Proceedings of the Indian National Science Academy (2026) 92:1282–1296
https://doi.org/10.1007/s43538-025-00488-x

RESEARCH PAPER

A novel attention based RNN-Kalman filter for accurate position forecasting framework

N. V. Maheswara Rao[1] · V. B. S. Srilatha Indira Dutt[2] · B. T. Krishna[3]

Received: 22 May 2025 / Accepted: 17 June 2025 / Published online: 7 July 2025
© Indian National Science Academy 2025

Abstract
Precise object location is the essence of the operation of the contemporary navigation systems that are predominantly based on GPS and INS. The ultimate goal of the proposed work is to predict the accurate position of the objects by improving the performance of the navigation system. The navigation system consists of a global positioning system (GPS) and an Inertial Navigation System (INS). In this work, we have developed a novel Hybrid RNN based Kalman filter. In addition, a Chaotic-Levy-based Particle Swarm Optimization (PSO) enhances the training efficiency of the proposed model. MATLAB is used as the main tool for implementing the proposed model. The hybrid RNN based Kalman filter system, combined with Chaotic-Levy-based PSO, shows excellent performance in predicting object positions accurately. The model achieved a high correlation value of 0.999, a very low Mean Absolute Error (MAE) of 0.03, Mean Absolute Relative Error (MARE) of 0.125, and Mean Squared Error (MSE) of just 0.00139. These results reflect the model's strong ability to learn from navigation data and make highly reliable predictions across different scenarios.

Keywords Navigation system · GPS/INS integration · Chaotic-Levy PSO · Object position prediction

Introduction

It has emerged in the recent past to offer positioning services using low Earth orbit (LEO) mega-constellations (Wang and El-Mowafy 2022) in contemporary navigation systems based on GPS and INS. One of the most general methods is calculation of space object orbital orbits with Two-Line Element Sets (TLEs), utilized in such operations as satellite laser ranging (SLR) (Kovář et al. 2022). The receiver positioning accuracy, however, depends not only on spatial distribution of satellite constellations but also on additional measurement errors such as satellite clock errors, ephemeris errors, and atmospheric delays (Najder and Sośnica 2021). These include such techniques as wide-area and local-area differential corrections, navigation message augmentation, and error masking, which are used to reduce them (Kong et al. 2022). Thus, satellite positioning has become a natural component of numerous mobile devices such as smartphones and handheld navigation units (Uradziński and Bakuła 2020).

Despite extensive usage, its performance seriously suffers in partially shaded environments such as urban canyons, flyover shadows, and indoor factory areas (Li et al. 2020b). GPS outages pose an enormous challenge to GPS/INS integrated navigation systems as they have high reliance on satellite signals (Cao et al. 2023). When GPS signals degrade or fail, the system resorts to INS alone, resulting in exponentially growing errors and lower accuracy (Gao et al. 2022). Although cooperative positioning (CP) is a prime candidate, comparatively few have tackled real-world datasets or the problem of internode ranging outliers, limiting practical application (Liu et al. 2021). Additionally, parameters like incidence angle, narrow measuring range, low scanning resolution, and sparse point cloud data limit the efficiency

✉ N. V. Maheswara Rao
himahesh4u@gmail.com

V. B. S. Srilatha Indira Dutt
svemuri@gitam.edu

B. T. Krishna
tkbattula@gmail.com

[1] Department of ECE, Gayatri Vidhya Parishad College of Engineering for Women, Visakhapatnam, India

[2] Department of ECE, GITAM University, Visakhapatnam, India

[3] Department of ECE, JNTUK University, Kakinada, Andhra Pradesh, India

of target detection. Under urban canyons, the issue is compounded with less satellite visibility, thereby complicating robust positioning even further (Pagoti and Vemuri 2022).

The classic GPS positioning techniques usually fail in accurate positioning within complex and blocked settings, giving rise to smart algorithmic technologies (Behivoke et al. 2021). Random forest, for instance, has been proven very reliable in modeling boat GPS traces and identifying space-explicit fisheries activities (Yang et al. 2023). DL frameworks such as CNNs and LSTM networks are most often utilized to acquire spatial and temporal relationships, and their performance further enhanced by spatial–temporal attention mechanisms (Li et al. 2020a). Real-time error estimation for positioning has been improved through deep neural networks (DNNs) run on edge servers. Gradient boosting decision trees (GBDT) help in the estimation of pseudorange errors with features like satellite elevation, signal strength, and residuals (Sun et al. 2020). Variable projection algorithms designed with the Aitken method have been found effective in reducing the impact of multipath effects to enhance the accuracy of GPS (Cheng et al. 2024).

This paper aims Hybrid RNN-informed Kalman filter model for improving prediction performance in sequence data analysis. As one further improvement over training efficacy and optimization procedure, a Chaotic-Levy-based Particle Swarm Optimization (PSO) algorithm is integrated into the model. This integration takes advantage of the capacity of the RNN to capture temporal dependencies and the Kalman filter to give reliable state estimation, since the Chaotic-Levy PSO offers improved convergence and more optimal parameter values. The model is simulated through MATLAB, which provides a flexible and efficient platform for simulation and experimentation. To develop a hybrid RNN-based Kalman filter model that effectively captures temporal dependencies and performs optimal state estimation for accurate object position prediction in navigation systems.To optimize the model's training process using a Chaotic-Levy-based Particle Swarm Optimization algorithm, ensuring efficient convergence and improved learning performance.To implement a structured preprocessing pipeline that removes null values, cleans noise, and aligns sequences, ensuring consistent and high-quality input data for the model.

The following are the sections of this paper: The relevant literature is reviewed in Sect. "Literature survey", the proposed technique is described in Sect. "Proposed methodology", the experimental results are thoroughly analysed in Sect. "Results and discussion", and the key findings of the proposed work and itsconsequences are discussed in Sect. "Conclusion".

Literature survey

Paszek and Grzecha (2023) proposed an artificial neural network (ANN) and LSTM for predicting object locations based on inertial navigation and UWB system historical data. The proposed method provides a reliable positioning system with less than 10 cm error and precisely predicts 10 mobile object positions at low and high speeds.

Stojković, and Tadić (2023) proposed a method that employs polynomial regression combined with a LSTM neural network to effectively process long-term dependencies as well as sequence data.The dependability and precision of the LSTM-based model was compared with the traditional Kalman Filter and was shown to improve precision and reliability. Yoon et al. (2024) presented a Bluetooth Low Energy signal-based indoor positioning technique with CNN-LSTM. The new technique uses the distances between neighboring transmitters to estimate the location of the receiver. The proximity method using solely the strongest signal as a substitute for triangulation, is used to estimate the location if the signal strength of a threshold value is surpassed.

Guang et al. (2021) suggested a technique that used the use of LSTM for position data estimation based on position data from the GPS and inertial measurement unit (IMU) data. Both static and dynamic simulations are used to investigate the range of values for LSTM hyperparameters. Tao et al. (2021) proposed a novel multipath mitigation approach in the coordinate domain. They use an LSTM (long short-term memory)-CNN (convolutional neural network) approach to mine the deep multipath properties in GNSS coordinate series, to overcome the multipath production mechanism.

Liao et al. (2021) introduced a new method that combines GPS embedding with an attention-based BiLSTM neural network for more precise prediction of passenger destinations. Two methods of GPS embeddings are proposed to recognize multi-scale spatial characteristics and geographic proximity, successfully encoding the spatial context of the visited urban places. Two large-scale benchmark datasets show superior prediction performance in evaluation.

Lv et al. (2022) proposed a model that employs the PSO article to optimize the initial parameters of the BP neural network, solving the instability and overfitting problems usually related to traditional BP networks. Li et al. (2022) discovered the use of DL models, to understand how multipath propagation influences GNSS correlation results. It shows the better performance of neural network-based schemes under line-of-sight conditions compared to conventional correlation schemes.

Ruwali et al. (2020) described the use of DL models to predict ionospheric delays on GPS signals based on ground and space-based ionospheric sensor measurements. The

use of the algorithms to predict low-latitude ionospheric disturbances is an interest with the evolution of artificial intelligence and deep learning (Table 1). Existing literature comparing different techniques is shown in Table 1.

