

INVESTIGATING THE COMPLEMENTARY USE OF RADAR AND LIDAR FOR POSITIONING APPLICATIONS

E. Mounier^{1,2,*}, E. Dawson¹, M. Elhabiby³, M. Korenberg¹, A. Noureldin^{1,4}

¹ Queen's University, Kingston, ON, Canada - (eslam.abdelmoneem, emma.dawson, korenber, nourelda)@queensu.ca

² Ain Shams University, Cairo, Egypt

³ Micro Engineering Tech Inc., Alberta, Canada - elhabiby@meng-tech.com

⁴ Royal Military College, Kingston, ON, Canada

KEY WORDS: Positioning, LiDAR, Automotive Radars, Dynamic Objects Detection, Inertial Sensors.

ABSTRACT:

In the realm of Autonomous Vehicles (AVs), accurate, reliable and uninterrupted positioning capabilities are vital to ensure successful operations. Light Detection And Ranging (LiDAR) technology, capable of providing a high-fidelity 3D representation of the surrounding environment, has enabled numerous odometry-based positioning algorithms. These algorithms utilize a registration process to estimate relative motion from two successive 3D scans. However, the accuracy of the registration process can be compromised by the presence of dynamic objects, leading to significant translational and rotational deviations. On the other hand, Radar technology provides spatial and speed information. However, it is limited by spatial sparsity and susceptibility to noise. In this paper, we propose combining the complementary LiDAR and Electronic Scanning Radar (ESR) measurements, along with onboard motion sensors for improved navigation performance in complex and dynamic environments. This is achieved by employing a radar-based filtering mechanism that refines the LiDAR's point cloud mitigating the impact of dynamic objects. This results in a more robust registration process, which in turn enhances the LiDAR Inertial Odometry (LIO) solution. The proposed method was verified using real data collected from onboard motion sensors, a 3D LiDAR, and four ESRs from road tests conducted in downtown Calgary, Alberta, Canada. Our approach achieved an improved average horizontal positioning and heading RMSE of 0.43 meters and 0.25 degrees, respectively, compared to the 0.66 meters and 0.39 degrees observed with the standalone LIO solution. Moreover, submeter-level and lane-level accuracies were enhanced to 95% and 100% of the time, respectively, up from 85.7% and 94.9%.

1. INTRODUCTION

Autonomous Vehicles (AVs) necessitate accurate, reliable, and continuous positioning capabilities to ensure smooth and safe operations. Positioning and navigation services have long relied on the Global Navigation Satellite Systems (GNSS). However, GNSS may face numerous challenges such as signal attenuation, interference, multipath, and blockage, which may critically degrade its performance significantly, rendering it unable to maintain an accurate vehicle position (Elsheikh and Noureldin, 2020). Traditionally, Dead Reckoning (DR) algorithms employing onboard motion sensors such as Inertial Measurement Units (IMUs) and vehicle odometers have been employed to provide navigation information bridging GNSS limitations (Noureldin et al., 2013). Despite their effectiveness in the short-term, these algorithms, such as the Inertial Navigation System (INS), can accumulate errors from the integration of motion sensor measurements, particularly when using low-cost Micro-Electro-Mechanical Systems (MEMS) technology (Elsheikh and Noureldin, 2020).

Contemporary AVs are equipped with perception sensors, such as cameras, Radio Detection and Ranging (RADAR), and Light Detection and Ranging (LiDAR) sensors. These sensors are primarily employed for perception-related tasks such as object detection and context awareness (Marti et al., 2019). However, their availability has revolutionized a new generation of perception-based positioning and navigation algorithms (El-Sheimy and Li, 2021).

Among these perception technologies, LiDAR stands out as an exceptionally powerful system offering an unrivalled high-fidelity 3D representation of the surrounding environment in the form of a point cloud. Such capabilities enabled the development of a wide range of robust LiDAR-based positioning algorithms. Specifically, LiDAR Odometry (LO) methods, have been widely studied to estimate relative motion from two successive 3D scans employing a registration process (Vizzo et al., 2023). Moreover, LiDAR Inertial Odometry (LIO) utilizes complementary measurements from IMU to aid the registration process thus providing a more robust pose estimation (Xu and Zhang, 2021). Nevertheless, the accuracy of these methods can be affected by various challenges such as the presence of dynamic objects in the scene, which can compromise the registration process leading to significant translational and rotational deviations, thus degrading the positioning solution.

