Journal of Robotics and Control (JRC)
Volume 6, Issue 2, 2025
ISSN: 2715-5072 DOI: 10.18196/jrc.v6i2.26345

Enhanced Angle Estimation Using Optimized Artificial Neural Networks with Temporal Averaging in IMU-Based Motion Tracking

Phichitphon Chotikunnan [1], Wanida Khotakham [2*], Rawiphon Chotikunnan [3], Kittipan Roongprasert [4], Yutthana Pititheeraphab [5], Tasawan Puttasakul [6], Anantasak Wongkamhang [7], Nuntachai Thongpance [8]
[1, 2, 3, 4, 5, 6, 7, 8] College of Biomedical Engineering, Rangsit University, Pathum Thani, Thailand
Email: [1] phichitphon.c@rsu.ac.th, [2] wanida.k@rsu.ac.th, [3] rawiphon.c@rsu.ac.th, [4] kittipan.r@rsu.ac.th, [5] yutthana.p@rsu.ac.th, [6] tasawan.p@rsu.ac.th, [7] anantasak.w@rsu.ac.th, [8] nuntachai.t@rsu.ac.th
*Corresponding Author

***Abstract*—Accurate angle estimate is crucial for motion tracking systems, especially in biomedical applications like rehabilitation, prosthesis control, and wearable health monitoring. Traditional filters, such as the Kalman filter, frequently encounter difficulties with nonlinear noise and dynamic variations, hence constraining their resilience. This study presents feedforward artificial neural network (ANN) models as a highly accurate alternative by utilizing IMU sensor data from a gyroscope and accelerometer. The research contribution encompasses: (1) the creation of ANN architectures of diverse complexity, featuring an innovative (4×8) structure with temporal averaging for enhanced noise resilience; (2) a simulation-based assessment in comparison to the Kalman filter utilizing consistent performance metrics; and (3) an evaluation of execution-time viability for embedded applications. A dataset including 3,599 samples was acquired from an MPU6050 IMU and partitioned into 70% for training, 15% for validation, and 15% for testing. Model assessment was conducted utilizing mean absolute error (MAE) and root mean squared error (RMSE). The NN (4×8 + Averaging) model produced a minimum MAE of 0.2657 and an RMSE of 0.3691, indicating a 68% enhancement compared to the Kalman filter. Although compact models (2×4, 2×8) exhibited marginal improvements, deeper architectures demonstrated superior generalization and resilience, especially during dynamic motion phases. These results show that ANN-based estimators offer better accuracy and adaptability, making them a good choice for real-time biomedical uses. Future research will investigate hybrid ANN-Kalman designs and assess their performance across diverse motion types, including gait cycles and robotics.**

Keywords—Angle Estimation; Artificial Neural Network (ANN); Kalman Filter; Inertial Measurement Unit (IMU); Noise Robustness; Real-Time Estimation.

I. INTRODUCTION

Precise angle estimate is essential in several engineering applications, such as robots, aircraft systems, navigation platforms, and biomedical engineering, where accurate orientation tracking greatly impacts system performance and safety [1]–[3]. Accurate and robust orientation estimation techniques, including extended Kalman filters, sensor fusion methods, and adaptive controllers, are essential in applications such as autonomous drone navigation, indoor localization, and aerial robotics, where imprecise estimations may result in system instability, navigation errors, or operational failures [4]–[9]. These estimation methods were first created for robotics and aerospace, but they are now widely used in biomedical engineering, especially in robotic-assisted surgeries, rehabilitation devices, prosthetic limb orientation, and wearable health monitoring systems. This shows how useful they are in many engineering fields.

Traditional techniques for angle measurement frequently include inertial sensors, such as gyroscopes and accelerometers. These sensors have inherent limitations, including drift, susceptibility to high-frequency noise, and sensitivity to external disturbances [10]–[12]. A multitude of persons have employed sensor fusion systems that integrate inertial measuring units (IMUs), vision sensors, and several other sensor types to address these challenges. These techniques are particularly effective in dynamic environments, such as aerial robotics, drone localization, and Internet of Things (IoT) applications [13]–[17]. Nevertheless, these sensor fusion approaches continue to face challenges, particularly with unpredictable dynamics, sensor errors, and environmental variations. Several studies [18]–[25] have shown that advanced filtering methods, such as Gaussian processes, Kalman filters, and complementing filters, make signals much more reliable and accurate in these situations. These approaches are particularly effective for biomedical engineering applications, such as wearable health monitoring, gait analysis, and rehabilitation systems that need precise angle measurement during human movement and interaction with the external environment.

Sensor fusion methods, like the Kalman Filter (KF) and the Complementary Filter (CF), are used in many situations to reduce errors caused by sensor limitations. Recent improvements highlight its significant relevance, particularly in wearable devices for thorough health monitoring, where accurate sensor data gathering is crucial for reliable health assessments [24]. Research indicates that complementary filters are effective in orientation tracking using inertial sensors, including full-body motion tracking systems that incorporate magnetometer, accelerometer, and gyroscope (MARG) detectors [25]. Hybrid filtering methods, including bilateral filters, have been employed in several domains, such as image processing and recognition, due to their efficacy in reducing visual distortions [26].

The Kalman filter employs a probabilistic recursive approach and has proven effective in several robotics and sensor fusion applications [27]. Innovative methodologies for using the filter demonstrate its efficacy in mitigating intricate sensor noise and bias [28]. An illustration is the resilient error-state Kalman filtering utilized for determining the orientation of an inertial measurement unit (IMU). A lot of different Kalman filter models, dynamic phasor estimates, and improvements to Gaussian filtering have been used in thorough evaluations, showing that they are useful and flexible in nonlinear situations [29]–[31]. There are many studies that show the Kalman filter is effective at predicting how power systems, robotics, and neuron-based autoregressive models will change over time [32]–[34]. While SLAM (simultaneous localization and mapping) is important, so are factor extraction, robust statistical filtering, and figuring out what something is doing in real time when you don't know for sure [35]–[38]. Suggested improvements include Extended Kalman Filters (EKF), Unscented Kalman Filters (UKF), Dual Kalman Filters, and other adaptive Kalman variants [39]–[45]. Their strengths and weaknesses in handling nonlinear dynamics and non-Gaussian noise have led to the development of these methods. These complex methods can be used in many areas, such as short-term load forecasting in power systems [39], adaptive sliding window methods [40], invariant filtering for underwater navigation [41], and combining GNSS and INS to improve navigational accuracy [42]. To make them work better in dynamic settings, a lot of research has been done on advanced Kalman filters that are made to deal with parametric uncertainties, nonlinear model changes, and measurement errors [43]–[45]. Even though these improvements have been made, advanced Kalman-based filters are still not able to handle sudden changes in dynamics and noise fluctuations that were not expected because they depend on accurate system modeling. These constraints constrain their efficacy in medical applications, necessitating real-time and adaptive motion estimation. This kind of limitation can be seen in complex biomedical settings, as well as in vehicle navigation [46], initial alignment problems [47], real-time adaptive learning environments [48], environmental hydrological forecasting [49], and strong statistical estimation when there are model uncertainties [50]–[52]. This highlights the necessity for continuous advancement in adaptive and model-agnostic estimating techniques.

Recent improvements in estimation and control show that we can mix traditional filtering methods with artificial intelligence (AI) and machine learning (ML) to manage complicated and changing system behaviors. Methods like long short-term memory (LSTM) networks, Q-learning, fuzzy logic controllers, and Kalman filters have made systems more adaptable, leading to better performance in tasks like keeping permanent magnet synchronous motor (PMSM), autonomous systems, and monitoring the environment in real-time [53]–[58]. Hybrid methods that combine momentum-based filtering with fuzzy control have shown strong performance in tough situations, like in aerial robots and assistive technologies such as computer-controlled wheelchairs [59], [60]. These findings underscore the persistent necessity for enhancements in sensor fusion, particularly in fields such as biomedical engineering and robotics, where dependability and precision are paramount [61].

Notwithstanding the progress in filtering and fusion techniques, constraints persist. Sudden changes in sensor noise, unstable environments, and heavy reliance on models continue to weaken the reliability of traditional estimators. Additionally, many traditional filters require many changes to their settings and can be very demanding on computing resources, making them hard to use in situations with limited resources. These issues have underscored the increasing significance of adaptive, model-agnostic estimating frameworks like artificial neural networks (ANNs), which provide a robust alternative by facilitating real-time adaptation to sensor variability and noise.

In biomedical applications, inertial measurement unit (IMU) sensors have become promising instruments, especially for clinical gait analysis. Their mobility, user-friendliness, and capacity to operate outside laboratory settings render them optimal for continuous monitoring [62], [63]. At first, research using IMUs showed that they weren't excellent at measuring joint angles, especially in movements that aren't straightforward, but later studies have indicated that they are now more reliable and accurate [64]. A main issue, though, is getting the sensors perfectly lined up with the body parts, which requires careful setup and mapping of movements to a standard reference point [65], [66]. The overall performance of the system, including how consistently it works for different people and the same person over time, heavily depends on excellent calibration methods and combining data from multiple sensors [67]–[69].

Along with tracking movement, recent studies have looked into how IMUs can measure forces and moments in joints, like ground reaction forces (GRFs). However, calculations based on Newton's laws have limitations during times like when both feet are on the ground while walking. However, periods such as double support in ambulation intrinsically constrain Newtonian-based calculations. Machine learning approaches have been employed to infer joint dynamics directly from IMU data to solve these difficulties [70]–[75]. Previous models concentrated on singular components; however, recent methodologies have effectively forecasted comprehensive joint angles and forces throughout all lower-limb joints [76], [77].

Artificial neural networks are especially adept at these tasks because of their capacity to represent nonlinear input-output relationships between IMU data and biomechanical parameters. They are often trained on synchronized datasets comprising both IMU and 3D motion capture data to guarantee anatomical correctness. However, ANN-based estimators can be affected by mistakes in how the sensors are lined up when making predictions, a problem that might be lessened by using different training data that includes various alignment situations. Various artificial neural network (ANN) designs have been analyzed, including multilayer perceptrons (MLPs), convolutional neural networks (CNNs), and recurrent neural networks (RNNs), each presenting a distinct equilibrium of complexity, data prerequisites, and processing demands [78].

Multilayer Perceptrons (MLPs) have been shown to be effective for basic regression tasks, offering ease of use and low processing requirements [73], [75], [79]. Convolutional Neural Networks (CNNs), which were originally made for classifying images, have been used in biomechanical time series by changing Inertial Measurement Unit (IMU) data into two-dimensional images that can work with transfer learning systems [72], [80]. Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) models have robust temporal learning abilities. However, they need large amounts of data and significant computing power, which limits their use in embedded systems [63], [79], [81].

