{ij} e(x_i, x_j, z_{ij}) \quad (4)$$

$$x^* = \underset{x}{\operatorname{argmin}} F(x), \quad (5)$$

where x_i and x_j represent nodes, z_{ij} is the edge (or constraint) between them and $e(x_i, x_j, z_{ij})$ is the error function, which tells us how well the current node poses satisfy the constraint z_{ij} . Ω_{ij} is the information matrix (goodness) of the edge. Optimization aims at finding poses to the nodes that minimize the errors caused by the constraints. The edges can also be considered to be nonlinear springs, which pull the nodes to different directions and the optimal solution is the one with minimal energy. The problem is highly nonlinear with multiple local minima but with sufficiently good initial guess it can be solved numerically with least squares optimization.

In this work general graph optimization framework g2o developed by Kümmeler et al. (2011) with switchable constraints add-on developed by Stürnerhauf and Protzel (2012) is used for solving and applying corrections to the initial trajectory. To make the optimization robust to errors in the tree detection or wrong association of the tree detections switchable constraints are used. For a switchable constraint the result of the constraints error function is multiplied with a switch variable $s_{ij} \in [0,1]$. The use of switchable constraints requires use of additional parameter Ξ_{ij} , which is the information value of the switch value and is used for penalizing deactivation of the constraint. This is a necessity as otherwise computationally the best solution would be to deactivate all the constraints as then the error would be zero. With switchable variables the optimization can be formulated in following where γ_{ij} is the initial value (1) of the switch variable s_{ij} , and instead of optimizing only the poses we jointly optimize the switch variable s as well.

$$F(x, s) = \sum_{i,j \in C} (s_{ij} e(x_i, x_j, z_{ij}))^T \Omega_{ij} (s_{ij} e(x_i, x_j, z_{ij})) + (\gamma_{ij} - s_{ij})^T \Xi_{ij} (\gamma_{ij} - s_{ij}) \quad (6)$$

$$x^*, s^* = \underset{x,s}{\operatorname{argmin}} F(x, s), \quad (7)$$

In effect when some constraint disagrees too much (error caused by it is large and the other constraints do not allow the pose to move to reduce the error) with the other constraints the switch variable is driven towards zero to make its impact smaller (or to deactivate it altogether when $s_{ij} = 0$). In pose graph based SLAM two types of edges are commonly used; (1) odometric edges con-

nnecting the consecutive poses and (2) loop closing edges connecting spatially close but temporally far nodes. Odometry (visual or wheel based) is often used for calculating the relative transformation of the odometric edges. Laser scanner(s) or camera(s) observing the environment are then used for detecting movement by finding a transcurrent measurement to previously observed measurements through feature/landmark detection or pure scan matching. These detections are used for creating additional edges. While not error free, the explicit detection and handling of loop closures can greatly reduce the errors accumulated in the odometry.

In our case the initial trajectory computed from the GNSS and IMU measurements with Waypoint Inertial Explorer is used for generating the constraints between consecutive nodes with the information estimate for the constraints derived from the standard deviation in the position and orientation estimates. The standard deviation estimate is scaled based on the measurement distance to reach the value used. The scaling is needed because if the spatial length of the edge is not taken into account, relatively many nodes are created when the MLS moves slowly or is stationary. In such a case the applied corrections to the trajectory largely only affect these regions of the trajectory instead of spreading evenly. Similar effect could have been reached by only creating nodes to the pose graph when there is sufficient distance between the poses of the initial trajectory. The features detected from the point cloud are used as the loop closing edges and are used for correcting for the drift in the initial trajectory estimate.

As a last step, the corrected trajectory is aligned with the initial trajectory by connecting the optimized trajectory nodes to the corresponding nodes in the initial trajectory with the initial trajectory nodes and the optimized trajectory edges set to be fixed. The standard deviation of the initial trajectory is used to create information matrix for each edge connecting the optimized and initial trajectory so areas where GNSS signal is good, or show some confidence at least, (low standard deviation) are weighted more. This should result the best possible absolute positioning accuracy for the optimized trajectory as a whole.

5. Results

To test the trajectory correction pipeline for MLS data it was applied on data collected from three (3) independent boreal forest test plots. For evaluation the tree features extracted from the point clouds before and after the optimization were compared to the reference positions of the trees measured with terrestrial laser scans georeferenced with target spheres with known coordinates from geodetic network built for each plot. With errors in a range of 2–3 cm for the RTK-GNSS measurements, the absolute accuracy of the measured tree trunk positions should be of similar magnitude while the internal accuracy of the reference scans for the plots is estimated to be around 10 mm.

