Ouattara, Issouf; Visala, Arto

LiDAR-odometry based UAV pose estimation in young forest environment

Published in:
IFAC-PapersOnLine

DOI:
10.1016/j.ifacol.2022.11.121

Published: 22/11/2022

Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 55-32 (2022) 95–100

LiDAR-odometry based UAV pose estimation in young forest environment ⋆

Issouf Ouattara * **Vesa Korhonen** * **Arto Visala** *

* *Department of Electrical Engineering and Automation, Aalto University, Espoo, Finland*

Abstract: In this study, we propose a real-time pose estimation solution for Unmanned Aerial Vehicle in a seedling pine forest environment. Our method uses graph-based approach to fuse data from an onboard IMU sensors, a GNSS receiver and a 3D LiDAR. Features are detected from every LiDAR scan. A local map is built from the detected features and used to compute the LiDAR odometry in real time for the incoming scans. In order to obtain a robust estimate of the state of the vehicle, the noise covariance of the LiDAR odometry is updated at each iteration using the fitness score of the LiDAR. The proposed solution provides promising trajectory and velocity estimates even in GNSS denied scenario. Both the local and global consistencies of the estimated trajectory are encouraging.

Keywords: unmanned aerial vehicle, SLAM, real-time estimation, autonomous vehicle, forestry.

1. INTRODUCTION

In Finland, forestry plays an important role in the country's economy. Among the tree species in Finnish forest, pine trees occupy the largest proportion of the forest area (Valsta, 2017), making them one of the most important species. However, moose (Alces alces), also called elk in Europe, is a real threat to the seedling pine stands and can cause extensive damage to the stands. According to (Nevalainen et al., 2016), considering all kind of forest stands, the majority (75%) of moose damage occurred in pine-dominated stands. In order to prevent these damages, several techniques are used, among which is the spraying of non-toxic moose repellent chemical on the top of the young pine trees. This spraying is currently done manually, and is time and resource consuming. Using autonomous Unmanned Aerial Vehicles (UAVs) for spraying the chemical can be an economical alternative. However, this autonomous use of UAVs inside a seedling pine forest environment is challenging because of the presence of many obstacles (i.e. the canopy), and the need to precisely and selectively spray the repellent chemical on the top of individual young trees. It requires a full navigation framework including a pose estimation, obstacles detection, and a path planning solution. The aim of this study is to develop a solution that addresses the problem of estimating the pose of the UAV in the young pine forest as a basis for the development of autonomous UAVs for spraying non-toxic chemicals in the seedling pine forest in subsequent studies.

When navigating in a young forest environment that is surrounded in certain areas by large trees, the GNSS signal may degrade when the UAV flies closer to the large trees. In autonomous navigation scenarios, this sudden GNSS signal deterioration can lead to a loss of control. In this type of environment, it is also desirable to add extra sensor with additional information to improve the navigation performance. In this study, a LiDAR is used to try to address both concerns: improving the navigation performance when GNSS signal is available and coping with GNSS signal deterioration or complete loss.

UAV navigation is in general a well studied field of research. Data fusion techniques are used to fuse the LiDAR sensor data with other sensor data. This include filtering techniques with or without smoothing, and the Simultaneous Localization And Mapping (SLAM) solutions. Several filtering solutions based on the Kalman filter have been explored to fuse LiDAR odometry with other sensors. The study by (Hening et al., 2017) develops a state estimation solution for UAVs in a GNSS-degraded environment. The study combines data from a 3D LiDAR, a GNSS receiver and an IMU sensor. Feature points are extracted from successive point clouds and they are matched using the Iterative Closest Point (ICP) technique. An adaptive Extended Kalman Filter (EKF) is used to combine the position and velocity information from three blocks: a mapping block, a GNSS block, and an inertial navigation system (INS) block. The ICP scan matching error and the GNSS receiver horizontal and vertical dilution of precision are used to update the measurement covariance matrix. This idea of updating the measurement covariance matrices is used in our study. The study by (Chiella et al., 2019) combines data from a GNSS receiver, an Attitude and Heading Reference System (AHRS), and a 2D LiDAR to

⋆ This work is supported by Technology Industries of Finland Centennial Foundation and Jane and Aatos Erkko Foundation; The Future Makers.