On the other hand, Radar technologies, particularly Electronic Scanning Radars (ESRs), are now becoming common in contemporary vehicles, greatly enhancing features such as Adaptive Cruise Control (ACC) and Advanced Driver-Assistance Systems (ADAS). Furthermore, their ability to provide valuable spatial and speed information about the surrounding environment has motivated interest in the positioning and navigation fields (Dawson et al., 2022). Despite the advantages, ESR typically has limitations in terms of sparse spatial information and susceptibility to noise, which can constrain the effectiveness of the positioning algorithms.

In this paper, motivated by the complementary characteristics of

* Corresponding author

LiDAR and ESR measurements, we aim to investigate efficient ways to combine their information to provide an accurate point cloud from LiDAR augmented with speed information from ESR. The enhanced point cloud is to be utilized by an LIO system to mitigate the impact of dynamic objects, resulting in an accurate and reliable estimation of navigation states including position, velocity, and attitude.

2. METHODOLOGY

The proposed system utilizes information from onboard motion sensors, LiDAR, and ESRs for the goal of improving navigation accuracy and robustness in challenging environments with highly dynamic surroundings and prolonged GNSS outages. The block diagram in Figure 1 illustrates the interconnections between the system modules. Firstly, the mechanization module utilizes onboard motion sensors to compute initial navigation states. Secondly, the radar-based filtering module identifies and filters out dynamic objects from the LiDAR point cloud. Lastly, the odometry module uses the refined point cloud aligning it with previous LiDAR scans, producing improved position and attitude estimates. The system modules and their interaction are thoroughly explained in the following subsections.

2.1 Mechanization

In this work, we employ the Vehicle Sensors Dead Reckoning (VSDR) algorithm, which we previously introduced in (Mounier et al., 2022) for navigation states estimation. The VSDR algorithm utilizes data from the IMU, including accelerations and angular velocities, in conjunction with the vehicle's forward velocity obtained from the odometer. At its core, the VSDR algorithm relies on the Inertial Navigation System (INS) mechanization process to compute navigation states, including position, velocity, and attitude in an East-North-Up (ENU) local-level frame of navigation (l -frame) (Noureldin et al., 2013). The vehicle's position is represented in geodetic coordinates in terms of latitude (φ), longitude (λ), and altitude (h), denoted as $\mathbf{r}^l = [\varphi, \lambda, h]^T$. The vehicle's velocity in the ENU directions is denoted as $\mathbf{v}^l = [v_e, v_n, v_u]^T$. Attitude, represented by Euler angles for pitch (p), roll (r), and azimuth (Az), orients the vehicle's body frame (b -frame) to the l -frame via the rotation matrix $R_b^l = [R_z(-Az)R_x(p)R_y(r)]$, where $R_z(-Az)$ represents rotation around the z -axis (upward-axis) by the azimuth angle Az , $R_x(p)$ represents rotation around the x -axis (transverse-axis) by pitch angle p , and $R_y(r)$ represents rotation around the y -axis (forward-axis) by roll angle r .

The standard INS mechanization equations can be described by a set of continuous-time differential equations that express the

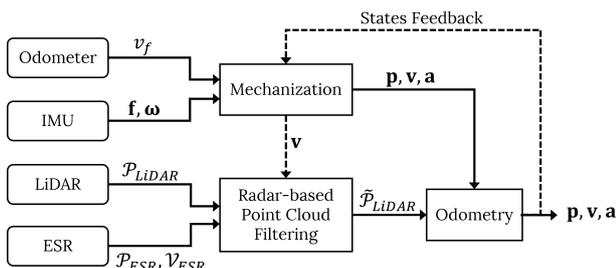


Figure 1. A block diagram depicting the proposed method.

rate of change of the navigation states:

$$\begin{bmatrix} \dot{\mathbf{r}}^l \\ \dot{\mathbf{v}}^l \\ \dot{R}_b^l \end{bmatrix} = \begin{bmatrix} D^{-1}\mathbf{v}^l \\ R_b^l \mathbf{f}^b - (2\Omega_{ie}^l + \Omega_{el}^l)\mathbf{v}^l + \mathbf{g}^l \\ R_b^l (\Omega_{ib}^b - \Omega_{il}^b) \end{bmatrix} \quad (1)$$