In the reference data only the planar locations of the tree trunks were available so only the planar corrections obtained could be compared to the reference. The results in Table 2 compare the obtained absolute coordinates of the detected trees to the reference. In Table 3 the positions of the detected trees are aligned with reference using simple 2D translation and rotation to see what the internal accuracy of the resulting point cloud is, and to reveal how much the global shape of the data is distorted. The rigid transformation for the alignment was found by minimizing the squared distance between the reference trees and the centroid of the corresponding tree observations from the optimized MLS data.

The initial mean distance tells that the GNSS-IMU system used could maintain absolute accuracy surprisingly well in the tested boreal forest conditions; the worst mean distance of 0.65 m was

Table 2

Planar accuracy of the initial and optimized test data. 'Observations' column gives the number of observations N_{Obs} used for evaluation while the 'Trees' column tells the number of individual trees N_{Trees} of which these observations were from. Distance is calculated between each observation and its reference pose. 'Distance mean' and 'Distance STD' are calculated over the observations of each tree and the number listed is the average of these values.

Dataset	Trees N_{Trees}	Observations N_{Obs}	Distance mean (m)		Distance STD (m)		Distance max (m)	
			Initial	Optimized	Initial	Optimized	Initial	Optimized
A (1001–1004)	130	508	0.646	0.466	0.222	0.008	1.537	0.945
B (1048–1051)	109	312	0.182	0.070	0.084	0.007	0.662	0.192
C (1108–1111)	152	455	0.359	0.206	0.077	0.005	0.779	0.408

Table 3

Planar accuracy of the test data aligned to the reference locations. 'Observations' column gives the number of observations N_{Obs} used for evaluation while the 'Trees' column tells the number of individual trees N_{Trees} of which these observations were from. Distance is calculated between each observation and its reference pose. 'Distance mean' and Distance STD' are calculated over observations of each tree and the number listed is the average of these values.

Dataset	Trees N_{Trees}	Observations N_{Obs}	Distance mean (m)		Distance STD (m)		Distance max (m)	
			Initial	Optimized	Initial	Optimized	Initial	Optimized
A (1001–1004)	130	508	0.434	0.059	0.19	0.008	1.232	0.247
B (1048–1051)	109	312	0.178	0.058	0.075	0.007	0.616	0.187
C (1108–1111)	152	455	0.182	0.038	0.073	0.004	0.523	0.138

found for dataset A. For the other two areas B and C the initial trajectory was of even better quality. Internal consistency, i.e. initial standard deviation of the tree stem locations was about 0.2 m for dataset A and better than 0.1 m for the datasets B and C. This shows that even if the absolute solution for the trajectory is shifted, the GNSS-IMU positioning could maintain reasonable accuracy over periods of 1200 s (see also Table 1) under the boreal forest canopy. The standard deviations for the optimized data show significant improvement in the internal accuracy down to a millimeter level, though some larger mismatches remain. Maximum errors are reduced approximately 50% or more from the initial state.

When the optimized datasets were aligned with the reference data we could see the conformity of the data: the proposed method could correct the data down to a centimeter level over the whole plot area (Table 3). However, the location and orientation of the optimized data as a whole may drift from the correct as the trajectory is in practice not fixed to any known point during the optimization. This is not a problem for a plot level tree mapping and inventory analysis. When other sources of data are fused into the process, a few landmarks may be necessary to guarantee mutual match – or alternatively the optimized MLS data is matched again to some additional 3D data set, e.g. point cloud data from TLS or ALS.

To evaluate the quality of the correction in elevation component, we compare the difference in elevation of the MLS trajectory at each location where it crosses itself (in planar) with an assumption that the sensor altitude should be more or less the same at a certain location. In practice there may be a small additional uncertainty due to system/ATV tilt on sloped terrain. The results are summarized in Table 4.

The results show a vast overall improvement in the geometric conformity of the point clouds in all the three cases. The most significant improvement is larger than an order of magnitude

decrease in the standard deviation of the distance between the observations and the reference. The small STD value tells us that the individual tree observations align well with each other, which is also evident in Fig. 6. It is also seen from Table 2 that the larger the initial drift the more probably the absolute position residual remains off.

When comparing the unaligned and aligned optimized data, the best performance is found for the dataset B where there is only small difference between the two. In that data set the error estimate for the initial trajectory was much smaller (30–50%) than that for the other two trajectories. The difference between the two other datasets with worse accuracy in the initial trajectory is understandable as no trajectory correction can help with global absolute positioning if the GNSS data is not accurate at any location on the plot.

As can be seen from Fig. 5 the proposed optimization improves significantly the internal consistency of the data. From the red lines in the plots we could see the significant improvement in the internal quality of the data. The method could also solve for large short term positional drifts reliably.