The research in GNSS positioning, object location prediction, and multipath mitigation with DL models has progressed a lot, yet there are still some gaps. Most research works address particular environments (e.g., indoor, urban) or sensor data, which reduces the applicability of the models to other environments. More reliable real-time performance under dynamic and very complex situations, where the environment conditions may change drastically, is another open area. Additionally, although some models have high accuracy, they may do so at the expense of computational complexity. Combining a variety of data sources, for example, inertial sensors, GPS, and environmental parameters, is still unexplored territory, and these models' scalability in real-world applications is not yet fully leveraged. The resolution of these issues will produce more efficient and dependable positioning systems.

Table 1 Comparison of existing works

Existing work	Aim	Methods	Advantages	Disadvantages
Guang et al. (2021)	Estimate position information using GPS and IMU data	LSTM for position estimation, compared with Kalman Filter and inertial navigation system	Suitable for both static and dynamic scenarios, improves accuracy over Kalman filter	Limited to specific scenarios, hyperparameter tuning needed for best results
Kabir et al. (2025)	Improve positioning accuracy in GNSS systems using position corrections	PC-DeepNet with permutation-invariant DNN to estimate position corrections, NLOS and multipath features	Outperforms existing models in accuracy, lower computational complexity, handles GNSS challenges effectively	Depends on accurate feature extraction, performance may vary with satellite visibility
Li et al. (2022)	Study multipath propagation effects on GNSS correlation	DL models to filter out multipath noise and improve LOS signal separation	Better performance in line-of-sight (LOS) settings, filters noise effectively	May not perform well in multipath-dominated environments, limited to GNSS data
Liao et al. (2021)	Improve passenger destination prediction using GPS data	BiLSTM with GPS embedding and attention mechanism for destination prediction	Better accuracy in predicting destinations, incorporates spatial and temporal context	Limited to urban areas, depends on quality of GPS data
Lv et al. (2022)	Improve clock offset prediction using a BP neural network	HPSO-BP neural network optimized by particle swarm for clock offset prediction	Improved prediction accuracy, avoids overfitting, optimized using swarm optimization	Complex model, may require significant computational resources
Paszek and Grzechca (2023)	Forecast object locations using inertial navigation and UWB data	LSTM neural network for position prediction with inertial navigation and UWB data	High accuracy (less than 10 cm), reliable in forecasting positions at low and high speeds	Not applicable for all environments, specific to the tested conditions
Ruwali et al. (2020)	Predict ionospheric delays for GPS signals	LSTM, GRU, LSTM-ConvNet applied to TEC data for ionospheric delay prediction	Utilizes AI to predict ionospheric disturbances, improves low-latitude GPS performance	Requires large datasets and may not work well in areas with weak GPS signal reception
Stojković and Tadić (2023)	Predict and interpolate object location coordinates	LSTM with polynomial regression for sequential data, real-world vehicle and GPS data	Superior accuracy over Kalman Filter, handles GPS signal dropouts, real-time operation	Might require high computational power, dependent on data quality
Tao et al. 2021)	Mitigate multipath effects in GNSS coordinate series	CNN-LSTM for mining deep multipath characteristics in GNSS signals	Reduces multipath interference in GNSS, real-time forecasting	Potential computational cost for real-time operation in dense environments
Yoon et al. (2024)	Indoor positioning using Bluetooth Low Energy signals	CNN-LSTM model for estimating distances using signal strengths from nearby transmitters	Adaptively chosen placement to avoid interference, uses strong signal for positioning	Limited to indoor environments, may be affected by signal interference

Proposed methodology

The architecture is a hybrid deep learning architecture for effective sequence forecasting and time-series analysis. It starts with input data, which undergo preprocessing which involves null data removal, data cleaning, and sequence alignment. These operations render the data complete, noise-free, and in the appropriate format for modeling in the temporal domain. After preprocessing, data is processed by a hybrid system consisting of two Recurrent Neural Networks (RNNs) in Kalman filter form. Hybrid Kalman filter based on RNN utilizes the strengths of the use of temporal relationship by RNNs and optimal state estimation of the Kalman filter to enhance the model robustness against noise and outliers. The result of this hybrid system is subsequently optimized through a Chaotic-Levy-based Particle Swarm Optimization algorithm. The novel PSO method combines chaotic dynamics and Levy flight mechanics to deliver increased global search ability and prevent premature convergence, leading to improved parameter tuning and improved model performance. The final output generated by this pipeline can be made accessible for utilization in predictive analysis or classification tasks. Overall, this architecture is most suitable for intricate sequence-based problems such as biomedical signal processing, finance forecasting, or anomaly detection in time-series data. Figure 1 demonstrates the architecture of the proposed method.

Preprocessing

Preprocessing is also a crucial initial step in any data-driven model, aimed at enhancing the quality and uniformity of input data before they are passed into the learning framework. Preprocessing entails the removal of missing, noisy, or irrelevant data to pass clean and formatted input to the model. Preprocessing for this work comprises null data elimination, cleaning of data, and sequence alignment to preserve temporal consistency and minimize noise. These operations serve to normalize the INS and GPS data to align it across samples. Preprocessing operates more or less well in improving model stability along with accuracy in the model, minimizing computing overhead. Preprocessing is the basis for continued and fruitful feature extraction in the subsequent phase.

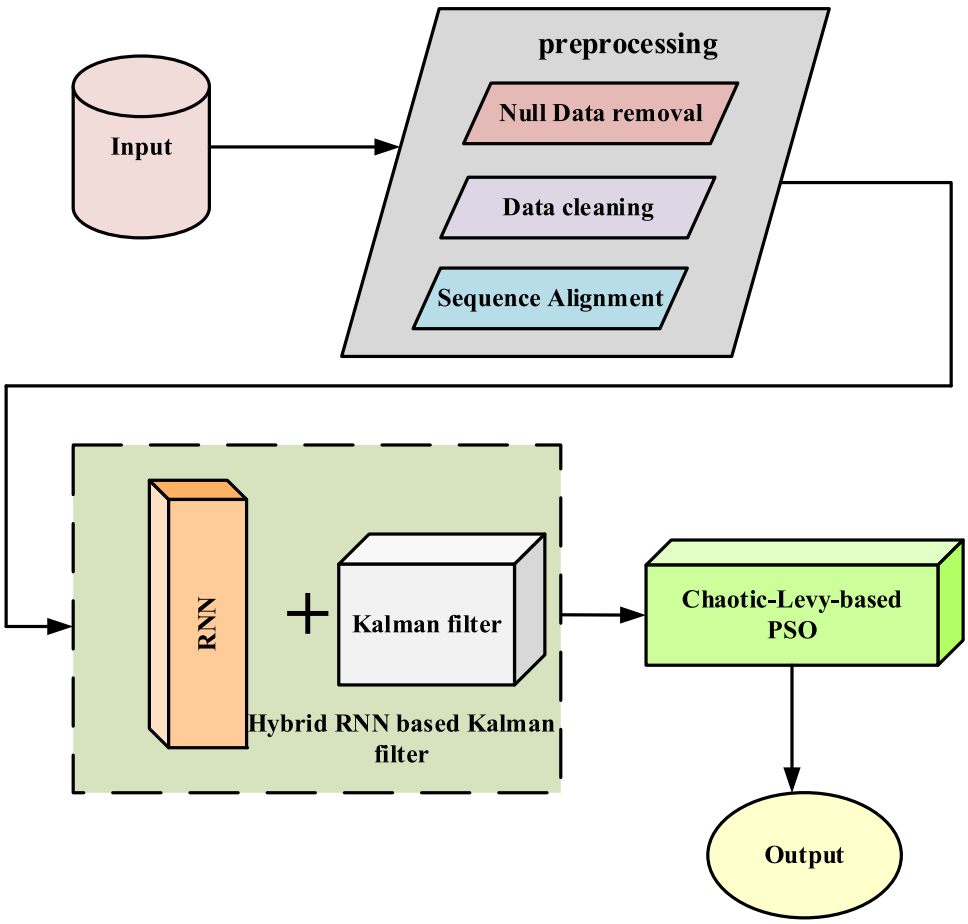


Fig. 1 Proposed architecture

Null data removal

The initial operation under data preprocessing is Null Data Removal, or the management of missing or incomplete data. Missing data may be due to many possible reasons such as sensor malfunction, loss of signal, or breakdown of data transmission. If the gaps are not addressed, the model may provide incorrect forecasts or cannot be relied upon. To deal with missing data, you can either delete the data points with missing values or fill them in using estimates. Common imputation techniques are to replace missing values with the mean, median, or other methods like linear interpolation. Proper treatment of null data makes the model operate on complete and valid data, enhancing its accuracy and trustworthiness.

