
Paulo Ricardo Marques de Araujo ¹, Mohamed Elhabiby ², Sidney Givigi ², and Aboelmagd Noureldin ²

¹Queens University
²Affiliation not available

October 30, 2023

Abstract

Autonomous and intelligent vehicles are multi-sensor systems operating in various environments and conditions. Due to their characteristics, inertial measurement units (IMUs) are typically the core component of such systems. However, these sensors rapidly accumulate errors due to biases and noise, degrading the positioning solution. Therefore, this paper presents a positioning solution that only uses three gyroscopes and one radar. The proposed method was tested using low-cost sensors in different scenarios, such as open-sky, urban and indoor areas. The key components of the method are the invariant Kalman filters and the use of deep neural networks to estimate the forward velocity of the car using the radar readings. The method was tested on a custom dataset, and our integrated solution accurately estimates the vehicle’s 3D position, velocity, and orientation. We achieved, on average, a 1.45% translational error in the tested scenarios, making the proposed method a robust alternative to current IMU-based positioning methods.

Paulo Ricardo Marques de Araujo, Graduate Student, IEEE, Mohamed Elhabiby, Sidney Givigi, Senior Member, IEEE, and Aboelmagd Noureldin, Senior Member, IEEE

Abstract—Autonomous and intelligent vehicles are multi-sensor systems operating in various environments and conditions. Due to their characteristics, inertial measurement units (IMUs) are typically the core component of such systems. However, these sensors rapidly accumulate errors due to biases and noise, degrading the positioning solution. Therefore, this paper presents a positioning solution that only uses three gyroscopes and one radar. The proposed method was tested using low-cost sensors in different scenarios, such as open-sky, urban and indoor areas. The key components of the method are the invariant Kalman filters and the use of deep neural networks to estimate the forward velocity of the car using the radar readings. The method was tested on a custom dataset, and our integrated solution accurately estimates the vehicle’s 3D position, velocity, and orientation. We achieved, on average, a 1.45 % translational error in the tested scenarios, making the proposed method a robust alternative to current IMU-based positioning methods.

Index Terms—Autonomous vehicles, deep learning, invariant extended Kalman filter, inertial navigation.

I. INTRODUCTION

Accurate and reliable positioning systems are imperative for autonomous vehicles (AVs) to function safely and effectively. This is especially true for self-driving cars operating in populated areas. Positioning systems provide real-time information on the vehicle’s position, orientation, and velocity, which enables the vehicle controller to make informed decisions, and precisely control its movements [1].

Typically, inertial measurement units (IMUs) are the core of positioning systems [2]. IMUs generally have three orthogonal gyroscopes and three orthogonal accelerometers. The former provides angular velocity measurements, while the latter provides linear acceleration measurements. By using both information, the system can infer the position and orientation of the vehicle. Commercial IMUs can vary in quality and accuracy, which can be especially critical for low-cost sensors normally used by the automotive industry as they cause significant drifts in the final solution [3].

An alternative for positioning using an IMU is the Reduced Inertial Sensor System (RISS) [4]. It uses a reduced set of sensors and an odometer to estimate the state of the vehicle more accurately. Recent publications have achieved compelling results with this method [5]–[9]. However, RISS has limitations. It relies on odometers, which are slow sensors. Therefore, RISS may not be the ideal solution for AVs. Also, some attitude angles are computed using accelerometers, which results in noisy and unreliable estimates.

Using a different reduced set of sensors, deep neural networks, and signal-processing tools, we propose a new method to serve as the core component for positioning systems embedded in modern vehicles and future AVs. Our contributions are as follows:

- we introduce the kinematic model and the strategy to estimate the forward velocity of a car using low-cost radar, deep learning and signal-processing tools;
- we implement invariant extended Kalman filters that exploits the proposed model and combines it with relevant
sensors of position corrections for ground AVs (e.g., GNSS and 3D maps);
- we demonstrate the performance of the proposed approach against IMU-based methods on a custom dataset that explored different urban environments.

The results shown in this paper demonstrate that the proposed approach outperformed the IMU-based method (see Appendix B for further details) in all tests. The proposed method is also flexible to work with any source of forward velocity. In this paper, all equations of the filters are detailed, and adaptations or extensions may be done in a smooth way.

The remainder of the document is organized as follows. Section II offers a review of recent related work, giving an idea of the potential applications of the proposed method, as discussed in the following sections. Next, section III presents the kinematic equations of the vehicle, the challenges associated with low-cost radars, and the strategy employed to tackle them. In Section IV, we model invariant Kalman filters for relevant applications aiming AVs. All the experiments and achieved results are detailed in Section V. Finally, Section VI concludes the paper.