Issouf Ouattara is a PhD Student with the Autonomous Systems research group, Department of Electrical Engineering and Automation, School of Electrical Engineering, Aalto University, PO Box 15500, FIN-00076 Aalto, Finland (e-mail: issouf.ouattara@aalto.fi).

Vesa Korhonen is an Operations Engineer at the Department of Electrical Engineering and Automation, Aalto University, PO Box 15500, FIN-00076 Aalto, Finland(e-mail: vesa.korhonen@aalto.fi).

Arto Visala is a Professor with the Autonomous Systems research group, Department of Electrical Engineering and Automation, School of Electrical Engineering, Aalto University, PO Box 15500, FIN-00076 Aalto, Finland (e-mail: arto.visala@aalto.fi).

10.1016/j.ifacol.2022.11.121

estimate the position and velocity of a UAV in a sparse forest environment. Tree trunks are used as features to match the successive scans. An Unscented Kalman Filter (UKF) is used to combine the relative measurements from the LiDAR and the absolute measurements from the GNSS receiver. The approach provides a low computational filtering solution but it cannot deal with medium or long term drift in case of total lack of GNSS signal. An early study by (Cui et al., 2014), uses also tree trunks for scan matching. A Kalman filter is used for motion estimation and the GraphSLAM method is used to correct for long term drift. However, the computational cost of graph-based SLAM method is of concern, particularly for an UAV system with small energy resources and limited computational power. Recent advances in graph-based SLAM theory make it suitable for real-time use without the need for an extra filtering scheme. One such breakthrough is the Incremental Smoothing And Mapping using the Bayes tree (ISAM2) method (Kaess et al., 2012). The study by Shan et al. (2020) uses the ISAM2 framework for pose estimation and SLAM in indoor and outdoor scenarios. For the matching of the LiDAR scans, the study uses edge and corner features that are not necessarily well defined in the environment of concern in our study. The normal distribution transform (NDT) (Magnusson et al., 2007; Biber and Strasser, 2003) is used for scan matching in our study. Before the NDT step however, the point cloud is downsampled using the tops of the young trees as features. This guarantees that the most important shapes in the environment are preserved while the point cloud is downsampled. Our study uses the ISAM2 algorithm that fuses the reading from an IMU, the LiDAR odometry, and optionally GNSS data when it is available. The scan matching is based on the concept of keyframes and local map used in several previous studies. In our study, the keyframes are used to build a local map. However, the pose is estimated at each scan in real-time, not only at the keyframe poses.

The contribution of this study is three-fold. Firstly, although there exist previous studies for UAV state estimation in forest environments, the number of these studies is relatively smaller than studies for state estimation of UAV in other environments such as indoors and underground mining. Our study therefore contributes to the evaluation of state estimation techniques of UAV in forest environments. Secondly, by relying on the young tree tops and their surrounding points, the approach is able to downsample the point cloud without demanding computation while preserving important shape features. Thirdly, in our study, the NDT scan matching is used and the state estimation algorithm is not based on the assumption of the existence of tree trunks. The proposed method can therefore be used in young forest where tree trunks cannot be differentiated in a LiDAR point cloud data. In fact, the propose solution can be used in any environment were keypoints can be detected and the surrounding points of the keypoints represent some shapes. With the choice of the components of our solution, a real-time estimate is achieved without compromising its accuracy.

The remainder of this paper is organized as follows: Section 2 describes the methods, in section 3 the experiment and the results are presented, and finally a conclusion is drawn in Section 4.