Conversely, feedforward artificial neural network models offer a more efficient option. They are more amenable to training, use fewer resources, and are especially appropriate for real-time applications where model simplicity and inference speed are critical. Unlike deep sequential models such as LSTM or complex fuzzy systems, feedforward ANNs can give accurate angle estimates in changing situations without needing complicated setups or a lot of data preparation. As a result, they are a practical and flexible choice for analyzing motion using IMUs, particularly in wearable health devices and prosthetic limbs that operate under everyday conditions.

The research contribution consists of three components. This paper presents feedforward artificial neural network models to address the constraints of the Kalman filter in dynamic and noisy contexts. Secondly, the suggested ANN models, especially the NN (4×8) model configuration, attained a 68% decrease in MAE and a 63% reduction in RMSE relative to Kalman filters. This study introduces an adaptive angle estimation system designed for real-time applications in prosthetic limb control, wearable health monitoring, and balance-assistive robots. The research builds on the work of Chotikunnan et al. [61], who looked at Kalman and complementary filters for angle estimation but faced challenges due to sudden noise changes and nonlinear factors in medical settings. The goal is to make angle measurements more accurate in both calm and noisy situations, since these conditions often cause Kalman-based filters to perform poorly when sensor data and noise change quickly. As a result, we looked at feedforward artificial neural networks with a (4×8) structure, using mean absolute error (MAE) and root mean squared error (RMSE) to measure how well they perform. The suggested solution addresses current issues without requiring extensive datasets or complex membership-function designs, facilitating real-time medical applications. This method uses noise patterns to build a strong and adaptable way to combine sensor data, which is helpful for things like controlling prosthetic limbs, monitoring health with wearables, and keeping robots balanced. The results show that feedforward artificial neural networks work better than traditional filtering methods, meaning they are more reliable when sensor conditions change.

II. Research Method

This research analyzes the creation and assessment of Artificial Neural Network (ANN) models for precise angle estimation utilizing inertial sensor data. This study transitions from conventional techniques, including the Kalman Filter, frequently utilized in previous research (including Chotikunnan et al. [61]), to data-driven methodologies. The proposed ANN models seek to improve the accuracy of angle estimation in noisy environments while ensuring computational efficiency for embedded devices.

Fig. 1 delineates the comprehensive technique encompassing sensor design, data acquisition, preprocessing, artificial neural network training, and final assessment. The procedure starts with the initialization of the MPU6050 at a sampling period of 13 milliseconds. During the Preprocessing step, data from the gyroscope (ω_{gyro}) and accelerometer (θ_{Accele}) are utilized to calculate covariance values, perhaps employing a moving average window (+Avg). Two principal estimation methodologies are employed: a Kalman Filter as a benchmark (not illustrated in the flowchart but utilized for performance evaluation) and various configurations of artificial neural networks (ANNs). The success of the Kalman and ANN methods is evaluated by training and testing the networks, using metrics such as mean absolute error (MAE), root mean squared error (RMSE), and correlation coefficients (R-values). The optimal method is determined at the conclusion of the pipeline.

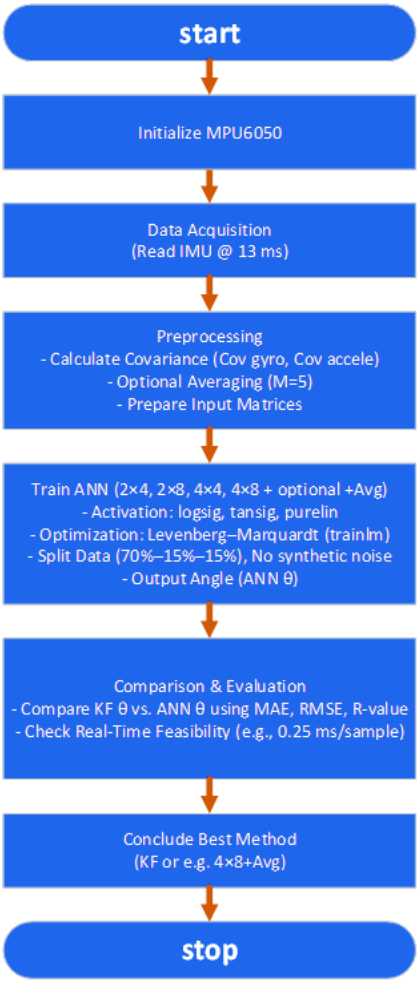

Fig. 1. Flowchart of the proposed methodology

The utilized sensor platform is the MPU6050, an Inertial Measurement Unit (IMU) including a 3-axis gyroscope and a 3-axis accelerometer. Data are recorded at 13 ms intervals, yielding unprocessed angular signals (ω_{gyro}, θ_{Accele}). A preprocessing step finds noise patterns by looking at changes over time and might use a moving average filter (M=5) to reduce sudden noise. The chosen filter size aimed to strike a balance between smoothing (about 65 ms delay) and real-time responsiveness. The implementation of a 5-point moving average (with a 13 ms sampling interval) results in an effective delay of around 65 milliseconds, signifying the duration of the smoothing window rather than the processing time. Synthetic noise injection was not utilized, as the actual sensor data exhibited adequate disruption. A total of 3,599 samples were collected and allocated in a 70%–15%–15% ratio for training, validation, and testing datasets. These attributes are subsequently organized as input vectors for the models. This phase is essential for capturing real-time sensor variability while ensuring minimal latency.

The basic model uses a feedforward neural network structure, improved with the Levenberg–Marquardt (*trainlm*) algorithm, which works well for nonlinear regression tasks. Four primary network architectures are employed: (2×4), (2×8), (4×4), and (4×8), indicating the quantity of neurons in the first and second hidden layers. The utilized activation functions are logsig for the initial hidden layer, tansig for the subsequent hidden layer, and purelin for the output layer. When more smoothing is needed, the study recommends adding time-averaged inputs (called +Avg), which reduces rapid fluctuations and leads to more reliable predictions.

Performance metrics such as MAE, RMSE, and R-values are employed to evaluate each design. The 4×8 model, without averaging, achieves the highest overall correlation coefficient (R = 0.99935), indicating remarkable agreement between predicted and actual angle targets. Models that use averaging handle noise better and adjust well to different situations, especially when there are sudden movements; however, they experience a slight delay because of the changing window.

In conjunction with the ANN-based technique, a Kalman filter method serves as a comparative benchmark. This filter relies on linear-Gaussian assumptions and has adjustable settings for process noise (Q) and measurement noise (R), which can also be calculated based on changes in the sensors. The Kalman filter yields dependable estimates in several systems; yet, abrupt noise spikes and complex data might diminish its efficacy in real-time biomedical applications.

Upon data collection, preparation, and model training, the final stage involves comparing the outcomes of the Kalman filter with the predictions generated by the ANN. Metrics such as Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) quantify discrepancies, whereas R-values assess the degree of correlation. Metrics like Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) represent absolute and squared deviations, respectively, whereas R-values evaluate correlation. The method further assesses real-time viability. The (4×8) ANN can process an individual sample in around 0.25 ms on microcontroller-class hardware, demonstrating its suitability for embedded applications.

The pipeline shown in Fig. 1 shows that ANN-based designs provide a flexible, adjustable, and quick way to estimate angles, especially in changing or noisy settings common in wearable biomedical devices and robotics. They exceed conventional filters like the Kalman Filter in several situations, and their versatility, such as including sensor covariance or averaging, enhances robust performance among real-world sensor disturbances.

A. Utilization of Gyroscope and Accelerometer in the MPU6050 Case Study

For this research, the MPU6050 sensor module is the main source of inertial data used to train and test artificial neural network (ANN) models. The MPU6050 is a 6-degree-of-freedom (6DOF) inertial measurement unit (IMU) that combines a 3-axis gyroscope and a 3-axis accelerometer onto a small chip. It is economical, easy to calibrate, and interoperable with several microcontroller systems via the I2C interface.

Fig. 2. MPU6050

The module's embedded Digital Motion Processor (DMP) can concurrently detect angular velocity and acceleration in real time. This is achieved using a 16-bit analog-to-digital converter (ADC). The gyroscope measures angular velocity (ω_{gyro}) up to ±500°/s, whilst the accelerometer records tilt-related acceleration values up to ±4g. These ranges are appropriate for applications requiring moderate to high-speed movement, such as balancing robots or wearable tracking systems.

Raw gyroscopic data are transformed into physical values by the conventional conversion procedure.

$$\omega_{gyro} = \frac{LSB_{read} - LSB_{Zero\ rotation\ value}}{Nominal\ sensitivity} \quad (1)$$

LSB_{read} signifies the current sensor measurement, $LSB_{Zero\ rotation\ value}$ indicates the baseline offset at rest, and sensitivity is defined by the sensor's configuration. An accelerometer-based tilt estimation derives the angle using gravitational acceleration.

$$\theta_{Accele} = arctan\left(\frac{LSB_{read} - LSB_{Zero\ rotation\ value}}{Nominal\ sensitivity}\right) \quad (2)$$

Angular signals from the MPU6050 that haven't been processed are used as inputs for both the basic Kalman filter processing and the suggested ANN-based angle estimation models. The MPU6050 works reliably in noisy and vibration-sensitive environments, as shown by tests that simulate shocks and surface disturbances. This makes it a good choice for building strong machine learning models in embedded systems.

B. Kalman Filter as Baseline Model for Angle Estimation

The Kalman filter acts as a standard for assessing its effectiveness compared to ANN-based methods. The filter is a recursive method used to determine the state of dynamic systems by integrating noisy sensor input across time. It has two principal phases: (1) Time Update (Prediction) and (2) Measurement Update (Correction). Even though changing Q and R helps reduce noise, assuming the system is linear and follows a Gaussian pattern can lead to errors in situations that are nonlinear or not Gaussian. Moreover, when real-time demands are rigorous, the filter's iterative matrix inversions must remain computationally feasible, as seen in Fig. 3.

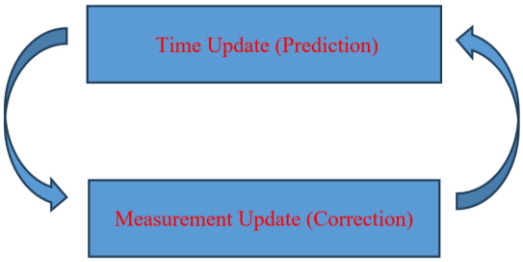

Fig. 3. Process of the Kalman Filter

The approach utilizes angular velocity data from the gyroscope (ω_{gyro}) and angle estimations from the accelerometer ((θ_{Accele}), received from the MPU6050 inertial sensor. The estimate becomes more precise when two critical components, process noise covariance (Q) and measurement noise covariance (R), derived from signal variance, are computed dynamically. This adaptive mechanism enables the Kalman filter to react to environmental changes, including vibrations or abrupt movements. It is essential to acknowledge that the Kalman filter assumes linear system models and Gaussian noise distributions. In practical situations characterized by highly nonlinear dynamics or non-Gaussian noise, performance may deteriorate, necessitating the exploration of alternate strategies, such as those based on artificial neural networks.

