Experimental assessment of IMU data processing techniques for a backpack mobile laser scanning system

Letícia F. Castanheiro [1], Antonio M. G. Tommaselli [1], Heikki Hyyti [2], Antero Kukko [2]

[1] Department of Cartography, São Paulo State University (UNESP) at Presidente Prudente, São Paulo, Brazil – (leticia.ferrari, a.tommaselli)@unesp.br
[2] Department of Remote Sensing and Photogrammetry, Finnish Geospatial Research Institute (FGI), National Land Survey of Finland, Espoo, Finland – (heikki.hyyti, antero.kukko)@nls.fi

Keywords: MEMS IMU, inertial navigation system, LiDAR, mobile mapping, local trajectory, SLAM.

Abstract

Positioning techniques are fundamental in many automation tasks with several applications. In GNSS-denied environments like in dense forests, other alternatives are required, such as inertial and visual navigation. However, Inertial Measurement Units (IMUs) data, mainly those from microelectromechanical-system (MEMS), are noisy, which affects the orientation estimation. MEMS IMUs have been employed in mobile laser scanning systems due to their compact design and low-cost solutions for short-term navigation. In this paper, we have compared three IMU processing techniques freely available: MAH (Mahony et al., 2009), MAD (Madgwick et al., 2011) and DCM (Hyyti and Visala, 2015). These techniques implemented different approaches to estimate the attitude. They were experimentally assessed with data from a backpack mobile laser scanning system, which is composed of an OS0-128 Ouster LiDAR equipped with an internal IMU. We have used data from a 5-second trajectory segment aiming to evaluate the attitude and position estimation for a local path. The results showed that the DCM algorithm maintained a consistent velocity for 5 seconds, achieving a positional error of 1.4 m, 0.06 m, and 1.05 m along the X-, Y- and Z-axis, respectively. In contrast, MAD and MAH showed a position error over 20 m, 7 m and 3 m along the X-, Y- and Z-axis, respectively, which was affected by the velocity drift.

1. Introduction

Accurate positioning is critical in many automation technologies, including robotics, autonomous vehicles, and precision agriculture (Aguiar et al., 2020). However, achieving suitable positioning in challenging environments, such as forestry, remains an issue. Traditional positioning systems, such as the Global Navigation Satellite System (GNSS) offer reliable georeferenced data, but in situations where there are multipath effects and signal blocking, their accuracy is frequently reduced. Kaartinen et al. (2015) showed that GNSS sensors achieved an accuracy of approximately 0.7 meters under the canopy, which is unsuitable for most tasks in smart agriculture and forest management.

Inertial Measurement Units (IMUs) have emerged as an alternative for positioning in such challenging scenarios. IMUs provide acceleration and angular velocity measurements, which can be used to estimate the platform’s attitude and position. One advantage of IMUs is that they do not rely on external signals, making them a better option in GNSS outage environments. In addition, IMUs have a high update rate, which is beneficial for dynamic applications. Therefore, the integration of IMUs with Simultaneous Localization and Mapping (SLAM) algorithms has become a focus of research and development (Tagliabue et al., 2021, Júnior et al., 2022, Kim et al., 2023, Faitli et al., 2023, Zhang et al., 2024).

SLAM algorithms are designed to incrementally generate a map of the environment while determining the pose of a mobile system (Cadena et al., 2016). The mobile system acquires data from sensors, such as digital cameras or LiDAR, mounted on the platform and the position and attitude are estimated by matching the detected features in the sensor data (e.g. images, point clouds) while moving the platform (Durrant-Whyte and Bailey, 2006). SLAM methods can be divided according to the sensor type. Visual SLAM (vSLAM) is based on sequential optical images while LiDAR SLAM uses point clouds to estimate the platform pose.

When combined with IMU data, SLAM algorithms, whether visual or LiDAR-based, benefit from high-frequency and continuous inertial measurements, which provide an initial trajectory estimation, thus improving the accuracy of localisation and mapping, mainly in GNSS-denied environments (Gao et al., 2024). This initial trajectory estimation can significantly improve SLAM performance in the matching step by eliminating incorrect correspondences and providing initial state estimation. The benefits of using IMU are still more evident for LiDAR data since the point clouds suffer from motion distortions and trajectory errors gradually accumulated, mainly in high-speed or complex motion, requiring complementary sensors to provide initial values of the platform’s attitude and positions or closed-loop correction in SLAM methods (Zhang et al., 2024).