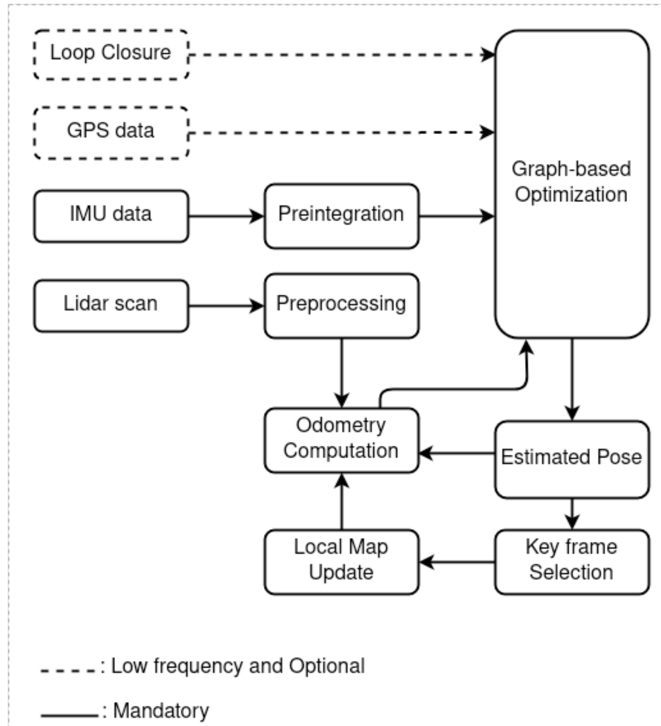

Fig. 1. Overall Pose Estimation Process. The mandatory measurements are the data from the IMU the LiDAR.

2. METHODS

This section introduces the proposed approach which is based on the notion of factor graph. A factor graph, is a bipartite graph consisting of factors connected to variables. The variables represent the unknown UAV state in this study. The factors represent probabilistic constraints on the variables, derived from measurements or prior knowledge (Dellaert, 2012). The main factors used in this study are the LiDAR odometry, and the IMU preintegration measurements. The GNSS data is also used as a factor when it is available. Additionally, a loop closure measurement is used when available to correct large drift. The LiDAR sensor provides a sparse point cloud data at a frequency of about 10Hz. This data is processed in real-time by first downsampling it around feature points. The processed cloud is used to compute the odometry with respect to a local map. The obtained odometry is fused with printegrated IMU measurement in a factor graph. Loop closure is monitored and added to the factor graph to allow for drift correction. The estimated state of the vehicle is composed of the attitude of the vehicle represented by a rotation matrix $R^{BW} \in SO(3)$, the position vector p^W, the velocity vector v^W, and the bias of the accelerometer and the gyroscope. The attitude represented by the rotation matrix is the attitude of the body frame of the vehicle with respect to the world frame which in our case is a local North-East-Down (NED) centered at the initial position of the UAV. The position and velocity are expressed in the local world frame. The overall process of estimating the state is shown in Fig. 1.

2.1 Point Cloud processing

When using a LiDAR sensor for pose estimation, the sensor collects point cloud data at regular intervals of time. These point clouds can be large and the real-time processing is an issue. In order to reduce the number of points in a single scan, features can be detected and used for subsequent steps or the point cloud can be down sampled using methods like voxel grid downsampling (Shan et al., 2020; Cui et al., 2014; Chiella et al., 2019). In the present study, when a scan is received, the tops of the young trees are detected as features using a local maximum filter in the vertical z direction and they are kept as the downsampled point cloud. To this downsampled point cloud are added the neighbouring points of the detected young tree tops. The use of these neighbouring points has two purposes. The first purpose is to use them in order to remove detected local maximum points that may not be tree tops. The second purpose is to enrich the downsampled point cloud. Using the neighbouring points and eigen analysis of the scatter matrix in Eqs. 1, the local maximum points that may not be young tree tops are removed by only keeping local maximum points that have large variations in the principal directions. This approach is done by keeping only points for which the smallest eigen value of $\Sigma(p)$ is larger than a chosen threshold. This is inspired from the study by Zhong (2009) for the detection of intrinsic shape signatures (ISS) keypoints in point cloud data. But contrary to the study by Zhong (2009), we keep points with a similar spread along the principal directions.

$$\Sigma(p) = \frac{1}{N} \sum_{q \in \mathcal{N}(p)} (q - \mu_p)(q - \mu_p)^T \tag{1}$$

In the Eqs. 1, p is a detected tree top; $\mathcal{N}(p)$ is the set of neighbouring points of p; N is the number of points in $\mathcal{N}(p)$; and μ_p is the mean of the points in $\mathcal{N}(p)$.