Data cleaning

Data Cleaning is the second step, in which we are concerned with correcting errors, removing inconsistencies, and cleaning noise from the dataset. GPS and INS data sometimes have sudden spikes or drops that are not representative of actual sensor reading. Those outliers will mislead the model and make it less accurate. To address this, techniques like smoothing, i.e., employing moving averages or Kalman filtering, remove fluctuations so that only meaningful data is used. Also, outliers—points that are well outside the remaining data—are identified and removed or corrected. This cleaning makes the dataset trustworthy, accurate, and ready to be processed by the model.

Sequence alignment

Sequence Alignment guarantees that GPS and INS system data are properly aligned in time. GPS and INS data are typically measured at varying frequencies or with minor timing offsets, resulting in misalignment of the two sets of data. Misalignment can result in poor predictions, since the model would not be provided with an appropriate sequence of events. To address this, we resample using resampling methods to match the timestamps and interpolation to input missing data points. This phase ensures both data sets are properly aligned, ensuring the model has something to learn from a contiguous sequence of events over time. Temporal models such as RNN depend on sequenced data; hence, a proper alignment is important.

Hybrid RNN-based Kalman filter

The proposed Hybrid RNN based Kalman filter model is shown in Figure 2. The preprocessed data is input to a hybrid Kalman Filter based RNN where the Kalman Filter estimates the initial state and the RNN learns temporal relationships to alleviate prediction errors. This merging expands the precision and flexibility of the system in nonlinear and dynamic settings. The RNN keeps on improving the Kalman output, which allows for stronger and real-time estimation (Fig. 2).

Recurrent Neural Networks (RNNs) are utilized to enhance the estimation accuracy of the state using the eradication of residual errors produced by the Kalman Filter (KF). The Kalman Filter is effective in handling noise and uncertainty in sequence data but relies on assumptions as well as model parameters that do not necessarily

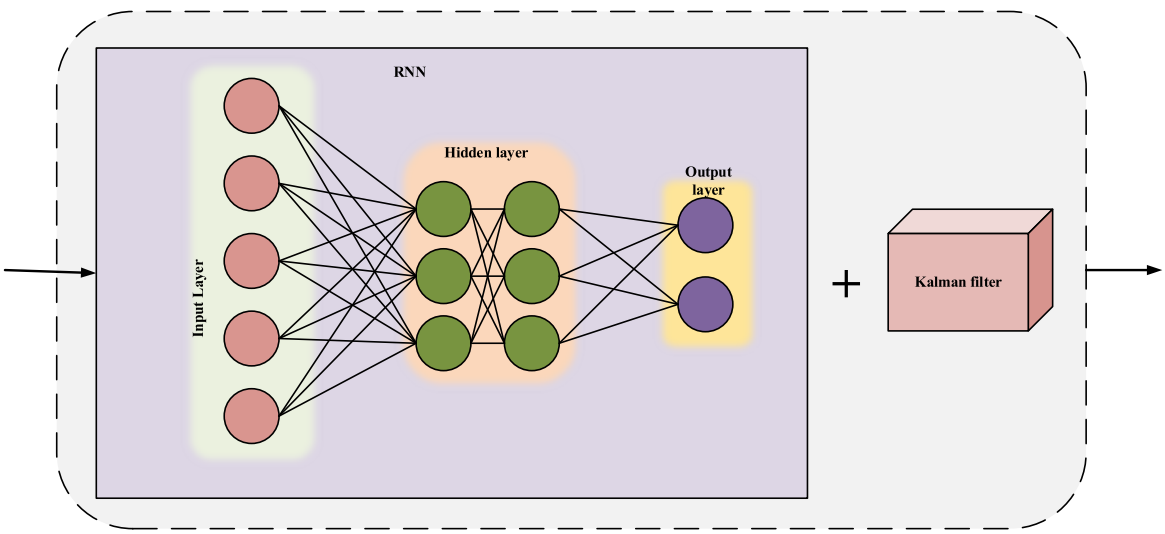


Fig. 2 Hybrid RNN based Kalman filter

Table 2 X-position prediction evaluation of 70% training

	X-position					
	MAE	RMSE	MSE	MARE	MAPE	Correlation
Proposed	0.031224	0.037021	0.001371	0.214809	21.4809	0.999319
CNN	0.071547	0.082562	0.006816	0.6901	69.01	0.996631
LSTM	0.099643	0.115536	0.013349	0.507449	50.74488	0.993549
MLP	0.149202	0.171825	0.029524	1.090819	109.0819	0.985835

Table 3 Y-position prediction evaluation of 70% training

	Y-position					
	MAE	RMSE	MSE	MARE	MAPE	Correlation
Proposed	0.03135	0.037317	0.001393	0.077372	7.737195	0.999312
CNN	0.071874	0.083039	0.006895	0.178105	17.81054	0.996618
LSTM	0.109063	0.126783	0.016074	0.301536	30.15361	0.992526
MLP	0.151323	0.173897	0.03024	0.401692	40.16917	0.985697

Table 4 Z-position prediction evaluation of 70% training

	Z-position					
	MAE	RMSE	MSE	MARE	MAPE	Correlation
Proposed	0.031643	0.037742	0.001424	0.085749	8.574911	0.999291
CNN	0.072512	0.083699	0.007006	0.195807	19.58069	0.996558
LSTM	0.107492	0.125658	0.01579	0.295224	29.52235	0.992485
MLP	0.151955	0.174624	0.030493	0.380627	38.06272	0.985198

capture the nonlinear and dynamic nature of real systems. Therefore, the estimated outputs—such as the Jacobian matrix or system state—could be different from the real values. To overcome this shortcoming, a time-conscious model developed using RNN is incorporated into the system. The RNN is trained to acquire the temporal relationships and error patterns of the KF, thus giving a correction mechanism in real time $x = \left(x^1, x^2, \ldots, x^n\right)$, an output vector $y = \left(y^1, y^2, \ldots, y^n\right)$, and an output vector in the hidden layer a = $(a^1, a^2, \ldots, a^T)$:

$$a^t = \tanh\left(W_{ax}x^t + W_{aa}a^{t-1} + b_a\right) \quad (1)$$

$$\hat{y}^t = softmax\left(W_{ya}a^t + b_y\right) \quad (2)$$

where W_{ax} stands for the present input weight matrix; b_a stands for the buried layer bias in the present RNN unit; W_{aa} stands for the input weight matrix from the prior hidden layer state. Currently, the output weight matrix is called W_{ya}; in the present RNN unit, b_y is the output bias.

In this hybrid approach, the input to the RNN is a temporally ordered feature vector derived from the input signal or image data:

$$S'(k) = [s_1(k), s_2(k), ..., s_m(k)]_{8\times m} \quad (3)$$

The real-time RNN output provides an estimated corrected state or Jacobian-like matrix:

$$J'(k) = [x_1(k), x_2(k), ..., x_m(k)]_{48\times m} \quad (4)$$

This improved output may serve as the optimized mapping of image features and control signals in a visual servo system, denoised physiological signals in biomedical systems, or the latent dynamic state in time-series models. Through cooperative training of the RNN-Kalman Filter (RKF) model, the system is able to learn and approximate the generalized inverse of the transformation matrix and then apply it to the characteristic error matrix. This hybrid modeling method greatly enhances the accuracy, convergence rate, and robustness of the system under dynamic and uncertain environments. Chaotic-Levy-based PSO is then used at the end to fine-tune the weights of the RNN and the Kalman filter model, resulting in global optimization and improved generalization performance.