This document presents the discrete-time formulation of the Kalman filter.

Prediction Step

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \quad (3)$$

$$P_k^- = AP_{k-1}A^T + BQB^T \quad (4)$$

Correction Step

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \quad (5)$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (6)$$

$$P_k = (I - K_k H)P_k^- \quad (7)$$

where $\hat{x}_k$ represents the expected angle, $u_k = \omega_{gyro}$ specifies the angular velocity input obtained from the gyroscope, and $z_k = \theta_{Accele}$ indicates the angle measurement derived from accelerometer data.

The essential parameters for implementation are specified as $A = 1$, $B = 0.013$, $Q = 1 \times 10^{-12}$,$R = 1 \times 10^{-12}$, $H = 1$, and $I = 1$. These settings are selected for compatibility with real-time embedded applications utilizing the MPU6050 and Arduino Uno platform.

Fig. 3 illustrates the formulation of the Kalman filter. Furthermore, Fig. 4 presents an improved configuration that processes the accelerometer data through an averaging filter before it enters the Kalman block. This is implemented to reduce transient noise and enhance the stability of the estimations. The Kalman filter provides precise angle estimations despite variations in noise levels.

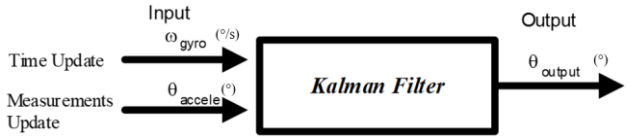

Fig. 4. Process of the Kalman Filter

Figuring out the right values for the Kalman filter settings, specifically the process noise covariance (Q) and measurement noise covariance (R), is essential for the filter to work well. The values are calculated dynamically based on the variance of sensor data to represent real-time noise characteristics. Q represents the variability of gyroscope observations, whereas R shows the variance in accelerometer data. This data-driven methodology enables the Kalman filter to adjust its assessment of internal uncertainty, thereby enhancing the accuracy and dependability of angle estimation in dynamic environments.

The Kalman filter works best with linear models and assumes that noise is Gaussian, which makes it less effective when there are sudden or very nonlinear changes in sensor data. Q and R are initially calculated using previous sensor data to establish a baseline. As the system evolves over time, particularly at each time step $k<N$, the covariance values are continuously updated according to the following formulas (8) and (9).

$$Cov(Q) = \frac{1}{N-1}\sum_{i=1}^{N}(\omega_i - \bar{\omega})^2 \quad (8)$$

$$Cov(R) = \frac{1}{N-1}\sum_{i=1}^{N}(\theta_i - \bar{\theta})^2 \quad (9)$$

This ongoing recalibration guarantees that the Kalman filter can adapt to fluctuations in sensor noise over time. This capability is essential in practice, as motion and vibration patterns are seldom static and may occasionally diverge considerably from the linear-Gaussian assumptions that form the basis of the Kalman framework.

C. Artificial Neural Network Models for Angle Estimation

To improve the precision of angle estimation from noisy inertial signals, this study implements several artificial neural network (ANN) architectures based on a feedforward neural network design. The input dataset consists of four principal variables: the angular measurements from the gyroscope (θ_{gyro}) and accelerometer (θ_{accele}), and the corresponding covariance values from each sensor (Cov_{gyro}), (Cov_{accele}). These input signals are concatenated into a feature matrix $I \in \mathbb{R}^{1\times N}$, where N is the number of time samples. The target output $T \in \mathbb{R}^{1\times N}$ corresponds to the ground-truth angle used for supervised learning.

Each ANN is structured with three layers: two hidden layers with nonlinear activation functions, log-sigmoid (logsig) and tan-sigmoid (tansig), and a final output layer with a linear activation function (purelin). The network function can be mathematically expressed as

$$\hat{y} = f_3(W_3 \cdot f_2(W_2 \cdot f_1(W_1 \cdot I + b_1) + b_2) + b_3) \quad (10)$$

Where $\hat{y}$ is the predicted angle output obtained from the neural network, W_i and b_i represent the weight matrices and bias vectors of layer i, respectively, and $f_1 = logsig$, $f_2 = tansig$, $f_3 = purelin$ are the activation functions used at each corresponding layer, enabling the network to map input data to the desired angle estimation through nonlinear transformations and linear output scaling.

The Levenberg–Marquardt (*trainlm*) optimization technique, a second-order approach, is employed for training, particularly effective for smaller datasets in nonlinear least-squares situations. Even though the Levenberg–Marquardt method converges quickly, it can use a lot of memory and may overfit when the dataset is too small, so it's important to validate the results to reduce this risk. The training parameters include a maximum epoch count of 2000 and a performance goal of mean squared error (MSE) = 1×10^{-12}, ensuring convergence toward a highly accurate model.

Four primary architectures are designed: (2×4), (2×8), (4×4), and (4×8), indicating the number of neurons in the first and second hidden layers, respectively. These architectures are visually represented in Fig. 5 and Fig. 6, where Fig. 5 shows the basic ANN model with two inputs (θ_{gyro} and θ_{accele}) and Fig. 6 illustrates the extended model that includes four inputs by incorporating the covariance values of both sensors. This expansion from two to four inputs enables the network to account not only for the raw angle measurements but also for the uncertainty associated with each sensor, thereby enhancing its robustness in noisy conditions. These configurations are systematically evaluated to assess how network depth and width affect regression accuracy in the presence of sensor noise. Notably, the (4×8) architecture achieves the highest regression performance, with an overall correlation coefficient of R = 0.99935, highlighting its superior ability to model complex nonlinear relationships among the input signals.

Fig. 5. Basic ANN Model with Angular Inputs from Gyroscope and Accelerometer

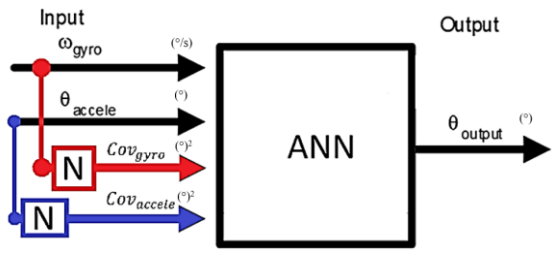

Fig. 6. Extended ANN Model Including Covariance Inputs for Sensor Fusion

Additionally, the study proposes an extended set of architectures that include temporal averaging of each input feature (i.e., moving average over a sliding window). These are denoted as (2×4+Avg), (2×8+Avg), (4×4+Avg), and (4×8+Avg). The averaged inputs provide smoothed versions of the raw data, reducing high-frequency fluctuations and improving robustness to transient disturbances. Formally, the moving average input for signal x at time k is computed as

$$x_{avg}(k) = \frac{1}{M} \sum_{i=k-M+1}^{K} x(i) \quad (11)$$

Where $M = 5$, which balances smoothing efficiency with a latency of around 65 ms and real-time responsiveness. Integrating these characteristics increases the input dimension to five or more, dependent upon implementation.

A total of 3,599 sensor samples were gathered at 13 ms intervals and allocated into 70% for training, 15% for validation, and 15% for testing. No synthetic noise was used, as the authentic sensor data exhibited adequate disruption. Empirical investigation indicates that models with averaged characteristics exhibit superior generalization during testing, especially in dynamic or non-stationary motion circumstances. This discovery indicates that the ANN can learn static correlations and adapt to temporal sensor activity, essential for real-time motion tracking. Initial real-time evaluations on a microcontroller-class system demonstrate that the (4×8) model can process a single sample in around 0.25 ms, indicating its viability for embedded biomedical or robotic applications.

Overall, these ANN models establish a practical and scalable solution for angle estimation tasks in embedded systems, especially where conventional filters such as Kalman or complementary filters may be limited.

Summary of ANN Architectures Table I outlines the primary ANN configurations assessed, employing the optional temporal averaging method (+Avg). The symbol "(m×n)" signifies m neurons in the first hidden layer and n neurons in the second hidden layer. The "+Avg" versions employ a moving average with a window size of $M = 5$ for all inputs before their transmission to the network.

TABLE I. SUMMARY OF ANN ARCHITECTURES AND ADDITIONAL PARAMETERS

Architecture	Hidden Layer 1 (neurons)	Hidden Layer 2 (neurons)	Averaging (M=5)
NN (2×4)	2	4	No
NN (2×8)	2	8	No
NN (4×4)	4	4	No
NN (4×8)	4	8	No
NN (2×4+Avg)	2	4	Yes
NN (2×8+Avg)	2	8	Yes
NN (4×4+Avg)	4	4	Yes
NN (4×8+Avg)	4	8	Yes

III. RESULTS AND DISCUSSION

This part talks about the results of the artificial neural network (ANN)-based angle estimation system and compares

them to the results of the traditional Kalman filter method. The efficacy of several ANN designs is evaluated for accuracy and noise resilience. The preprocessing techniques and noise characteristics of gyroscope and accelerometer data are evaluated as well.

Fig. 7 depicts the mechanical apparatus employed for angle measurement. The configuration guarantees precise data collection through regulated rotating motions. A steady support secures the IMU sensor to mitigate undesirable interferences. The data acquired from this configuration underpins the training and evaluation of the ANN models. Nevertheless, the generalizability of these discoveries to intricate, real-world scenarios may be restricted by the controlled character of the experimental setup.

Due to the limitations of Kalman filtering in addressing nonlinear noise fluctuations, ANN models were investigated as a substitute method for angle estimation. This investigation includes signal preprocessing, ANN training, and performance evaluation in contrast to conventional filtering methods. The results illustrate the efficacy of neural networks in enhancing the precision of estimation and mitigating sensor noise. The subsequent sections provide a critical analysis of the computational considerations, practical implications, and model trade.

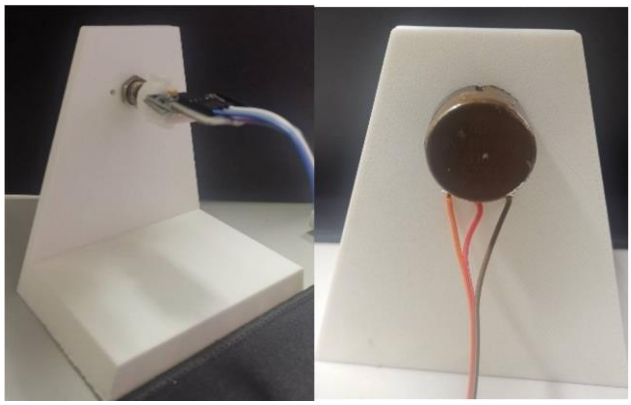

Fig. 7. Experimental Setup for Angular Measurement Using IMU Sensors

A. Signal Analysis and Noise Characteristics

This section looks at the noise patterns and signal features of accelerometer and gyroscope data, which are important for training and evaluating the ANN in predicting angles. The datasets comprise raw angle values from the gyroscope and accelerometer, along with a reference signal denoting the real angle, as seen in Fig. 8 and Fig. 9. The datasets were carefully divided into training (70%), validation (15%), and testing (15%) subsets to ensure thorough ANN training and unbiased performance assessment. Five variables provide the principal inputs of the neural network: the reference angle, the gyroscope angle, the accelerometer angle, the gyroscope covariance, and the accelerometer covariance. The gyroscope and accelerometer angles yield orientation measurements, while the covariance values reveal the uncertainty and noise characteristics of each sensor.