However, estimating attitude and position through IMU measurements poses some challenges (Hyyti and Visala, 2015). One significant issue is the accumulation of errors over time, known as drift. Estimating heading angles in dynamic environments, where motion direction frequently changes, adds further complexity. In addition, abrupt acceleration and rapid changes in orientation can affect the IMU solution. Another challenge is the sensitivity of IMUs to external factors such as temperature variations and magnetic interference, which can introduce errors in acceleration and angular velocity measurements, requiring calibration procedures. Several approaches and techniques have been proposed to address and mitigate these limitations of IMUs.

This paper experimentally assesses the techniques presented by Mahony et al. (2009), Madgwick et al. (2011), and Hyyti and Visala (2015) for attitude and position estimation using IMU

measurements. These techniques, whose codes are freely available, are evaluated for their potential integration into SLAM algorithms. The assessment focuses on the accuracy of attitude and position estimation based solely on IMU data, highlighting both their potential benefits and limitations in mobile mapping systems.

2. IMU data processing

An IMU is composed of a triaxial accelerometer and a triaxial gyroscope. The accelerometers measure linear acceleration along the three orthogonal directions (X, Y, Z) while the gyroscopes provide angular velocity around the three axes (X, Y, Z), which are proportional to the platform’s rotation movements (Shan and Toth, 2008). By integrating these measurements, the IMU can be used to estimate position and attitude, which is the 3D orientation of the platform concerning the Earth coordinate system.

The orientation space can be parameterised by Euler angles, quaternions, and rotation matrices (Dam et al., 1998). Euler angles provide an intuitive representation of the rotations by specifying successive rotations around the X, Y, and Z axes and requiring only three values for representation (for instance, yaw, pitch and roll angles). Nevertheless, Euler angles can suffer from the gimbal lock problem, which is a loss of one degree of freedom caused by the alignment of two rotation axes. Furthermore, different axis conventions can cause ambiguity and uniqueness in representations of the same orientation. Quaternions offer several significant advantages for estimating and representing attitude (Dam et al., 1998). First, rotations defined by quaternions are independent of the coordinate system, making them more robust for a navigation system. Second, a critical issue in Euler angle rotations, the gimbal lock, is avoided with the four-dimensional representation of quaternions. Therefore, there are no singularities in rotating when using quaternions. Finally, quaternions are simpler to interpolate, enabling smooth transitions between rotations and efficient computations, which are important for real-time applications in navigation systems. Another option is the use of rotation matrices that provide a direct transformation in the 3D space, maintaining orthogonality. However, rotation matrices need more memory to store their nine values with a 3x3 matrix. In practical applications, the choice among Euler angles, quaternions, and rotation matrices depends on the requirements. Usually, IMU processing algorithms employ quaternions more frequently due to their robustness and efficiency, but Euler angles are also used in certain contexts (Dam et al., 1998).

Besides the choice of the rotation representation, one significant challenge concerning IMU measurements processing is the heading angle estimation, which is typically solved using an extra sensor such as a triaxial magnetometer or satellite navigation. Although magnetometers and GNSS can offer a good solution, they also present some challenges. For instance, the LiDAR system induces a magnetic field due to the electronics affecting magnetometer measurements, and thus the earth’s magnetic field cannot be easily separated. Moreover, as previously discussed, GNSS signals can be blocked or weakened when traversing through indoor and vegetated areas. These problems are more challenging with microelectromechanical-system (MEMS) IMUs, which are noisy, and the results from their measurements often drift for long-term navigation, posing challenges compared to high-precision IMUs. Nevertheless, MEMS IMUs have been widely used in navigation systems since they are small, lightweight, and low-cost solutions for short-term navigation. Consequently, methods have been proposed to estimate the attitude and position of a mobile platform using only linear accelerations and angular velocities. In this paper, we present and assess three approaches (Mahony et al., 2009; Madgwick et al., 2011; and Hyyti and Visala, 2015).

Mahony’s (Mahony et al., 2009) and Madgwick’s (Madgwick et al., 2011) methods are non-Kalman filter techniques based on quaternion representation for attitude angles avoiding the complexities and singularities associated with Euler angles. Mahony’s algorithm calculates the quaternion derivative by integrating the gyroscope angular rate data and updates the orientation estimation over time. The accelerometer data provide the direction of gravity, which is used to compensate for the drift from gyroscope measurements. The solution is based on the Special Orthogonal group SO(3), which is the underlying Lie group structure for the space of rotation matrices (Hall, 2015).