II. RELATED WORK

IMUs are ubiquitous in positioning systems for land vehicles. For example, they can be used with vision [10], [11], LiDAR [12], radar [13], global navigation satellite system (GNSS) [14], and 3D maps [15], [16]. As maps have become important for positioning systems in areas where GNSS fails, researchers also use IMUs to aid the 3D map creation while navigating [17]. Recently, some researchers proposed a remarkable approach for accurate positioning using only the IMU. The method leverages deep-learning tools and state-of-the-art Kalman filters [18].

Another exciting research path investigates methods to enhance the quality of IMU measurements. Research has explored conventional signal-processing tools [19]. Recent deep-learning developments allowed for the achievement of promising results using supervised learning [20], and self-supervised learning that diminishes the need for specialized datasets [21].

All the aforementioned techniques and sensors have associated uncertainties. Combining estimates from these diverse systems requires filtering techniques such as Kalman filters. In recent years, the application of geometry theory for position estimation has been introduced [22]. The proposed framework gave rise to a class of invariant filters that rapidly became state-of-the-art for positioning problems. Invariant Kalman filters (IKFs) have been applied to different mobile platforms such as autonomous underwater vehicles (AUVs) [23], quadcopters [24], bipedal robots [25], aircrafts [26], and cars [27]. The properties of IKFs helped to solve various challenges in mobile navigation, such as Simultaneous Localization and Mapping (SLAM) [28] and visual-inertial odometry [29].

This work proposes a method that leverages invariant Kalman filters to minimize the impact of errors in the linearization process. Also, the proposed method replaced the accelerometers with a radar aided by a deep-learning-based denoiser, reducing the effect of biases in the integrated solution even more.

III. PROPOSED METHOD

A. Kinematic Model

The proposed method is based on attitude estimation using gyroscopes. Unlike IMU-based methods, the velocities in the IMU-frame, \(i\), are estimated using measurements from a radar mounted on the front bumper of the car. This method alleviates the system from error propagation due to acceleration integration as well as the presence of the gravity field for estimating the vehicle’s position. The proposed method is a Radar Inertial Dead-Reckoning (RIDR) positioning system.

Let us assume the vehicle is travelling following a generic path as shown in Figure 3. The forward velocity of the vehicle, \(v^f\), is measured using the radar in the IMU frame, \(i\). Let us further assume that the lateral and up velocities are null and, without loss of generality, that the velocity vector in the \(i\)-frame is defined as

\[
v_i^n = \begin{bmatrix} 0 & v_i^f & 0 \end{bmatrix}^T.
\]

The orientation, \(R \in \text{SO}(3)\), and position, \(p \in \mathbb{R}^3\), of the vehicle are expressed in the \(w\)-frame. Given \(R_0\) and \(p_0\), the proposed method estimates future states using gyroscopes and a radar. The velocity vector in the \(i\)-frame is projected to the \(w\)-frame using \(R\).

The gyroscopes provide noisy and biased angular velocity measurements, \(\omega_n\), that can be modelled as

\[
\dot{\omega}_n = \omega_n + b_\omega^n + \omega_n^\text{bias},
\]

where \(\omega_n \in \mathbb{R}^3\) is the true angular velocity, \(b_\omega^n \in \mathbb{R}^3\) is the bias drift and \(\omega_n^\text{bias} \in \mathbb{R}^3\) is zero-mean Gaussian noise. Usually, biases are modelled as random walk processes:

\[
b_\omega^{n+1} = b_\omega^n + \omega_n^\text{bias}.
\]

The radar also provides noisy measurements, i.e.,

\[
\dot{v}_n = v_i^n + w_n^\text{Rd}.
\]
where \( w_{ni}^f \in \mathbb{R}^3 \) is zero-mean Gaussian noise. Therefore, the following equations govern the kinematic model:

\[
R_{n+1} = R_n \exp \left( \left[ \tilde{w}_n - b_n^w - w_n^{w'} \right] dt \right) \tag{4}
\]
\[
p_{n+1} = p_n + R_n \left( \left[ \tilde{v}_n - w_n^{v'} \right] dt \right), \tag{5}
\]

where \( dt \) is the discrete sampling time, and \( \exp \) is the SO(3) exponential map. Finally, \( x_{i,x} \in \mathbb{R}^{3 \times 3} \) denotes the skew matrix associated with the cross product with a three-dimensional vector, i.e., for \( a, x \in \mathbb{R}^3 \), we have \( x_{i,x} a = x \times a \).

B. Forward Velocity Measurement

One of the advantages of the proposed system is the frequency of the integrated solution. Radars are often faster than odometers, e.g., the low-cost radar we use operates at 20 Hz, while the odometer available in the experimental setup operates at 3 Hz. Further technical details are presented in Section [V].