In contrast to the more steady gyroscope signals, accelerometer readings exhibit greater fluctuation, particularly during motion transitions, as seen by a study of the raw signals. Analyzing the noise covariance graphs clarifies the volatility. The accelerometer displays distinct pulses, signifying its sensitivity to external forces and motions. The gyroscope has little and consistent covariance, signifying enhanced dependability during uniform motion conditions. Temporal drift compromises the gyroscope's stability over time. The problem highlights the need for fusion systems that combine the benefits of both sensors and a fusion approach for the effective integration of sensor data. Fig. 8 illustrates the training phase, whereas Fig. 9 depicts the evaluation phase. Both demonstrate the effect of noise on the raw sensor data. The variance plots demonstrate the need to incorporate these properties during training to provide reliable performance under diverse signal conditions.

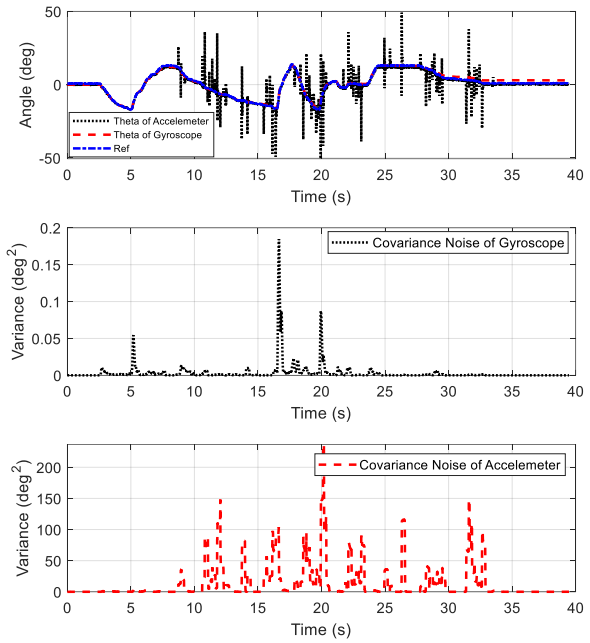

Fig. 8. Training Dataset for ANN: Gyroscope, Accelerometer, and Reference Angle Signals

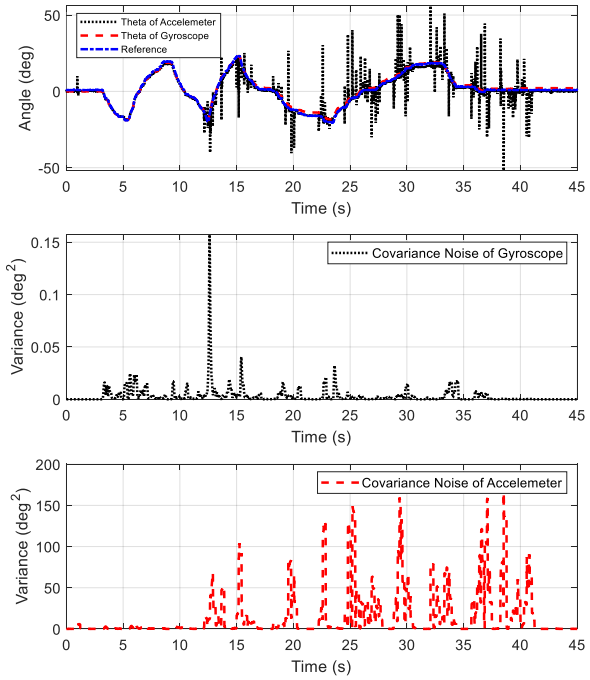

Fig. 9. Testing Dataset for ANN: Gyroscope, Accelerometer, and Reference Angle Signals

A fusion model was created employing artificial neural networks and various feedforward architectures to amalgamate sensor data, as seen in Fig. 10. The selected architectures (2×4, 2×8, 4×4, 4×8) provide a systematic assessment of the model's performance across different difficulties. Each neural networks consists of three layers utilizing nonlinear activation functions (*logsig* and *tansig*) in combination with a linear output layer. The chosen architectural changes are meant to see what happens to the model's ability to predict angles when things aren't going as planned when the hidden layer complexity changes.

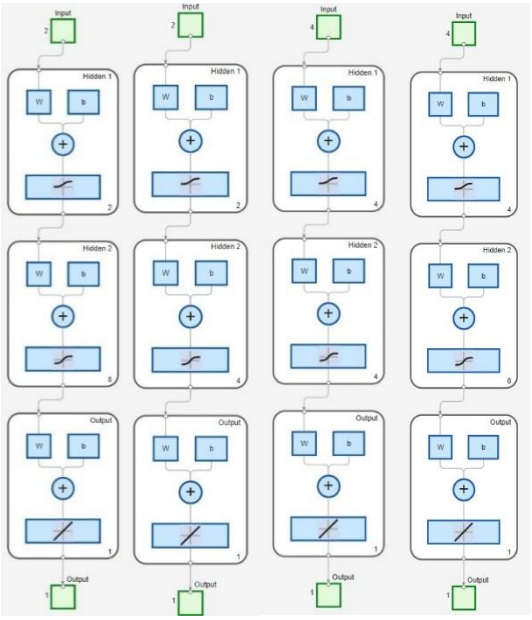

Fig. 10. ANN-Based Sensor Fusion Architectures for IMU Angle Estimation

Fig. 11 to Fig. 14 demonstrate the effectiveness of each ANN construction in regression tasks. Four subplots within each image depict the outcomes of training, validation, testing, and total regression. The graphs depict the anticipated output (calculated angle) with the actual data (target angle), in addition to a fitted linear regression line. The correlation coefficient (R-value) is a crucial performance metric in these diagrams, since it measures the linear connection between expected and actual outcomes. Estimates with values approaching 1 demonstrate more accuracy.

The feedforward neural networks with a (2×4) architecture, as seen in Fig. 11, achieved an R-value of 0.99704 for training, 0.99675 for validation, 0.9967 for testing, and an overall R-value of 0.99695. In many data contexts, these outcomes provide strong predictive accuracy with negligible variance.

The (2×8) design works because the regression results are better across all subgroups (0.99746 for training, 0.99743 for validation, 0.99731 for testing, and 0.99743 overall) (Fig. 12). The augmentation of neurons in the second layer enhanced learning and generalization. This resulted in improved angle calculations for both known and unknown data.

The (4×4) configuration depicted in Fig. 13 achieved an outstanding R-value of 0.99755 for training and 0.99752 overall, with both validation and testing values at 0.99745. This design proposes a configuration that achieves an ideal equilibrium between computational complexity and depth. This diminishes the likelihood of overfitting and improves predictive accuracy.

The regression performance of the (4×8) design, seen in Fig. 14, is ideal, with R-values of 0.99934 for training, 0.99935 for validation, and 0.99935 overall. The documented R-value diminishes somewhat to 0.99337. Notwithstanding a little reduction in generalization, the model exhibits exceptional performance throughout both training and validation. This may result from overfitting or heightened sensitivity to variations in the test data.

The R-values rise from Fig. 11 to Fig. 14. By adding more neurons and input dimensions, the network may be better able to figure out complex, nonlinear connections between random sensor outputs and reference angles. The regression results show that the ANN model works well and is right for systems that need to accurately integrate sensors and estimate angles in real time.

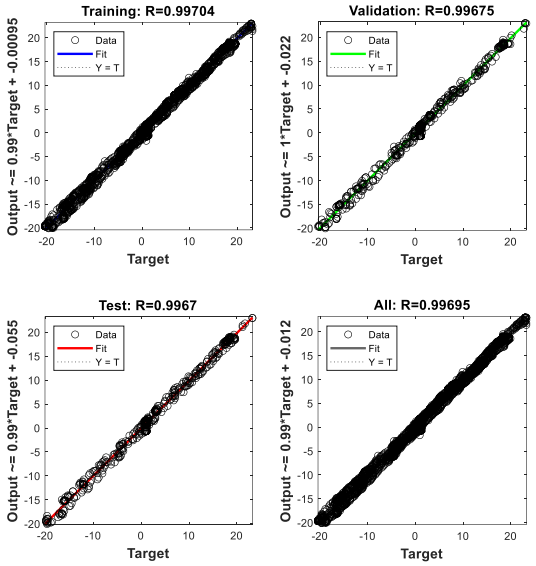

Fig. 11. Regression Performance of Feedforward Neural Network with 2×4 Architecture

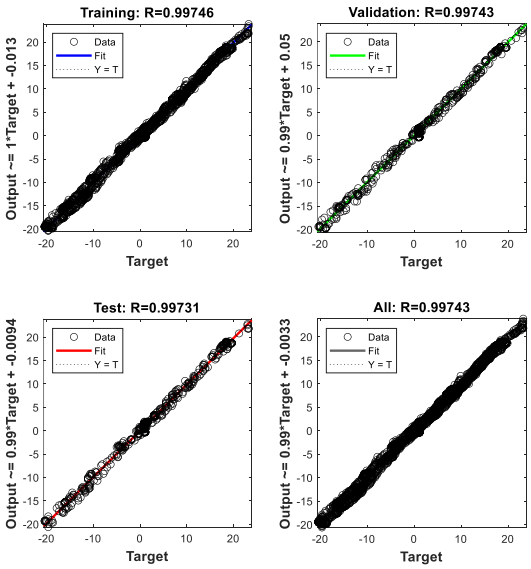

Fig. 12. Regression Performance of Feedforward Neural Network with 2×8 Architecture

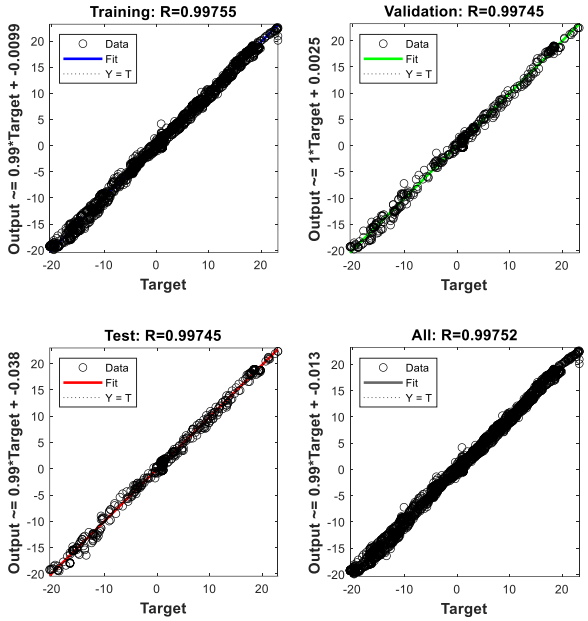

Fig. 13. Regression Performance of Feedforward Neural Network with 4×4 Architecture