Absolute position of the optimized data is highly dependent on the GNSS visibility on the plot. The thicker the canopy the more likely the absolute positioning of the MLS data requires in practice a few known points to make an alignment for mutual match with any additional data. For plotwise data analysis however the known points are not needed. The result also shows that in the boreal forest plots with 800–900 full-grown trees/ha the GNSS-IMU positioning gives on average better than 65 cm absolute accuracy over time periods of 1200 s.

Errors in the elevation are generally larger than that for the planar coordinates, and gives typically average errors doubled to the horizontal counterpart, maximum error being initially as large as 4 meters. As expected, the elevation performance of the GNSS-IMU system is more hampered on the lack of GNSS signal than that

Table 4

Elevation residual at trajectory crossings.

Dataset	Crossings N	Distance mean (m)		Distance STD (m)		Distance max (m)	
		Initial	Optimized	Initial	Optimized	Initial	Optimized
A (1001–1004)	80	1.404	0.119	0.910	0.252	3.930	1.336
B (1048–1051)	15	0.209	0.060	0.210	0.063	0.697	0.276
C (1108–1111)	25	0.507	0.054	0.312	0.070	1.02	0.294

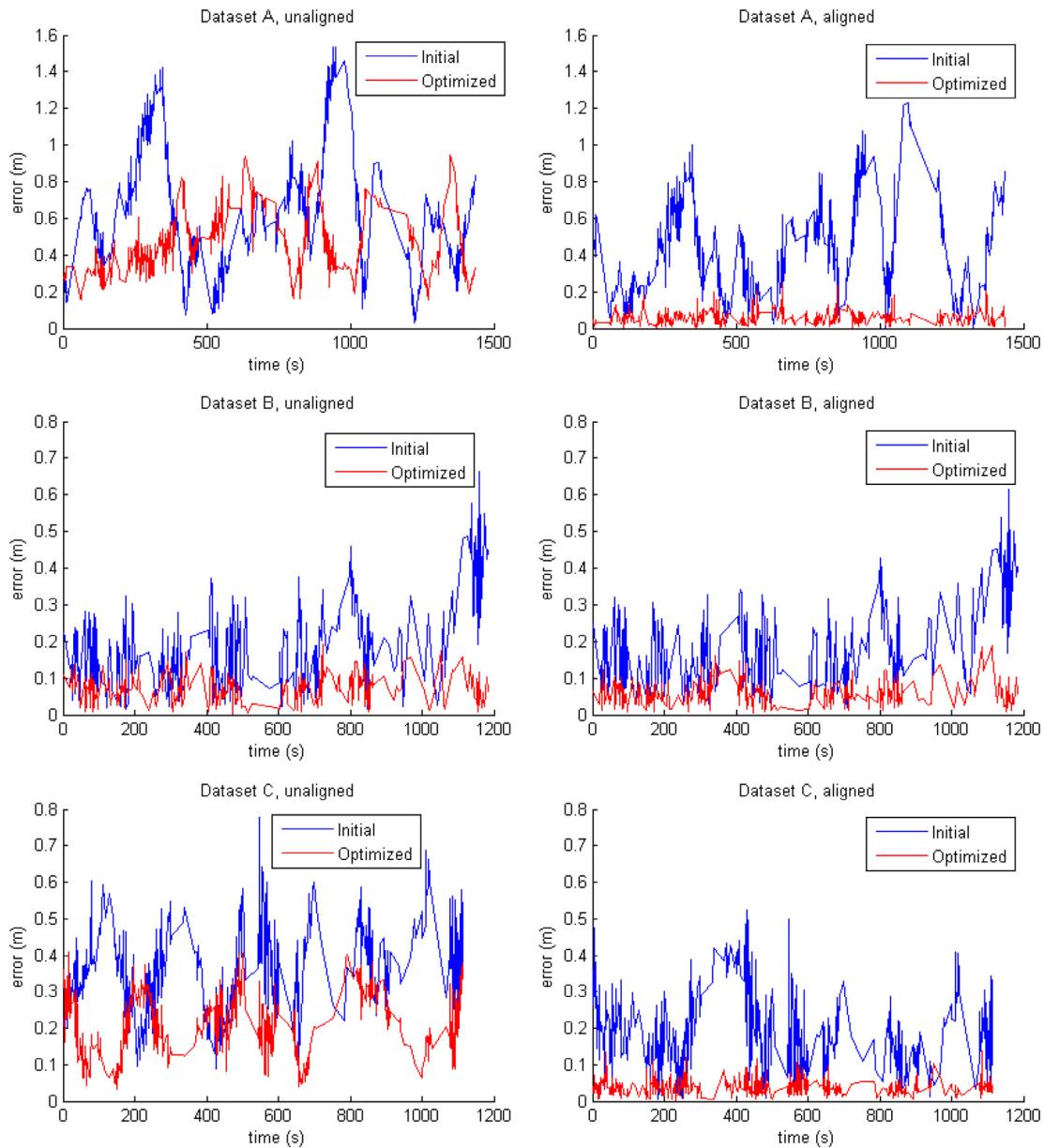


Fig. 5. Planar distances of the tree stem observations to the reference tree positions as a function of time. Left column shows results for the unaligned data, whereas the right column shows the readings for the data aligned with the reference.

of the horizontal position. Optimized elevation shows an order of magnitude worse accuracy in comparison to the horizontal residuals, being an average around 10 cm.