Madgwick’s algorithm is designed for a wearable inertial motion system that can be composed of a tri-axis gyroscope and accelerometers (IMU sensors), and Magnetic Angular Rate and Gravity (MARG) sensors. The MARG implementation combines gyroscope, accelerometer, and magnetometer data. This is done by adjusting the orientation estimate from the gyroscope data with corrections derived from the accelerometer and magnetometer data. In this work, we focused on Madgwick’s IMU-only implementation since our IMU is composed of only a gyroscope and accelerometer. Madgwick’s IMU algorithm allows accelerometer data to be used in an analytically derived and optimised gradient descent algorithm to compute the direction of the gyroscope measurement error as a quaternion derivative. First, the angular rate data from the gyroscopes are used to determine the quaternion derivative, which is the rate of change of the earth frame relative to the sensor frame. Then, the orientation is obtained by integrating this derivative over time. The accelerometer data are used to estimate the orientation relative to the direction of gravity. In the end, a gradient descent algorithm is employed to minimize the error between the measured and estimated orientations. This filter is better for high measurement frequencies (over 50Hz).

Hyyti and Visala (2015) presented an approach based on the extended Kalman filter (EKF) using Euler angles in the directions cosines matrix (DCM). EKF is the most common approach for attitude estimation, adapting the Kalman Filter (KF) for nonlinear systems. System dynamic is linearised using the first order of the Taylor series. The first-order linearisation can introduce significant errors in the mean and covariance of the state vector, especially in low-cost IMU that commonly include noises in the measurements leading to instability in the solution. However, it can provide an accurate attitude for local trajectory.

The technique presented by Hyyti and Visala (2015) was developed for low-cost MEMS IMUs, avoiding the need for additional sensors, such as magnetometers. The algorithm focuses on estimating roll and pitch angles and a minimal drift relative to the yaw angle. The direction of gravity is calculated by integrating the measured angular velocities using a partial DCM. The adaptative EKF adjusts measurement covariances dynamically, reducing errors caused by rapid nongravitational accelerations, which is common in mobile applications. Their approach also includes temperature calibration and an online bias estimator of gyroscope biases to handle large temporary accelerations and changes in sampling rate. Under challenging conditions, Hyyti’s and Visala’s approach was evaluated using two low-cost IMUs (MicroStrain Inertia-Link and a Spark-Fun 6DOF Digital IMU). The performance was compared to Mahony’s and Madgwick’s methods (Mahony et al., 2009,

Madgwick et al., 2011). Moreover, accurate reference measurements were obtained with the KUKA LWR 4 robot arm. The results showed that their method outperforms the others across different conditions, such as temperature changes and dynamic movements. The DCM achieved an RMSE of 1.57°, 0.56°, and 0.61° for yaw, pitch and roll angles.

Several methods for estimating attitude based on IMU data have been proposed in the literature and it is clear the challenges to maintain a consistent solution over time without additional sensors. For instance, the DCM algorithm does not provide an absolute heading (yaw angle) but rather a relative one, which may drift over time without external corrections. Hence, this solution can be used for local trajectory compensation in SLAM.

3. Experiments with a backpack mobile laser scanning system

3.1 Data acquisition

The experimental dataset was collected with a backpack mobile laser scanning system (MLS) (Figure 1). This backpack platform (Figure 1.a) is composed of an Ouster OS0-128 laser scanning system (Figure 1.b), equipped with an internal IMU (Figure 1.c). The internal IMU records data from three-axis gyroscopes and three-axis accelerometers providing the acceleration (in g units) and the angular velocity (in degrees per second) in the X, Y and Z axes at a frequency of 100 Hz. Additionally, the backpack platform includes a Dell OptiPlex 3070 computer placed inside the backpack for data recording and storing. More details about the backpack MLS and Ouster OS0-128 sensor can be found in Castanheiro et al. (2022) and Castanheiro et al. (2023).

The experimental dataset took place in an orange orchard, where the platform was carried by an operator while walking in a trajectory with a length of approximately 400 m (Figure 1.d). Aiming to evaluate the IMU processing algorithms for generating local trajectories as initial values, five (5) seconds of data were used in this study. Figure 1 shows (a) the developed backpack MLS, (b) an Ouster OS0-128 laser scanner, (c) its internal IMU, and (d) the test area.