where, Ω_{ib}^b is the skew-symmetric matrix of the angular velocities $\boldsymbol{\omega}^b = [\omega_x, \omega_y, \omega_z]^T$ measured by the gyroscopes in the b -frame, and Ω_{il}^b is a skew-symmetric matrix that accounts for the Earth's rotation rate and the orientation change due to the vehicle's motion. It should be noted that the quaternion approach is utilized, due to its computational efficiency and lack of singularities, to solve for R_b^l (Rogers, 2003). The vector $\mathbf{f}^b = [f_x, f_y, f_z]^T$ represents the measurements from the IMU accelerometers in the b -frame, and $\mathbf{g}^l = [0, 0, -g]^T$ is the gravity compensation vector. The term $2(\Omega_{ie}^l + \Omega_{el}^l)\mathbf{v}^l$ is a correction term compensating for the rate of change of velocity due to the stationary and non-stationary rotation components caused by Earth's rotation and the vehicle's motion, respectively. The transformation matrix D^{-1} scales the velocity vector in the l -frame by the meridian (R_M) and normal (R_N) Earth radii to geodetic coordinates, and it can be expressed as:

$$D^{-1} = \begin{bmatrix} 0 & \frac{1}{R_M+h} & 0 \\ \frac{1}{(R_N+h)\cos\varphi} & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2)$$

The VSDR algorithm extends the standard INS algorithm by leveraging the fact that the motion of a land vehicle is constrained to be in the forward direction, with no vertical or transverse motion (Georgy et al., 2009). This non-holonomic constraint can be enforced by utilizing forward speed measurements v_f from the vehicle odometer (when available) projected to the l -frame according to Equation (3), thus minimizing the velocity mechanization errors.

$$\mathbf{v}^l = \begin{bmatrix} v_e \\ v_n \\ v_u \end{bmatrix} = R_b^l \begin{bmatrix} 0 \\ v_f \\ 0 \end{bmatrix} = \begin{bmatrix} v_f \sin(Az) \cos(p) \\ v_f \cos(Az) \cos(p) \\ v_f \sin(p) \end{bmatrix} \quad (3)$$

2.2 Radar-based Filtering

The radar-based filtering process aims to eliminate points associated with dynamic objects from the LiDAR point cloud, by leveraging measurements from ESRs, thereby enhancing the registration process. First, ESRs scan the environment and obtain measurements such as range, azimuth, elevation, and radial speed for detected targets. For each ESR, the range, azimuth, and elevation information are projected from the spherical to Cartesian coordinates, forming a 3D point cloud. For an effective filtering process, the point cloud of the j^{th} ESR denoted as \mathcal{P}_{r_j} is transformed to the LiDAR sensor frame (s -frame). Specifically, the coordinates of the i^{th} target in the LiDAR s -frame, denoted as \mathbf{r}_i^s can be computed as:

$$\mathbf{r}_i^s = \mathbf{r}_{r_j}^s + R_{r_j}^s \mathbf{r}_i^{r_j} \quad (4)$$

where, $\mathbf{r}_{r_j}^s$ and $R_{r_j}^s$ represent the translation and rotation calibration parameters between the j^{th} ESR and the LiDAR, respectively, while $\mathbf{r}_i^{r_j}$ denotes the coordinate of the i^{th} target in the j^{th} ESR frame.

By leveraging the latest velocity estimate \mathbf{v}^l from the VSDR algorithm described in subsection 2.1, the velocity of each ESR, denoted as \mathbf{v}^{rj} , can be estimated in its own frame using Equation (5) as:

$$\mathbf{v}^{rj} = R_s^{rj} R_b^s R_l^b \mathbf{v}^l \quad (5)$$

where, R_l^b represents the rotation from the navigation l -frame to the vehicle's b -frame, R_b^s represents the rotation from the b -frame to the LiDAR s -frame, and R_s^{rj} represents the rotation from the s -frame to the j^{th} ESR frame. Following that, and based on the estimated ESR velocity, assuming that all detected targets are static, an estimation of the radial speed for the i^{th} target denoted as \hat{v}_i can be computed as:

$$\hat{v}_i = -\mathbf{v}^{rj} \cdot \begin{bmatrix} \cos(\phi_i) \cos(\theta_i) \\ \cos(\phi_i) \sin(\theta_i) \\ \sin(\phi_i) \end{bmatrix} \quad (6)$$

where ϕ_i and θ_i represent the elevation and azimuth angle of the i^{th} target with respect to the j^{th} ESR.

To identify dynamic targets, a Speed Discrepancy Index (SDI) is introduced, which can be computed for the i^{th} ESR target as:

$$SDI_i = |v_i - \hat{v}_i| \quad (7)$$

where v_i and \hat{v}_i represent the actual and estimated radial speeds respectively. Targets with high SDI values are identified as potentially dynamic, indicating a violation of the static target assumption. By applying a suitable threshold to the SDI values, targets with the highest potential to be dynamic are retained. The dynamic points identified based on the SDI values are spatially clustered in LiDAR s -frame using the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm (Schubert et al., 2017). Each obtained cluster represents a dynamic object, while clusters with a minimal number of points are considered outliers and rejected.

