

GPS/INS integration via faded memory Kalman filter

H. Alaeiyan, Mohammad Reza Mosavi*, A. Ayatollahi

Department of Electrical Engineering, Iran University of Science and Technology, Narmak, Tehran 16846-13114, Iran

(Communicated by Seyed Hossein Siadati)

Abstract

A common technique for navigation and positioning applications is the Global Positioning System (GPS)/Inertial Navigation System (INS) integration, which combines the strengths of GPS and INS to offer accurate and reliable information. As a standalone system, the performance of the INS deteriorates as time is passed. Kalman Filter (KF) is used for GPS/INS integration, and its performance is excellent for simple data. However, in a complex and natural set environment, its performance degrades when the system performs relatively long; therefore, resolving the long-time problem for the GPS/INS system is challenging. The novelty of this paper is GPS/INS integration with the Faded Kalman Filter (FKF). In the FKF, the measurement updates are weighted differently to adapt to changes in the system. This approach allows the filter to adapt to changes or uncertainties in the system dynamics. GPS/INS integration performance is significantly improved using this algorithm rather than a simple KF. An average of 45% reduces the positioning errors compared to traditional KF.

Keywords: GPS/INS integration, Kalman Filter, Faded Memory Filter
2020 MSC: 97M50

1 Introduction

An Inertial Navigation System (INS) is a technology designed to gauge the orientation, position, and velocity of a mobile entity, like a vehicle or bicycle. Its applicability extends to wearable devices, such as shoes or clothing, enabling the monitoring of movement and activity independently of external references, such as GPS or radio signals [5, 7, 22]. This sensing device comprises two primary components: an Inertial Measurement Unit (IMU), responsible for gauging linear acceleration and angular velocity, and a mechanization block that analyzes the IMU data to derive information on orientation, position, and velocity [20]. An INS provides several advantages, including functioning autonomously, operating automatically, ensuring reliability, and remaining unaffected by external disturbances [3]. However, an INS also has drawbacks, such as having high cost, complexity, and susceptibility to sensor errors that accumulate over time. On the other hand, GPS is a system that uses satellites to give location and time information to a receiver on or close to Earth. It computes the receiver's location by using trilateration, which is based on the distance and time difference between the receiver and the satellites. GPS offers many benefits, such as providing precise, dependable, worldwide, and available information.

By combining the strengths of GPS and INS, GPS/INS integration is a method that provides precise and dependable information for navigation and positioning applications. GPS/INS integration enhances positioning performance by

*Corresponding author

Email addresses: h_alaeiyan@elec.iust.ac.ir (H. Alaeiyan), m_moasvi@iust.ac.ir (Mohammad Reza Mosavi), ayatollahi@iust.ac.ir (A. Ayatollahi)

leveraging the complementary features of GPS and INS. GPS exhibits high precision; however, it is not always available or reliable, whereas INS is consistently available and reliable but possesses lower precision. Through the fusion of data from GPS and INS, GPS/INS integration achieves high precision, availability, and reliability in various situations and settings [20, 3, 18]. Addressing the challenge of GPS integration involves considering two key aspects. Firstly, enhancing the system with additional sensors, and secondly, designing and implementing suitable processing and estimation algorithms. The selection of sensors for integration is contingent on the specific application and technological requirements.

One common sensor employed in navigation systems for applications like autonomous cars, drones, robots, and satellites is the camera. Cameras provide vision-like sensory capabilities and facilitate the recognition of 2D information. The camera offers advantages such as cost-effectiveness, high resolution, and the capability to detect colors and shapes. However, it is not without limitations, including diminished performance in extreme weather conditions, limited range and angle of view, and susceptibility to occlusion and glare. Additionally, managing the data generated by the camera presents a challenging task, averaging between 20-40 MB/Sec [24]. Another sensor commonly used is LiDAR, offering higher accuracy than a camera for positioning. Nevertheless, LiDAR generates substantial data volumes, ranging from 10-70 MB/Sec., necessitating considerable computing power for real-time processing [16]. Additionally, a sensor designed to measure the speed of a moving vehicle is known as a Doppler Velocity Log (DVL). It provides precise and consistent speed measurements in any direction, irrespective of the vehicle's orientation or movement. However, its performance is influenced by weather and environmental factors. [12]. Other sensors can assist the system, such as a magnetometer [28], which measures the magnitude and direction of the magnetic fields. A barometer uses the atmospheric pressure to calculate the altitude [14]. Furthermore, 5G networks can support the GPS through some map applications or initialization states [13, 17]. Nevertheless, ensuring network coverage for the entire environment poses a challenging task, demanding the deployment of more base stations, which incurs significant costs. Consequently, augmenting sensors introduces numerous challenges, including escalating system costs, heightened computational load, increased storage utilization, elevated system failure rates, issues related to matching and timing, and the management of battery resources. As a result, alternative approaches such as processing algorithms have been proposed to address the challenges associated with integration.

The second approach involves data fusion and processing algorithms, commonly referred to as classic algorithms due to their analytical and iterative-based equations. The Kalman Filter (KF) stands out as the preferred choice among experts for various applications. KF is adept at operating in linear form, with the assumption that the system's noise follows a Gaussian distribution. An enhanced method, the Extended Kalman Filter (EKF), is particularly robust as it can account for the system's non-linearity by employing the Taylor series [20]. While the Extended Kalman Filter (EKF) is well-suited for handling non-linearity, it exhibits limitations in extreme scenarios. To address this, the Unscented Kalman Filter (UKF) has been introduced. The UKF employs a deterministic sampling method to propagate a Gaussian random variable through the non-linear system, in contrast to the EKF's utilization of a linearization method. The UKF offers advantages over the EKF, including greater precision, elimination of the need for Jacobians or Hessians, and the capability to handle non-linearities that may be discontinuous or non-differentiable [6]. In contrast, the Unscented Kalman Filter (UKF), while advantageous, is not exempt from limitations. It introduces challenges such as heightened computational demands, complexities associated with high-dimensional state vectors, and reliance on specific assumptions and approximations. To address these concerns, the Cubature Kalman Filter (CKF) has been developed as a method for enhancing non-linear estimation, particularly improving upon the limitations of the UKF [29]. CKF uses a set of cubature points to compute the mean and covariance of the state distribution of a non-linear system with additive Gaussian noise. CKF is one of the nearest approximations to the Bayesian filter and has some benefits over UKF, such as higher precision, no need for Jacobians or Hessians, and the ability to handle non-linearities that are discontinuous or non-differentiable [29, 19]. Moreover, there are new methods of KF, such as Kalman-Neural Network (NN), which improve the KF and EKF in robustness and non-linearity [9, 23]. Within these methodologies, learning algorithms play a crucial role in managing uncertain conditions. For instance, the Long Short-Term Memory (LSTM) network proves beneficial in estimating Kalman Filter (KF) gain, presenting an alternative to traditional equations [8, 2, 26, 4, 10, 1, 30, 11].