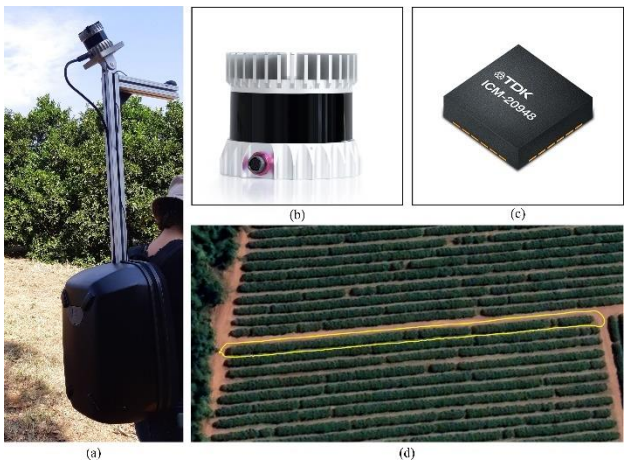

Figure 1. (a) the developed backpack platform, (b) the Ouster OS0-128 laser scanning system, (c) the internal IMU InvenSense, (d) the test area and the trajectory in yellow.

The Ouster sensor uses two coordinate systems: the sensor coordinate (depicted in green in Figure 2), and the laser coordinate (depicted in blue in Figure 2) systems. Both coordinate systems follow the right-hand convention (OUSTER, 2024). The sensor coordinate system is defined at the centre of the sensor on the bottom of the unit with X_S pointing opposite to the external connector, Y_S pointing to the left and Z_S pointing to the top of the sensor. The origin of the laser coordinate system is in the laser unit. The X-axis (X_L) is positive to the external connector, Y-axis (Y_L) turns the right-hand system, and Z-axis (Z_L) points to the top of the sensor (OUSTER, 2024).

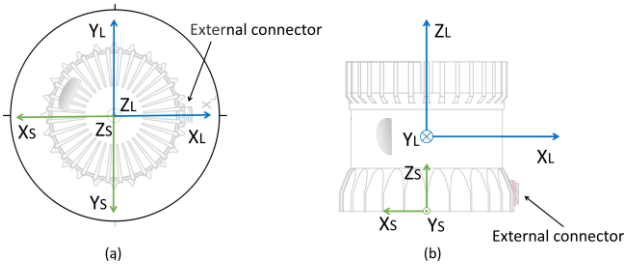

Figure 2. Both OS0-128 sensor (green) and laser (blue) coordinate systems in (a) the top view, and (b) the lateral view.

The transformation from the laser coordinate system to the sensor coordinate system is given by the homogeneous matrix (R_{LU}^{S}) presented in Equation (1). The origin is translated in the Z-axis and the X and Y axis are rotated 180° about the Z-axis, which corresponds to applying reflections to the X and Y axis (Figure 2). The Z translation is the height of the laser unit and the origin of the sensor coordinate system, which is 36.18 mm for the OS0-128 sensor.

$$R_{LU}^{S} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 36.18 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

The internal IMU is slightly displaced concerning the sensor coordinate system. Equation 2 presents the homogeneous transformation matrix from IMU to the sensor coordinate system, in which there are translations in the X, Y and Z-axis with the values 6.253 mm, -11.775 mm, and 7.645 mm, respectively (OUSTER, 2024).

$$R_{IMU}^{S} = \begin{bmatrix} 1 & 0 & 0 & 6.253 \\ 0 & 1 & 0 & -11.775 \\ 0 & 0 & 1 & 7.645 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

3.2 Accelerometer IMU calibration

MEMS accelerometers usually show some bias and gain errors. Therefore, it is important to calibrate the system before using the data in any attitude estimation algorithm to reduce the effects of these errors and achieve the best possible IMU performance. We used the approach for accelerometer calibration proposed by Won and Golnaraghi (2010). The IMU sensor was placed in six positions and held stationary. Figure 3 depicts the six (6) positions that the Ouster OS0-128 laser scanner was placed stationary to collect IMU data for the accelerometer calibration.