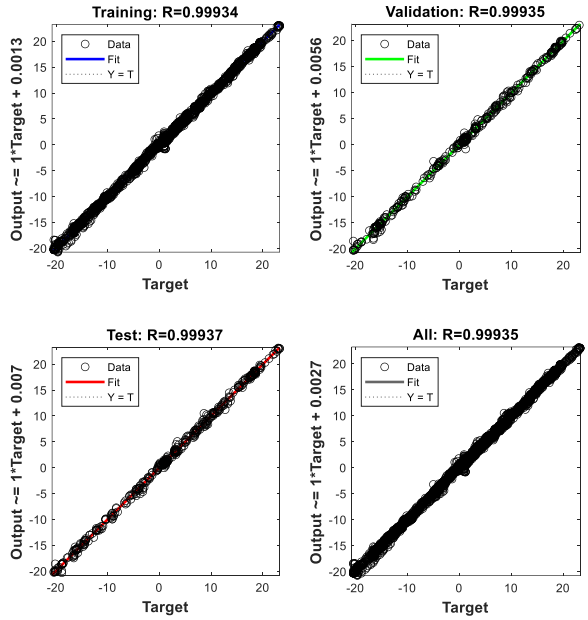

Fig. 14. Regression Performance of Feedforward Neural Network with 4×8 Architecture

B. ANN Output Comparison for Small Network Architectures

This section evaluates the effectiveness of artificial neural networks with simplified topologies. The estimation results utilizing ANN models with two hidden layers and diverse neuron topologies are presented in Fig. 15, Fig. 16, and Fig. 17. The results demonstrate that both models surpass the Kalman filter by diminishing noise and more adeptly capturing motion dynamics, while the averaging technique also improves stability.

Notwithstanding commendable overall performance, smaller ANN structures (2×4, 2×8) can fall short in accurately capturing swift beginning motion dynamics in comparison to the Kalman filter. Nonetheless, ANN models ultimately align more closely with the reference line over time, signifying enhanced adaptability to various motion patterns.

Fig. 15 presents a comprehensive comparison of Kalman filtering and neural network models with two hidden layers (2×4, 2×8). The results indicate that while the Kalman filter efficiently mitigates noise, neural networks provide a more reliable and accurate prediction over time, particularly during dynamic transitions.

Fig. 16 provides a comprehensive analysis of the first motion phase, concentrating on the duration from 2 to 10 seconds. Initially, the Kalman filter has higher performance relative to the neural networks since it responds more rapidly to early oscillations. Over time, the neural network models increasingly converge toward the intended reference line. This indicates that they can acquire knowledge and adjust to various motion patterns more effectively than the Kalman filter.

Fig. 17 illustrates the steady-state transition phase happening between 30 and 40 seconds. At this juncture, neural networks emerge as the superior choice due to their ability to maintain a more consistent trajectory with less departure from the reference angle. The Kalman filter, while effective in noise reduction, has considerable inaccuracies, particularly during transitional periods. The neural network models demonstrate increased adaptability, consistently aligning with the expected values.

Compared to Kalman filtering, the results show that neural networks are a more reliable and accurate way to estimate angles, especially when there are big changes in motion and noise. Averaging algorithms enhance the stability of neural network predictions, making them very suitable for practical applications.

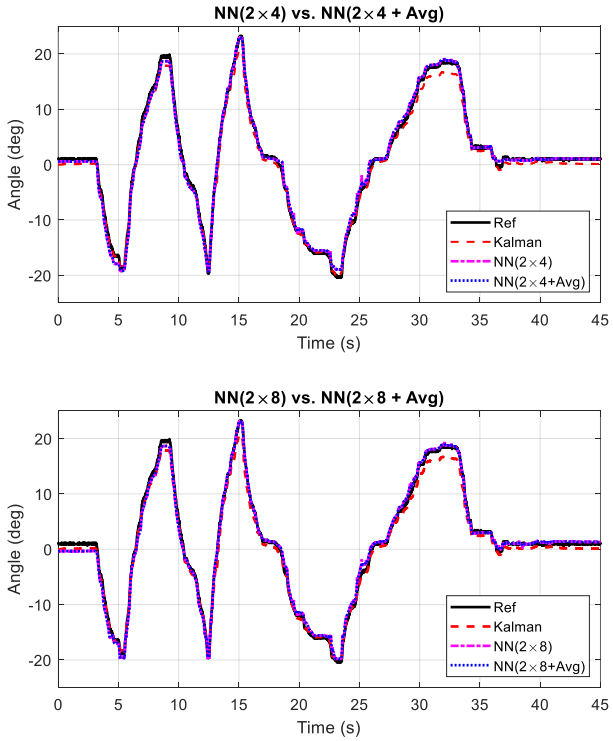

Fig. 15. Full-Time Comparison of Kalman Filter and Neural Networks (2×4, 2×8) for Angle Estimation

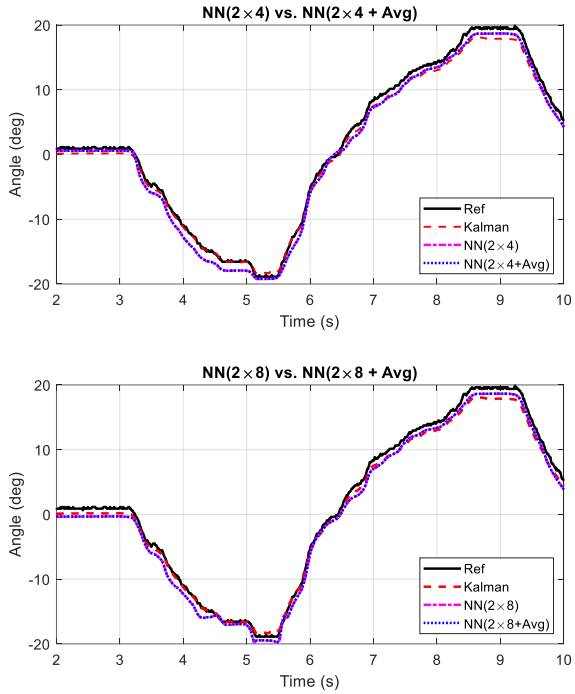

Fig. 16. Zoomed-in Analysis (2s-10s): Initial Motion Phase with NN (2×4, 2×8) and Kalman Filter

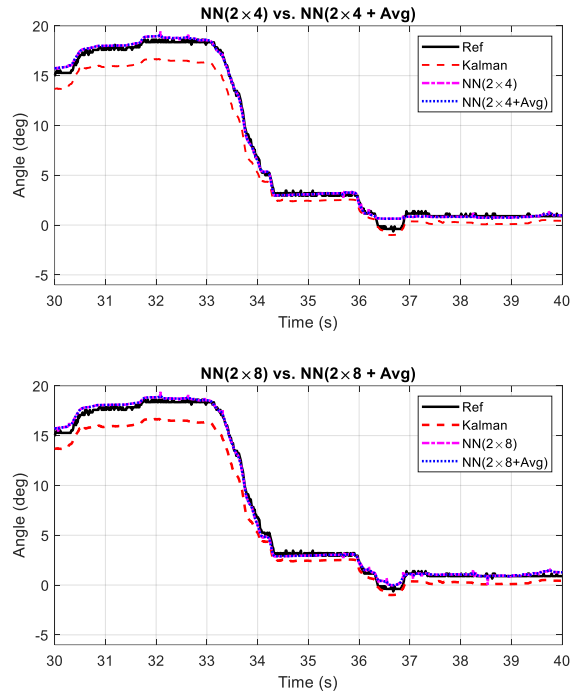

Fig. 17. Zoomed-in Analysis (30s-40s): Steady-State Transition with NN (2×4, 2×8) and Kalman Filter

C. ANN Output Comparison for Larger Network Architectures

Artificial neural network architectures with larger hidden layers (4×4, 4×8) provide improved estimation accuracy, as seen in Fig. 18, Fig. 19, and Fig. 20. Augmented neural networks improve robustness by stabilizing predictions and reducing estimation error in both static and dynamic environments. However, increased model complexity results in greater computation requirements that must be carefully assessed in real-time embedded system applications.

The neural network model (4×8 with averaging) effectively reduces angle errors caused by noisy inputs, resulting in accurate angle predictions. However, it's important to note that averaging adds a slight delay of about 65 milliseconds to processing time. It is essential to acknowledge that averaging incurs a little delay in processing time, around 65 milliseconds. Despite its advantages, the little processing delay caused by averaging (~65 ms) must be acknowledged.

The full-time comparison of Kalman filtering and neural network models with four hidden layers (4x4, 4x8) is shown in Fig. 18. The results show that larger neural networks are always better than the Kalman filter, especially when it comes to getting rid of signal noise and keeping smooth transitions.

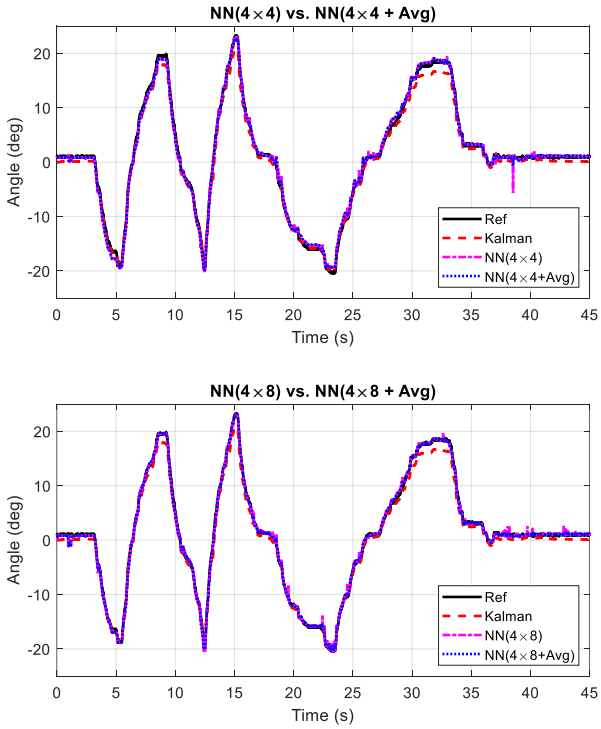

Fig. 18. Full-Time Comparison of Kalman Filter and Neural Networks (4×4, 4×8) for Angle Estimation

The initial motion phase is comprehensively examined in Fig. 19, with a particular emphasis on the period extending from 2 to 10 seconds. As evidenced by the results, the neural network (4×8) and neural network (4×8 plus averaging) models demonstrate superior stability in the presence of rapid motion changes. In contrast to the Kalman filter, which displays moderate oscillations, the larger ANN models effectively monitor the reference trajectory while suppressing undesirable fluctuations.