This paper introduces Faded Memory Kalman Filter (FMKF) to enhance the performance of the GPS/INS integration system in land-based navigation. In this scenario, the KF deteriorates and deviates or oscillates from true values over time. By fading the covariance of measurement noise, the system accuracy improves and the problems are solved. The remains of this article are organized as follows. In Section 2, the system architecture is fully described. In Section 3 the KF and FMKF methods are illustrated. Next, the experimental results of the proposed method performed and investigated are mentioned in Section 4. Finally, the conclusion is presented in Section 5.

2 System Methodology and Mathematical Model

The system architecture consists of three main components: data collectors, data processors, and core modules, as shown in Fig. 1. The data collectors, such as IMU and GPS, capture the physical parameters of the system. The data processors perform basic mathematical operations on the data. The core modules such as mechanization block, KF, and INS error modeling apply the complex mathematical models that determine the system behavior. Table 1 also provides the parameters and definitions for navigation used in the following sections.

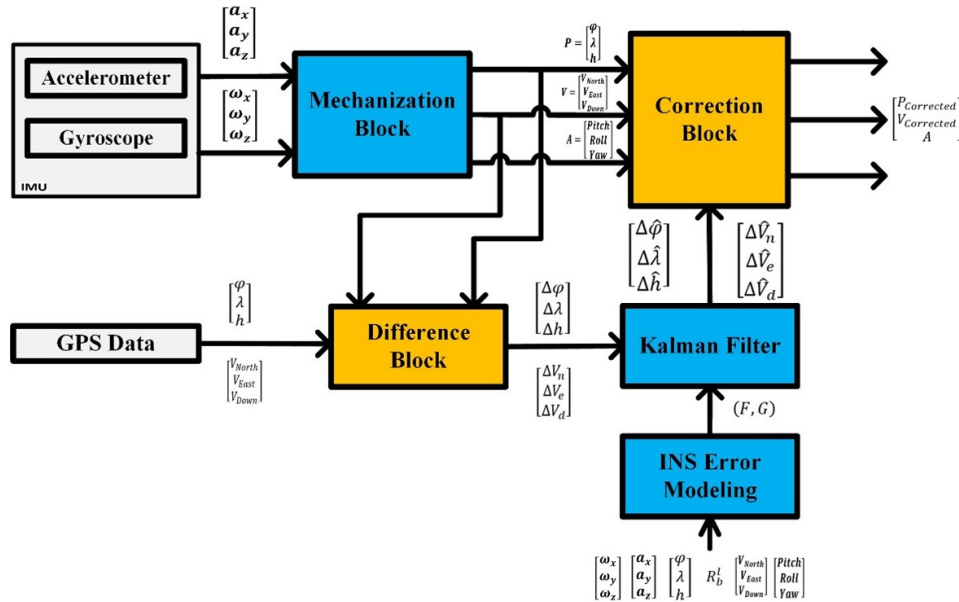


Figure 1: Overall architecture of the system

2.1 Data Collectors

The system consists of two sensors: IMU and GPS. The IMU measures the linear acceleration and angular velocity of the system in the body frame coordinates. The GPS calculates the position and velocity of the system in the global frame coordinates. The IMU has different sampling rates depending on the application.

2.2 Data Processing Elements

The data processing consists of elements such as difference block and correction block, as described in Table 2.

2.3 Main Blocks

This section consists of three main components: mechanization, INS error modeling, and KF. In the following, mechanization block and INS error modeling are described.

2.3.1 Mechanization Block

This block computes the attitude, velocity, and position of the vehicle using the accelerometer and gyroscope data. Eq. (2.1) shows the compact formula of the mechanization block:

$$\begin{bmatrix} \dot{\mathbf{r}}^l \\ \dot{\mathbf{v}}^l \\ R_l^b \end{bmatrix} = \begin{bmatrix} D^{-1} \mathbf{v}^l \\ R_b^l \mathbf{f}^b - (2\Omega_{ie}^l + \Omega_{el}^l) \mathbf{v}^l + g^l \\ R(q_w, q_x, q_y, q_z) \end{bmatrix} \quad (2.1)$$

The details of the Eq. (2.1) is described in Table 1. Figure 2 shows the internal structure of this block. The mechanization block receives the IMU's information $(f_{(x,y,z)}^b, w_{ib(x,y,z)}^b)$. It then calculates the quaternions (Eq. (2.2)) and the rotation matrix using the third row of Eq. (2.1). The rotation matrix has two purposes: obtaining the Euler

Parameters					
φ	Latitude	v_d	Down velocity	Euler angles (Attitude)	Pitch, roll, yaw
λ	Longitude	r^l	$[\varphi, \lambda, h]$	$f_{x,y,z}^b$ (Accelerometer)	$[f_x, f_y, f_z]^T$
h	Altitude from sea level	v^l	$[v_n, v_e, v_d]^T$	$w_{i b_{x,y,z}}^b$ (Accelerometer)	$[w_x, w_y, w_z]^T$
v_n	North velocity		Equatorial radius	$a = 6378137.0m$	
v_e	East velocity		Eccentricity	$e = 0.08181919$	