Measurements from these six positions were then used in the algorithm to iteratively optimize gains and biases for each axis of the accelerometer sensor (Won and Golnaraghi, 2010). Equation 3 shows the relation between the accelerometer outputs in each axis (S_{axis}) and the true acceleration in each axis (A_{axis}), where G_{axis} and B_{axis} are the true gain factor and bais, respectively, of

each axis. The estimated bias and gain factor were applied to the accelerometer data before processing in the IMU codes.

$$S_{axis} = G_{axis} \cdot A_{axis} + B_{axis} \quad (3)$$

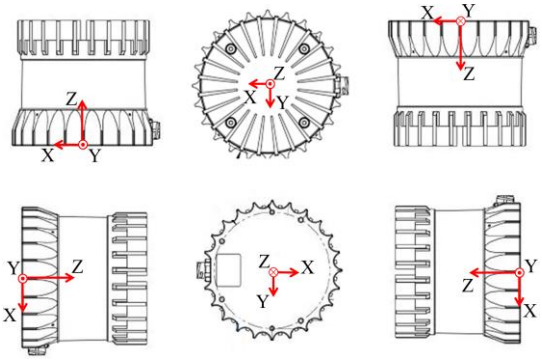

Figure 3. The six positions in which the Ouster laser scanner with the internal IMU was held stationary for calibration measurements.

3.3 Attitude and position estimation with IMU data

Three algorithms for attitude and position estimation based only on IMU data were compared. This algorithms are referred here as DCM (Hyyti and Visala, 2015), MAD (Madgwick et al., 2011), and MAH (Mahony et al., 2009). These techniques were implemented in MATLAB and C++. The source codes are available on GitHub, enabling further developments. Besides the attitude estimation, the DCM approach integrates temperature calibration and an online bias estimator, which was not employed in this work.

Each technique employs different approaches to estimate the attitude, as discussed in Section 2. In the end, the MAD and MAH algorithms deliver a list of quaternions, while the DCM method results in a list of roll, pitch and yaw (rpy) angles. Quaternions are easier to calculate and more efficient for inertial navigation systems, avoiding singularities. However, they are difficult to interpret physically compared to rpy angles. Therefore, the rotation matrix was firstly calculated using quaternions then rpy angles were extracted from the matrix. After orientation estimation, the positions were calculated in three steps. First, the translational accelerations were obtained using the previously computed rpy angles for each method. Then, these accelerations were integrated to obtain velocities and, finally, a subsequent integration was done to estimate the position.

The analyses were conducted by examining both the estimated velocities and positions aiming to obtain a short-term trajectory. The estimated velocities were evaluated considering the velocity acquisition, which was approximately 1.5 m/s. The positions were evaluated by comparing them with reference values obtained through the point cloud registration tool in CloudCompare (2016). This tool aligns and registers 3D point clouds with manual picking points, which was performed with five homologous points between the point clouds. To perform the comparison of the positions, the results obtained with the IMU-based algorithms were converted to the laser coordinate system, following Equation 2 and Equation 1, and then compared with the positions estimated with the CloudCompare tool.

3.4 Results and Discussions

The accelerometer calibration resulted in biases of 0.419, - 0.064, and -0.005 for the X, Y, and Z-axes, respectively, indicating a slight systematic error, mainly in the X-axis accelerometer. The gain factor for each axis was approximately 1, as expected. The accelerometer measurements were corrected from these biases and gains before estimating the attitude.

Figure 4 shows the computed translational velocities obtained with DCM, MAD, and MAH algorithms in (a) X, (b) Y, and (c) Z axes. The results for DCM are depicted using dash-dotted red lines, MAD results are shown with green lines and MAH results are represented by dashed blue lines.

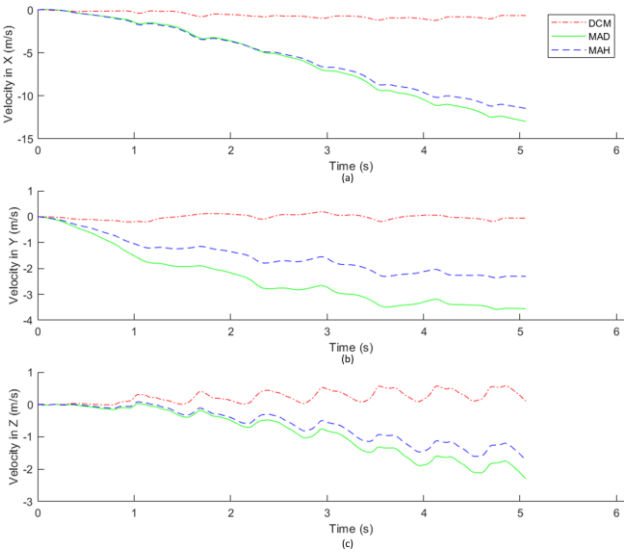

Figure 4. Computed translational velocities in (a) X-, (b) Y-, and (c) Z-axis for DCM, MAD, and MAH techniques.