The radar is mounted facing the ground with an angle, \( \theta \). This configuration enables the measurement of the Doppler velocity, \( v^D \), between the car and the ground. As the ground is always stationary, the radar can estimate the forward velocity of the vehicle. The mounting angle is unknown due to the holding device and the shape of the bumper, and it can be used to convert \( v^D \) to \( v^f \) using

\[
v_n^D = v_n^f \cdot \cos \theta. \tag{6}
\]

The mounting angle can be obtained using any forward velocity reference. Besides calibrating the mounting angle, it is crucial to consider the noise associated with the measurements.

To tackle all these challenges, the proposed approach leverages deep learning and signal processing methods to better estimate the forward velocity of a car. Fig. 4 depicts the denoiser used within the proposed framework. The core component of the denoiser is the autoencoder used to estimate the mounting angle and reduce the noise of a sequence of radar readings. The length of the readings can vary according to any project requirement. In this work, a sequence of 128 readings is used to balance performance and initialization time for real-time systems (a radar operating at 20 Hz takes approximately 6.4 s to collect the required data to be sent to the denoiser).

The denoiser architecture comprises a sequence of 6 one-dimensional convolutions with kernels of size 7. On the encoder side, the number of channels increases by \( 2^n \), where \( n \) is the order of the layer. The decoder mirrors the encoder, and the number of channels decreases until a single channel is achieved in the last layer. The model has approximately 27,600 parameters, and its estimated size is 0.9 MB. Each convolution layer is followed by a batch normalization layer [30] and a GELU activation layer [31]. The median filter uses a sliding window of size 51 to remove possible outliers that might appear, given the quality of the sensor.

IV. KALMAN FILTERING WITH ARBITRARY MEASUREMENTS

The Extended Kalman Filter (EKF) is an algorithm that estimates the state of a non-linear dynamic system with Gaussian noise. It is an extension of the Kalman filter designed for linear systems. The EKF linearizes the non-linear system at each time step, propagates the mean and covariance of the state estimate through the linearized system, and updates the estimate based on measurements. The EKF is widely used in various fields, such as control systems, robotics, and signal processing.

The dynamical discrete-time model is defined as

\[
x_{n+1} = f(x_n, u_n) + w_n, \tag{7}
\]

where \( x_n \) is the states to be estimated, \( u_n \) is the inputs of the system, and \( w_n \) is the process noise assumed to be zero-mean Gaussian with covariance \( Q_n \), i.e. \( w_n \sim N(0, Q_n) \).

The predictions are corrected using measurements of the states. The observation model is defined as

\[
y_n = h(x_n) + n_n, \tag{8}
\]

where the measurement noise is also zero-mean Gaussian, i.e. \( n_n \sim N(0, R_n) \). The covariance matrix can be defined following the sensors’ datasheets, user experience, and/or dedicated experiments.

After the filter initialization, the states are estimated using (7) with the noise turned off, and the covariance matrix, \( P \), is updated using

\[
P_{n+1} = F_n P_n F_n^T + G_n Q_n G_n^T, \tag{9}
\]

where \( F_n \) and \( G_n \) are the Jacobian matrices of \( f(\cdot) \) with respect to the states and the inputs, respectively. To update the previous estimation, the Kalman gain is computed to balance the weight between the previous estimation and the current measurement

\[
K_n = P_n H_n (H_n P_n H_n^T + N_n)^{-1}, \tag{10}
\]

where \( H_n \) is the Jacobian matrix of \( h(\cdot) \) with respect to the states. The previous states, \( x_n^- \), are corrected using

\[
x_n^+ = x_n^- + K_n (y_n - h(x_n^-)). \tag{11}
\]

Finally, the covariance matrix can be updated as follows:

\[
P_{n+1}^+ = (I - K_n H_n) P_n^- (I - K_n H_n)^T + K_n N_n K_n^T. \tag{12}
\]

where \( \cdot^- \) and \( \cdot^+ \) indicate the \emph{a priori} and \emph{a posteriori} states or covariances, i.e. before and after correcting the predicted variables. To implement an EKF, one needs to define \( f(\cdot) \), \( h(\cdot) \), and the covariance matrices \( Q_n \) and \( N_n \).

A. Defining the Dynamical Model \( f \)

The evolution of the states is given by (3)-(5).

B. Defining the Measurement Model \( h \)

RIDR is flexible and easy to use with Kalman filters. Considering the arrangement presented in Fig. 3 a wide variety of measurement models can be admitted, such as:

1) Position in the \( w \)-frame using GNSS: \( p_n \)
2) Position in the \( w \)-frame using 3D maps: \( p_n \)
3) Position in the \( i \)-frame using known landmarks \( r_n^m \):

\[
R_n^T (r_n^m - p_n)
\]
4) Attitude in the $i$-frame using magnetometer: $R_n^T \beta$
5) Attitude in the $w$-frame using 3D maps: $R_n$

This work exploits two sources of information of utmost importance for autonomous vehicles: GNSS and 3D maps. The GNSS receiver measures the position, and the 3D maps are used with the LiDAR assembled in the car. The LiDAR generates a local point cloud that is compared to the city map using the iterative closest point (ICP) algorithm to measure the position and the orientation of the vehicle \[15\].