Title	Definition and formula
Local-level frame (l)	Fixed and locally oriented coordinate system attached to the Earth's surface.
Body frame (b)	Moving and rotating coordinate system attached to the vehicle or sensor.
Earth-Centered Earth-Fixed (ECEF) (e)	Rotating coordinate system attached to the Earth's center.
ω_{il}^l	Rotation of the Earth about its spin axis ($\omega_{ie}^e = 15 \circ /hr$) which is interpreted in the local-level frame.
\dot{r}^l and \dot{v}^l	The derivative of those vector.
Rotation matrix computation (q_w, q_x, q_y, q_z)	$R_l^b = \begin{bmatrix} q_w^2 + q_x^2 - q_y^2 - q_z^2 & 2(q_x q_y - q_w q_z) & 2(q_x q_z + q_w q_y) \\ 2(q_x q_y + q_w q_z) & q_w^2 - q_x^2 + q_y^2 - q_z^2 & 2(q_y q_z - q_w q_x) \\ 2(q_x q_z + q_w q_y) & 2(q_y q_z - q_w q_x) & q_w^2 - q_x^2 - q_y^2 + q_z^2 \end{bmatrix}$ $R_l^b = (R_l^b)^T = (R_l^b)^{-1}$
Gravity computation	$g^l = [0 \quad 0 \quad \gamma]^T$ $\gamma = a_1(1 + a_2 \sin^2 \varphi + a_3 \sin^4 \varphi) + (a_4 + a_5 \sin^2 \varphi)h + a_6 h^2$ $a_1 = 9.7803267714 \text{ m/s}^2, a_2 = 0.0052790414, a_3 = 0.0000232718,$ $a_4 = -0.0000030876910891 \text{ 1/s}^2, a_5 = 0.0000000043977311 \text{ 1/s}^2,$ $a_6 = 0.0000000000007211 \text{ 1/(m}^2\text{s}^2\text{)}.$
Position transition matrix	$D^{-1} = \begin{bmatrix} \frac{1}{(R_N + h) \cos \varphi} & 0 & 0 \\ 0 & \frac{1}{R_M + h} & 0 \\ 0 & 0 & -1 \end{bmatrix}$

Table 1: Parameters and definitions

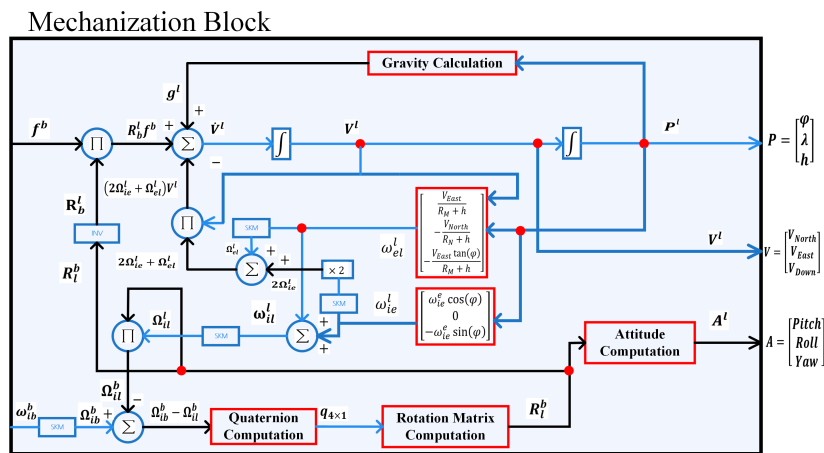


Figure 2: Details of mechanization block

Table 2: Data processing elements.

Name	Description	Formula and structure
Difference Block	This block determines the KF inputs. The KF inputs are the first six equations when the GPS is valid. When the GPS is invalid, the KF inputs are the last two equations.	$\begin{cases} \Delta\varphi_{INS-GPS} = \varphi_{INS} - \varphi_{GPS} _k \\ \Delta\lambda_{INS-GPS} = \lambda_{INS} - \lambda_{GPS} _k \\ \Delta h_{INS-GPS} = h_{INS} - h_{GPS} _k \\ \hline (\Delta V_{North})_{INS-GPS} = (V_{North})_{INS} - (V_{North})_{GPS} _k \\ (\Delta V_{East})_{INS-GPS} = (V_{East})_{INS} - (V_{East})_{GPS} _k \\ (\Delta V_{Down})_{INS-GPS} = (V_{Down})_{INS} - (V_{Down})_{GPS} _k \end{cases}$
Correction Block	This block corrects the output of the INS. The conditions of the equations is same as difference block.	$\begin{cases} \varphi_{Corrected} = \varphi_{INS} - \Delta\varphi_{Kalman} _k \\ \lambda_{Corrected} = \lambda_{INS} - \Delta\lambda_{Kalman} _k \\ h_{Corrected} = h_{INS} - \Delta h_{Kalman} _k \\ \hline (\Delta V_{North})_{Corrected} = (V_{North})_{INS} - (\Delta V_{North})_{Kalman} _k \\ (\Delta V_{East})_{Corrected} = (V_{East})_{INS} - (\Delta V_{East})_{Kalman} _k \\ (\Delta V_{Down})_{Corrected} = (V_{Down})_{INS} - (\Delta V_{Down})_{Kalman} _k \end{cases}$

angles and transforming the $(f_{(x,y,z)}^b)$ from the body frame to the local-level frame. Next, it integrates the gravity, acceleration, and $\omega_{eI}^l, \omega_{ie}^l$ data to obtain the velocity. Then, it derives the position from the velocity.

Also, the gravity block calculates the Earth's gravity at a specific location, as described in Table 1. The quaternion computation block derives the quaternion to obtain the rotation matrix (R_i^b) . It is given by Eq. (2.2) [27, 21]:

$$\mathbf{q}_{4 \times 1}|_{k+1} = \mathbf{q}_{4 \times 1}|_k + 0.5[2(\cos \frac{\theta}{2} - 1)I_{4 \times 4} + \frac{2}{\theta} \sin \frac{\theta}{2} S(\omega)]\mathbf{q}_{4 \times 1}|_k \quad (2.2)$$

where $\theta = \sqrt{(\omega_{lb,x}^b)^2 + (\omega_{lb,y}^b)^2 + (\omega_{lb,z}^b)^2}$ and $S(\omega)$ is given by Eq. (2.3):

$$S(\omega) = \begin{bmatrix} -(\Omega_{lb}^b)_{3 \times 3} & (\omega_{lb}^b)_{3 \times 1} \\ (\omega_{lb}^b)^T_{1 \times 3} & 0 \end{bmatrix}_{4 \times 4} \quad (2.3)$$

Moreover, for finding the Euler's angles, the attitude computation block derives the *pitch*, *roll* and *yaw* angles, described in Eq. (2.4):

$$pitch = \tan^{-1} \frac{R_b^l(3, 2)}{R_b^l(3, 3)}, roll = \sin^{-1} -R_b^l(3, 1), yaw = \tan^{-1} \frac{R_b^l(2, 1)}{R_b^l(1, 1)} \quad (2.4)$$

where $R_b^l(i, j)$ denotes the i^{th} row of the j^{th} column value of the rotation matrix.