The platform was maintained stationary in the first second, expecting a velocity of approximately zero. After that, the platform was carried at a velocity of approximately 1.5 m/s. Considering this situation, the DCM algorithm had a better performance maintaining more constant velocity across all axes while MAD and MAH algorithms drifted (Figure 4). The velocity along the X-axis, the main data acquisition direction, reached around 13 m/s with MAD and MAH methods (Figure 4.a). In contrast, the DCM algorithm resulted in a velocity of approximately 1.3 m/s on the X-axis, which is consistent with the expected acquisition velocity. The velocities in the Y- and Z-axis for the DCM method remained relatively stable, ranging from 0 to 0.5 m/s (Figure 4.b and Figure 4.c). Figure 5 shows, in more detail, the translational velocities calculated through the DCM solution. In Figure 5, red, green and blue lines represent the translational velocities obtained through the DCM algorithm in the X, Y, and Z axes respectively.

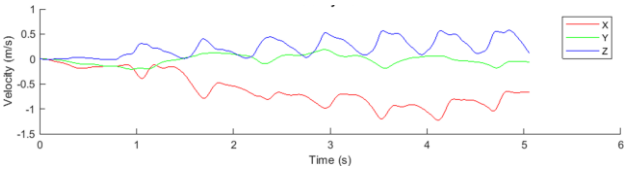

Figure 5. Translational velocities obtained with the DCM algorithm in the X (in red), Y (green), and Z (blue) axes.

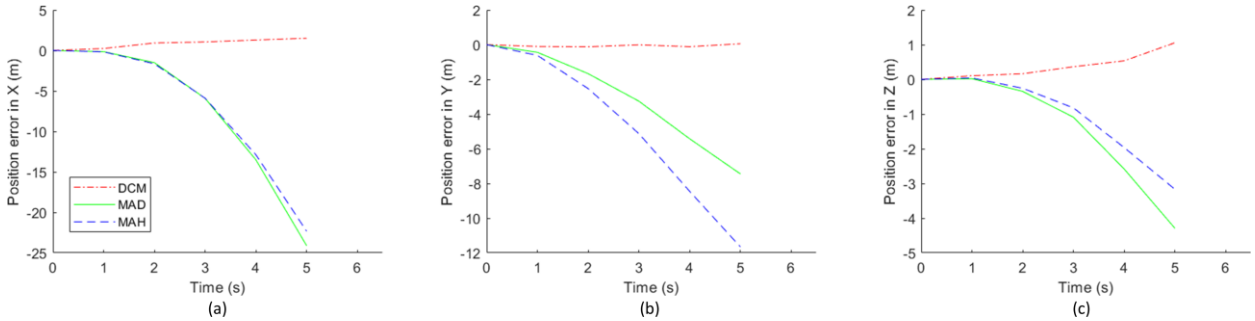

Figure 6. Positional errors obtained with DCM, in dash-dotted red lines, MAD, in green line, and MAH, in dashed blue lines, techniques along the (a) X, (b) Y, and (c) Z axes.

Figure 6 shows the positional errors (in meters) across the X, Y, and Z axes over the 5 seconds. The DCM positional errors are depicted using dash-dotted red lines. The MAD errors are represented by a solid green line, while the MAH algorithm’s errors are shown with dashed blue lines.

Table 1 shows the displacements in X, Y and Z obtained after 5 seconds for DCM, MAD and MAH’s algorithm solutions compared with the trajectory calculated from point cloud registration (PCR).

	DCM	MAD	MAH	PCR
X (m)	-3.10	-29.45	-27.59	-4.5
Y (m)	-0.08	-12.01	-7.73	-0.14
Z (m)	1.15	-4.33	-3.16	0.10

Table 1. Computed translational position for DCM, MAD and MAH’s solutions and point cloud registration (PCR) after 5 seconds.

Regarding the data acquisition, the large displacement in the X-axis for all methods was expected, since it is the main trajectory direction. However, MAD and MAH methods showed greater discrepancies in the position after 5 seconds when compared to PCR (Figure 6). This is consistent with the observed drift in their velocities (Figure 4). The DCM algorithm presented better performance, with displacement values close to those obtained with the PCR. This indicates that the DCM method better handles the drift over time. This is highlighted in Figure 6, which shows the positional error in (a) X, (b) Y and (c) Z axes, calculated for attitude angles (rpy) estimated with DCM, MAD and MAH algorithms.