Following the preprocessing step, the resulting downsampled point cloud, despite having fewer points, will still keep important shapes information of the environment. Another advantage is that these young tree tops and their surrounding points can be used both as landmarks and as potential obstacles in our subsequent research studies.

2.2 Scan Matching and LiDAR Odometry

A local map is used for computing the odometry of the current LiDAR scan. The local map is updated by adding and removing keyframes from it so that it is always composed of a fixed number of keyframes. A current scan is selected as a keyframe when the drone has traveled a given distance or turned to some heading value and when the scan matching of that scan with the local map has a fitness score lower than a certain value. A traveled distance is computed by integrating the distances between consecutive scans on the trajectory. A scan selected as a keyframe is added to the local map using a voxel filter. In this study, unlike others studies, the keyframes are only used for updating the local map and the pose of the UAV is computed at every scan in real-time. This is important because the autonomous flight in a young forest environment needs accurate pose availability at medium to high frequency for obstacle avoidance. This is also important because a relatively cheap IMU is used and cannot be relied upon for a long period prediction. Using the local map, the odometry of the current LiDAR scan is computed using the normal distribution transform (NDT) method. The scan is downsampled during the processing step so as to maintain only important shapes information. This makes the use of NDT in a real-time fashion possible and results in a good odometry estimation. The NDT scan matching is initialized using the previous pose estimation thereby improving its convergence. The NDT method is used in this study in order to avoid a costly feature matching process. To further boost the speed of the scan matching process, parallel processing based on the work by (Koide et al., 2019) is used.

2.3 IMU Pre-integration

The IMU sensor measures the linear acceleration and the angular velocity of the vehicle in the body frame of the sensor which is taken to be also the vehicle body frame. The measurements are corrupted with additive noise considered to be Gaussian and a slowly varying bias. They are modeled following the Eqs. 2a, 2b. In these equations, a_t^B and w_t^B are respectively the acceleration and angular velocity of the vehicle measured by the IMU in the body frame; a_t^W is the true acceleration expressed in the world frame; g^W is the acceleration of gravity expressed in the world frame; R^{BW} is the rotation matrix from the world frame to the body frame and is equal to the inverse of the rotation matrix representing the attitude of the vehicle in the world frame; w_t is the true angular velocity expressed in the body frame; b_t^a and b_t^g are respectively the accelerometer bias and the gyroscope bias; n_t^a and n_t^g are white noise affecting, respectively the measurements from the accelerometer and the gyroscope.

$$a_t^B = R^{BW}(a_t^W - g^W) + b_t^a + n_t^a \tag{2a}$$

$$\omega_t^B = \omega_t + b_t^g + n_t^g \tag{2b}$$

Between two scan readings, several dozens of IMU measurements are acquired. These measurements between two LiDAR scans can be integrated using the notion of preintegration. We refer to the two consecutive scans as the previous scan s_i and the current scan s_j. In the preintegration theory, the integration of IMU measurement is done in the local frame of the scan s_i. This is done by using the relationship shown in Eqs. 3. Using these relationships, local constraints between the two LiDAR scan poses is derived. These constraints represented by Eqs. 4 are indeed expressed in the local frame of the scan s_i (Forster et al., 2016). The expressions in Eqs. 4 have complex dependencies on the gyroscope and accelerometer noises. The study in (Forster et al., 2016) developed a theory to deal with the noise terms.

$$\begin{aligned}
R_j &= R_i \prod_{k=i}^{j-1} Exp((\omega_k - b_k^g - n_k^{gd})\Delta t) \\
v_j &= v_i + g\Delta t_{ij} + \sum_{k=i}^{j-1} R_k(a_k - b_k^a - n_k^{ad})\Delta t \\
p_j &= p_i + \sum_{k=i}^{j-1} [v_k \Delta t + \frac{1}{2} g \Delta t^2 + \frac{1}{2} R_k (a_k - b_k^a - n_k^{ad})\Delta t^2]
\end{aligned} \tag{3}$$