Fig. 20 illustrates the steady-state transition phase, which has a duration of 30 to 40 seconds. The neural network (4×8 plus averaging) model is very good at reducing the angular disturbances caused by high-noise signals. In addition to guaranteeing high accuracy and preventing abrupt deviations, the averaging technique further smoothes the approximated angles. In comparison to other models, the NN (4×8 + Averaging) model approach is particularly well-suited for applications that necessitate precision angle surveillance, as it offers a more refined estimation.

These results show that adding more complexity to ANN models makes them much more useful in changing environments, especially when they are combined with techniques for averaging. The NN (4×8 + averaging) model is very strong; it can reduce noise-induced angle deviations better than both the standard Kalman filter and less-sophisticated ANN configurations.

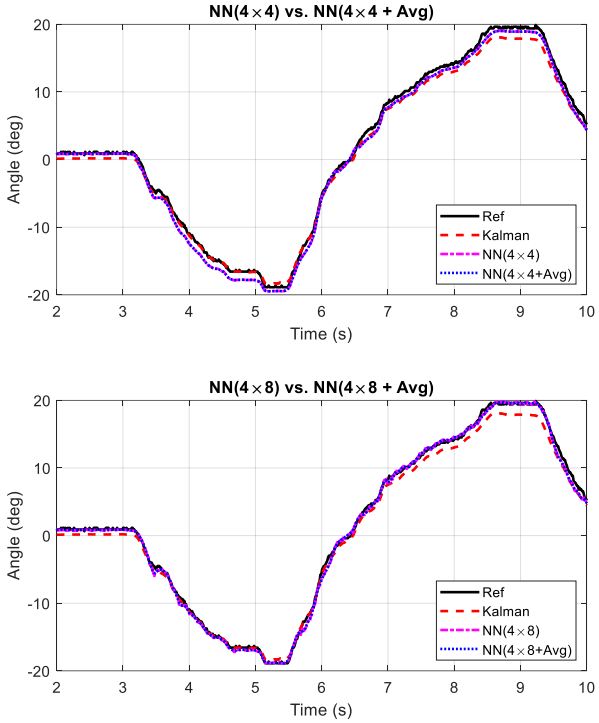

Fig. 19. Zoomed-in Analysis (2s-10s): Initial Motion Phase with NN (4×4, 4×8) and Kalman Filter

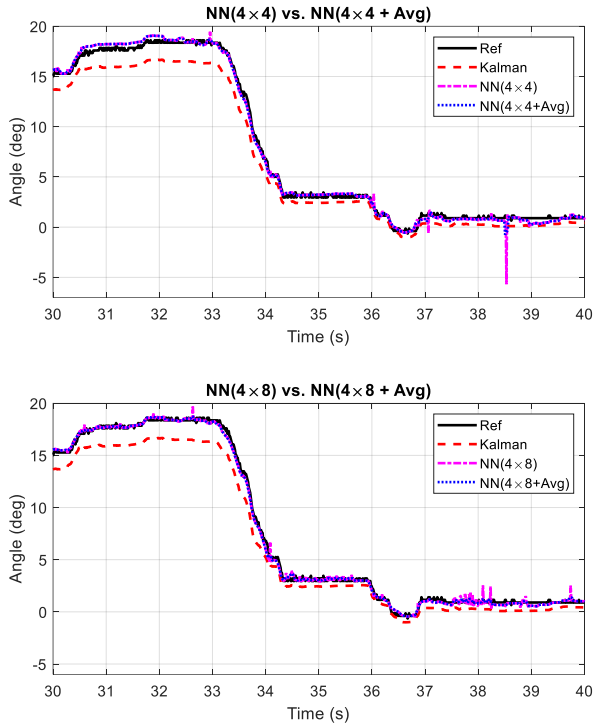

Fig. 20. Zoomed-in Analysis (30s-40s): Steady-State Transition with NN (4×4, 4×8) and Kalman Filter

D. Error Metrics Comparison

Estimation strategies were evaluated quantitatively using mean absolute error (MAE) and root mean squared error (RMSE). Table II summarizes the results, demonstrating the superior accuracy of all ANN models relative to the Kalman filter. The NN (4×8 + Averaging) model had the lowest MAE (0.2657) and RMSE (0.3691).

TABLE II. ERROR METRICS COMPARISON

Method	MAE	RMSE
Kalman Filter	0.8354	1.0055
NN (2x4)	0.5125	0.681
NN (2x8)	0.5915	0.7397
NN (4x4)	0.4411	0.5901
NN (4x8)	0.2849	0.4112
NN (2x4+Avg)	0.5105	0.6773
NN (2x8+Avg)	0.5878	0.7335
NN (4x4+Avg)	0.4307	0.5672
NN (4x8+Avg)	0.2657	0.3691

The NN (4×8 + Averaging) model has outstanding predictive ability, with a minimum MAE of 0.2657 and an RMSE of 0.3691. This means that making the model more complicated lowers the effects of noise on variations, which improves the accuracy of estimates, especially when used with an averaging method.

The results indicate that neural networks offer significantly enhanced accuracy, although Kalman filtering continues to be a fundamental approach. The superiority of models with larger hidden layers (4×8) over those with smaller configurations (2×4, 2×8) underscores the significance of neural networks depth in enhancing angle estimation. Averaging methods consistently decrease errors across diverse ANN configurations.

The findings indicate that deep learning methods provide a reliable substitute for conventional filtering techniques, particularly in situations where sensor noise poses a constraint. The NN (4×8 + Averaging) model did very well, showing that ANN topologies can be changed to lower noise and improve real-time angle estimation.

E. Discussion on ANN Advantages and Future Improvements

The findings unequivocally indicate that artificial neural networks (ANNs) much surpass conventional Kalman filters, especially in dynamic or high-noise environments. Kalman filters depend on linear assumptions and established noise models, but artificial neural networks may adjust to nonlinearities and sensor uncertainties via learnt representations. This capability enhances the robustness of ANNs in situations with intricate motion dynamics.

However, ANN models pose practical challenges, particularly in real-time embedded systems that face constraints in processor power, memory, and energy consumption. The NN (4×8 averaging) methodology, while precise, necessitates around 0.25 milliseconds per sample on microcontroller-class hardware. This delay is permissible in several real-time systems but may remain a limitation in ultra-low-power biomedical equipment.

Future research should explore hybrid methodologies that integrate ANN models with Kalman filters in either cascaded or parallel configurations to address these problems. An artificial neural network (ANN) may be employed for noise modeling or prediction, whereas a Kalman filter is utilized for real-time correction. These hybrid systems can offer the flexibility and adaptability of artificial neural networks in conjunction with the speed and reliability of traditional filters.

Furthermore, the generalizability of ANN models must be corroborated using a broader range of datasets. Recent tests were performed under regulated rotation utilizing a singular IMU sensor. Testing these models in real-life situations, such as different walking styles, fall detection scenarios, and unpredictable movement settings, will improve how we measure their strength and reliability.

Methods like as transfer learning, online learning, and network pruning can diminish training overhead and resource consumption, enhancing the practicality of deploying ANN models. Researchers may investigate sophisticated designs such as recurrent neural networks (RNNs) or transformer-based models to improve sequential motion tracking and temporal forecasting.

The proposed ANN-based models are very adaptable, reliable, and accurate, making them highly suitable for using IMUs in tracking movement for health monitoring, self-driving systems, and robotics that operate in real-time. Future endeavors must confront limits of scalability, dataset variety, and hardware feasibility to guarantee wider use in actual systems.

IV. CONCLUSION

This research demonstrated the effectiveness of artificial neural networks (ANNs) in accurately calculating angles using inertial measurement unit (IMU) data. The suggested ANN models did much better than the old Kalman filter, particularly when using bigger network designs and averaging over time. The neural network (4×8 + averaging) achieved the lowest error rates, exhibiting enhanced stability and robustness in handling noisy and dynamic conditions. These discoveries are very relevant to biomedical engineering (BME), particularly in gait analysis, prosthetic limb management, rehabilitation robots, and wearable health monitoring systems. Accurate angle measurements improve motion tracking, posture assessments, and real-time feedback, hence increasing patient care and assistive technology. Even though the ANN approach has many benefits, it needs a lot of computing power and takes a long time to train, which makes it difficult to use in real-time medical devices. Future research ought to concentrate on addressing these issues by developing systems that operate efficiently with constrained resources, employing a combination of various filtering techniques (such as integrating artificial neural networks with Kalman filters), and investigating more sophisticated deep learning architectures (including recurrent neural networks or transformer models). In summary, using ANN for angle estimation is a better option than traditional filtering methods because it shows better accuracy, is less affected by interference, and has potential uses in biology and engineering. Addressing computational and data-related limitations through targeted future research will enhance the use and effectiveness of ANN techniques in real-world applications.

ACKNOWLEDGMENT

The researcher acknowledges the financial assistance the research team received from the research institute, academic services center, and college of biomedical engineering at Rangsit University. The Ethics Review Board of Rangsit University has evaluated the study, reference number RSUERB2025-006, and confirmed that the research does not involve human subjects. Moreover, AI-driven methods (QuillBot Premium) were utilized for grammatical verification, paraphrasing, and linguistic augmentation to ensure the accuracy and clarity of the text.

REFERENCES

[1] M. Barbary and M. H. Abd ElAzeem, "Drones tracking based on robust cubature Kalman-TBD-multi-Bernoulli filter," *ISA Transactions*, vol. 114, pp. 277–290, 2021, doi: 10.1016/j.isatra.2020.12.042.

[2] K. Ansari and P. Jamjareegulgarn, "Effect of weighted PDOP on performance of linear Kalman filter for RTK drone data," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–4, 2022, doi: 10.1109/LGRS.2022.3204323.

[3] M. L. Hoang, M. Carratù, V. Paciello, and A. Pietrosanto, "Fusion Filters between the No Motion No Integration Technique and Kalman Filter in Noise Optimization on a 6DoF Drone for Orientation Tracking," *Sensors*, vol. 23, no. 12, p. 5603, 2023, doi: 10.3390/s23125603.

[4] Á. Odry, I. Kecskes, P. Sarcevic, Z. Vizvari, A. Toth, and P. Odry, "A novel fuzzy-adaptive extended Kalman filter for real-time attitude estimation of mobile robots," *Sensors*, vol. 20, no. 3, p. 803, 2020, doi: 10.3390/s20030803.