2.3.2 INS Error Modeling

The system model is given by Eq. (2.5):

$$\delta \dot{\mathbf{x}}_{15 \times 1} = F_{15 \times 15} \delta \mathbf{x}_{15 \times 1} + \mathbf{g}_{15 \times 1} w \quad (2.5)$$

The state vector $(\delta \mathbf{x})$ includes error components of the position, velocity, and attitudes, as well as accelerometer biases and gyroscope drifts, attained by the second-order of Gauss-Markov equations [21]. The details of the Eq. (2.5) is explained in Table 3.

3 Faded Memory Kalman Filter

The first part of the KF is the system model, equal to Eq. (2.5). The second part of the KF is the measurement model expressed in Eq. (3.1). The details of Eq. (3.1) is described in Table 4.

$$\delta z_{6 \times 1} = H_{6 \times 15} \delta x_{15 \times 1} + \eta_{6 \times 1}. \quad (3.1)$$

Table 3: Equation (2.5)'s description

Title	Description and Formula
Expanded version of δx	$\delta x_{15 \times 1} = \begin{bmatrix} \delta r_{3 \times 1}^l & \delta v_{3 \times 1}^l & \underline{\epsilon}_{3 \times 1}^l & \delta \omega_{3 \times 1}^l & \delta f_{3 \times 1}^l \end{bmatrix}^T$ $= \begin{bmatrix} \underbrace{\delta \varphi, \delta \lambda, \delta h}_{\text{Position Errors}}, & \underbrace{\delta v_n, \delta v_e, \delta v_d}_{\text{Velocity Errors}}, & \underbrace{\delta p, \delta r, \delta A}_{\text{Attitude Errors}}, & \underbrace{\delta \omega_x, \delta \omega_y, \delta \omega_z}_{\text{Gyro Errors}}, & \underbrace{\delta f_x, \delta f_y, \delta f_z}_{\text{Accelerometer Errors}} \end{bmatrix}$
Noise distribution vector (g)	$\underline{g} = [\underline{\sigma}_r, 1 \times 3 \quad \underline{\sigma}_v, 1 \times 3 \quad \underline{\sigma}_\epsilon, 1 \times 3 \quad \underline{\sigma}_\omega, 1 \times 3 \quad \underline{\sigma}_f, 1 \times 3]^T$ <p>$\underline{\sigma}_r, 1 \times 3; \underline{\sigma}_v, 1 \times 3$ and $\underline{\sigma}_\epsilon, 1 \times 3$ are knowledge base and depend on the sensors and applications with quantity $[10^{-7}, 10^{-12}]$.</p> <p>$\underline{\sigma}_\omega, 1 \times 3$ and $\underline{\sigma}_f, 1 \times 3$ are obtained from the second-order of Gauss-Markov equations.</p>
F is the dynamic coefficient matrix	$F = \begin{bmatrix} 0_{3 \times 3} & F_r & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & F_v & 0_{3 \times 3} & R_b^l \\ 0_{3 \times 3} & F_\epsilon & 0_{3 \times 3} & R_b^l & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & F_\omega & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & F_f \end{bmatrix}_{15 \times 15}$ $F_r = \begin{bmatrix} \frac{1}{(R_N+h) \cos \phi} & 0 & 0 \\ 0 & \frac{1}{R_M+h} & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad F_\epsilon = \begin{bmatrix} \frac{1}{R_M+h} & 0 & 0 \\ 0 & \frac{-1}{R_N+h} & 0 \\ 0 & \frac{-\tan \varphi}{R_N+h} & 0 \end{bmatrix} \quad F_f = \begin{bmatrix} -\beta_{fx} & 0 & 0 \\ 0 & -\beta_{fy} & 0 \\ 0 & 0 & -\beta_{fz} \end{bmatrix}$ $F_v = \begin{bmatrix} 0 & f_d & -f_e \\ -f_d & 0 & -f_n \\ f_e & f_n & 0 \end{bmatrix} \quad F_\omega = \begin{bmatrix} -\beta_{\omega x} & 0 & 0 \\ 0 & -\beta_{\omega y} & 0 \\ 0 & 0 & -\beta_{\omega z} \end{bmatrix} \quad \begin{cases} \beta_\omega(x, y, z) \\ \beta_f(x, y, z) \end{cases} = \text{Gauss Markov Equation}$

After describing the navigation-based KF's equations, The KF steps are summarized in Table 5. The KF has two main steps (prediction and correction) and one minor (initialization or prior information).

A system model may have a finite validity period, beyond which it fails to capture the system dynamics accurately[25]. To mitigate this issue, attenuating the older measurements is preferred over time, which can be done by incrementally increasing the noise covariance R of those measurements to reduce their impact on the current estimate. In mathematical terms:

$$R_k = R \alpha^{-(k+1)} \quad (3.2)$$

where α is a positive constant and fading factor. Over time k becomes greater and R_k gradually decreases, thereby giving greater weight to the most recent data.

4 Results and Discussions

This paper has described the in the previous sections, and the next sections will report the experiments and results. The scenario is car navigation in urban land with the MPU6050 sensor (IMU, and u-blox 6 GPS. The IMU has a rate of 100 Hz, and the GPS has a rate of 1 Hz. This sensor is common and is used for commercial purposes. Table 6 shows the information and 2D path of the trajectories. Moreover, trajectories 1, 3, 4, 5, and 6 are local datasets (near campus) with lower velocity and position changes. Trajectories 2, 7, 8, and 9 are highway datasets with higher velocity and some traffic congestion.