In Eqs. 3 and 4, R_i, v_i, and p_i are respectively the rotation matrix representing the attitude of the vehicle, the velocity, and the position at the time when the scan s_i is measured. R_j, v_j, and p_j are the same quantities at the time when the scan s_j is measured. $\Delta t_{ij} = \sum_{k=i}^{j-1} \Delta t$ where Δt is the time difference between two IMU measurements; n_k^{gd} and n_k^{ad} are the discrete version of the gyroscope noise and the accelerometer noise, respectively. The noise parameters (i.e. the variances) are computed using the Allan Variance method.

$$\begin{aligned} \Delta R_{ij} &= R_i^T R_j \\ \Delta v_{ij} &= R_i^T (v_j - v_i - g\Delta t_{ij}) \\ \Delta p_{ij} &= R_i^T (p_j - p_i - v_i \Delta t_{ij} - \frac{1}{2} g \Delta t_{ij}^2) \end{aligned} \tag{4}$$

2.4 Factor graph and Optimization

In this study, the IMU preintegration measurements and the LiDAR odometry are used as factors in the factor graph. It is common to only add factors when a new keyframe is detected to avoid a large factor graph. The IMU sensor can then be used for high rate prediction. This, however, requires an accurate IMU sensor especially if the keyframes are distant in time from each other. In this study, a factor is added to the factor graph whenever a new LiDAR scan is available. This allows to have a high rate estimate of the pose of the UAV. The ISAM2 algorithm is used for the graph optimization. It uses the Bayes tree data structure to only update a small part of the graph that is affected by new measurements. Because only the state of the vehicle is considered as unknowns in the factor graph without landmarks, the ISAM2 algorithm only update the solutions to the last part of the factor graph that is affected by the new measurements. To update the local map however, we proceed to a full batch optimization that is done at lower frequency. If several loop closures are used, the computation cost can become excessive because several states might be affected by the loop closure measurements. In this study, when the current state is close to a visited portion of the trajectory, a loop closure is added only if that visited portion is at some chosen number of keyframes away from the current state. This avoids adding small loops or a large number of loop closures. The loop closure is added to the factor graph as LiDAR odometry measurement. Because the loop closure does not need to be detected in real time, it is computed using two successive scan matching methods. First an NDT scan matching is used to provide a first estimate that is used in the iterative closest point (ICP) algorithm for refinement. This non real-time loop closure update is still important for future states because it affects the local map which is used to compute the lidar odometry measurement. The scan matching measurement for the pose of a given scan can result in low fitness score, however, the matching might still be wrong. To detect those erroneous matching, the resulting velocity estimate from the optimization is compared to the velocity of the previous pose. If the difference is higher than a given threshold, the estimate is replaced by IMU preintegration prediction and the scan matching measurement is removed from the graph.

Fig. 2. The unmanned aerial vehicle platform with the 3D LiDAR and the external GNSS module

3. EXPERIMENT AND RESULTS

3.1 Experiment

The goal of the proposed method is to accurately determine the state of an Unmanned Aerial Vehicle (UAV) for a future development of autonomous navigation solution in seedling pine forest environment. In this study, the UAV is flown in a young pine forest located at Lohja which is a town and municipality in the southern region of Finland. Given that the young forest is bordered by big trees and few big trees can also be found at some places, the GNSS signal can suffer from temporarily signal degradation or signal loss that can lead to a crash of the vehicle. It is therefore necessary to add a new measurement for the state estimation of the UAV.

The DJI Matrice 100 UAV shown in Fig. 2. It is equipped with the Velodyne VLP16 light "PUCK" LiDAR, which is a 3D LiDAR with 360° horizontal field of view and 30° (−15° to 15°) vertical field of view. The GARMIN GPS 18x 5Hz is used as an external GNSS receiver instead of the proprietary GNSS receiver from DJI. This choice is motivated by the fact that in addition to the position and the velocity information, the horizontal dilution of precision (HDOP) and the vertical dilution of precision (VDOP) can be read from this external GNSS receiver. These values can significantly change in a forest environment. They are used in this study to approximate the position covariance of the GNSS receiver.

An external and relatively cheap Epson IMU is used to provide angular rate and acceleration data at about 200Hz. The latter could also be read from the flight controller unit (FCU). But reading these data at sufficiently high rate can overwhelm the FCU and have a negative impact on its primary function, which is to stabilize and control the motion of the vehicle.