The constraints in the elevation direction are formed indirectly through using the offset from the detected ground, and as the detection algorithm fails to correctly find the ground level at times (mostly due to fast elevation drifting), the process results in pairing features captured from wrong elevations. In addition, the terrain of the test plots is uneven so height difference of several centimeters during the trajectory crossings can be expected, even for accurate trajectory, as the MLS system can have different orientation due to the sloped terrain. Thus the analysis may include to some extent additional uncertainty into the elevation performance estimation.

Fig. 6 demonstrates the initial condition of the data for the test area C and the effect of the performed optimization. In Fig. 7 the optimized point cloud from plot B is illustrated showing the final data product to be input for further tree inventory and modeling processing. Though there still remain minor errors in the data,

individual tree level data processing is expected to be reliably performed without a fear of confusing nearby tree copies. Terrain data has obvious gaps from the scanner blind angle, and from the shadowing caused by the terrain undulation, tree stems and undergrowth. The shadowing defects are however largely compensated by the multi-pass data collection.

6. Discussion

For some cases the optimization method could not solve for the elevation errors to a sufficient confidence. This seems to be especially related to cases where the applied ground classification method fails to extract the ground points correctly from the point cloud data due to measurement noise patterns and especially to data distortions due to fast elevational drifts. The ground classification method searches for the ground by triangulation starting from low points in the data, and has an assumption that at certain spa-