C. The Invariant Extended Kalman Filter

To demonstrate the potential of RIDR, this work uses a recent version of the EKF, the invariant EKF (IEKF). The IEKF exploits symmetry in the state space to derive state-invariant Jacobians \[22\], \[32\]. The IEKF has shown remarkable results for inertial navigation in recent years \[18\], \[23\], \[25\], \[29\], \[33\], \[34\]. The implementation workflow of the IEKF is the same of the EKF. The reader can refer to Appendix A for some background on Lie theory and to Appendix B for the equations used in the tested filters.

V. EXPERIMENTAL RESULTS

A. Experimental Setup

The proposed method was evaluated using a dataset developed with the experimental setup available at the Navigation and Instrumentation laboratory of Queen’s University (Fig. 5). The experimental car has various sensors, such as stereo cameras, GNSS receivers, LiDAR, and automotive radars.

For the filters we present in this work, we use the IMU inside the Stereolabs Zed2i camera, and an OmniPresense OPS241-A radar. The low-cost radar operates at $20$ Hz with a resolution of $0.3$ m/s, and the low-cost IMU operates at $100$ Hz, but it is dynamically downsampled to match the frequency of the radar. Experiments conducted in the laboratory reviewed a bias instability around $18$ deg/h for the gyroscopes and $14$ mg for the accelerometers.

The reference solution to evaluate the results was built using a Novatel PwrPak7 unit connected to a KVH 1750 IMU. The data was processed using the Novatel Inertial Explorer software and data collected with a base station to generate Real-Time Kinematics positioning (RTK).

B. Radar Denoiser Training

The model was implemented using Pytorch and trained using the aforementioned dataset. It contemplates various trajectories recorded in Canadian cities such as Kingston, Toronto, Calgary, and Edmonton. Around $75$ km of data was collected, generating more than $160,000$ radar measurements. The data was divided into small sequences of $128$ readings.

The training spanned $400$ epochs, using the Rectified ADAM as the optimizer ($\lambda = 1e^{-3}$, $\delta = 1e^{-5}$). The loss function was defined as

$$L = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

where $N$ is the total number of observations, $y_i$ is the actual value of the $i^{th}$ observation, and $\hat{y}_i$ is the predicted value of the $i^{th}$ observation. A scheduler was employed to reduce $\lambda$ if the validation loss does not improve after $100$ epochs. The batch size was defined as $256$.

Fig. 6 presents the performance of the model compared to the popular Savitzky-Golay filter \[35\] in a testing trajectory. It can be seen that the proposed method achieves robust results...
while calibrating the mounting angle, denoising the signal and detecting when the vehicle is stationary.

C. Evaluation Metrics and Compared Methods

To evaluate the performance of RIDR with and without the IEKF, this work uses two error metrics proposed in [36], [37]:

1) Relative Translation Error ($t_{rel}$): it is determined by calculating the average relative translation increment error for every possible sub-sequence of distances ranging from 100 m to 800 m and expressing this error as a percentage of the total distance travelled.

2) Relative Rotational Error ($r_{rel}$): the average relative rotational increment error is calculated for all possible sub-sequences of distances ranging from 100 m to 800 m. This error is expressed in deg/m.

3) Absolute Trajectory Error (ATE): it measures the root-mean-square error between the predicted 3D trajectory and the ground truth.

RIDR works as an alternative to the IMU-based method. Therefore, its performance is compared to INS with and without the IEKF. The reader can refer to the Appendix for further details about IEKF implementation for INS.

The tests were performed in real environments where GNSS and 3D maps were available. Some challenges, such as GNSS outages and 3D map gaps, are present in the tested environments.

D. Implementation Details

The IMU was tested indoors for bias instability and noise density specification. Four hours of data were collected and analyzed using Allan Variance tools. The retrieved information was used to build the process covariance matrices. In this work, $Q_n$ is kept constant throughout the trajectories and dynamic tuning is left for future work. However, the measurement covariance matrices are set dynamically. The GNSS receiver provides the standard deviations of the measurements, and they are used to build the $N_n$ whenever the GNSS readings are used. On the other hand, ICP outputs similar information for the registration solution, which is used to build $N_n$ whenever the 3D maps are used. GNSS and 3D maps measurements were given with respect to the $w$-frame.

E. Trajectory Results

All trajectories presented in this section followed the same protocol. The parameters were set as described in the previous section, and the trajectories started in open-sky areas to allow the initialization of the filters using GNSS.

This work explored three different real urban scenarios:

- open-sky: most of the trajectory happens in an open area with good GNSS coverage;
- mixed: the trajectory explores areas with good GNSS and 3D map coverage but faces challenges such as outages and gaps;
- dense: most of the trajectory happens in dense downtown areas with long periods of GNSS outages, but 3D maps are available;
- indoor: the trajectory starts outside before entering an indoor parking garage where 3D maps are available.