A Raspberry Pi 4 Model B board is used as a companion computer to log data from the FCU, and the different external sensors. The Robot Operating System (ROS) is used for the communication between the different hardware, sensors and the software subsystems during the data collection. The software for the state estimation is developed using C++ language and the graph-based optimization algorithm is the ISAM2 algorithm. The GTSAM library is used in this study (Dellaert, 2012). The developed software is currently run on a laptop computer using only 3 CPU cores.

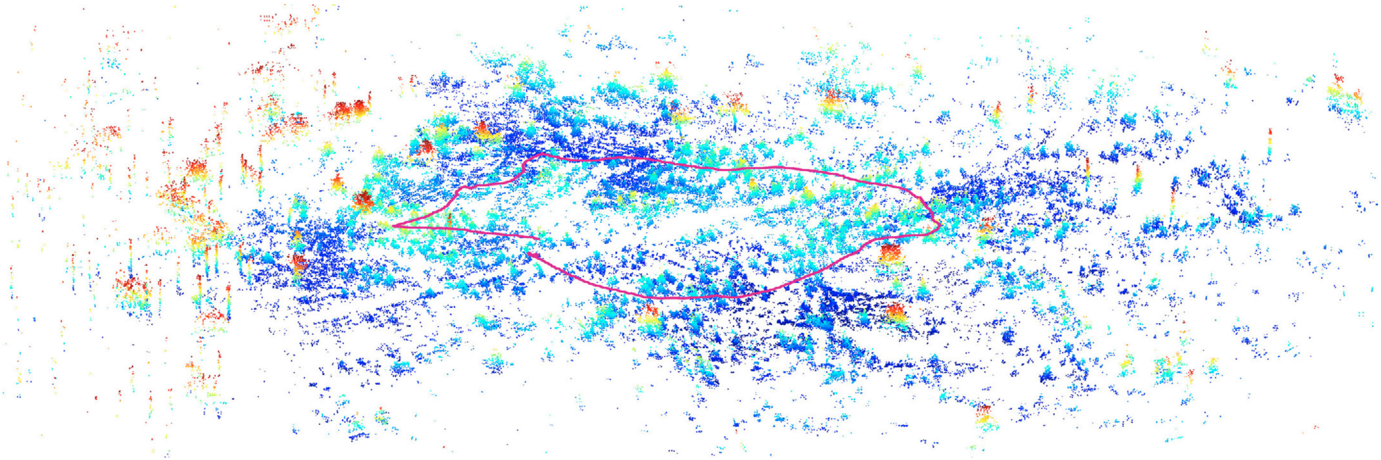

Fig. 3. A sample map of the environment obtained during the pose estimation along the *Trajectory 1*

3.2 Results

The proposed solution is tested on several trajectories each being a data recorded during a single flight of the UAV. We have not compared the proposed solution to the state of the art solutions because the available open source implementations that have been tried out fail to achieve coherent results and needed extensive parameters adjustment which is out of the scope of this work. We evaluate the proposed solution without the use of GNSS sensor on the entire trajectory. This is important to show how well the solution can cope with complete lost of GNSS signal. Fig. 4 and Fig. 5 show sample estimated trajectories by our solution (LiDAR+IMU) and a reference trajectory generated using a low cost RTK system. Figs. 6 and 7 show the corresponding velocity estimates. The estimated trajectory is the set of real-time estimated positions. The estimated error expressed as root mean square errors (RMSE) for the trajectories and the velocities are presented in table 1. The trajectory error is expressed using the absolute position error (APE) and the relative position error (RPE) which measure, respectively, the absolute and relative consistency of the estimated trajectory. The APE and RPE have been computed using the EVO toolbox Grupp (2017). The solution was also tested by injecting the GNSS data at the start of the trajectory and manually interrupting the GNSS signal during some time window to simulate the loss of GNSS signal. But the result did not show any significant difference with the ones presented above because the LiDAR odometry is already providing a better estimate and is not suffering from large drift. Therefore that result is not shown here. To appreciate the consistency of the position estimate, a sample of the local maps used during the state estimation are recorded and plotted. Fig. 3 shows a sample map of the environment. This map is only a concatenation of several local maps, it can therefore be considered as the real-time estimate of the map of the environment and not a full batch optimized map.