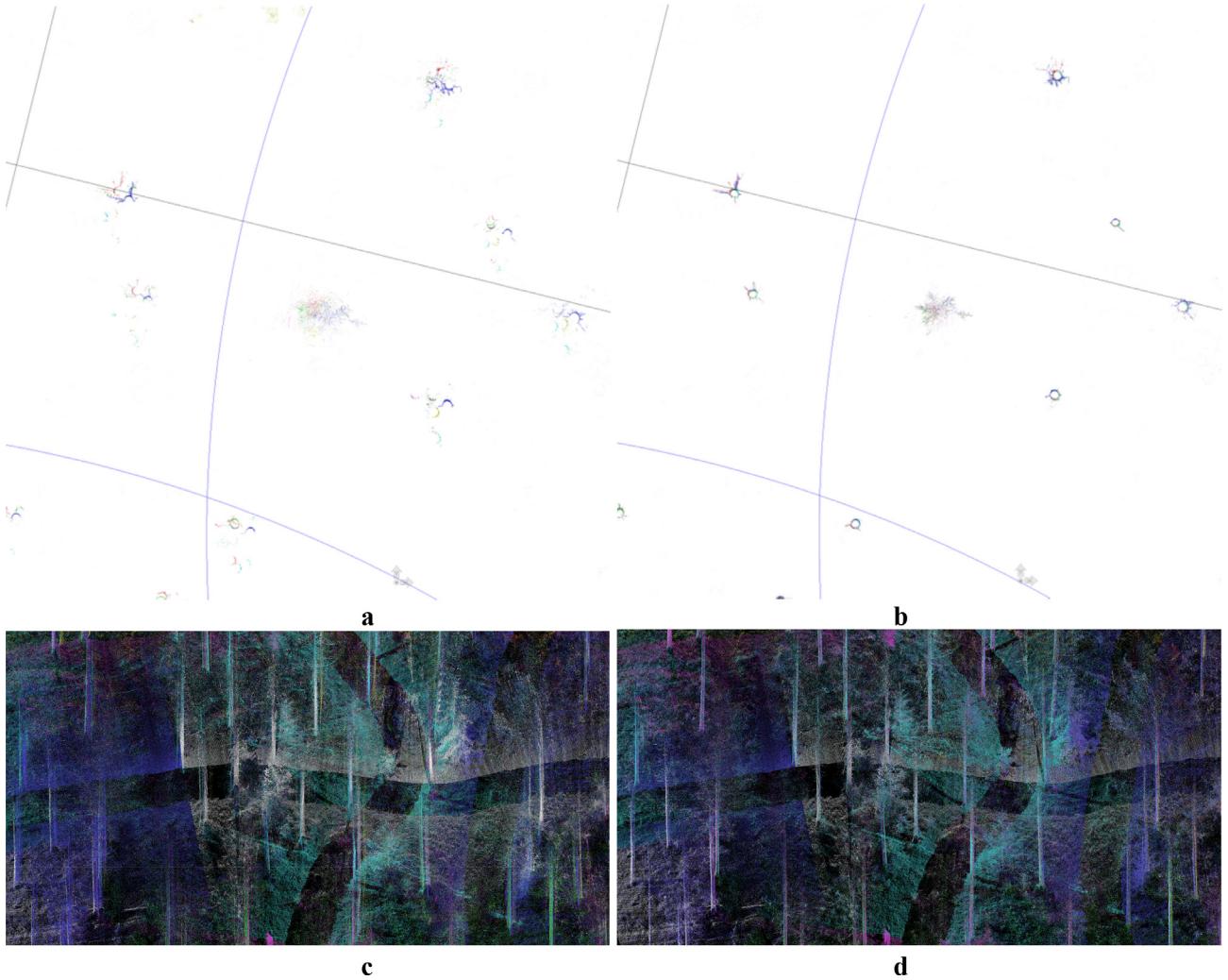


Fig. 6. Captures of the point cloud data illustrating the drift in the initial trajectory solution (left) and after (right) the trajectory optimization for the test area C (1108–1111). Top row shows the data from the ground following slices at 300–350 cm above the ground representing the raw tree stem observations. Bottom row shows a section of the 3D point cloud from the area with each data block colored distinctly for illustration of the progress of the data collection. The initial data shows multiple copies of each tree (a,c). After the trajectory optimization the measurements taken up at different times align with each other (b,d).

tial location there is only one ground surface layer present. However in practice there are cases where the MLS trajectory has large short term elevation drift and the ATV is simultaneously making a sharp turn. This results in representation of multiple layers of ground points at relatively short time periods in the overlapping data. Also at some cases the noise filtering could not handle all patterned noise, and those remaining below the actual ground cause the local ground detection to fail, or to produce odd shaped ground artefacts. As the ground classification is performed over a data block (ca. 32 s of data in each) at a time, such situations ruin the ground detection integrity. This has direct effect on the stem feature quality and the method's ability to solve for the elevation correction at such locations reliably.

To overcome this, we need to modify our ground classification to cope with short term drifting, improve noise removal, but also sensor upgrade may provide improvement to this problem. The most straightforward approach would be processing the ground detection on sub-block partitions of data, representing e.g. 10–15 s of traverse in each, or even on each profile, and extract the stem features accordingly.

On average 33% of the constraints added by the tree detections were switched off during the trajectory optimization (Constraints/Disabled constraints; area A: 1494/467, area B: 1155/429, and area

C: 1261/382). This is a significant part of the feature mass, and reduces the level of correction attained, which may have of critical influence on robustness in cases where the number of features is reduced due to object visibility or in presence of scarce features. Visual analysis shows that nearly all of these feature detections were correct in planar, but many had some noticeable mismatch error in the vertical direction caused by inaccuracies in the ground detection. Experimenting with different optimization parameters revealed that making the switches more resistant to closing improved the planar accuracy (up to 50% reductions in the residual errors) but either made the trajectory height component jittery or left most of the height uncorrected altogether. Settings chosen for the results shown in this paper were a compromise to keep the trajectory smooth. Currently switches get closed easily and this removes some correct constraints also, but this is preferable as we can still correct most of the errors, and as even one bad constraint left switched on can ruin large part of the trajectory.

There is recognized to be room for improvement in the method. Currently only a small subset of the original points is used, and the use of circle matching for stem features may not be optimal as tree stems rarely are perfectly circular. An additional step to refine the trajectory by matching the points of temporally different parts of the point cloud with iterative closest point (ICP), or some other