Fig. 7 shows the robustness of RIDR standalone and integrated with the left-invariant extended Kalman filter (LIEKF) in an open-sky scenario. As discussed in section III-A, RIDR does not use accelerometers as they cause a significant drift when they are low-cost sensors (e.g., INS and LIEKF-INS face this problem).

The second sequence explored a mixed area. In this trajectory, GNSS and 3D maps are available, but not continuously (Fig. 8 (a)). The third sequence was recorded in the dense area of downtown Calgary, where reliable GNSS measurements do not exist. Portions of the central area have 3D maps used for position and orientation corrections. The overall results obtained in this sequence are presented in Fig. 8 (b).

The last sequence explored an indoor parking garage. In this scenario, little time was spent outside to initialize the filters. As soon as the vehicle hit the parking garage ramp, GNSS measurements were out, and only 3D maps were used for estimation. Inside the garage, there are gaps in the map, similar
Fig. 8. Trajectory results in a mixed (seq. 02) and in a dense area (seq. 03) of the Navinst dataset.

Fig. 9. Trajectory results in an indoor parking garage (seq. 04) of the Navinst dataset.

to the other scenarios presented in this work. The velocity profile in this scenario differs from all other trajectories as the car moved slowly and intermittently, configuring a challenge for the low-cost radar employed in the experimental setup. Despite this fact, the integrated solution using RIDR achieved the best results.

Further details about the trajectories and the obtained results can be found in Table I. The trajectory results were averaged, and clearly, the proposed integrated method achieved the best scores. The LIEKF-RIDR achieved an averaged relative translation error of 1.45% for a total of 16.48 km (when considering all the trajectories). This is a remarkable achievement given the low quality of the sensors used and the challenges faced along the trajectories. The biases of the accelerometers certainly impacted the LIEKF-IMU solution, which can also be noticed in the IMU-only solution. The proposed method held the solution closer to the reference most of the time and kept the evaluation metrics more stable, keeping the averaged absolute trajectory error below 2 m.

F. Discussion

As presented in previous sections, the absence of accelerometers increases the robustness of RIDR. The integration of RIDR and the LIEKF proved to be a powerful tool for land vehicle navigation. This combination could bridge different challenges in real urban environments.

We note that RIDR can be slower when compared to IMUs, i.e., the data rate of the radar regulates the data rate of the system. However, the results presented show that RIDR equipped with invariant filters is a reliable and robust alternative for onboard navigation solutions for autonomous vehicles. For example, consider a vehicle travelling in a city at 50 km/h ≈ 13.89 m/s. Let us assume that RIDR is operating at 20 Hz, i.e., a new state estimate every 0.05 s. In this case, the car travelled 0.69 m without any prediction. It is a sub-meter achievement that can equip a multi-sensor system during missions to keep the car in the planned trajectory. Furthermore, as automotive radars are evolving quickly, this limitation might be solved soon. For example, if radars or any other source of forward velocity provides measurements at 50 Hz, the travelled distance drops to 0.28 m, potentially improving safety.

Another important aspect to note is the presence of a deep learning module in the RIDR workflow. This technology may be hard to certify for commercial or industrial applications [38]. However, it can offer a baseline for developing more advanced or combined signal-processing techniques.

VI. CONCLUSION

The proposed method performed better than IMU-based methods in all trajectories, including the results using GNSS and 3D map measurements. The scenarios presented in this work represent real and critical situations that autonomous
vehicles might face in urban environments. This work can be adapted to other applications to replace complete IMU-based methods. For example, based on the results presented in the previous section, RIDR can be used in surveying vehicles that map cities. GNSS outages have less effect on RIDR’s positioning solution, potentially improving the accuracy of the map. In future work, we would like to address the biases issue using other deep learning methods to further improve the integrated solution.

APPENDIX A

A matrix Lie group \( \mathcal{G} \) is a set of \( n \times n \) matrices, verifying the following properties:

\[ I \in \mathcal{G}, \]
\[ X^{-1} \in \mathcal{G}, \quad \forall X \in \mathcal{G}, \quad (14) \]
\[ X \cdot Y \in \mathcal{G}, \quad \forall X, Y \in \mathcal{G}. \quad (15) \]

\( \mathcal{G} \) is generally a curved space, and for a point, \( X \in \mathcal{G} \), \( T_X \mathcal{G} \) is denoted as the tangent space at \( X \) with dimension \( d \) equals to the degrees of freedom of \( \mathcal{G} \). The tangent space at the identity element of the group is called Lie algebra, and it is denoted as \( \mathfrak{g} \). It consists of the tangent vectors to smooth paths in \( \mathcal{G} \) where they pass through \( I \). A system evolving through time can be seen as a point moving on the curved space of \( \mathcal{G} \).