Finally, the dynamic objects represented by the cluster centroid coordinates are used to filter the LiDAR point cloud \mathcal{P}_s . This is achieved by constructing a KD-tree and performing a Radius Nearest Neighbour Search (RNNS) step to associate the centroids of the dynamic objects with the points in \mathcal{P}_s . The associated points are then filtered out, resulting in a refined LiDAR point cloud denoted as $\tilde{\mathcal{P}}_s$ which can be used by the LIO algorithm for enhanced performance.

2.3 LiDAR Inertial Odometry (LIO)

The LIO module utilizes the VSDR inertial-based estimations as pose (position and attitude) initialization for the odometry process between successive LiDAR scans. A registration algorithm is then employed yielding a transformation matrix, which if applied would correct the initial VSDR pose estimates.

Upon obtaining a LiDAR point cloud, it undergoes several preprocessing steps inspired by our previous work in (Mounier et al., 2022). Noisy points beyond the scanning range or with attenuated laser returns are filtered out. Outliers are further eliminated by analyzing the standard deviations of the mean distances of the neighbouring points. Spatial cropping is performed confining the point cloud within predefined limits. Additionally, a voxel grid filtering is applied to the point cloud to decrease the number of points thus reducing the computational requirements for subsequent steps. Next, the preprocessed LiDAR point cloud is transformed from

the LiDAR s -frame into the local navigation l -frame. This transformation can be described for the i^{th} point in the LiDAR's point cloud as:

$$\mathbf{r}_i^l = \mathbf{r}_b^l + R_b^l(\mathbf{r}_s^b + R_s^b \mathbf{r}_i^s) \quad (8)$$

where \mathbf{r}_i^l represents the coordinates vector of point i in the l -frame, \mathbf{r}_b^l and R_b^l represents the coordinates vector and the rotation matrix of the vehicle position b with respect to the l -frame initially obtained from VSDR. \mathbf{r}_s^b and R_s^b are the calibration parameters (lever-arm and rotation matrix) of the LiDAR sensor s with respect to the b -frame. \mathbf{r}_i^s represents the coordinates vector of the i^{th} point in the s -frame.

With the transformed point cloud as the source cloud, it is matched with the point cloud from the previous LiDAR scan (target cloud) using the Iterative Closest Point (ICP) algorithm (Besl and McKay, 1992). The ICP algorithm estimates a 4×4 transformation matrix \hat{T} expressed as:

$$\hat{T} = \begin{bmatrix} \hat{R} & \hat{\mathbf{t}} \\ 0 & 1 \end{bmatrix} \quad (9)$$

where the translation vector $\hat{\mathbf{t}} = [\hat{t}_x, \hat{t}_y, \hat{t}_z]^T$ and rotation matrix $\hat{R} = [\hat{R}_z(\theta_z) \hat{R}_x(\theta_x) \hat{R}_y(\theta_y)]$ if applied will tightly align the two point clouds. The ICP algorithm iterates over two steps to optimize \hat{T} . First, it finds a correspondence set $K = (\mathbf{p}, \mathbf{q})$ from the target and source point clouds respectively. The transformation matrix \hat{T} is then updated by minimizing the correspondence set objective function. In this work, we use the ICP algorithm from the open-source *Open3D* python library by (Zhou et al., 2018), where the objective function can be expressed as:

$$E(\hat{T}) = \sum_{(\mathbf{p}, \mathbf{q}) \in K} (\mathbf{p} - \hat{T}\mathbf{q})^2 \quad (10)$$

Since all point clouds are pre-transformed to the l -frame, the output of the registration process is a corrective transformation matrix also in the l -frame. Applying the obtained transformation will provide a corrected position and attitude navigation states. To ensure the robustness of the odometry corrections, the registration results are evaluated based on the size of the correspondence set K and the Root Mean Squared Error (RMSE) of the point-to-point distance within the correspondence set. Corrections are rejected if either the RMSE or the size of the correspondence set does not fall within specified thresholds.