To assess the performance of the FKF and KF, Root Mean Square Error (RMSE) and Signal-to-Error-Ratio (SER) are used. RMSE show the error from true path in XYZ based on meter. The formulation of the RMSE is described in Eq. (4.1) and total RMSE is described in Eq. (4.2):

$$\text{RMSE}_{x,y,z} = \sqrt{\frac{1}{\text{Number of Samples}} \sum_{i=1}^{\text{Number of Samples}} (\text{True Value}_i - \text{Estimated value}_i)^2} \quad (4.1)$$

$$\text{RMSE}_{\text{Total}} = \sqrt{\text{RMSE}_x^2 + \text{RMSE}_y^2 + \text{RMSE}_z^2} \quad (4.2)$$

On the other hand, the SER shows the power of the signal to error and is calculated by Eq. (4.3). The MSS is the mean-squared value of the actual data and MSE is the mean-squared value of the prediction error [15].

$$\text{SER}_{x,y,z} = 10 \log_{10} \left| \frac{\text{MSS}}{\text{MSE}} \right|_{x,y,z} \quad (4.3)$$

Table 4: Details of the Eq. (3.1)

Title	Formula	Description
State vector	$\delta x_{15 \times 1}$	Contains the INS's errors.
Measurement vector	$\delta z = \begin{bmatrix} r_{INS}^l - r_{GPS}^l \\ v_{INS}^l - v_{GPS}^l \end{bmatrix} = \begin{bmatrix} \phi_{INS} - \phi_{GPS} \\ \lambda_{INS} - \lambda_{GPS} \\ h_{INS} - h_{GPS} \\ v_{n,INS} - v_{n,GPS} \\ v_{e,INS} - v_{e,GPS} \\ v_{d,INS} - v_{d,GPS} \end{bmatrix}$	The difference of the GPS's and INS's position and velocity.
Measurement noise	$\eta_{6 \times 1}$	Zero-mean with covariance R .
Measurement design matrix	$H = [I_{6 \times 6} \quad 0_{6 \times 9}]_{6 \times 15}$	Measurements directly correspond to the position and velocity error states.
Variances of the measured states	$R = \begin{bmatrix} \sigma_\phi^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_\lambda^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_h^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{v_n}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{v_e}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{v_d}^2 \end{bmatrix}_{6 \times 6}$	Covariance of measurement noise and is selected knowledge based.
The state prediction covariance matrix	$P = \begin{bmatrix} \sigma_r^2 I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & \sigma_v^2 I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & \sigma_e^2 I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & \sigma_\omega^2 I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & \sigma_f^2 I_{3 \times 3} \end{bmatrix}_{15 \times 15}$	All σ terms are 3×3 diagonal matrices associated with the position, velocity, attitude, gyroscope's bias/drift, and accelerometer's bias/drift.

Table 5: KF time update prediction and correction equations (k is the discrete time step)

Definition	Formula	Description
Initialization: A priori information	R_0, Q_0 $\hat{\delta x}_0, P_0$	The matrix R_0, Q_0 are estimated on the basis of prior experience with the system and tuned to get the best estimates of the states.
Prediction	System dynamic model: $\hat{\delta x}_k = F_{k k-1} \hat{\delta x}_{k-1} + g_{k-1}^T w_{k-1}$ $w_k \sim N(0, Q_k)$ State vector prediction: $\hat{\delta x}_k^- = F_{k k-1} \hat{\delta x}_{k-1}^+$ Covariance prediction: $P_k^- = F_{k k-1} P_{k-1}^- F_{k k-1}^T + g_{k-1}^T Q_{k-1} g_{k-1}$	The $F_{k k-1}$ is computed and then, using this matrix, the initial state is propagated from the epoch $k-1$ to k , which is denoted by $\hat{\delta x}_k^-$. The covariance of the predicted state P_k^- is based on $F_{k k-1}$, the previous value of the state covariance (P_{k-1}^-), the last value of the process noise covariance (Q_{k-1}), and the noise distribution matrix (g_{k-1}^T).
Correction	Measurement model: $\delta z_k = H_k \delta x_k + \eta_k$ $\eta_k \sim N(0, R_k)$ Kalman gain matrix: $K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$ Corrected state estimation: $\hat{\delta x}_k^+ = \hat{\delta x}_k^- + K_k (c - H_k \hat{\delta x}_k^-)$ Corrected covariance matrix: $P_k^+ = (I - K_k H_k) P_k^- (I - K_k H_k)^T + K_k R_k K_k^T$	The Kalman gain (K_k) depends on the a priori error covariance (P_k^-), the process noise covariance (R_k), and the design matrix (H_k). The estimated (or the a priori) state ($\hat{\delta x}_k^-$) is corrected whenever a measurement is received. This is based on the difference of the predicted measurement ($H_k \hat{\delta x}_k^-$) and the real measurement (δz_k). This difference contains the new information that forms the basis for the correction. updating the a priori error covariance (P_k^-) to the a posteriori error covariance (P_k^+) for indicating the level of trust in the corrected estimate ($\hat{\delta x}_k^+$), which is proportional to gain K_k and P_k^- .

In this research different constant numbers for α are utilized and $\alpha = 2$ is chosen by trial and error method for fading factor. In this regard, the performance of the system for these nine trajectories is described in Table 7.

Fig. 3 and Fig. 4 show the latitude and longitude values for all trajectories, respectively. As Table 7, Fig. 3, and Fig. 4 indicate, the simple KF and FKF have the same performance for Tr. 3, Tr. 4, and Tr. 5. This is due to the short duration and stable values of the system that follows the proper trajectory closely. However, as mentioned before, the system becomes more complex. Dynamic changes affect its performance, as seen at the end of Tr. 1, Tr. 2, and Tr. 6. When the trajectory is longer and more complex, the KF exhibits weak performance, as in Tr. 7, Tr. 8, and Tr. 9. In a traditional KF, every sensor measurement has the same weight in the estimation process. This implies that older measurements still influence the current estimates equally. Therefore, the predictions may not match the current conditions accurately. In contrast, a fading KF reduces the impact of older measurements by assigning them lower weights based on a fading factor. Thus, the system performance for all trajectories improves compared to the

Table 6: Information and path of trajectories.