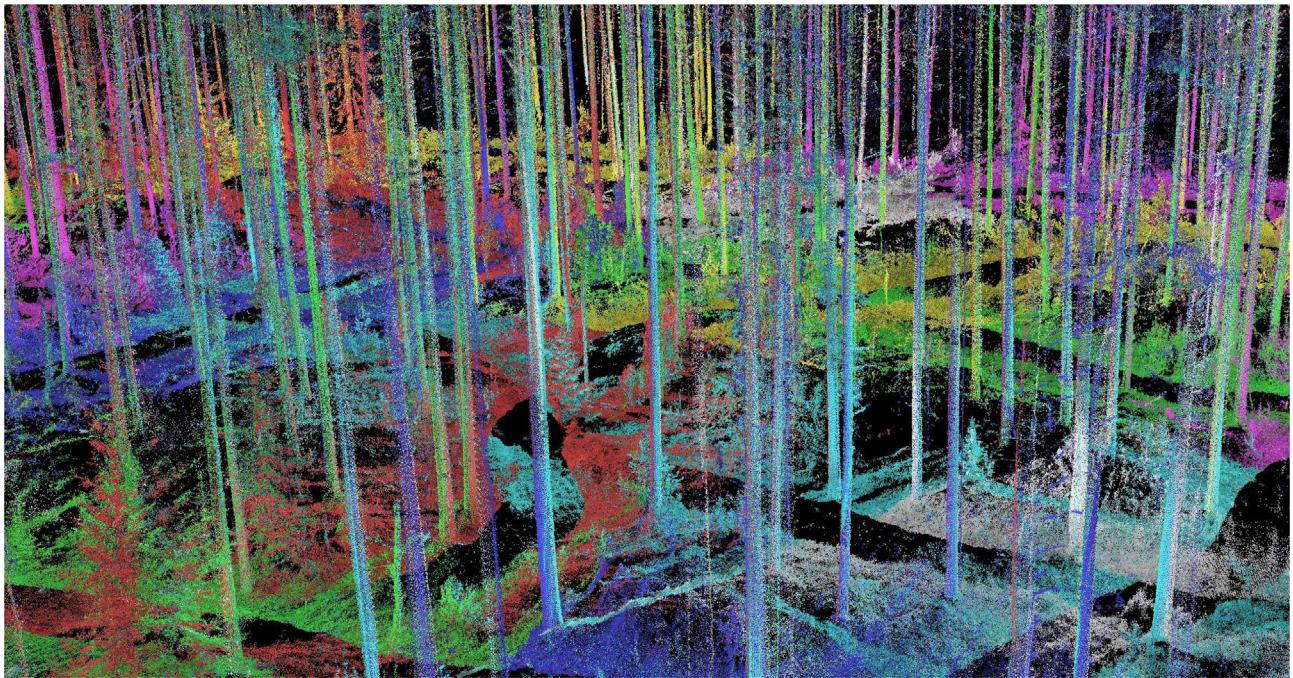


Fig. 7. Optimized point cloud for the forest plot B shows the accurately matched tree stems ready for further stem modeling and inventory work.

similar algorithm could possibly largely remove the remaining errors. Using the method as an initial step to correct the trajectory for the larger positional and angular errors is beneficial as iterative closest point algorithm is prone to settling for local minima instead of finding the correct transformation. By starting from an optimized solution close to the correct one the chances for finding converging fine-tune corrections to the trajectory locally using ICP are expected to greatly improve.

7. Conclusions

We have developed a process pipeline for MLS data from boreal forests to correct trajectory data for errors induced by obstructed GNSS signal under mature boreal forest canopy and report results achieved using the method. The experiments were conducted on data from ATV mounted RoamerR2 MLS system, which allows for efficient collection of 3D tree inventory data from the ground. The point cloud data was obtained with cross-track vertical laser scanning and post-processed GNSS-IMU positioning. We present the processing pipeline and report optimization results for the data from three test plots of nominal size 64 m by 64 m, and analyze the results against tree location reference derived from TLS data georeferenced to spherical landmarks with known locations on each plot.

The results show that the method could vastly improve the quality of the produced optimized point cloud. The internal consistency of tree stem locations is reduced to millimeter level in terms of distance residual standard deviation: for plot A 0.008 m, for B 0.007 m and C 0.005 m. The methods ability in providing correction in absolute coordinates is found slightly weaker, but this is of secondary importance from individual tree modeling point of view. There were no absolute landmarks used to constraint the optimization from locational drifting, and the results show that with the larger the initial drift the larger the remaining absolute positional offset may be expected.

The attained level of correction in the internal accuracy of the point cloud enables us to use all of the non noise measured points when extracting individual tree inventory parameters, such as

diameter-at-breast-height and tree height. The improved point cloud data permit stem curve modeling of individual trees unambiguously. We could also expect more accurate object level matching to be successful after running the optimization method on forest MLS data as the ambiguity in pairing adjacent tree stems is significantly reduced.

The applied feature detection is less accurate in constraining the elevation direction than what is achieved in the planar domain. For correcting the elevation the features in use also require us to trade-off some of the performance for planar corrections as use of them causes a large part of the constraints to be disabled during the optimization. Use of additional features, such as ground batches and horizontal structures, which could better tie the correction for error in elevation to the optimization pipeline, is a straightforward task and should enable us to remove the remaining major errors in the near future.