3. EXPERIMENTS

To evaluate the effectiveness of the proposed method, the Navigation and Instrumentation (NavINST) research lab's multi-sensor experimental vehicular setup depicted in Figure 2 was used. The setup comprises a 3D Velodyne Puck LITE LiDAR equipped with 16 laser pulse projector channels, capturing a 360-degree horizontal field of view and a 30-degree vertical field of view at a rate of 10 Hz. Four automotive-grade ESRs are mounted at the vehicle's outer corners, offering a 130-degree horizontal field of view and a 15-degree vertical field of view, and providing measurements of target elevation, azimuth, and radial velocity at approximately 20 Hz. The low-cost MEMS-based IMU from the Zed-2i front-mounted camera provides accelerometer and gyroscope measurements at

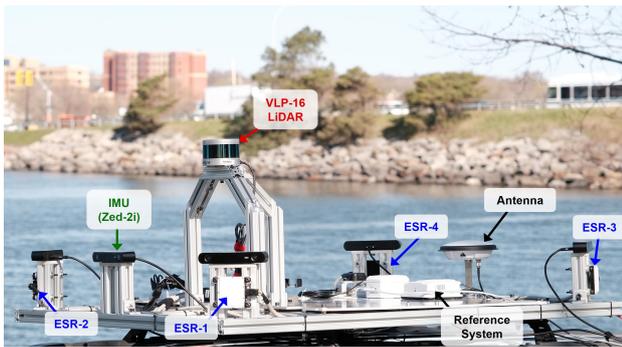


Figure 2. The NavINST multi-sensor vehicle setup highlighting the primary sensors.

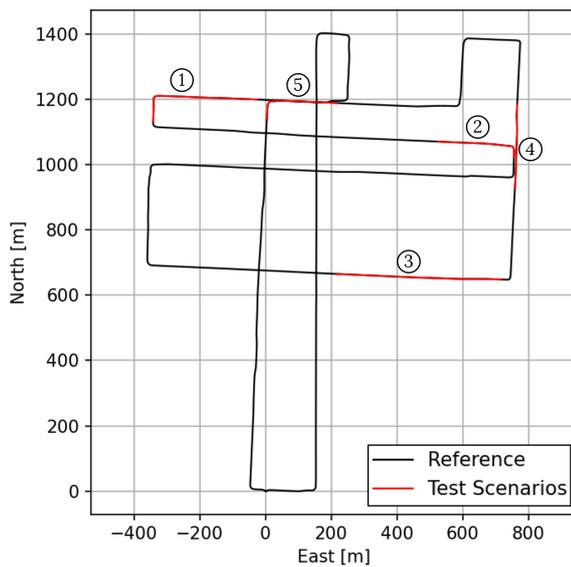


Figure 3. The test trajectory visualized in black with the test scenarios highlighted in red.

a rate of 50 Hz. Forward speed from the vehicle's odometer is obtained using an On-Board Diagnostics version II (OBD II) module at 3 Hz. The setup has a Novatel PwrPak7-E1 reference system, which combines an OEM7 SPAN GNSS receiver with a high-precision KVH-1750 IMU, enabling the generation of a high-accuracy navigation reference solution at 50 Hz through the incorporation of corrections from RTK base stations.

The experimental setup utilizes the Robot Operating System (ROS) for synchronized data recording (Quigley et al., 2009). Extrinsic sensors calibration steps were performed to estimate calibration transformation matrices (translations and rotations) for accurate measurement transformation across different sensors. Initial calibration parameters were obtained from the system's mechanical design and rough measurements, then further refined following the work in (Domhof et al., 2019). A real-road test was conducted to collect multi-sensor information from the aforementioned sensors in downtown Calgary, Alberta, Canada as shown in Figure 3. The test trajectory spanned approximately 35 minutes and covered a total distance of 9.1 km.

4. RESULTS

The performance evaluation of the proposed method focuses on scenarios with multiple dynamic objects present in the LiDAR scene, which can potentially hinder the registration process and degrade odometry corrections, thus the overall navigation solution especially during the absence of position corrections during GNSS outages. Five test scenarios were selected based on the analysis of the test trajectory data, where the presence of several dynamic objects allowed for thorough testing of the proposed method. These scenarios, highlighted in red on the test trajectory in Figure 3, spanned a duration of 3.75 minutes and covered a total distance of 1.67 km.

In each test scenario, the VSDR algorithm described in subsection 2.1 was employed to obtain a navigation solution using the low-cost IMU measurements and odometer speed updates. The LIO module utilized the initial navigation estimates from VSDR to assist the registration process by transforming the current point cloud and providing an initial guess for the registration algorithm. The transformation is then refined via the ICP registration process aligning the current and previous LiDAR point clouds, resulting in corrected position and attitude navigation states. We refer to this solution as the LIO solution. Similarly, another LIO solution was executed, but it utilized a LiDAR point cloud filtered based on the radar-based filtering module described in subsection 2.2. This filtered solution is denoted as RF-LIO. Both the LIO and RF-LIO solutions were obtained at a rate of 50 Hz, and their performance was compared to the reference solution obtained from the Novatel system to evaluate the benefits achieved when utilizing RF-LIO compared to the standalone LIO solution.