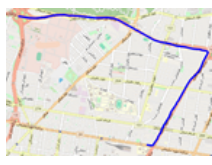
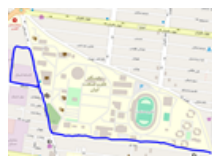






Title	Inform.	Path	Inform.	Path	Inform.	Path
Time series Duration (s)	Trajectory 1 418 s		Trajectory 2 554 s		Trajectory 3 244 s	
Time series Duration (s)	Trajectory 4 90 s		Trajectory 5 107 s		Trajectory 6 498 s	
Time series Duration (s)	Trajectory 4 1163 s		Trajectory 5 1407 s		Trajectory 6 911 s	

Table 7: Result of KF, FKF, and without any algorithm for all trajectories.

Path	Duration	Type of algorithm	SER X (dB)	SER Y (dB)	SER Z (dB)	RMSE X (m)	RMSE Y (m)	RMSE Z (m)	RMSE Total (m)
Tr. 1	418 s	Without anything	51	80	60	8559	369	3619	9299
		KF	89	89	94	108	137	71	188
		FKF	103	108	107	22	14	14	29
Tr. 2	554 s	Without anything	48	81	59	11788	324	3762	12377
		KF	92	105	97	75	22	49	92
		FKF	97	106	104	44	19	22	52
Tr. 3	244 s	Without anything	74	67	83	590	1726	253	1841
		KF	103	112	106	22	9	16	28
		FKF	103	112	107	20	9	15	26
Tr. 4	90 s	Without anything	95	85	86	57	225	183	295
		KF	104	115	105	18	6	19	26
		FKF	105	116	106	17	5	17	24
Tr. 5	107 s	Without anything	83	87	84	205	171	224	348
		KF	103	109	107	20	14	15	28
		FKF	104	109	107	19	14	14	27
Tr. 6	498 s	Without anything	67	63	68	1417	2724	1467	3381
		KF	102	108	108	22	14	13	29
		FKF	104	109	111	18	13	9	23
Tr. 7	1163 s	Without anything	42	46	45	23717	18230	18662	35257
		KF	85	85	80	162	205	335	424
		FKF	95	100	98	53	33	41	74
Tr. 8	1407 s	Without anything	39	39	47	32803	44953	15701	57821
		KF	80	83	81	317	270	328	530
		FKF	95	100	94	53	40	68	95
Tr. 9	911 s	Without anything	55	48	53	5266	15759	8224	18539
		KF	85	89	86	168	138	174	278
		FKF	97	100	106	43	40	17	61

traditional KF. Table 8 illustrates the improvement from simple KF to FKF. The average improvement of FKF is 45% for all trajectories.

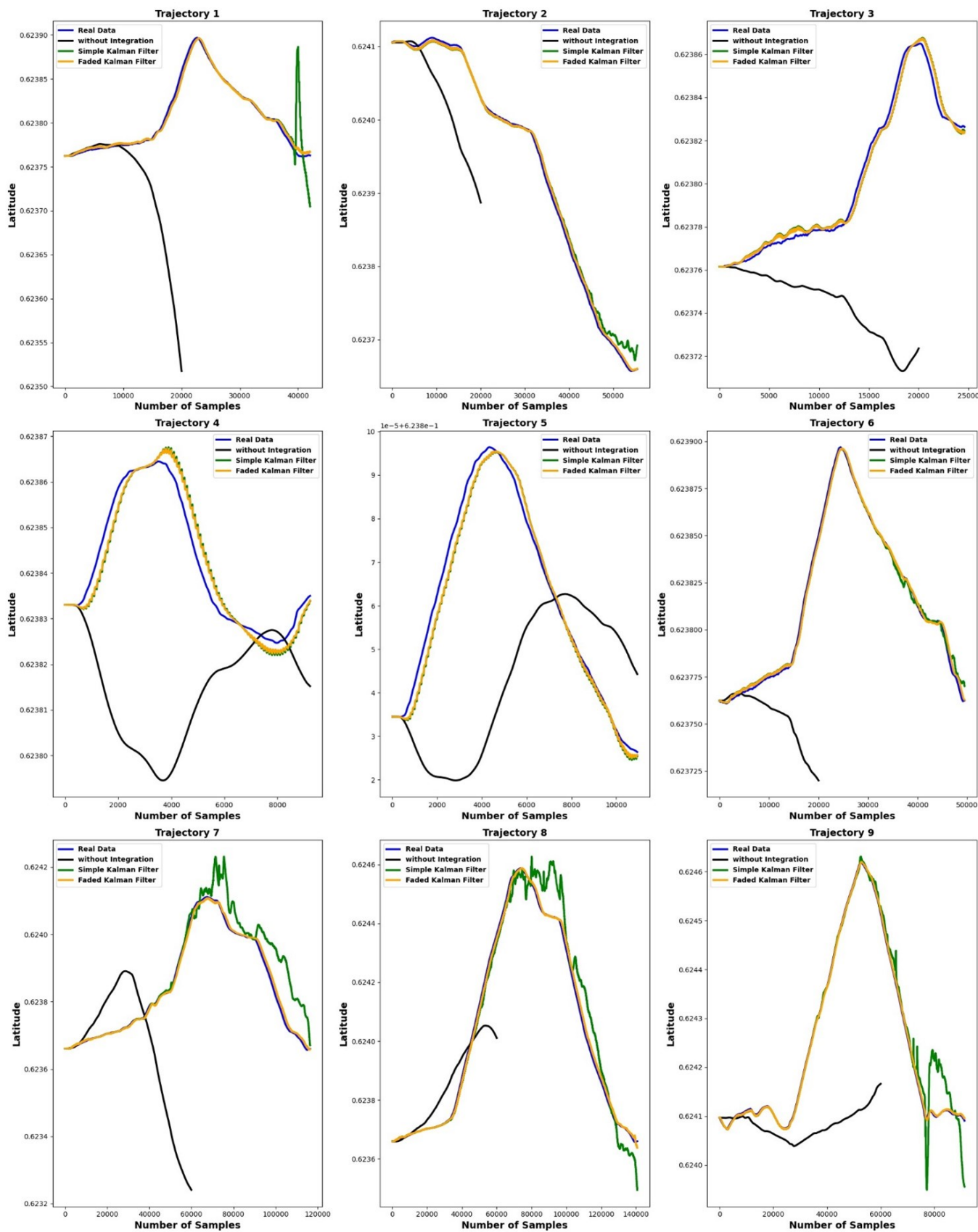


Figure 3: Latitude (in radian) figures of all trajectories.