By incorporating a separate set of features to correct for the elevation drift, the tree features currently used could be reserved for only constraining the horizontal drift of the trajectory. This would decrease the number of disabled constraints and enable us to capitalize most of the detected stem features in making the optimization more accurate. A ground detection algorithm better suited to handle raw data with considerable elevation errors and below-ground patterned noise within relatively short periods of time is needed.

The proposed method is based on sparse features and graph optimization, and is tested on MLS data on forestry application with supposedly the most challenging GNSS signal loss conditions. However, the method could be also used in any general MLS case, e.g. for correcting urban and road asset point cloud data as similar pole features are abundant in those environments. Additional features, such as building and other structural corners, provide robust constraints for trajectory optimization in built and industrial environments that often pose a challenge for the GNSS based positioning. Thus the proposed method is seen to provide an important asset for a diversity of MLS data applications.

The phase-shift ranging scanner mounted on RoamerR2 and used in the experiments for this paper generate noise in dense veg-

eration, and does not provide a reliable ranging counts from within or behind the canopy hampering the desired tree and ground modeling. It is also to be noted that since the data collection for this study the instrument manufacturer has released a major firmware update to improve the ranging performance of the scanner. Use of laser scanner based on pulsed time-of-flight ranging, such as Riegl VUX-1HA currently in use with the next generation AkhkaR3 and RoamerR3 systems of the FGI with multiple echo detection, should also improve performance of the method in terms of data processing, and adaptation of the proposed method for a backpack mapping will be the topic of the following paper.

The amount of the laser points on stems was adequate for feature detection, and further for solving the corrections for the trajectory drift accurately. The ranging noise combined with the angular resolution of the scanner prevented the use of faraway features and ground points as the points have large distance between each other and begin to be filtered away in the noise removal step. In future research we aim to investigate the use of the method on forestry and urban data collected with AkhkaR3 backpack laser scanning system. In the forestry context we will also compare how the resulting tree parameters extracted from an MLS point cloud optimized with the proposed method compares to the tree parameters extracted from point measurements gathered with terrestrial or UAV based methods.

Acknowledgements

This study was made possible by financial funding from the Academy of Finland for “Multi-spectral personal laser scanning for automated environment characterization (300066)”, “Centre of Excellence in Laser Scanning Research (CoE-LaSR) (272195)”, “Interaction of Lidar/Radar Beams with Forests Using Mini-UAV and Mobile Forest Tomography” (259348), and in part the research leading to these results has received funding from the European Community’s Seventh Framework Programme ([FP7/2007-2013]) under grant agreement no. 606971. Strategic Research Council at the Academy of Finland is acknowledged for financial support “Competence Based Growth Through Integrated Disruptive Technologies of 3D Digitalization, Robotics, Geospatial Information and Image Processing/Computing Point Cloud Ecosystem” (293389).