The effectiveness of the radar-based filtering process is qualitatively demonstrated in Figure 4. Figure 4a showcases the LiDAR point cloud in black, with the point cloud from the four ESRs transformed into the LiDAR frame and colour-coded based on the Speed Discrepancy Index (SDI). The ESR point clouds are filtered based on an SDI threshold, followed by a DBSCAN clustering process, resulting in the identification of dynamic objects represented by coloured markers, as shown in Figure 4b. Finally, the dynamic objects centroids are fed to a KD-tree-based search algorithm to find the corresponding dynamic points from the LiDAR point clouds highlighted in red in Figure 4c.

For the sake of comparison, test scenario 5 was selected to demonstrate the navigation performance of RF-LIO versus LIO solutions as shown in Figure 5. Both solutions closely resemble the reference trajectory; however, closer examination in Figures 5b and 5c reveals that the RF-LIO solution exhibits fewer drifting errors and closely follows the true trajectory, especially towards the end of the scenario. This is further supported by Figure 6, which highlights the performance difference between the RF-LIO and LIO solutions compared to the reference solution in terms of horizontal position errors, with an RMSE of 0.27 meters for RF-LIO and 0.53 meters for LIO. Additionally, the horizontal heading (azimuth) shows fewer drifting errors in the RF-LIO solution with an RMSE of 0.21 degrees, compared to 0.39 degrees for the LIO solution.

The comprehensive analysis of all test scenarios consistently demonstrated the significant positive impact of the radar-based filtering stage on enhancing the robustness of the LIO solution in the presence of dynamic objects. The thorough performance evaluation, summarized in Table 1, validated several key

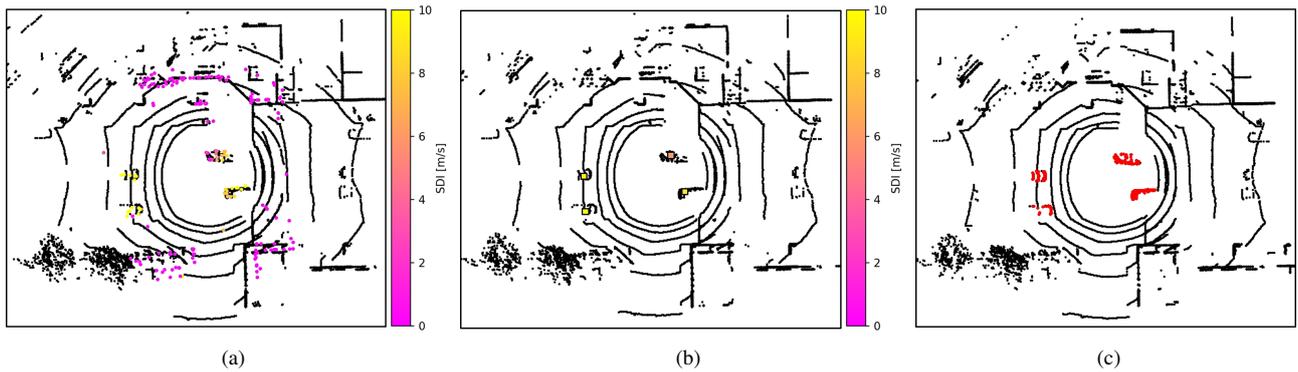


Figure 4. The stages of the Radar-based LiDAR point cloud filtering process: (a) LiDAR point cloud in black and ESRs point cloud colour-coded based on SDI. (b) Dynamic objects identified through SDI thresholding and DBSCAN clustering. (c) Extracted dynamic object points highlighted in red using the KD-tree RNNs algorithm.

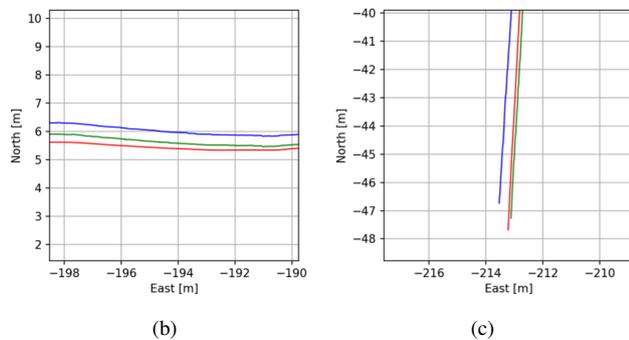
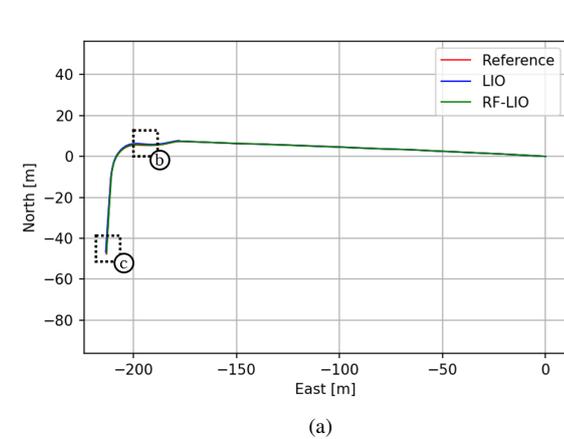


Figure 5. (a) Test scenario comparing the 2D navigation performance. Close-up views of the 2D trajectory are shown in (b) and (c).