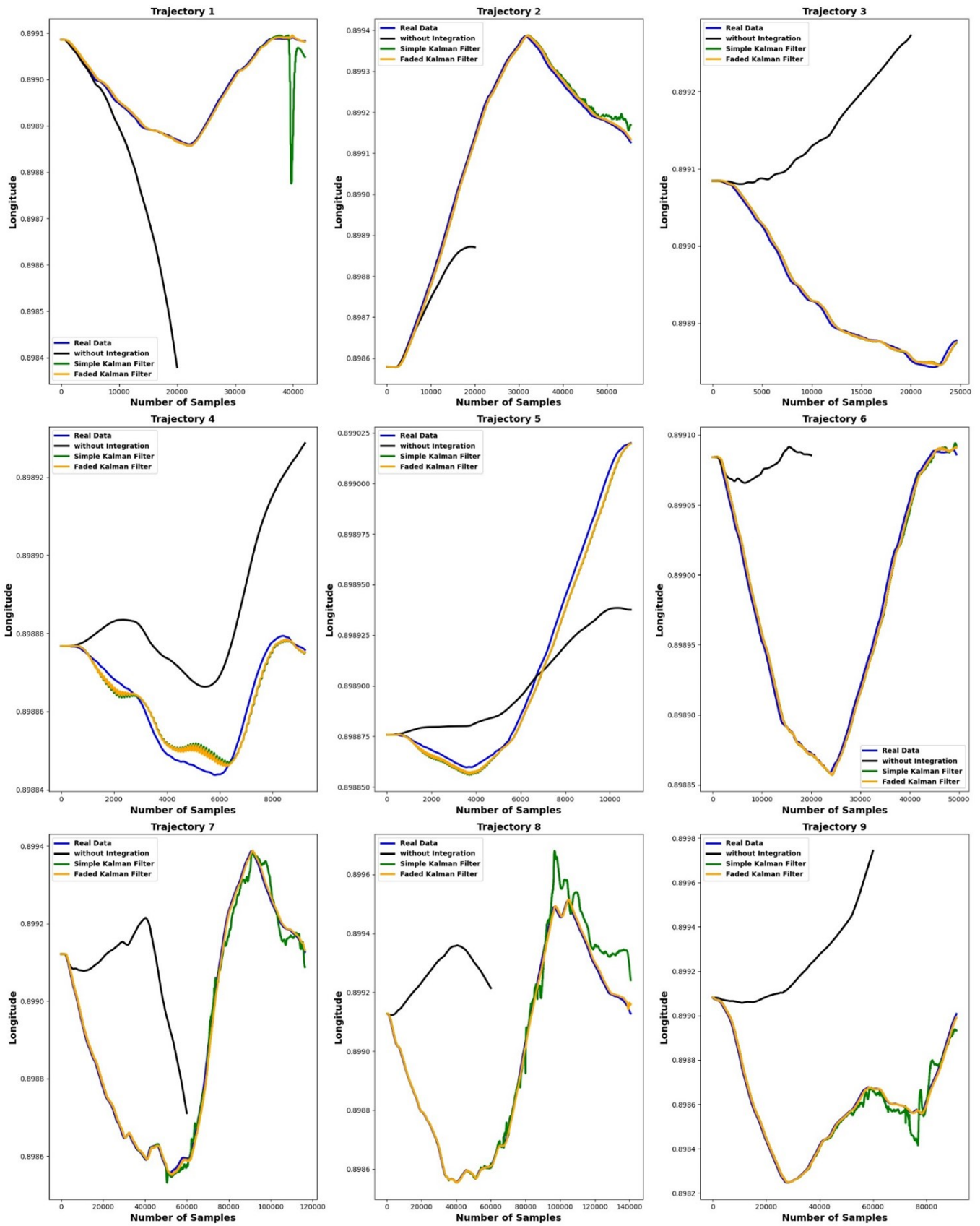


Figure 4: Longitude (in radian) figures of all trajectories.

Table 8: results of improvement for all trajectories.

Trajectory	Tr. 1	Tr. 2	Tr. 3	Tr. 4	Tr. 5	Tr. 6	Tr. 7	Tr. 8	Tr. 9
Duration (s)	418 s	554 s	244 s	90 s	107 s	498 s	1163 s	1407 s	911 s
Improvement (%)	84%	43%	7%	7%	3%	20%	82%	82%	78%

5 Conclusion

This article proposed the FMKF method to enhance the positioning accuracy of GPS/INS integration. Implementing a faded KF involves modifying the traditional KF equations to introduce fading factors for update steps. The fading factors control how much influence past observations have on the current estimation. The FKF is used on a real dataset with multiple trajectories at various speeds and complexities. The performance of FKF is compared with traditional KF in GPS/INS integration. The results showed that the FKF can reduce position errors by an average of 45% compared to simple KF.