The general form of the tangent vectors living in this space can be found by differentiating the defining equation of the group. The tangent space can be locally identified with a Euclidean vector space in \( \mathbb{R}^d \). Thus, for \( \xi \in \mathbb{R}^d \), \( \xi^\wedge \in \mathfrak{g} \) is its corresponding element of \( \mathfrak{g} \). Therefore, for a Lie group \( \mathcal{G} \) and its associate Lie Algebra \( \mathfrak{g} \), the following linear mapping is defined:

\[ (\cdot)^\wedge : \mathbb{R}^{\dim \mathcal{G}} \mapsto \mathfrak{g}. \quad (17) \]

This linear map takes elements of the tangent space of \( \mathcal{G} \) at the identity to the corresponding matrix representation. The opposite mapping is defined as

\[ (\cdot)^\vee : \mathfrak{g} \mapsto \mathbb{R}^{\dim \mathcal{G}}. \quad (18) \]

The matrix Lie group \( \text{SO}(3) \) is the group of \( 3 \times 3 \) rotation matrices that preserve orientation. Its Lie algebra is the space of skew-symmetry matrices:

\[ \text{SO}(3) = \{ R, \ R R^T = I_{3 \times 3}, \det(R) = 1 \}, \quad (19) \]
\[ \text{so}(3) = \{ A, \ A = -A^T \} , \quad (20) \]
\[ \xi^\wedge = [\xi^\vee]_x = \begin{bmatrix} 0 & -\xi_3 & \xi_2 \\
\xi_3 & 0 & -\xi_1 \\
-\xi_2 & \xi_1 & 0 \end{bmatrix}. \quad (21) \]

The closed-form expression for the exponential map is given by

\[ \exp(\xi) = I_{3 \times 3} + \frac{\sin(\|\xi\|)}{\|\xi\|} [\|\xi\|]_x + \frac{2\sin(\|\xi\|/2)}{\|\xi\|^2} [\|\xi\|^2]_x. \quad (22) \]

The logarithmic map is computed using

\[ \|t\| = \cos^{-1}\left(\frac{\text{Tr}(t^2) - 1}{2}\right), \quad (23) \]
\[ \xi = \frac{1}{2\sin(\|t\|)} \begin{bmatrix} R_{32} - R_{23} \\
R_{13} - R_{31} \\
R_{21} - R_{12} \end{bmatrix}. \quad (24) \]

The matrix Lie group \( \text{SE}(3) \) is the group of rigid motions defined as

\[ \text{SE}(3) = \left\{ \begin{bmatrix} R & p \\
0_{1 \times 3} & 1 \end{bmatrix} , \ R \in \text{SO}(3), \ p \in \mathbb{R}^3 \right\} , \quad (25) \]
\[ \mathfrak{se}(3) = \left\{ \begin{bmatrix} \xi^R & \xi^p \\
0_{1 \times 3} & 0 \end{bmatrix} , \ R^R, \xi^p \in \mathbb{R}^3 \right\}. \quad (26) \]
\[ (\xi^R)^\wedge = S = \begin{bmatrix} [\xi^R]_x & \xi^p \\
0_{1 \times 3} & 0 \end{bmatrix} \quad (27) \]

Its exponential map is defined as

\[ \exp(\xi^R) = I_{4 \times 4} + S + \frac{1 - \cos(\|\xi^R\|)}{\|\xi^R\|^2} S^2 \]
\[ + \frac{\|\xi^R\| - \sin(\|\xi^R\|)}{\|\xi^R\|^3} S^3. \quad (28) \]

Finally, the matrix Lie group \( \text{SE}_2(3) \) is an extension of \( \text{SE}(3) \). It was named as the group of double direct spatial isometries [22].

\[ \text{SE}_2(3) = \left\{ \begin{bmatrix} R & v & p \\
0_{1 \times 3} & 1 & 0 \\
0_{1 \times 3} & 0 & 1 \end{bmatrix} , \ R \in \text{SO}(3), \ v, p \in \mathbb{R}^3 \right\} , \quad (29) \]
\[ \mathfrak{se}_2(3) = \left\{ \begin{bmatrix} [\xi^R]_x & \xi^v & \xi^p \\
0_{1 \times 3} & 0 & 0 \end{bmatrix} , \ R^R, \xi^v, \xi^p \in \mathbb{R}^3 \right\}. \quad (30) \]
\[ (\xi^R)^\wedge = S = \begin{bmatrix} [\xi^R]_x & \xi^v & \xi^p \\
0_{1 \times 3} & 0 & 0 \\
0_{1 \times 3} & 0 & 0 \end{bmatrix}. \quad (31) \]
The Jacobian matrix $H$ is obtained from the sensitivity of the innovation vector to be used with the Kalman gain.