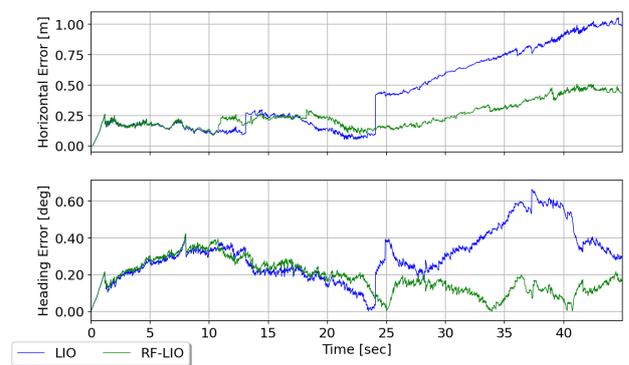


Figure 6. Performance comparison highlights.

observations. Firstly, the proposed method achieved notable enhancements in horizontal positioning accuracy, yielding an average RMSE of 0.43 compared to the standalone LIO solution. Additionally, the submeter and lane-level positioning statistics exhibited considerable improvements, increasing from averages of 86% and 95% to 95% and 100%, respectively. Moreover, the proposed method achieved enhanced heading accuracy, with an average error of 0.25 degrees compared to 0.39 degrees. These improvements are particularly valuable in scenarios with prolonged GNSS outages. However, it is important to acknowledge that the vertical statistics exhibited less consistency, primarily due to the inherent challenges associated with effectively removing dynamic objects while accurately preserving the points representing the ground plane. As a result, these challenges can impact the accuracy of vertical registration estimations, including pitch, roll, and height corrections, leading to increased susceptibility to variations.

Test Scenario #	1		2		3		4		5		Average	
Distance traveled [m]	389		260		501		257		259		333	
	LIO	RF-LIO	LIO	RF-LIO	LIO	RF-LIO	LIO	RF-LIO	LIO	RF-LIO	LIO	RF-LIO
Horizontal RMSE [m]	1.12	0.68	0.38	0.28	0.77	0.46	0.52	0.44	0.53	0.27	0.66	0.43
Vertical RMSE [m]	0.56	0.73	1.34	1.16	0.14	0.10	1.03	0.28	0.74	1.05	0.76	0.66
Horizontal distance traveled error [%]	0.29	0.18	0.15	0.11	0.15	0.09	0.20	0.17	0.20	0.10	0.20	0.13
Sub-meter-level accuracy [%]	70.80	80.10	100	100	71.80	100	88.97	95.24	97.00	100	85.71	95.07
Lane-level (< 1.5 m) accuracy [%]	74.60	100	100	100	100	100	100	100	100	100	94.92	100
Pitch RMSE [deg]	0.56	0.47	0.39	0.31	0.06	0.08	0.59	0.32	0.83	1.00	0.49	0.44
Roll RMSE [deg]	0.10	0.16	0.10	0.09	0.34	0.33	0.08	0.12	0.77	0.90	0.28	0.32
Azimuth RMSE [deg]	0.85	0.26	0.24	0.23	0.27	0.31	0.26	0.22	0.33	0.21	0.39	0.25

Table 1. Statistical analysis of pose estimation errors.

5. CONCLUSION

This paper has investigated the performance and benefits of involving Electronic Scanning Radar (ESR) measurements with LiDAR to enhance the positioning performance of land vehicles in challenging urban environments with GNSS outages and dense and dynamic surroundings. The proposed method utilizes the Vehicle Sensors Dead Reckoning (VSDR) algorithm computing initial navigation states, a radar-based filtering module effectively identifying and filtering out dynamic objects from the LiDAR point cloud, based on Speed Discrepancy Index (SDI) analysis. The refined point cloud is then utilized by the LiDAR Inertial Odometry (LIO) algorithm to align it with previous scans, providing precise position and attitude corrections to minimize the mechanization drifting errors.