The comparison in Table 1 and Figure 6 shows the lower positional errors provided by the DCM method. After 5 seconds, errors larger than 20 m along the X-axis were obtained with the MAD and MAH techniques, while an error of 1.4 m along the X-axis was achieved with the DCM method. The errors were less magnitude along the Y- and Z-axis for all methods, but still DCM presented lower discrepancies. The errors obtained with DCM are feasible for initial trajectory values in SLAM processing. Moreover, a shorter trajectory can be used to decrease positional errors.

The results suggest that the MAD and MAH algorithms may be useful for a short time trajectory, but their solutions drift more over time. The DCM method maintained the velocity more consistently along the 5-second trajectory, showing to be an attractive option for mobile applications.

4. Conclusion

There is a recognised benefit to using IMU measurements in LiDAR-SLAM algorithms. However, attitude estimation from IMU measurements is not straightforward due to the challenges posed by noisy IMU data. Therefore, our study presented three IMU measurement processing approaches (Mahony et al., 2009; Madgwick et al., 2011; Hyyti and Visala, 2015) and evaluated their performance in attitude and position estimation for local trajectory (5 seconds). The analysis showed the DCM (Hyyti and Visala, 2015) algorithm outperformed the MAD (Madgwick et al., 2011) and MAH (Mahony et al., 2009) algorithms in maintaining a consistent velocity during 5 seconds and, as a consequence, better position. Positional error analysis showed the consistency of the DCM algorithm, with acceptable error after 5 seconds, when compared to the position estimated with point cloud registration. In contrast, MAD and MAH showed a position error over 20 m, which was affected by the velocity drift. For shorter-term paths probably the position error will be smaller, which can be assessed in future.

While MEMS IMUs are known for their noise and drift over long-term navigation, our study showed that the DCM approach can still provide acceptable results for a local trajectory. Future works should focus on coupling the IMU measurements and refining these results in SLAM algorithms. The matching step can also be enhanced by using local trajectory to reduce space search and improve the feature correspondences. Additionally, this comparison can be extended to more complex navigation scenarios including turns and varying velocities.

ACKNOWLEDGEMENTS

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) - Finance Code 001 (Grant: 88887.310313/2018-00, 88887.695922/2022-00), by the National Council for Scientific and Technological Development, CNPq (grants n. 141550/2020-1 and 303670/2018-5) and by the São Paulo Research Foundation, FAPESP (Grant: 2021/06029-7), and was conducted during a mobility visit to UNITE Flagship, Finnish Geospatial Research Institute (Academy of Finland 337656) supported by grants “Understanding Wood Density Variation Within and Between Trees Using Multispectral Point Cloud Technologies and X-ray microdensitometry” (331708) and “Digital technologies, risk management solutions and tools for mitigating forest disturbances” (353264) by the Academy of Finland.

References

Aguiar, A.S., Dos Santos, F.N., Cunha, J.B., Sobreira, H. and Sousa, A.J., 2020. Localization and mapping for robots in

agriculture and forestry: A survey. *Robotics*, 9(4), p.97. https://doi.org/10.3390/robotics9040097

Cadena, C., Carlone, L., Carrillo, H., Latif, Y., Scaramuzza, D., Neira, J., Reid, I. and Leonard, J.J., 2016. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Transactions on robotics*, 32(6), pp.1309-1332. https://doi.org/10.1109/TRO.2016.2624754

Castanheiro, L.F., Tommaselli, A.M.G., Machado, M.V., Santos, G.H., Norberto, I.S. and Reis, T.T., 2022. The Use of a Wide Fov Laser Scanning System and a SLAM Algorithm for Mobile Applications. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, 43, pp.181-187. https://doi.org/10.5194/isprs-archives-XLIII-B1-2022-181-2022

Castanheiro, L.F., Tommaselli, A.M.G., Garcia, T.A.C., Campos, M.B. and Kukko, A., 2023. Point Cloud Registration Using Laser Data from an Orange Orchard. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* 48, pp.71-77. https://doi.org/10.5194/isprs-archives-XLVIII-1-W1-2023-71-2023

CloudCompare, 2016. GPL software, Version 2.12.4. cloudcompare.org.

Dam, E.B., Koch, M. and Lillholm, M., 1998. *Quaternions, interpolation and animation*. Copenhagen, Denmark: Datalogisk Institut, Københavns Universitet.