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} + \frac{1}{\|\xi^R\|} \begin{bmatrix} 2 \xi^R_1 \xi^R_2 \xi^R_3 \\ -\xi^R_1 \xi^R_2 + \xi^R_3 \\ \xi^R_1 \xi^R_3 + \xi^R_2 \\ \xi^R_2 \xi^R_3 - \xi^R_1 \\ 0 \xi^R_1 + \xi^R_2 \\ 0 \xi^R_1 - \xi^R_2 \\ 0 \xi^R_1 \xi^R_3 + \xi^R_2 \\ 0 \xi^R_1 \xi^R_3 - \xi^R_2 \end{bmatrix} S^2$$

\[ (32) \]

**Appendix B**

The Invariant Extended Kalman Filter is an error state filter. The errors are linearized in the Lie algebra of the group used to represent the state of the system. It can have two variations named left- and right-invariant \^[22], [32]. This works exploits measurements in the $w$-frame, leading to the development of left-invariant filters. This appendix presents the left-invariant filters tested in this work.

**A. Radar Inertial Dead-Reckoning System**

The state of the system, i.e. $x_n = (R_n, p_n)$, is embedded in the matrix Lie group SE(3). The gyroscopes bias, $b^\omega_n$ is treated as a vector. The linearized error is defined as:

$$e_n = \begin{bmatrix} \xi^R_n \xi^p_n \xi^b_n \end{bmatrix}^T \sim \mathcal{N}(0, P_n)$$

\[ (33) \]

The states are propagated using (3)-4, and the covariance is propagated using (9). The $G_n$ matrix is a $9 \times 9$ identity matrix, and the $F_n$ is defined as follows:

$$F_n = \begin{bmatrix} - [\omega_n - b^\omega_n]_x & 0_{3 \times 3} & -I_{3 \times 3} \\ -[\dot{\nu}_n]_x & [\omega_n - b^\omega_n]_x & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix} dt$$

\[ (34) \]

The correction step exploits two measurement models. For position correction using GNSS readings, the model becomes:

$$y_n = \begin{bmatrix} p_n & 1 \end{bmatrix}^T$$

\[ (35) \]

leading to the Jacobian matrix $H_n$ equals to:

$$H_n = \begin{bmatrix} 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}$$

\[ (36) \]

The innovation is computed as follows:

$$z_n = x_n^{-1} \cdot y_n - b, \quad b = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T$$

\[ (37) \]

In the case of position and attitude correction using 3D maps, the model becomes:

$$y_n^R = R_n$$

$$y_n^p = \begin{bmatrix} p_n & 1 \end{bmatrix}^T$$

\[ (38), (39) \]

The innovation involves mapping the rotation matrix to the Lie algebra. The individual results are stacked into a final innovation vector to be used with the Kalman gain.

$$z_n^R = \begin{bmatrix} \log (\hat{R}_n \cdot Y_n^{R}) \end{bmatrix}^T$$

\[ (40) \]

$$z_n^p = x_n^{-1} \cdot y_n - b, \quad b = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T$$

\[ (41) \]

$$z_n = \begin{bmatrix} z_n^R & z_n^p \end{bmatrix}^T$$

\[ (42) \]

where $\hat{R}_n$ is the rotation matrix extracted from the state matrix, and $\log$ is the SO(3) logarithmic map. Finally, the Jacobian matrix $H_n$ becomes:

$$H_n = \begin{bmatrix} I_{6 \times 6} & 0_{6 \times 3} \end{bmatrix}$$

\[ (43) \]

The error is then computed and injected in the states as follows:

$$e_n = K_n z_n$$

$$\dot{X}_n^+ = X_n^+ \cdot \exp(e_n^X)$$

$$b_n^+ = b_n^- + e_n^b$$

\[ (44), (45), (46) \]

where $\exp$ is the SE(3) exponential map, and $e_n^X \in \mathbb{R}^6$ and $e_n^b \in \mathbb{R}^3$ are sub-vectors of $e_n$ regarding the matrix Lie group and the bias vector, respectively.

**B. Inertial Measurement Unit System**

Besides the gyroscopes, INS uses the accelerometers available in IMUs to estimate the velocity of the system. Similar to the gyroscopes, the accelerometers provide noisy and biased readings:

$$a_n = a_n + b_n^\omega + w_n^a$$

\[ (47) \]

The evolution of the states is modelled as follows:

$$R_n+1 = R_n \exp \left( [ [\omega_n - b_n^\omega - w_n^\omega] dt \right)$$

$$v_n+1 = v_n + (R_n (a_n - b_n^a - w_n^a) + g) dt$$

$$p_n+1 = p_n + v_n dt$$

\[ (48), (49), (50) \]

where $g \in \mathbb{R}^3$ is the gravitational vector. The biases are also modelled as random walk processes:

$$b_n^{\omega+1} = b_n^\omega + w_n^{\omega}$$

$$b_n^{a+1} = b_n^a + w_n^{a}$$

\[ (51), (52) \]