References

- Alismail, H., Baker, L.D., Browning, B., 2014. Continuous trajectory estimation for 3D slam from actuated lidar. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 6096–6101.
- Axelsson, P., 2000. DEM generation from laser scanner data using adaptive TIN models. *Int. Arch. Photogram. Remote Sensing XXXIII (Part B4)*, 110–117.
- Bailey, T., Durrant-Whyte, H., 2006. Simultaneous localization and mapping (SLAM): part II. *IEEE Robot. Autom. Magazine* 13 (3), 108–117.
- Besl, P.J., McKay, N.D., 1992. A method for registration of 3-D shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* 14 (2), 239–256. <http://dx.doi.org/10.1109/34.121791>.
- Borrmann, D., Elseberg, J., Lingemann, K., Nüchter, A., Hertzberg, J., 2008. Globally consistent 3D mapping with scan matching. *J. Robot. Auton. Syst. (JRAS)* 56 (2), 130–142.
- Bosse, M., Zlot, R., Flick, P., 2012. Zebedee: Design of a spring-mounted 3-d range sensor with application to mobile mapping. *IEEE Trans. Robot.* 28 (5), 1104–1119.
- Cabo, C., Kukko, A., García-Cortés, S., Kaartinen, H., Hyppä, J., Ordoñez, C., 2016. An algorithm for automatic road asphalt edge delineation from mobile laser scanner data using the line clouds concept. *Remote Sensing* 8 (9), 740.
- Carbone, L., Aragues, R., Castellanos, J.A., Bona, B., 2014. A fast and accurate approximation for planar pose graph optimization. *Int. J. Robot. Res.* 33 (7), 965–987.
- Chen, Yuwei, et al., 2016. Scan matching technology for forest navigation with map information. In: Position, Location and Navigation Symposium (PLANS), 2016 IEEE/ION. IEEE, pp. 198–203.
- Durrant-Whyte, H., Bailey, T., 2006. Simultaneous localization and mapping: part I. *IEEE Robot. Autom. Mag.* 13 (2), 99–110.
- Elseberg, J., Borrmann, D., Nüchter, A., 2013. Algorithmic solutions for computing accurate maximum likelihood 3D point clouds from mobile laser scanning platforms. *Remote Sensing* 5 (11), 5871–5906.
- Frese, U., Hirzinger, G., 2001. Simultaneous localization and mapping – a discussion. In: Proceedings of the IJCAI Workshop on Reasoning with Uncertainty in Robotics, Seattle, USA, pp. 17–26.
- Kaartinen, H., Hyppä, J., Vastaranta, M., Kukko, A., Jaakkola, A., Yu, X., Pyörälä, J., Liang, X., Liu, J., Wang, Y., Kaijaluo, R., Melkas, T., Holopainen, M., Hyppä, H., 2015. Accuracy of kinematic positioning using Global Satellite Navigation Systems under forest canopies. *Forests* 6 (9), 3218–3236.
- Kümmerle, R., Grisetti, G., Strasdat, H., Konolige, K., Burgard, W., 2011. g2o: A general framework for graph optimization. In: IEEE International Conference on Robotics and Automation (ICRA), 2011, pp. 3607–3613.
- Lehtola, V.V., Virtanen, J.P., Vaaja, M.T., Hyppä, H., Nüchter, A., 2016. Localization of a mobile laser scanner via dimensional reduction. *ISPRS J. Photogram. Remote Sensing* 121, 48–59.
- Liang, X., Kankare, V., Hyppä, J., Wang, Y., Kukko, A., Haggrén, H., Yu, X., Kaartinen, H., Jaakkola, A., Guan, F., Holopainen, M., Vastaranta, M., 2016. Terrestrial laser scanning in forest inventories. *ISPRS J. Photogram. Remote Sensing* 115, 63–77.
- Liang, X., Hyppä, J., Kukko, A., Kaartinen, H., Jaakkola, A., Yu, X., 2014. The use of a mobile laser scanning system for mapping large forest plots. *IEEE Geosci. Remote Sens. Lett.* 11 (9), 1504–1508.
- Liu, T., Carlberg, M., Chen, G., Chen, J., Kua, J., Zakhor, A., 2010. Indoor localization and visualization using a human-operated backpack system. In: International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2010, pp. 1–10.
- Nüchter, A., Elseberg, J., Schneider, P., Paulus, D., 2010. Study of parameterizations for the rigid body transformations of the scan registration problem. *J. Comput. Vision Image Understand. (CVIU)* 114 (8), 963–980.
- Olofsson, K., Holmgren, J., Olsson, H., 2014. Tree stem and height measurements using terrestrial laser scanning and the RANSAC algorithm. *Remote Sensing* 6 (2014), 4323–4344. <http://dx.doi.org/10.3390/rs6054323>.
- Qian, C., Liu, H., Tang, J., Chen, Y., Kaartinen, H., Kukko, A., Hyppä, J., 2017. An integrated GNSS/INS/LiDAR-SLAM positioning method for highly accurate forest stem mapping. *Remote Sensing* 9 (1), 3.
- Rusu, R.B., Cousins, S., 2011. 3D is here: Point Cloud Library (PCL). In: IEEE International Conference on Robotics and Automation (ICRA), 2011 Shanghai, China, May 2011.
- Sünderhauf, N., Protzel, P., 2012. Switchable constraints for robust pose graph slam. *Intell. Robot. Syst. (IROS) 2012*, 1879–1884.
- Tang, J., Chen, Y., Kukko, A., Kaartinen, H., Jaakkola, A., Khoramshahi, E., Hyppä, H., 2015. SLAM-aided stem mapping for forest inventory with small-footprint mobile LiDAR. *Forests* 6 (12), 4588–4606.
- Thrun, S., Montemerlo, M., 2006. The GraphSLAM algorithm with applications to large-scale mapping of urban structures. *International Journal on Robotics Research* 25 (5/6), 403–430.
- Thrun, S., 2002. Probabilistic robotics. *Commun. ACM – Robots: Intell., Versatility, Adaptivity* 45 (3), 52–57.
- Zhang, J., Singh, S., 2014. LOAM: Lidar Odometry and Mapping in Real-time. In: Proceedings of Robotics Science and Systems (RSS). Berkeley, CA, USA.