[5] F. J. González-Castaño, F. Gil-Castineira, D. Rodriguez-Pereira, J. Á. Regueiro-Janeiro, S. Garcia-Mendez, and D. Candal-Ventureira, "Self-corrective sensor fusion for drone positioning in indoor facilities," *IEEE Access*, vol. 9, pp. 2415–2427, 2020, doi: 10.1109/ACCESS.2020.3048194.

[6] Z. Dai and L. Jing, "Lightweight extended Kalman filter for MARG sensors attitude estimation," *IEEE Sensors Journal*, vol. 21, no. 13, pp. 14749–14758, 2021, doi: 10.1109/JSEN.2021.3072887.

[7] F. Marino and G. Guglieri, "Beyond Static Obstacles: Integrating Kalman Filter with Reinforcement Learning for Drone Navigation," *Aerospace*, vol. 11, no. 5, p. 395, 2024, doi: 10.3390/aerospace11050395.

[8] W. An, T. Lin, and P. Zhang, "An Autonomous Soaring for Small Drones Using the Extended Kalman Filter Thermal Updraft Center Prediction Method Based on Ordinary Least Squares," *Drones*, vol. 7, no. 10, p. 603, 2023, doi: 10.3390/drones7100603.

[9] S. Srey and S. Srang, "Adaptive Controller Based on Estimated Parameters for Quadcopter Trajectory Tracking," *International Journal of Robotics and Control Systems*, vol. 4, no. 2, pp. 480–501, 2024, doi: 10.31763/ijrcs.v4i2.1342.

[10] I. Kurniasari and A. Ma'arif, "Implementing PID-Kalman Algorithm to Reduce Noise in DC Motor Rotational Speed Control," *International Journal of Robotics and Control Systems*, vol. 4, no. 2, pp. 958–978, 2024, doi: 10.31763/ijrcs.v4i2.1309.

[11] R. I. Alfian, A. Ma'arif, and S. Sunardi, "Noise reduction in the accelerometer and gyroscope sensor with the Kalman filter algorithm," *Journal of Robotics and Control*, vol. 2, no. 3, pp. 180–189, 2021, doi: 10.18196/jrc.2375.

[12] Y. Zhu, J. Liu, R. Yu, Z. Mu, L. Huang, J. Chen, and J. Chen, "Attitude solving algorithm and FPGA implementation of four-rotor UAV based on improved Mahony complementary filter," *Sensors*, vol. 22, no. 17, p. 6411, 2022, doi: 10.3390/s22176411.

[13] A. Basiri, V. Mariani, and L. Glielmo, "Improving visual SLAM by combining SVO and ORB-SLAM2 with a complementary filter to enhance indoor mini-drone localization under varying conditions," *Drones*, vol. 7, no. 6, p. 404, 2023, doi: 10.3390/drones7060404.

[14] H. Dong, J. Liu, C. Wang, H. Cao, C. Shen, and J. Tang, "Drone detection method based on the time-frequency complementary enhancement model," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1-12, 2023,doi: 10.1109/TIM.2023.3328072.

[15] N. Srinidhi, J. Shreyas, and E. Naresh, "Establishing Self-Healing and Seamless Connectivity among IoT Networks Using Kalman Filter," *Journal of Robotics and Control*, vol. 3, no. 5, pp. 646–655, 2022, doi: 10.18196/jrc.v3i5.11622.

[16] B. Skorohod, "Finite Impulse Response Filtering Algorithm with Adaptive Horizon Size Selection and Its Applications," *Journal of Robotics and Control*, vol. 3, no. 6, pp. 836–847, 2023, doi: 10.18196/jrc.v3i6.16058.

[17] V. Shenoy and S. Vekata, "Estimation of Liquid Level in a Harsh Environment Using Chaotic Observer," *Journal of Robotics and Control*, vol. 3, no. 5, pp. 566–582, 2022, doi: 10.18196/jrc.v3i5.16183.

[18] Z. Zainudin and S. Kodagoda, "Gaussian Processes-BayesFilters with Non-Parametric Data Optimization for Efficient 2D LiDAR Based People Tracking," *International Journal of Robotics and Control Systems*, vol. 3, no. 2, pp. 206–220, 2023, doi: 10.31763/ijrcs.v3i2.901.

[19] T. Habib, "Magnetometer-Only Kalman Filter Based Algorithms for High Accuracy Spacecraft Attitude Estimation (A Comparative Analysis)," *International Journal of Robotics and Control Systems*, vol. 3, no. 3, pp. 433–448, 2023, doi: 10.31763/ijrcs.v3i3.988.

[20] M. Salwa and I. Krzysztofik, "Application of filters to improve flight stability of rotary unmanned aerial objects," *Sensors*, vol. 22, no. 4, p. 1677, 2022, doi: 10.3390/s22041677.

[21] W. T. Higgins, "A comparison of complementary and Kalman filtering," *IEEE Transactions on Aerospace and Electronic Systems*, no. 3, pp. 321–325, 1975, doi: 10.1109/TAES.1975.308081.

[22] P. Narkhede, S. Poddar, R. Walambe, G. Ghinea, and K. Kotecha, "Cascaded complementary filter architecture for sensor fusion in attitude estimation," Sensors, vol. 21, no. 6, p. 1937, 2021, doi: 10.3390/s21061937.

[23] M. Al Borno *et al*., "OpenSense: An open-source toolbox for inertial-measurement-unit-based measurement of lower extremity kinematics over long durations," *Journal of NeuroEngineering and Rehabilitation*, vol. 19, no. 1, p. 22, 2022, doi: 10.1186/s12984-022-01001-x.

[24] V. Vijayan, J. P. Connolly, J. Condell, N. McKelvey, and P. Gardiner, "Review of wearable devices and data collection considerations for connected health," *Sensors*, vol. 21, no. 16, p. 5589, 2021, doi: 10.3390/s21165589.

[25] S. O. Madgwick, S. Wilson, R. Turk, J. Burridge, C. Kapatos, and R. Vaidyanathan, "An extended complementary filter for full-body MARG orientation estimation," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 4, pp. 2054–2064, Aug. 2020, doi: 10.1109/TMECH.2020.2992296.

[26] W. Liang, J. Long, K. C. Li, J. Xu, N. Ma, and X. Lei, "A fast defogging image recognition algorithm based on bilateral hybrid filtering," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 17, no. 2, pp. 1–16, 2021, doi: 10.1145/3391297.

[27] C. Urrea and R. Agramonte, "Kalman filter: historical overview and review of its use in robotics 60 years after its creation," *Journal of Sensors*, vol. 2021, p. 9674015, 2021, doi: 10.1155/2021/9674015.

[28] R. V. Vitali, R. S. McGinnis, and N. C. Perkins, "Robust error-state Kalman filter for estimating IMU orientation," *IEEE Sensors Journal*, vol. 21, no. 3, pp. 3561–3569, 2021, doi: 10.1109/JSEN.2020.3026895.

[29] M. Khodarahmi and V. Maihami, "A review on Kalman filter models," *Archives of Computational Methods in Engineering*, vol. 30, no. 1, pp. 727–747, 2023, doi: 10.1007/s11831-022-09815-7.

[30] J. Khodaparast, "A review of dynamic phasor estimation by non-linear Kalman filters," *IEEE Access*, vol. 10, pp. 11090–11109, 2022, doi: 10.1109/ACCESS.2022.3146732.

[31] A. K. Singh, "Major development under Gaussian filtering since unscented Kalman filter," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 5, pp. 1308–1325, 2020, doi: 10.1109/JAS.2020.1003303.

[32] H. Liu, F. Hu, J. Su, X. Wei, and R. Qin, "Comparisons on Kalman-filter-based dynamic state estimation algorithms of power systems," *IEEE Access*, vol. 8, pp. 51035–51043, 2020, doi: 10.1109/ACCESS.2020.2979735.

[33] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, "Contact-aided invariant extended Kalman filtering for robot state estimation," *International Journal of Robotics Research*, vol. 39, no. 4, pp. 402–430, 2020, doi: 10.1177/0278364919894385.

[34] Y. T. Bai, X. Y. Wang, X. B. Jin, Z. Y. Zhao, and B. H. Zhang, "A neuron-based kalman filter with nonlinear autoregressive model," *Sensors*, vol. 20, no. 1, p. 299, 2020, doi: 10.3390/s20010299.

[35] I. Ullah, X. Su, X. Zhang, and D. Choi, "Simultaneous localization and mapping based on Kalman filter and extended Kalman filter," *Wireless Communications and Mobile Computing*, vol. 2020, p. 2138643, 2020, doi: 10.1155/2020/2138643.

[36] P. Poncela, E. Ruiz, and K. Miranda, "Factor extraction using Kalman filter and smoothing: This is not just another survey," *International Journal of Forecasting*, vol. 37, no. 4, pp. 1399–1425, 2021, doi: 10.1016/j.ijforecast.2021.01.027.

[37] Y. Huang, Y. Zhang, Y. Zhao, P. Shi, and J. A. Chambers, "A novel outlier-robust Kalman filtering framework based on statistical similarity measure," *IEEE Transactions on Automatic Control*, vol. 66, no. 6, pp. 2677–2692, 2020, doi: 10.1109/TAC.2020.3011443.

[38] M. Impraimakis and A. W. Smyth, "An unscented Kalman filter method for real time input-parameter-state estimation," *Mechanical Systems and Signal Processing*, vol. 162, p. 108026, 2022, doi: 10.1016/j.ymssp.2021.108026.

[39] S. Sharma, A. Majumdar, V. Elvira, and E. Chouzenoux, "Blind Kalman filtering for short-term load forecasting," *IEEE Transactions on Power Systems*, vol. 35, no. 6, pp. 4916–4919, 2020, doi: 10.1109/TPWRS.2020.3018623.

[40] Y. Huang, F. Zhu, G. Jia, and Y. Zhang, "A slide window variational adaptive Kalman filter," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 12, pp. 3552–3556, 2020, doi: 10.1109/TCSII.2020.2995714.

[41] E. R. Potokar, K. Norman, and J. G. Mangelson, "Invariant extended kalman filtering for underwater navigation," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5792–5799, 2021, doi: 10.1109/LRA.2021.3085167.

[42] W. Wen, T. Pfeifer, X. Bai, and L. T. Hsu, "Factor graph optimization for GNSS/INS integration: A comparison with the extended Kalman filter," *NAVIGATION: Journal of the Institute of Navigation*, vol. 68, no. 2, pp. 315–331, 2021, doi: 10.1002/navi.421.

[43] K. D. Rocha and M. H. Terra, "Robust Kalman filter for systems subject to parametric uncertainties," *Systems & Control Letters*, vol. 157, p. 105034, 2021, doi: 10.1016/j.sysconle.2021.105034.