The state of the system is embedded in the matrix Lie group $SE_2(3)$. The biases can be stacked to form a vector, i.e. $b_n = [b_n^{\omega^T} b_n^{a^T}]^T \in \mathbb{R}^6$. The linearized error becomes:

$$e_n = \begin{bmatrix} \xi_n^{R^T} & \xi_n^{p^T} & \xi_n^{b^\omega^T} & \xi_n^{b^a^T} \end{bmatrix}^T \sim \mathcal{N}(0, P_n)$$

\[ (53) \]

The $G_n$ matrix is a $15 \times 15$ identity matrix, and $F_n$ is defined as follows:
\[ F_n = I_{15 \times 15} + \begin{bmatrix} - \tilde{\omega}_n \times & 0_{3 \times 3} & - I_{3 \times 3} & 0_{3 \times 3} \\ - \tilde{a}_n \times & - \tilde{\omega}_n \times & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & I_{3 \times 3} & - \tilde{a}_n \times & 0_{3 \times 3} \\ 0_{6 \times 3} & 0_{6 \times 3} & 0_{6 \times 3} & 0_{6 \times 3} \end{bmatrix} dt \]  

where \( \tilde{\omega} = \dot{\omega}_n - b_n^\omega \) and \( \tilde{a} = \dot{a}_n - b_n^a \). The measurement models are similar to those presented in the previous section. However, due to the different state space, the Jacobian matrix \( H_n \) for position corrections using GNSS readings becomes:

\[ H_n = \begin{bmatrix} 0_{3 \times 6} & I_{3 \times 3} & 0_{3 \times 6} \end{bmatrix} \]  

In case of position and attitude correction using 3D maps, the Jacobian matrix is defined as:

\[ H_n = \begin{bmatrix} I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & I_{3 \times 3} \end{bmatrix} \]  

The error is then computed and injected in the states as follows:

\[ e_n = K_n z_n \]  

\[ \chi_n^+ = \chi_n^- \cdot \exp(e_n^\chi) \]  

\[ b_n^b = b_n^b + e_n^b \]  

where \( e_n^\chi \in \mathbb{R}^9 \) and \( e_n^b \in \mathbb{R}^6 \) are sub-vectors of \( e_n \) regarding the matrix Lie group and the bias vector, respectively.

REFERENCES


Paulo Ricardo Marques de Araujo received an M.Sc. degree in Engineering and Innovation Management from the Federal University of ABC, Santo André, SP, Brazil. He is currently a Ph.D. candidate in the Electrical and Computer Engineering department at Queen’s University, Kingston, ON, Canada. His research interests include autonomous systems and robotics, machine learning, and digital manufacturing.

Mohamed Elhabiby was the Treasurer of the Geodesy Section at the Canadian Geophysical Union from 2008 to 2014. He is currently an Associate Professor with the Faculty of Engineering, Ain Shams University, Cairo, Egypt. He is also the Executive Vice President and Co-Founder of Micro Engineering Tech Inc., Calgary, AB, a high-tech international company specialized in high-precision engineering and instrumentation, mobile mapping, laser scanning, deformation monitoring, and GPS/INS integrations. He is a Leader of an Archaeological Mission in the Area of the Great Pyramids, Cairo. He received the Astech Awards. Avenue Magazine names him as one of the Top 40 under 40. He is the Chair of WG 4.1.4: Imaging Techniques, Sub-Commission 4.1: Alternatives and Backups to GNSS. He chaired the Geocomputations and Cyber Infrastructure Oral Session at the Canadian Geophysical Union annual meeting from 2008 to 2012.

Sidney Givigi (SM’14) received a Ph.D. in Electrical and Computer Engineering from Carleton University, Ottawa, ON, Canada. He is now an Associate Professor with the School of Computing of Queen’s University, Kingston, ON. Sidney’s research interests mainly focus on machine learning, autonomous systems, and robotics.

Aboelmagd Noureldin received the B.Sc. degree in electrical engineering and the M.Sc. degree in engineering physics from Cairo University, Egypt, in 1993 and 1997, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Calgary, AB, Canada, in 2002. He is a Professor at the Department of Electrical and Computer Engineering, Royal Military College of Canada (RMCC), with Cross Appointments at the School of Computing and the Department of Electrical and Computer Engineering, Queen’s University. He is also the Founder and the Director of the Navigation and Instrumentation Research Lab at RMCC. He has published two books, four book chapters, and more than 270 papers in journals, magazines, and conference proceedings. His research interests include global navigation satellite systems, wireless positioning and navigation, indoor positioning, and multi-sensor fusion targeting applications related to autonomous systems, intelligent transportation, road information services, crowd management, and the vehicular Internet of Things. His research led to 13 patents and several technologies licensed to the industry in position, location, and navigation systems.