To verify the effectiveness of the proposed method, a real road test was conducted in downtown areas of Calgary, utilizing actual measurements from onboard motion sensors, four ESRs, and LiDAR. The test scenarios encompassed various dynamic surroundings scenarios, reflecting realistic urban driving conditions. Our results demonstrated the effectiveness of the proposed method in extracting and filtering dynamic objects from the LiDAR point cloud, resulting in enhanced registration and improved LIO positioning accuracy in densely dynamic scenarios. Specifically, the incorporation of ESR measurements has led to enhanced horizontal positioning accuracy and increased sustainability of submeter and lane-level positioning accuracies compared to the standalone LIO solution.

In the future, several strategies can be explored in order to achieve further improvements. This includes investigating advanced sensor fusion techniques such as Extended Kalman Filtering (EKF) and Unscented Kalman Filtering (UKF) to optimally integrate information from multiple measurement sources, thereby providing an optimally integrated navigation solution. Additionally, the radial speed information from static detected targets from ESR can be utilized to estimate and refine the vehicle's velocity which can further minimize mechanization drifting velocity errors, leading to enhanced overall system accuracy. Such improvements, if optimally combined, would lead to an enhanced positioning system capable of sustaining decimeter-level positioning accuracy during prolonged GNSS outages.

ACKNOWLEDGMENT

This research is supported by grants from the Natural Sciences and Engineering Research Council of Canada (NSERC) under grant numbers: RGPIN-2020-03900, ALLRP-560898-20, and CREATE-MSS Program.

REFERENCES

- Besl, P. J., McKay, N. D., 1992. Method for registration of 3-d shapes. *Sensor fusion IV: control paradigms and data structures*, 1611, Spie, 586–606.
- Dawson, E., Rashed, M. A., Abdelfatah, W., Noureldin, A., 2022. Radar-Based Multisensor Fusion for Uninterrupted Reliable Positioning in GNSS-Denied Environments. *IEEE Transactions on Intelligent Transportation Systems*.
- Domhof, J., Kooij, J. F., Gavrila, D. M., 2019. An extrinsic calibration tool for radar, camera and lidar. *2019 International*

Conference on Robotics and Automation (ICRA), IEEE, 8107–8113.

El-Sheimy, N., Li, Y., 2021. Indoor navigation: State of the art and future trends. *Satellite Navigation*, 2(1), 1–23.

Elsheikh, M., Noureldin, A., 2020. Multisensor precise positioning for automated and connected vehicles. *Connected and Autonomous Vehicles in Smart Cities*, CRC Press, 351–385.

Georgy, J., Noureldin, A., Korenberg, M. J., Bayoumi, M. M., 2009. Low-cost three-dimensional navigation solution for RISS/GPS integration using mixture particle filter. *IEEE Transactions on vehicular technology*, 59(2), 599–615.

Marti, E., De Miguel, M. A., Garcia, F., Perez, J., 2019. A review of sensor technologies for perception in automated driving. *IEEE Intelligent Transportation Systems Magazine*, 11(4), 94–108.

Mounier, E., Abdelaziz, S. K., de Araujo, P. R. M., Taghavikish, S., Elhabiby, M., Noureldin, A., 2022. Utilizing lidar registration on 3d high accuracy digital maps for robust positioning in gnss challenging environments. *Proceedings of the 35th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2022)*, 1366–1376.

Noureldin, A., Karamat, T. B., Georgy, J., 2013. *Fundamentals of inertial navigation, satellite-based positioning and their integration*. Springer.

Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Wheeler, R., Ng, A. Y. et al., 2009. Ros: an open-source robot operating system. *ICRA workshop on open source software*, 3number 3.2, Kobe, Japan, 5.

Rogers, R. M., 2003. *Applied mathematics in integrated navigation systems*. 1, Aiaa.

Schubert, E., Sander, J., Ester, M., Kriegel, H. P., Xu, X., 2017. DBSCAN revisited, revisited: why and how you should (still) use DBSCAN. *ACM Transactions on Database Systems (TODS)*, 42(3), 1–21.

Vizzo, I., Guadagnino, T., Mersch, B., Wiesmann, L., Behley, J., Stachniss, C., 2023. KISS-ICP: In Defense of Point-to-Point ICP Simple, Accurate, and Robust Registration If Done the Right Way. *IEEE Robotics and Automation Letters*.

Xu, W., Zhang, F., 2021. Fast-lío: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter. *IEEE Robotics and Automation Letters*, 6(2), 3317–3324.

Zhou, Q.-Y., Park, J., Koltun, V., 2018. Open3D: A Modern Library for 3D Data Processing. *arXiv:1801.09847*.