[44] M. Song, R. Astroza, H. Ebrahimian, B. Moaveni, and C. Papadimitriou, "Adaptive Kalman filters for nonlinear finite element model updating," *Mechanical Systems and Signal Processing*, vol. 143, p. 106837, 2020, doi: 10.1016/j.ymssp.2020.106837.

[45] H. Fang, M. A. Haile, and Y. Wang, "Robust extended Kalman filtering for systems with measurement outliers," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 2, pp. 795–802, 2021, doi: 10.1109/TCST.2021.3077535.

[46] G. Hu, B. Gao, Y. Zhong, and C. Gu, "Unscented Kalman filter with process noise covariance estimation for vehicular INS/GPS integration system," *Information Fusion*, vol. 64, pp. 194–204, 2020, doi: 10.1016/j.inffus.2020.08.005.

[47] W. Wang, N. He, K. Yao, and J. Tong, "Improved Kalman filter and its application in initial alignment," *Optik*, vol. 226, p. 165747, 2021, doi: 10.1016/j.ijleo.2020.165747.

[48] A. Tsiamis and G. J. Pappas, "Online learning of the kalman filter with logarithmic regret," *IEEE Transactions on Automatic Control*, vol. 68, no. 5, pp. 2774–2789, 2022, doi: 10.1109/TAC.2022.3207670.

[49] Y. Sun, W. Bao, K. Valk, C. C. Brauer, J. Sumihar, and A. H. Weerts, "Improving forecast skill of lowland hydrological models using ensemble Kalman filter and unscented Kalman filter," *Water Resources Research*, vol. 56, no. 8, p. e2020WR027468, 2020, doi: 10.1029/2020WR027468.

[50] Y. Huang, G. Jia, B. Chen, and Y. Zhang, "A new robust Kalman filter with adaptive estimate of time-varying measurement bias," *IEEE Signal Processing Letters*, vol. 27, pp. 700–704, 2020, doi: 10.1109/LSP.2020.2983552.

[51] S. Yi and M. Zorzi, "Robust kalman filtering under model uncertainty: The case of degenerate densities," *IEEE Transactions on Automatic*

Control, vol. 67, no. 7, pp. 3458–3471, 2021, doi: 10.1109/TAC.2021.3106861.

[52] M. Bai, Y. Huang, B. Chen, and Y. Zhang, "A novel robust Kalman filtering framework based on normal-skew mixture distribution," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 52, no. 11, pp. 6789–6805, 2021, doi: 10.1109/TSMC.2021.3098299.

[53] L. Ma *et al.*, "Fault-tolerant control based on modified eXogenous Kalman filter for PMSM," Journal of Electrical Engineering & Technology, vol. 18, pp. 1313–1323, 2023, doi: 10.1007/s42835-022-01223-y.

[54] Y. Cheng, Y. Li, K. Li, X. Liu, C. Liu, and X. Hao, "Fusing LSTM neural network and expanded disturbance Kalman filter for estimating external disturbing forces of ball screw drives," *Robotics and Computer-Integrated Manufacturing*, vol. 89, p. 102776, 2024, doi: 10.1016/j.rcim.2024.102776.

[55] Y. Kang, Z. Qiu, X. Huang, Z. Kong, F. Gu, and A. D. Ball, "Field simultaneous estimation of residual unbalance and bearing dynamic coefficients for double-disk rotor-bearing system using dual augmented Kalman filter," *Journal of Sound and Vibration*, vol. 577, p. 118325, 2024, doi: 10.1016/j.jsv.2024.118325.

[56] A. Srichandan, J. Dhingra, and M. K. Hota, "An improved Q-learning approach with Kalman filter for self-balancing robot using OpenAI," *Journal of Control, Automation and Electrical Systems*, vol. 32, no. 6, pp. 1521–1530, 2021, doi: 10.1007/s40313-021-00786-x.

[57] E. Rabb and J. J. Steckenrider, "Walking trajectory estimation using multi-sensor fusion and a probabilistic step model," Sensors, vol. 23, no. 14, p. 6494, 2023, doi: 10.3390/s23146494.

[58] J. Zhao, J. Li, and J. Zhou, "Research on two-round self-balancing robot SLAM based on the gmapping algorithm," *Sensors*, vol. 23, no. 5, p. 2489, 2023, doi: 10.3390/s23052489.

[59] H. A. O. Mohamed, G. Nava, G. L'Erario, S. Traversaro, F. Bergonti, L. Fiorio, et al., "Momentum-based extended Kalman filter for thrust estimation on flying multibody robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 1, pp. 526–533, 2021, doi: 10.1109/LRA.2021.3129258.

[60] M. Kiew-ong-art *et al.*, "Comparative study of Takagi-Sugeno-Kang and Madani algorithms in Type-1 and Interval Type-2 fuzzy control for self-balancing wheelchairs," *International Journal of Robotics and Control Systems*, vol. 3, no. 4, pp. 643–657, 2023, doi: 10.31763/ijrcs.v3i4.1154.

[61] P. Chotikunnan *et al*., "Comparative Analysis of Sensor Fusion for Angle Estimation Using Kalman and Complementary Filters," *International Journal of Robotics and Control Systems*, vol. 5, no. 1, pp. 1–21, 2024, doi: 10.31763/ijrcs.v5i1.1674.

[62] Y. Adesida, E. Papi, and A. H. McGregor, "Exploring the role of wearable technology in sport kinematics and kinetics: A systematic review," *Sensors*, vol. 19, no. 7, p. 1597, 2019, doi: 10.3390/s19071597.

[63] R. D. Gurchiek, N. Cheney, and R. S. McGinnis, "Estimating biomechanical time-series with wearable sensors: A systematic review of machine learning techniques," *Sensors*, vol. 19, no. 23, p. 5227, 2019, doi: 10.3390/s19235227.

[64] D. Kobsar *et al*., "Validity and reliability of wearable inertial sensors in healthy adult walking: a systematic review and meta-analysis," *Journal of NeuroEngineering and Rehabilitation*, vol. 17, p. 62, 2020, doi: 10.1186/s12984-020-00685-3.

[65] G. Wu *et al*., "ISB recommendation on definitions of joint coordinate system of various joints for the reporting of human joint motion—part I: ankle, hip, and spine," *Journal of Biomechanics*, vol. 35, no. 4, pp. 543–548, 2002, doi: 10.1016/S0021-9290(01)00222-6.

[66] R. V. Vitali and N. C. Perkins, "Determining anatomical frames via inertial motion capture: A survey of methods," *Journal of Biomechanics*, vol. 106, p. 109832, 2020, doi: 10.1016/j.jbiomech.2020.109832.

[67] M. Caruso *et al*., "Analysis of the accuracy of ten algorithms for orientation estimation using inertial and magnetic sensing under optimal conditions: One size does not fit all," *Sensors*, vol. 21, no. 7, p. 2543, 2021, doi: 10.3390/s21072543.

[68] E. Palermo, S. Rossi, F. Marini, F. Patanè, and P. Cappa, "Experimental evaluation of accuracy and repeatability of a novel body-to-sensor calibration procedure for inertial sensor-based gait analysis," *Measurement*, vol. 52, pp. 145–155, 2014, doi: 10.1016/j.measurement.2014.03.004.

[69] X. Robert-Lachaine, H. Mecheri, C. Larue, and A. Plamondon, "Accuracy and repeatability of single-pose calibration of inertial measurement units for whole-body motion analysis," *Gait & Posture*, vol. 54, pp. 80–86, 2017, doi: 10.1016/j.gaitpost.2017.02.029.

[70] A. Ancillao, S. Tedesco, J. Barton, and B. O'Flynn, "Indirect measurement of ground reaction forces and moments by means of wearable inertial sensors: A systematic review," *Sensors*, vol. 18, no. 8, p. 2564, 2018, doi: 10.3390/s18082564.

[71] F. J. Wouda *et al*., "Estimation of vertical ground reaction forces and sagittal knee kinematics during running using three inertial sensors," *Frontiers in Physiology*, vol. 9, p. 218, 2018, doi: 10.3389/fphys.2018.00218.

[72] E. Dorschky, M. Nitschke, C. F. Martindale, A. J. Van den Bogert, A. D. Koelewijn, and B. M. Eskofier, "CNN-based estimation of sagittal plane walking and running biomechanics from measured and simulated inertial sensor data," *Frontiers in Bioengineering and Biotechnology*, vol. 8, p. 604, 2020, doi: 10.3389/fbioe.2020.00604.

[73] H. Lim, B. Kim, and S. Park, "Prediction of lower limb kinetics and kinematics during walking by a single IMU on the lower back using machine learning," *Sensors*, vol. 20, no. 1, p. 130, 2019, doi: 10.3390/s20010130.

[74] B. J. Stetter, F. C. Krafft, S. Ringhof, T. Stein, and S. Sell, "A machine learning and wearable sensor based approach to estimate external knee flexion and adduction moments during various locomotion tasks," *Frontiers in Bioengineering and Biotechnology*, vol. 8, p. 9, 2020, doi: 10.3389/fbioe.2020.00009.

[75] M. Mundt *et al*., "Estimation of gait mechanics based on simulated and measured IMU data using an artificial neural network," *Frontiers in Bioengineering and Biotechnology*, vol. 8, p. 41, 2020, doi: 10.3389/fbioe.2020.00041.

[76] E. Rapp, S. Shin, W. Thomsen, R. Ferber, and E. Halilaj, "Estimation of kinematics from inertial measurement units using a combined deep learning and optimization framework," *Journal of Biomechanics*, vol. 116, p. 110229, 2021, doi: 10.1016/j.jbiomech.2021.110229.

[77] L. Chen *et al*., "AI-driven deep learning techniques in protein structure prediction," *International Journal of Molecular Sciences*, vol. 25, no. 15, p. 8426, 2024, doi: 10.3390/ijms25158426.

[78] M. Mundt *et al*., "Prediction of lower limb joint angles and moments during gait using artificial neural networks," *Medical & Biological Engineering & Computing*, vol. 58, pp. 211–225, 2020, doi: 10.1007/s11517-019-02061-3.

[79] W. R. Johnson, J. Alderson, D. Lloyd, and A. Mian, "Predicting athlete ground reaction forces and moments from spatio-temporal driven CNN models," *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 3, pp. 689–694, 2019, doi: 10.1109/TBME.2018.2854632.

[80] W. R. Johnson, A. Mian, M. A. Robinson, J. Verheul, D. G. Lloyd, and J. A. Alderson, "Multidimensional ground reaction forces and moments from wearable sensor accelerations via deep learning," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 1, pp. 289–297, 2021, doi: 10.1109/TBME.2020.3006158.

[81] M. Mundt, A. Koeppe, F. Bamer, S. David, and B. Markert, "Artificial neural networks in motion analysis—applications of unsupervised and heuristic feature selection techniques," *Sensors*, vol. 20, no. 16, p. 4581, 2020, doi: 10.3390/s20164581.