References

- [1] E.S. Abdolkarimi, G. Abaei, and M.R. Mosavi, *A wavelet-extreme learning machine for low-cost INS/GPS navigation system in high-speed applications*, GPS Solutions **22** (2018), no. 1, 15.
- [2] E.S. Abdolkarimi and M.R. Mosavi, *Wavelet-adaptive neural subtractive clustering fuzzy inference system to enhance low-cost and high-speed INS/GPS navigation system*, GPS Solutions **24** (2020), no. 2, 35.
- [3] E.S. Abdolkarimi and M.R. Mosavi, *A low-cost integrated MEMS-based INS/GPS vehicle navigation system with challenging conditions based on an optimized IT2FNN in occluded environments*, GPS Solutions **24** (2020), no. 4, 1–19.
- [4] E.S. Abdolkarimi and M.R. Mosavi, *A modified neuro-fuzzy system for accuracy improvement of low-cost MEMS-based INS/GPS navigation system*, J. Wireless Personal Commun. **129** (2023), 1369–1392.
- [5] E.S. Abdolkarimi, M.R. Mosavi, S. Rafatnia, and D. Martin, *A hybrid data fusion approach to AI-assisted indirect centralized integrated SINS/GNSS navigation system during GNSS outage*, IEEE Access **9** (2021), 100827–100838.
- [6] S. Cao, H. Gao, and J. You, *In-flight alignment of integrated SINS/GPS/polarization/geomagnetic navigation system based on federal UKF*, Sensors **22** (2022), no. 16, 5985.
- [7] H. Alaeiyan, M.R. Mosavi, and A. Ayatollahi, *Hybrid noise removal to improve the accuracy of inertial sensors using lifting wavelet transform optimized by genetic algorithm*, Alexandria Engin. J. **80** (2023), 326–341.
- [8] A. Ebrahimi, M. Nezhadshahbodaghi, M.R. Mosavi, and A. Ayatollahi, *An improved GPS/INS integration based on EKF and AI during GPS outages*, J. Circ. Syst. Comput. **33** (2023), no. 3, 1–23.
- [9] M.A. El-Gendy, A.A. Atwan, and A.M. Moussa, *An integrated adaptive Kalman filter for improving the reliability of navigation systems*, J. Appl. Geodesy **16** (2022), no. 1, 1–14.
- [10] M.A. El-Gendy, A.A. Atwan, and A.M. Moussa, *GNSS/INS integration based on machine learning LightGBM model for vehicle navigation*, Appl. Sci. **12** (2022), no. 11, 5565.
- [11] W. Fang, J. Jiang, S. Lu, Y. Gong, Y. Tao, Y. Tang, P. Yan, H. Luo, and J. Liu, *A LSTM algorithm estimating pseudo measurements for aiding INS during GNSS signal outages*, Remote Sens. **12** (2020), no. 2, 256.
- [12] G. Fukuda and N. Kubo, *Application of initial bias estimation method for inertial navigation system (INS)/Doppler velocity log (DVL) and INS/DVL/gyrocompass using micro-electro-mechanical system sensors*, Sensors **22** (2022), no. 14, 5334.
- [13] J. Gante, L. Sousa, and G. Falcao, *Dethroning GPS: Low-power accurate 5G positioning systems using machine learning*, IEEE J. Emerg. Select. Topics Circ. Syst. **10** (2020), no. 2, 240–252.
- [14] C. Hajiyev, U. Hacizade, and D. Cilden-Guler, *Integration of barometric and GPS altimeters via adaptive data fusion algorithm*, Int. J. Adapt. Control Signal Process. **35** (2021), no. 1, 3–17.

- [15] S. Haykin, *Kalman Filtering and Neural Networks*, John Wiley & Sons, 2004.
- [16] H. Jiang, C. Shi, T. Li, Y. Dong, Y. Li, and G. Jing, *Low-cost GPS/INS integration with accurate measurement modeling using an extended state observer*, *GPS Solutions* **25** (2021), no. 1, 1–15.
- [17] G. Li, *Development of cold chain logistics transportation system based on 5G network and Internet of things system*, *Microproc. Microsyst.* **80** (2021), 103565.
- [18] D. Li, X. Jia, and J. Zhao, *A novel hybrid fusion algorithm for low-cost GPS/INS integrated navigation system during GPS outages*, *IEEE Access* **8** (2020), 53984–53996.
- [19] J. Li, J. Wang, Y. Zhang, and S. Wang, *Robust variational Bayesian method-based SINS/GPS integrated system*, *Measurement* **163** (2020), 107917.
- [20] J. Liu and G. Giu, *Vehicle localization during GPS outages with extended Kalman filter and deep learning*, *IEEE Trans. Instrument. Measur.* **70** (2022), 1–10.
- [21] A. Nouredin, T.B. Karamat, and J. Georgy, *Fundamentals of Inertial Navigation, Satellite-Based Positioning and their Integration*, Springer Science & Business Media, 2012.
- [22] A.G. Quinchia, G. Falco, E. Falletti, F. DAVIS, and C. Ferrer, *A comparison between different error modeling of MEMS applied to GPS/INS integrated systems*, *Sensors* **13** (2013), no. 8, 9549–9588.
- [23] G. Revach, N. Shlezinger, N. Xiaooyong, A.L. Escoriza, V. Sloun, J.G. Ruud, and C.E. Yonina, *KalmanNet: Neural network aided Kalman filtering for partially known dynamics*, *IEEE Trans. Signal Process.* **70** (2022), 1532–1547.
- [24] L. Rui, G. Klaus, C. Pengyu, and J. Nan, *Collaborative positioning method via GPS/INS and RS/MO multi-source data fusion in multi-target navigation*, *Survey Rev.* **54** (2022), no. 383, 95–105.
- [25] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*, John Wiley & Sons, 2006.
- [26] S. Wang, J. Li, Y. Zhang, and J. Wang, *Deep learning-enabled fusion to bridge GPS outages for INS/GPS integrated navigation*, *IEEE Sensors J.* **22** (2022), no. 9, 8974–8985.
- [27] J.L. Weston and D.H. Titterton, *Modern inertial navigation technology and its application*, *Electron. Commun. Engin. J.* **12** (2000), no. 2, 49–64.
- [28] Y. Zhang, J. Li, J. Wang and S. Wang, *Multi-rate strong tracking square-root cubature Kalman filter for MEMS-INS/GPS/polarization compass integrated navigation system*, *Control Engin. Practice* **118** (2021), 105184.
- [29] S. Zhao, Y. Zhou, and T. Huang, *A novel method for AI-assisted INS/GNSS navigation system based on CNN-GRU and CKF during GNSS outage*, *Remote Sens.* **14** (2022), no. 18, 4494.
- [30] Z. Zhi, D. Liu, and L. Liu, *A performance compensation method for GPS/INS integrated navigation system based on CNN-LSTM during GPS outages*, *Measurement* **188** (2022), 516–529.