Chaotic-Levy-based particle swarm optimization (CL-PSO)

For further enhancing the performance and the ability to generalize of the RNN-Kalman Filter module, Chaotic-Levy-based Particle Swarm Optimization (CL-PSO) is used. A Chaotic-Levy-based Particle Swarm Optimization (CL-PSO) algorithm is employed to optimize the

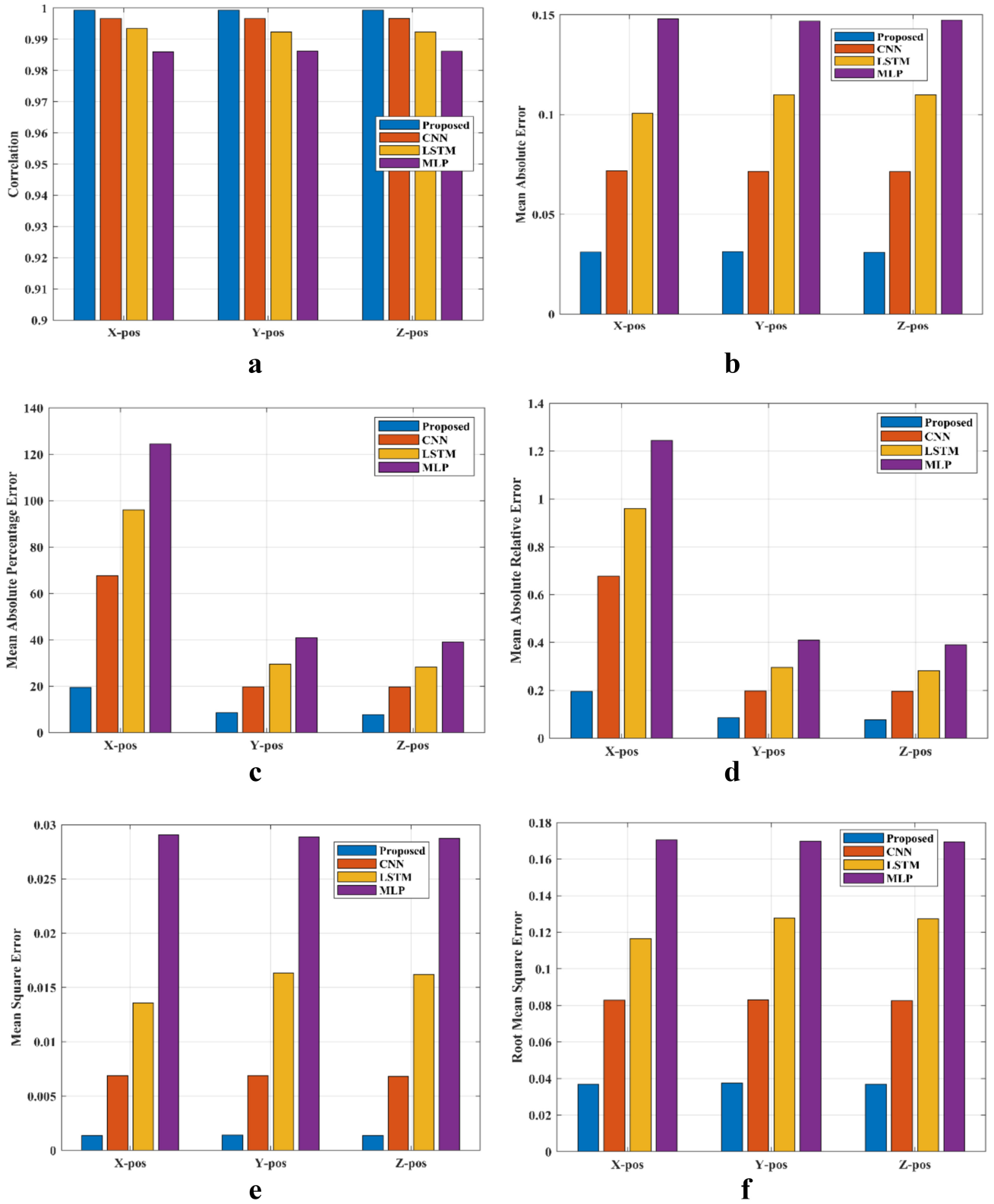


Fig. 3 **a** Correlation **b** MAE **c** MAPE **d** MARE **e** MSE **f** RMSE

performance of the model in training. The PSO method was created by J. Kennedy and R. C. Eberhard to solve nonlinear problems. It is a population-based, stochastic, self-adaptive optimization technique inspired by nature.

Table 5 X-position prediction evaluation of 80% training

	X-position					
	MAE	RMSE	MSE	MARE	MAPE	Correlation
Proposed	0.031727	0.037635	0.001416	0.256256	25.62563	0.999295
CNN	0.07138	0.08258	0.006819	0.717825	71.7825	0.996621
LSTM	0.101372	0.117246	0.013747	0.931887	93.18872	0.993343
MLP	0.149465	0.17196	0.02957	0.82649	82.64904	0.98573

Table 6 Y-position prediction evaluation of 80% training

	Y-position					
	MAE	RMSE	MSE	MARE	MAPE	Correlation
Proposed	0.031221	0.03689	0.001361	0.086152	8.615244	0.999325
CNN	0.073411	0.084503	0.007141	0.194983	19.49833	0.996523
LSTM	0.10837	0.126459	0.015992	0.284618	28.4618	0.992568
MLP	0.147601	0.170963	0.029228	0.400549	40.05489	0.986114

Table 7 Z-position prediction evaluation of 80% training

	Z-position					
	MAE	RMSE	MSE	MARE	MAPE	Correlation
Proposed	0.031196	0.037267	0.001389	0.081713	8.171324	0.999309
CNN	0.07085	0.082496	0.006806	0.184394	18.43945	0.996634
LSTM	0.109495	0.127431	0.016239	0.285357	28.53574	0.99245
MLP	0.149516	0.171548	0.029429	0.387789	38.77895	0.98566

This is how the PSO algorithm operates. First, initial velocities are assigned to the produced beginning particles. At each particle's location, the goal function is assessed. The optimal position and function value are identified. The following stage involves choosing new velocities based on recent velocity, individual particle positions, and the locations of their neighboring particles. Iterative updates are then made to the particle locations, velocities, and neighbors. By adding the velocity to the old location, the new location is evaluated. To keep particles within boundaries, locations are changed. Until the algorithm reaches a stopping threshold, the process is repeated multiple times.

The PSO algorithm explores the space of an optimization problem by changing the trajectories of individual members, called particles because they create partwise paths in a quasistochastic fashion. The random and deterministic components make up the two main components of the swarming particle's motion. With this approach, each potential response is handled as a particle with a random position and velocity in the search space. The search space is the collection of all likely solutions to the problem that requires optimization. The best position and velocity of each particle are determined by the optimal solution (fitness) in the solution space. In the PSO algorithm, the i-th particle uses the following equations to update its position and velocity at each t-th step:

$$V_i^{i+1} = wV_i^t + r_1C_1\left(P_{best} - X_i^t\right) + r_2C_2\left(G_{best} - X_i^t\right) \quad (5)$$

Prior velocity, cognitive learning, and social learning are the first, second, and third terms in the statement above, respectively.

$$X_i^{t+1} = X_i^t + V_i^{t+1} \quad (6)$$

where C_1 and C_2 indicate the user-predefined acceleration constant, X_i^t and V_i^t indicate the position and velocity vector of the i-th particle in the swarm, and w is the inertial weight to preserve the balance between local and global search capability. The random numbers r_1 and r_2 are produced between 0 and 1. The i-th particle's global best position inside the swarm is denoted by G_{best}, while P_{best} is its personal best position at time t.

P_{best} denotes the personal best position of the particle, G_{best} represents global best position of particular particle in a search space.