4. CONCLUSION AND FUTURE WORK

In this study, a solution for an unmanned aerial vehicle state estimation is proposed. This solution will be used in later studies for developing autonomous navigation solu-

Table 1. RMSE errors of trajectories and velocities

Trajectories	*Trajectory 1*	*Trajectory 2*
position error (APE)	0.2471	0.3123
position error (RPE)	0.1521	0.2646
X-axis velocity (m/s)	0.0945	0.1310
Y-axis velocity (m/s)	0.0794	0.1180
Z-axis velocity (m/s)	0.0706	0.0556

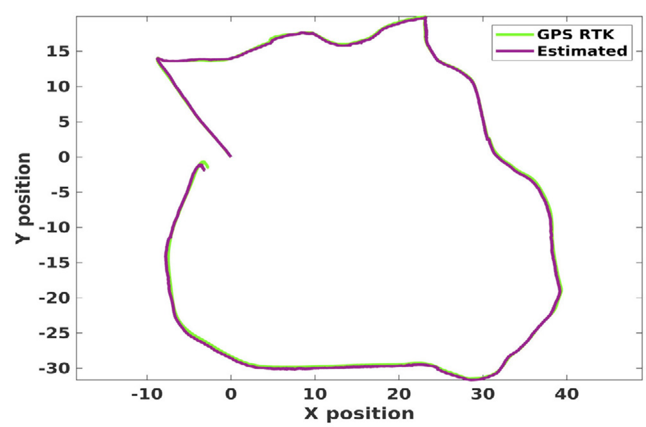

Fig. 4. The estimated *Trajectory 1*

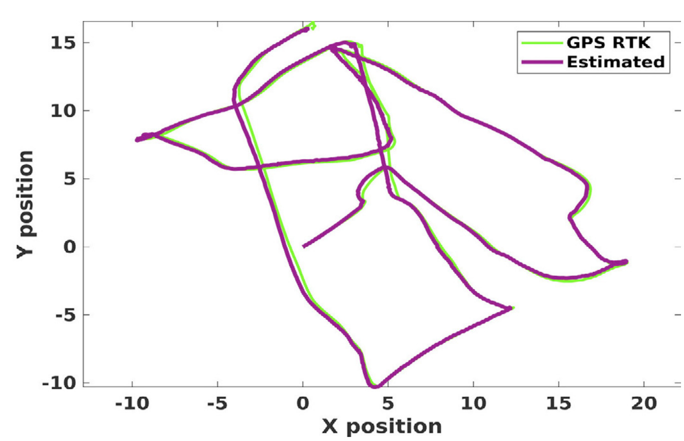

Fig. 5. The estimated *Trajectory 2*

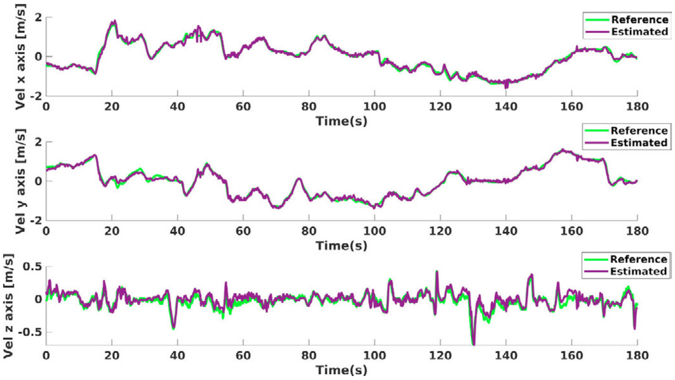

Fig. 6. Estimated velocity in x, y , and z directions for *Trajectory 1*. The reference velocity is the RTK GNSS reported velocity.