Durrant-Whyte, H. and Bailey, T., 2006. Simultaneous localization and mapping: part I. *IEEE robotics & automation magazine*, 13(2), 99-110. doi.org/10.1109/MRA.2006.1638022

Faitli, T., Hakala, T., Kaartinen, H., Hyyppä, J. and Kukko, A., 2023. Real-time lidar-inertial positioning and mapping for forestry automation. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, 48, pp.145-150. https://doi.org/10.5194/isprs-archives-XLVIII-1-W1-2023-145-2023

Gao, J., Sha, J., Li, H. and Wang, Y., 2024. A Robust and Fast GNSS-Inertial-LiDAR Odometry with INS-Centric Multiple Modalities by IESKF. *IEEE Transactions on Instrumentation and Measurement*, 73, p. 8501312. https://doi.org/10.1109/TIM.2024.3351253

Hall, B. C. (2015). *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction* (2nd ed.). Springer.

Hyyti, H. and Visala, A., 2015. A DCM Based Attitude Estimation Algorithm for Low-Cost MEMS IMUs. *International Journal of Navigation & Observation*, *2015*(1), p.503814. https://doi.org/10.1155/2015/503814

Júnior, G.P.C., Rezende, A.M., Miranda, V.R., Fernandes, R., Azpúrua, H., Neto, A.A., Pessin, G. and Freitas, G.M., 2022. EKF-LOAM: An adaptive fusion of LiDAR SLAM with wheel odometry and inertial data for confined spaces with few geometric features. *IEEE Transactions on Automation Science and Engineering*, 19(3), pp.1458-1471. https://doi.org/10.1109/TASE.2022.3169442

Kaartinen, H., Hyyppa, J., Vastaranta, M., Kukko, A., Jaakkola, A., Yu, X., Pyorala, J., Liang, X., Liu, J., Wang, Y., Kaijaluoto, R., Melkas, T., Holopainen, M., Hyyppa, H., 2015. Accuracy of Kinematic Positioning Using Global Satellite Navigation Systems under Forest Canopies. *Forests*, 6, 3218-3236, 6(9), 3218–3236. https://doi.org/10.3390/f6093218

Kim, B., Jung, C., Shim, D.H. and Agha-mohammadi, A.A., 2023. Adaptive keyframe generation based LiDAR inertial odometry for complex underground environments. *In 2023 IEEE international conference on robotics and automation (ICRA)*, pp. 3332–3338. https://doi.org/10.1109/ICRA48891.2023.10161207

Mahony, R., Hamel, T., Trumpf, J. and Lageman, C., 2009. Nonlinear attitude observers on SO (3) for complementary and compatible measurements: A theoretical study. *In Proceedings of the 48h IEEE Conference on Decision and Control (CDC)* held jointly with 2009 28th Chinese Control Conference (pp. 6407-6412). IEEE. https://doi.org/10.1109/CDC.2009.5399821

Madgwick, S.O., Harrison, A.J. and Vaidyanathan, R., 2011. Estimation of IMU and MARG orientation using a gradient descent algorithm. *In 2011 IEEE international conference on rehabilitation robotics* (pp. 1-7). IEEE. https://doi.org/10.1109/ICORR.2011.5975346

OUSTER. Ouster Firmware User Manual Firmware v3.1.0 for all Ouster sensors. San Francisco, 02 May 2024. Available in: https://ouster.com/downloads/ (30 June 2024).

Shan, J. and Toth, C.K., 2008. *Topographic laser ranging and scanning – Principles and Processing*. Taylor and Francis Group. 590 pp.

Tagliabue, A., Tordesillas, J., Cai, X., Santamaria-Navarro, A., How, J.P., Carlone, L. and Agha-mohammadi, A.A., 2021. Lion: Lidar-inertial observability-aware navigator for vision-denied environments. *In Experimental Robotics: The 17th International Symposium.* pp. 380–390. https://doi.org/10.1007/978-3-030-71151-1_34

Won, S.H.P. and Golnaraghi, F., 2009. A triaxial accelerometer calibration method using a mathematical model. *IEEE transactions on instrumentation and measurement*, 59(8), pp.2144-2153. https://doi.org/10.1109/TIM.2009.2031849

Zhang, Y., Shi, P. and Li, J., 2024. 3D LiDAR SLAM: A survey. *The Photogrammetric Record*, 39(186). https://doi.org/10.1111/phor.12497