The addition of chaotic maps can enhance the exploration and exploitation capabilities of the algorithm.

$$d_{t+1} = Sin\left(\pi d_t\right) \qquad (0,1) \quad (7)$$

Levy flight (LF) is derived from the following formula:

$$LF\left(x\right) = \frac{u \times \sigma}{|v|^{\frac{1}{\beta}}} \quad (8)$$

In the Levy Flight equation, u and v are random numbers between 0 and 1, while σ is the scale parameter and β = 1.5

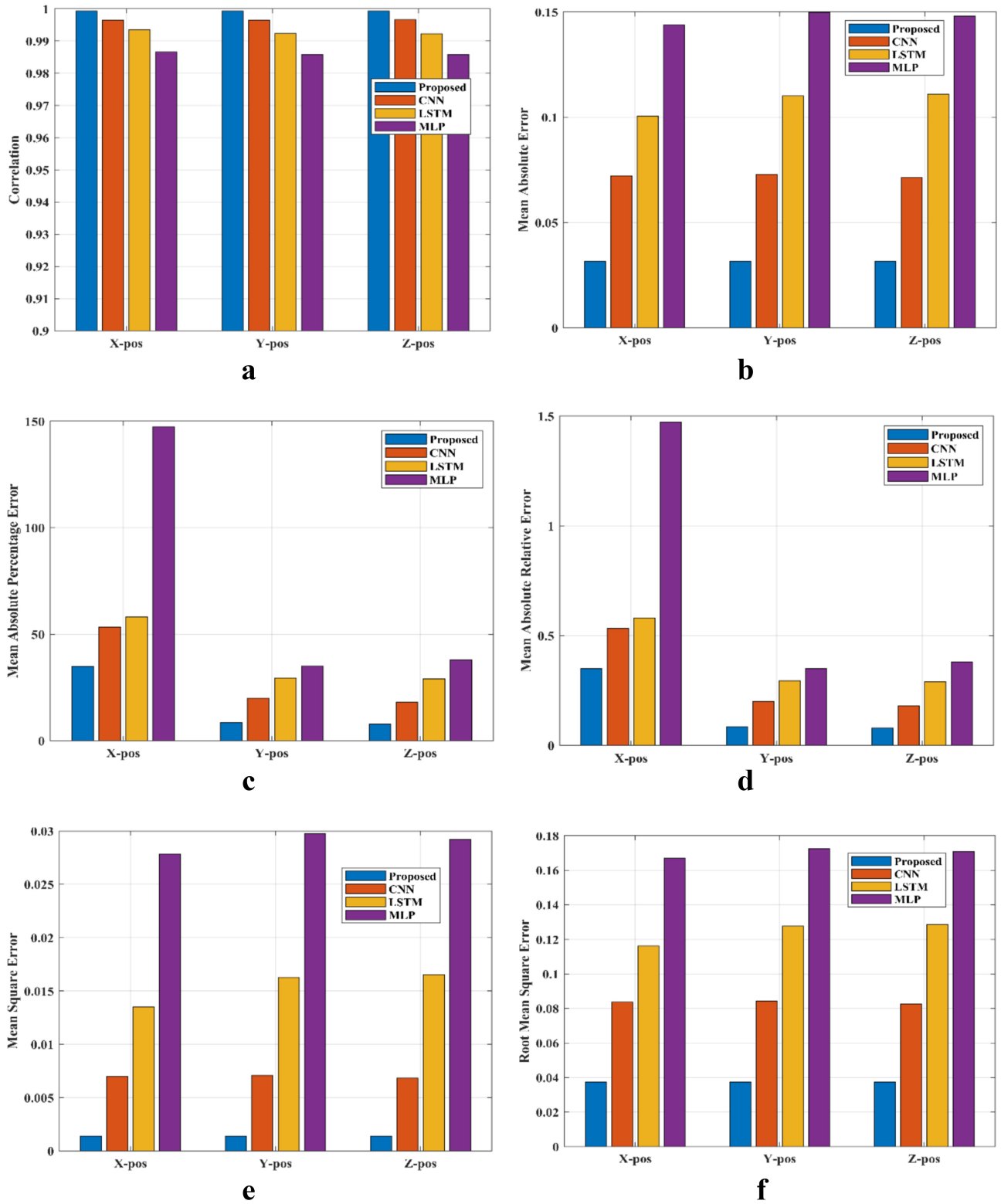


Fig. 4 **a** Correlation **b** MAE **c** MAPE **d** MARE **e** MSE **f** RMSE

defines the tail behavior of the distribution. The term $|v|^{\frac{1}{\beta}}$ represents the stochastic step size.

The chaotic map was improved by LF is given in Eq. (9).

$$d_{t+1} = Sin\left(\pi d_t\right) * LF\left(x\right) \tag{9}$$

For improving the efficiency, the Chaotic-Levy-based Particle Swarm Optimization (CL-PSO) algorithm is employed.

$$X_i^{t+1} = X_i^t + V_i^{t+1} * d_{t+1}$$

The Chaotic-Levy-based Particle Swarm Optimization (CL-PSO) enhances global search ability and avoids local minima by introducing chaotic sequences and Lévy flights. It improves convergence speed and model accuracy by efficiently exploring the solution space. This leads to better parameter tuning and robustness in noisy GPS-INS navigation data.

This optimization layer fine-tunes the parameters of the entire model, leading to more accurate and reliable position predictions, ultimately supporting efficient and intelligent navigation.The final output of the optimized Hybrid RNN based Kalman filter model is the accurate predicted position of the objects. This prediction leverages the learned spatial and temporal features from the GPS/INS data, with the attention mechanism focusing on the most relevant historical information.

Results and discussion

This section contains results and a discussion of the proposed model. This paper presents a Novel Attention based RNN based Kalman filter for Accurate Position Forecasting Framework. The performance effectiveness of the recommended technique is assessed using different metrics like Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Square Error (MSE), Mean Absolute Relative Error (MARE), Mean Absolute Percentile Error (MAPE).

Performance analysis of the models

The performance analysis consistently demonstrates the superiority of the proposed hybrid RNN based Kalman filter model, enhanced by chaotic-Levy-based PSO, for predicting X, Y, and Z positions. Across different training data percentages (70% and 80%), the proposed model exhibits significantly lower error rates (MAE, RMSE, MSE, MARE, MAPE) and near-perfect correlation compared to individual CNN, LSTM, and MLP models. This highlights the effectiveness and robustness of the proposed architecture in accurately predicting object positions, showcasing its advantage in the navigation system context.

This Table 2 is a performance comparison of various models' ability to predict X-position using 70% of the data for training. The Proposed model surpasses the others in all measures. It shows a Mean Absolute Error of 0.031224, a RMSE of 0.037021, and a MSE of 0.001371, all smaller in magnitude compared to its counterparts. Additionally, its MARE is 0.214809, and its MAPE is 21.4809, meaning proportionally much smaller errors. Most importantly, the Proposed model also has a significant correlation coefficient of 0.999319, meaning an almost perfect linear correlation between its predictions and true X-positions. Conversely, the MLP model fares the worst, with highest errors thereof, such as a MAE of 0.149202, and a correlation of 0.985835. In general, the Proposed model demonstrates better predictive performance in the X-position.

This Table 3 shows the comparison of different models to predict Y-position, trained on 70% of the dataset. Following the same pattern with X-position predictions, the Proposed model outperforms in all the metrics. It has the minimum Mean Absolute Error at 0.03135, Root Mean Square Error at 0.037317, and Mean Squared Error at 0.001393, indicating its accuracy. The Proposed model also has the lowest MARE of 0.077372 and a MAPE of 7.737195, reflecting very slight proportionate deviations from the true Y-positions. Additionally, its correlation coefficient is a remarkable 0.999312, demonstrating a very strong positive linear relationship between its predictions and the actual Y-values. On the other hand, MLP model again has the poorest performance with the greatest errors and the lowest correlation of 0.985697. The Proposed model is obviously the best for predicting Y-position.

This Table 4 shows a comparison of several models for forecasting Z-position, with 70% of data applied for training. Similar to X and Y positions, the Proposed model also performs the best on all the evaluation measures. It has the best MAE of 0.031643, RMSE of 0.037742, and MSE of 0.001424, reflecting very accurate forecasts. The Proposed model also has the lowest MARE at 0.085749 and a MAPE of 8.574911, indicating negligible relative and percentage-based prediction errors. In addition, its correlation coefficient of 0.999291 is very high, validating a very strong linear relationship with the true Z-positions. The MLP model is always the weakest, with the largest errors and the lowest correlation of 0.985198. The Proposed model is evidently the best for precise prediction of Z-position. Figure 3a–f shows graphical representation of XYZ position prediction with 70% training in terms of the measured parameters.