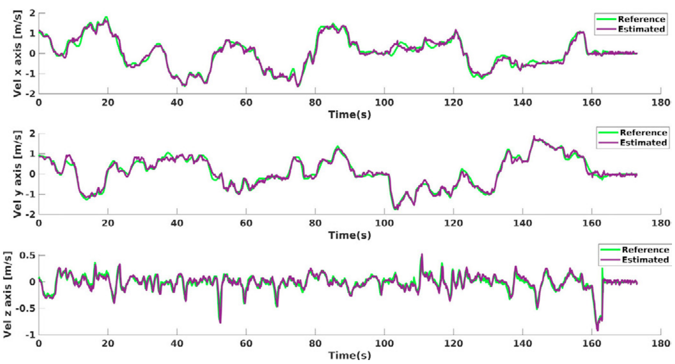

Fig. 7. Estimated velocity in x, y , and z directions for *Trajectory 2*. The reference velocity is the RTK GNSS reported velocity.

tion for precise and selective spraying of non-toxic moose repellent chemical in seedling pine forest environment. The proposed solution uses the notion of factor graph to fuse data from an IMU sensor, a sparse LiDAR sensor, and optionally a GNSS sensor. In this environment, tree trunks are not available to be used as features for the scan matching. The NDT scan matching is therefore used on point clouds which have been downsample using selected features of the environment. The proposed solution can effectively estimate the state of the vehicle including the trajectory and the velocity. The estimated trajectory has a good local consistency which is vital for control and obstacle avoidance. The global consistency of the estimated trajectory also shows an encouraging result which is as good as a typical GNSS-based position estimate. It has to be noted that the test flights that have been done are relatively short due to operational constraints and the hardware energy autonomy constraint. Longer flight tests will be performed to fully appreciate the proposed solution.

REFERENCES

Biber, P. and Strasser, W. (2003). The normal distributions transform: a new approach to laser scan matching. In *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, volume 3, 2743–2748 vol.3.

Chiella, A.C.B., Machado, H.N., Teixeira, B.O.S., and Pereira, G.A.S. (2019). Gnss/lidar-based navigation of an aerial robot in sparse forests. *Sensors*, 19(19). doi:10.3390/s19194061. URL `https://www.mdpi.com/1424-8220/19/19/4061`.

Cui, J.Q., Lai, S., Dong, X., Liu, P., Chen, B.M., and Lee, T.H. (2014). Autonomous navigation of uav in forest. In *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, 726–733.

Dellaert, F. (2012). Factor graphs and gtsam: A hands-on introduction.

Forster, C., Carlone, L., Dellaert, F., and Scaramuzza, D. (2016). On-manifold preintegration for real-time visual-inertial odometry. *IEEE Transactions on Robotics*, PP. doi:10.1109/TRO.2016.2597321.

Grupp, M. (2017). evo: Python package for the evaluation of odometry and slam. `https://github.com/MichaelGrupp/evo`.

Hening, S., Ippolito, C., Krishnakumar, K., Stepanyan, V., and Teodorescu, M. (2017). 3d lidar slam integration with gps/ins for uavs in urban gps-degraded environments.

Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J.J., and Dellaert, F. (2012). isam2: Incremental smoothing and mapping using the bayes tree. *The International Journal of Robotics Research*, 31(2), 216–235. doi:10.1177/0278364911430419. URL `https://doi.org/10.1177/0278364911430419`.

Koide, K., Miura, J., and Menegatti, E. (2019). A portable three-dimensional lidar-based system for long-term and wide-area people behavior measurement. *International Journal of Advanced Robotic Systems*, 16. doi:10.1177/1729881419841532.

Magnusson, M., Lilienthal, A., and Duckett, T. (2007). Scan registration for autonomous mining vehicles using 3d-ndt. *Journal of Field Robotics*, 24(10), 803–827. doi:10.1002/rob.20204. URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.20204`.

Nevalainen, S., Matala, J., Korhonen, K., Ihalainen, A., and Nikula, A. (2016). Moose damage in national forest inventories (1986-2008) in finland.

Shan, T., Englot, B., Meyers, D., Wang, W., Ratti, C., and Rus, D. (2020). Lio-sam: Tightly-coupled lidar inertial odometry via smoothing and mapping. *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 5135–5142.

Valsta, L. (2017). Overview of mixed forests in finland.

Zhong, Y. (2009). Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *2009 IEEE 12th International Conference on Computer Vision Workshops*, 689–696. doi:10.1109/ICCVW.2009.5457637.