This Table 5 compares different models for X-position prediction, with a higher percentage of training data at 80%. The Proposed model still outperforms others, with the lowest MAE at 0.031727, RMSE at 0.037635, and MSE at 0.001416. These values confirm very accurate

Table 8 Combined XYZ prediction evaluation (70% Training)

	MAE	RMSE	MSE	MARE	MAPE	Correlation
Proposed	0.031406	0.03736	0.001396	0.125977	12.59767	0.999307
CNN	0.071978	0.0831	0.006906	0.354671	35.46708	0.996602
LSTM	0.105399	0.122659	0.015071	0.368069	36.80695	0.992853
MLP	0.150827	0.173449	0.030086	0.624379	62.43793	0.985577

Table 9 Combined XYZ prediction evaluation (80% Training)

80–20	MAE	RMSE	MSE	MARE	MAPE	Correlation
Proposed	0.031381	0.037264	0.001389	0.141374	14.1374	0.99931
CNN	0.071881	0.083193	0.006922	0.365734	36.57342	0.996593
LSTM	0.106412	0.123712	0.015326	0.500621	50.06209	0.992787
MLP	0.148861	0.17149	0.029409	0.538276	53.82763	0.985835

predictions. Though its MAPE rose to 25.62563 from the 70% training condition, it is still lower than other models by far. The Proposed model also has an extremely high correlation coefficient of 0.999295, which indicates a strong linear relationship to actual X-positions. The MLP model invariably has the poorest performance with larger error measures, e.g., a Mean Absolute Error of 0.149465, and a lower correlation of 0.98573. This again verifies the Proposed model's performance in X-position prediction with increased training data.

This Table 6 compares the performance of various models in Y-position prediction, with an increased training data allocation of 80%. The Proposed model uniformly registers better performance on all parameters. It obtains the best MAE of 0.031221, Root Mean Square Error of 0.03689, and MSE of 0.001361, reflecting very precise and accurate predictions. The Proposed model also has the lowest MARE of 0.086152 and a MAPE of 8.615244, which indicates very small proportional errors. Its correlation coefficient is still very high at 0.999325, which indicates a very strong linear positive correlation with the actual Y-positions. In contrast, the MLP model still performs the worst, with the highest error values and the lowest correlation of 0.986114. This table again consolidates the Proposed model's efficacy and resilience in Y-position prediction even with increased training data.

This Table 7 provides a comparison of different Z-position predictive models using an 80% training data ratio. The Proposed model is shown to have outstanding performance at all times, and it maintains the lead in all measurements. It has the lowest MAE at 0.031196, RMSE at 0.037267, and MSE at 0.001389. These are indicators of high prediction precision and low dispersion of errors. In addition, the Proposed model has the smallest MARE of 0.081713 and a MAPE Error of 8.171324, suggesting very tiny proportional errors in its Z-position predictions. Its correlation coefficient is still extremely high at 0.999309, affirming an extremely strong positive linear relationship between its predictions and the true Z-positions. The MLP model, on the other hand, always provides the poorest performance, indicating the largest error values and the least correlation of 0.98566. This strengthens the Proposed model's better and stable ability for precise Z-position prediction even with a greater training dataset. Figure 4a–f shows graphical representation of XYZ position prediction with 80% training in terms of the measured parameters.

Overall prediction comparison

This Table 8 is a total estimate of XYZ position forecasts, 70% having been utilized for training. The Proposed model always demonstrates superior performance when all three axes are treated as an entity. It possesses the lowest MAE of 0.031406, RMSE of 0.03736, and MSE of 0.001396, which signifies highly precise and accurate overall position forecast. In addition, the Proposed model posts the lowest MARE of 0.125977 and a Mean Absolute Percentage Error of 12.59767, indicating minimal proportional errors on all dimensions. Its correlation coefficient is still unusually high at 0.999307, affirming an extremely strong linear association between its combined predictions and true XYZ positions. Conversely, the MLP model always has the poorest performance with the largest values of error and the lowest correlation overall of 0.985577. This table effectively indicates the Proposed model's thorough effectiveness for predicting combined XYZ position.

This Table 9 shows the overall XYZ position prediction performance evaluation using a higher training data split of 80%. The Proposed model still outperforms in all the combined metrics. It has the lowest MAE of 0.031381, RMSE of 0.037264, and MSE of 0.001389. These consistently low error rates reflect its high overall precision and accuracy of predicting 3D positions. Even with its Mean Absolute Percentage Error of 14.1374, it is still lower compared to the remaining models. The Proposed model also possesses a very high correlation

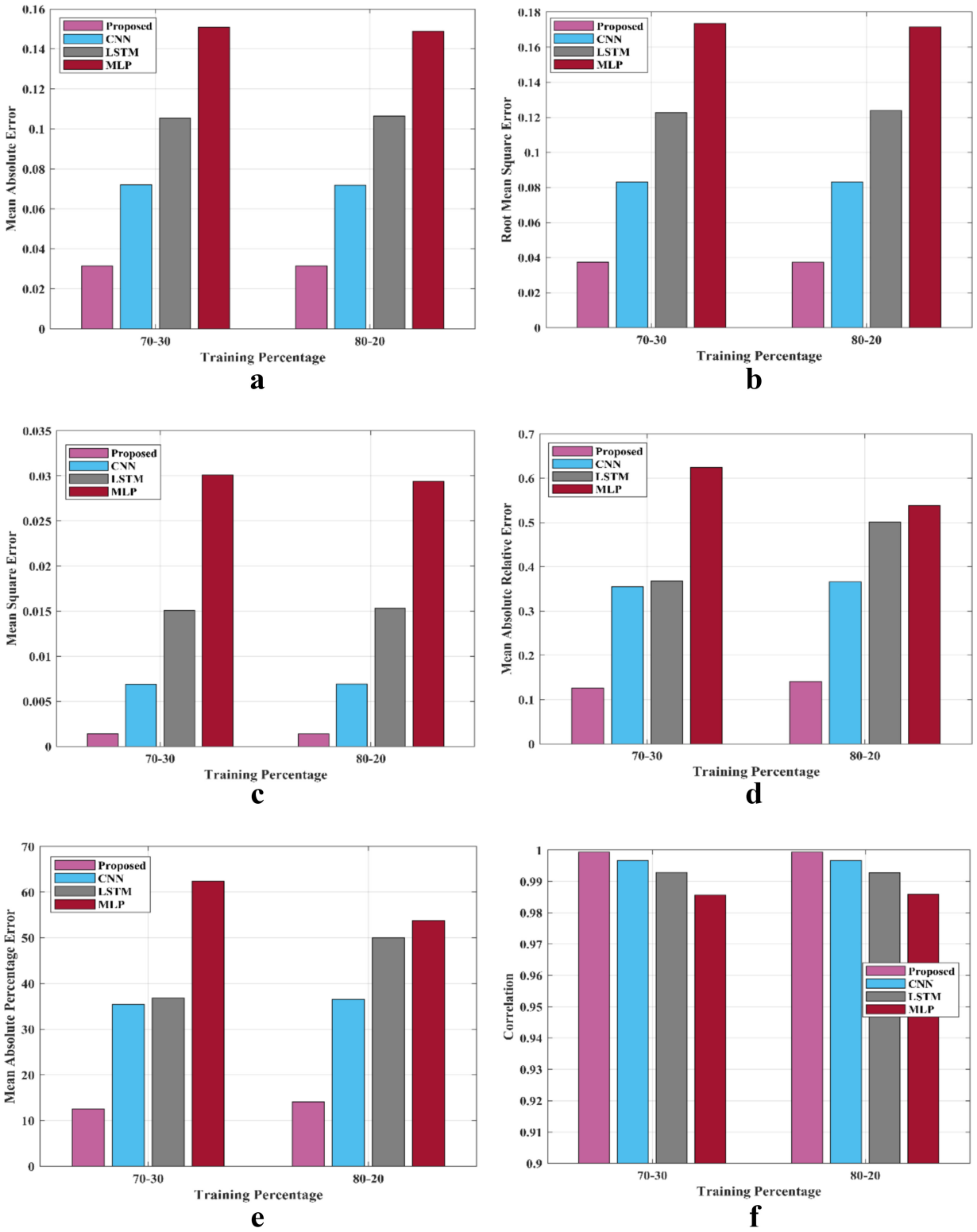


Fig. 5 **a** MAE **b** RMSE **c** MSE **d** MARE **e** MAPE **f** Correlation

coefficient of 0.99931, which confirms there is a very strong linear relationship between its overall XYZ predictions and actual values. MLP model, however, still shows the worst performance with the highest error measurements and lowest of 0.985835 correlation. This table further supports the Proposed model's effectiveness

and strength for generalized XYZ position prediction, even under the presence of a larger training dataset. Figure 5a–f shows graphical representation of XYZ position prediction with 70% and 80% training in terms of the measured parameters.

Figure 6a–d represents XYZ position prediction over various time instants for different models with 70% training.

Figure 7a–d represents XYZ position prediction over various time instants for different models with 80% training.

Conclusion

This paper introduces a new hybrid Kalman filter model constructed on RNN to improve the object position prediction accuracy in GPS/INS navigation. The proposed model is capable of capturing temporal relationships using recurrent neural networks and leverages the strength of the Kalman filter to handle the non-linear and dynamic characteristics of the real world. A pre-processing pipeline that is systematic guarantees that input data are well-conditioned, time-aligned, and prepared for the analysis of sound. The platform was developed with MATLAB and tested with conventional metrics like MAE, RMSE, MSE, MARE, and MAPE. The outcome

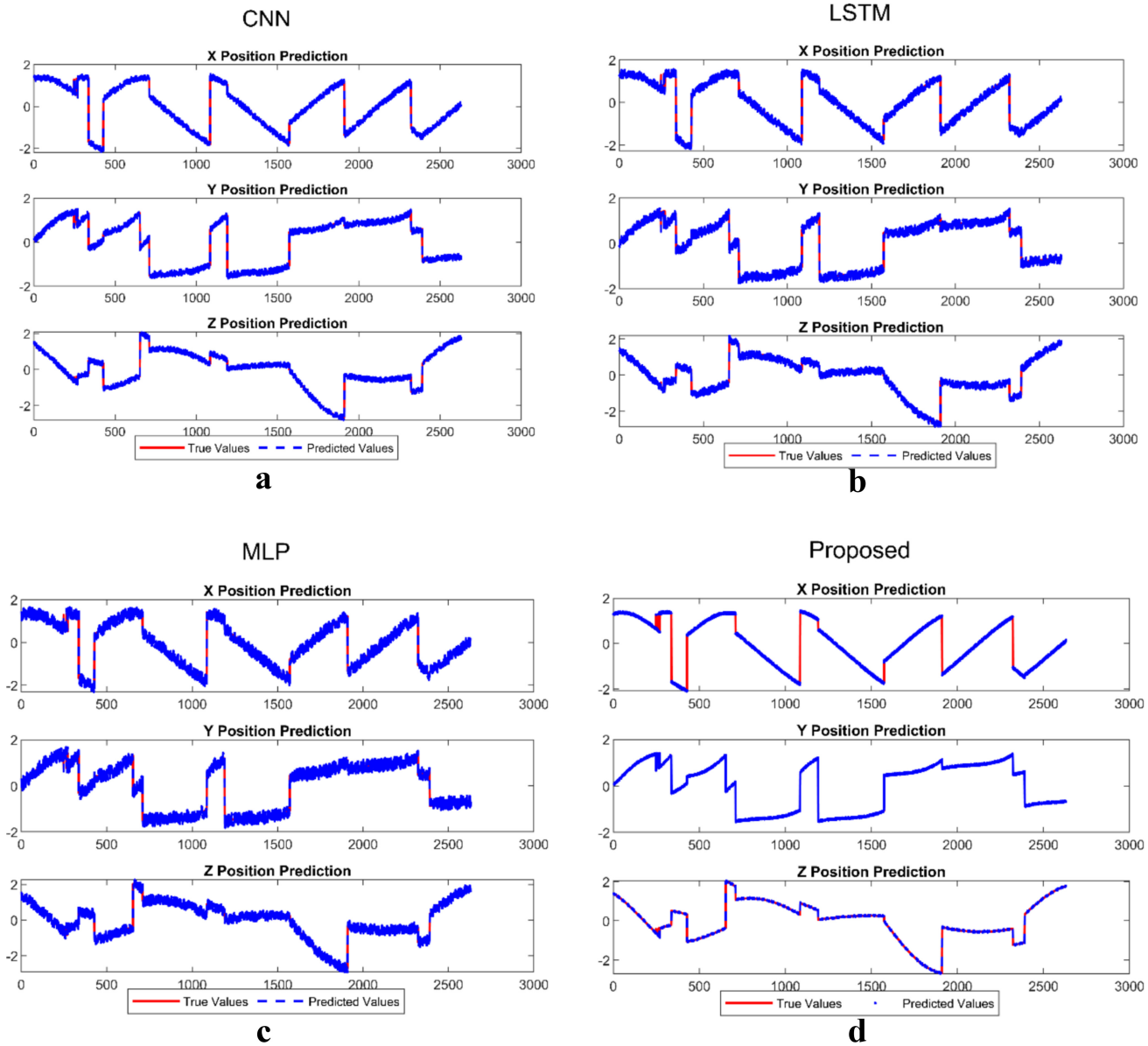


Fig. 6 XYZ position prediction over various time instants for different models with 70% training

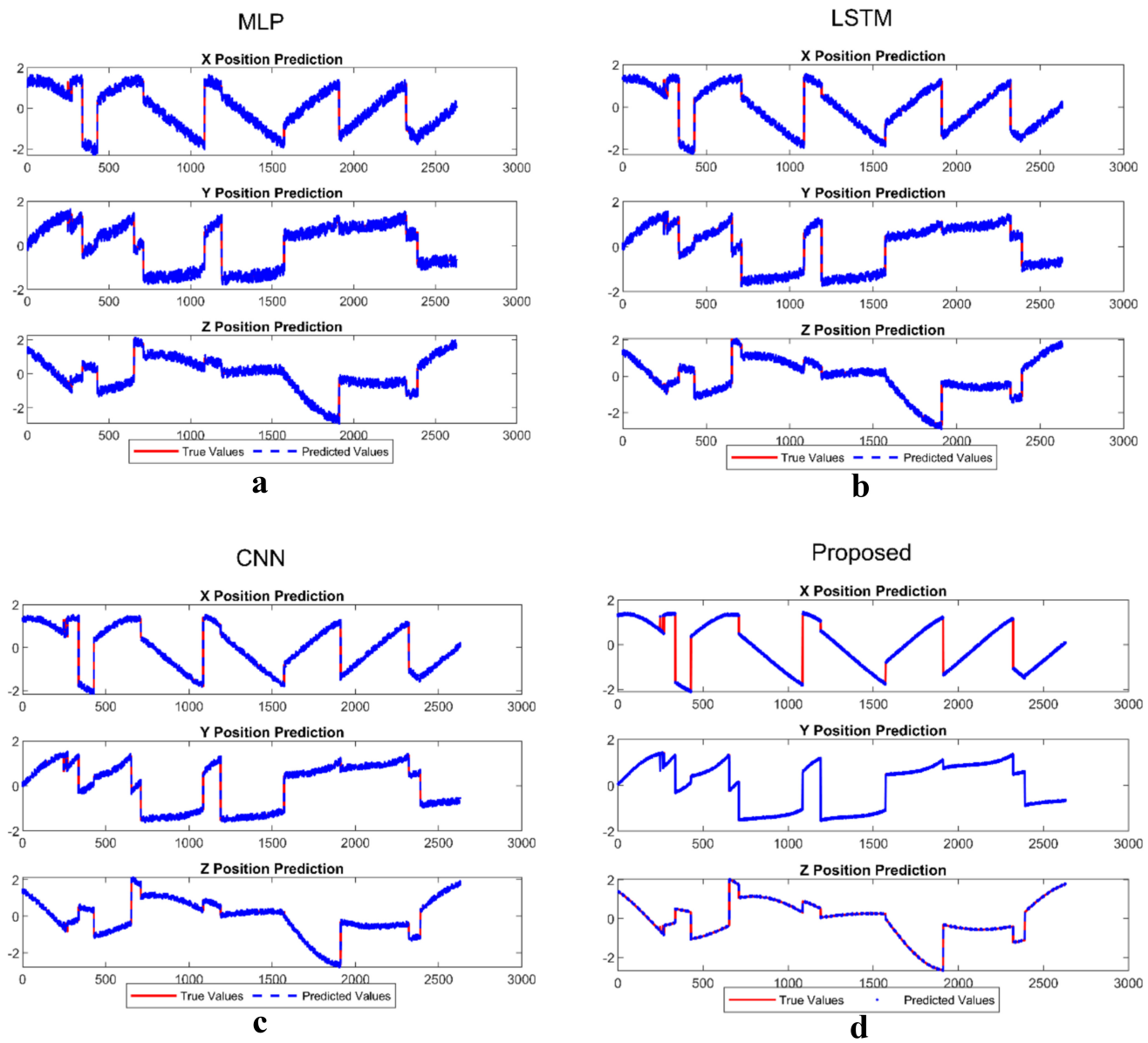


Fig. 7 XYZ position prediction over various time instants for different models with 80% training

validates that hybrid model performs better than conventional CNN, LSTM, and MLP models across all three axes (X, Y, and Z) of estimation of position. The suggested method continued to yield nearly ideal correlation values (0.999 +) and much lower levels of errors, verifying its stability and reliability. RNN with Kalman filtering provides adaptive correction of errors and learning in sequential data and is thus well suited for real-time navigation. Overall, the model's excellent prediction accuracy and low computational cost make it a good candidate for complete deployment in smart positioning systems, especially for GPS-degraded or denied environments like urban canyons, tunnels, or indoor buildings. Future work will involve the addition of other modalities of sensors like LiDAR and UWB for increased robustness. Real-time deployment on embedded devices will be explored for autonomous mobility. The model is also generalizable to multi-agent tracking in dynamic environments.

Funding The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Data availability The referred papers will be available on request.

Declarations

Conflict of interest The authors declare no conflict of interest.

References

Behivoke, F., Etienne, M.P., Guitton, J., Randriatsara, R.M., Ranaivoson, E., Léopold, M.: Estimating fishing effort in small-scale fisheries using GPS tracking data and random forests. Ecol. Ind. **123**, 107321 (2021)

Cao, Y., Bai, H., Jin, K., Zou, G.: An GNSS/INS integrated navigation algorithm based on PSO-LSTM in satellite rejection. Electronics **12**(13), 2905 (2023)

Cheng, L., Chen, J., Liu, Y., Zhu, Q.: Variable projection algorithm for GPS positioning in multipath environments based on Aitken acceleration method. IEEE Trans. Ind. Inf. **20**(4), 6404–6412 (2024)

Gao, Y., Jing, H., Dianati, M., Hancock, C.M., Meng, X.: Performance analysis of robust cooperative positioning based on GPS/UWB integration for connected autonomous vehicles. IEEE Trans. Intell. Veh. **8**(1), 790–802 (2022)

Guang, X., Gao, Y., Liu, P., Li, G.: IMU data and GPS position information direct fusion based on LSTM. Sensors **21**(7), 2500 (2021)

Kabir, M.H., Hasan, M.A., Islam, M.S., Ko, K., Shin, W.: PC-DeepNet: A GNSS positioning error minimization framework using permutation-invariant deep neural network. IEEE Sens. J. (2025)

Kong, T., Ma, L., Ai, G.: Research on improving satellite positioning precision based on multi-frequency navigation signals. Sensors **22**(11), 4210 (2022)

Kovář, P., Puričer, P., Kovářová, K.: Study of the two-line element accuracy by 1U CubeSat with a GPS receiver. Sensors **22**(8), 2902 (2022)

Li, C., Fu, Y., Yu, F.R., Luan, T.H., Zhang, Y.: Vehicle position correction: a vehicular blockchain networks-based GPS error sharing framework. IEEE Trans. Intell. Transp. Syst. **22**(2), 898–912 (2020a)

Li, D., Jia, X., Zhao, J.: A novel hybrid fusion algorithm for low-cost GPS/INS integrated navigation system during GPS outages. IEEE Access **8**, 53984–53996 (2020b)

Li, H., Borhani-Darian, P., Wu, P., Closas, P.: Deep neural network correlators for GNSS multipath mitigation. IEEE Trans. Aerosp. Electron. Syst. **59**(2), 1249–1259 (2022)

Liao, C., Chen, C., Xiang, C., Huang, H., Xie, H., Guo, S.: Taxi-passenger's destination prediction via GPS embedding and attention-based BiLSTM model. IEEE Trans. Intell. Transp. Syst. **23**(5), 4460–4473 (2021)

Liu, W., Li, Z., Sun, S., Gupta, M.K., Du, H., Malekian, R., Li, W.: Design a novel target to improve positioning accuracy of autonomous vehicular navigation system in GPS denied environments. IEEE Trans. Ind. Inform. **17**(11), 7575–7588 (2021)

Lv, D., Liu, G., Ou, J., Wang, S., Gao, M.: Prediction of GPS satellite clock offset based on an improved particle swarm algorithm optimized BP neural network. Remote Sens. **14**(10), 2407 (2022)

Najder, J., Sośnica, K.: Quality of orbit predictions for satellites tracked by SLR stations. Remote Sens. **13**(7), 1377 (2021)

Pagoti, S.K., Vemuri, S.I.D.: Development and performance evaluation of CorrentropyKalman Filter for improved accuracy of GPS position estimation. Int. J. Intell. Netw. **3**, 1–8 (2022)

Paszek, K., Grzechca, D.: Using the LSTM neural network and the UWB positioning system to predict the position of low and high speed moving objects. Sensors **23**(19), 8270 (2023)

Ruwali, A., Kumar, A.S., Prakash, K.B., Sivavaraprasad, G., Ratnam, D.V.: Implementation of hybrid deep learning model (LSTM-CNN) for ionospheric TEC forecasting using GPS data. IEEE Geosci. Remote Sens. Lett. **18**(6), 1004–1008 (2020)

Stojković, P., Tadić, P.: Object location prediction in real-time using LSTM neural network and polynomial regression (2023). arXiv preprint arXiv:2311.13950

Sun, R., Wang, G., Cheng, Q., Fu, L., Chiang, K.W., Hsu, L.T., Ochieng, W.Y.: Improving GPS code phase positioning accuracy in urban environments using machine learning. IEEE Internet Things J. **8**(8), 7065–7078 (2020)

Tao, Y., Liu, C., Chen, T., Zhao, X., Liu, C., Hu, H., Xin, H.: Real-time multipath mitigation in multi-GNSS short baseline positioning via CNN-LSTM method. Math. Probl. Eng. **2021**(1), 6573230 (2021)

Uradziński, M., Bakuła, M.: Assessment of static positioning accuracy using low-cost smartphone GPS devices for geodetic survey points' determination and monitoring. Appl. Sci. **10**(15), 5308 (2020)

Wang, K., El-Mowafy, A.: LEO satellite clock analysis and prediction for positioning applications. Geo-Spat. Inf. Sci. **25**(1), 14–33 (2022)

Yang, B., Guo, Z., Wang, L., He, J., Xia, B., Vakily, S.: Updated global navigation satellite system observations and attention-based convolutional neural network–long short-term memory network deep learning algorithms to predict landslide spatiotemporal displacement. Remote Sens. **15**(20), 4971 (2023)

Yoon, J.H., Kim, H.J., Lee, D.S., Kwon, S.K.: Indoor positioning method by CNN-LSTM of continuous received signal strength indicator. Electronics **13**(22), 4518 (2024)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.