Received 1 August 2023, accepted 13 August 2023, date of publication 17 August 2023, date of current version 25 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3305990

RESEARCH ARTICLE

A Parallel Recurrent Neural Network for Robust Inertial and Magnetic Sensor-Based 3D Orientation Estimation

JI SEOK CHOI[1], CHANG JUNE LEE[2], AND JUNG KEUN LEE[1], (Member, IEEE)

[1]School of ICT, Robotics and Mechanical Engineering, Hankyong National University, Anseong 17579, South Korea

[2]Department of Integrated Systems Engineering, Hankyong National University, Anseong 17579, South Korea

Corresponding author: Jung Keun Lee (jklee@hknu.ac.kr)

ABSTRACT The precise orientation estimation of moving objects in 3D space is crucial for the inertial and magnetic measurement unit (IMMU)-based motion capture applications. Disturbance components such as external acceleration and magnetic disturbance in the sensor signal deteriorate the estimation accuracy. While conventional filters such as Kalman filters and complementary filters successfully deal with these issues, there are still much room to improve the estimation performance. One alternative approach involves training an end-to-end neural network (NN) using raw IMMU datasets based on ground truth measurements. In this study, we propose an end-to-end NN to estimate a 3D orientation over time without an additional conventional filter. The architecture of the proposed NN comprises two separate recurrent NNs in a parallel configuration. The proposed parallel network can independently estimate two vectors, corresponding to the attitude and heading, and then combine the two vectors to form a direction cosine matrix (DCM) that represents the 3D orientation. The proposed DCM-based NN has been experimentally verified under various disturbance conditions. The orientation estimation accuracy of the proposed method was superior to those of the conventional filters as well as that of the quaternion-based NN.

INDEX TERMS Direction cosine matrix, inertial and magnetic sensor, recurrent neural network, 3D orientation estimation.

I. INTRODUCTION

The inertial and magnetic measurement unit (IMMU), which comprises an accelerometer, gyroscope, and magnetometer has been increasingly applied as a motion sensor owing to the development of the microelectromechanical systems (MEMS) technology. It has been implemented in various applications such as robotics [1], [2], unmanned aerial vehicles [3], [4], [5], rehabilitation, and sports application [6], [7], [8], [9], where motion tracking and analysis are required. 3D orientation estimation using an IMMU is especially crucial in inertial motion capture systems. The IMMU-based orientation estimation technique has been applied in various fields since it presents the advantage of measuring the 3D orientation of an object in real time without space restrictions by attaching a sensor to the object.

The gyroscope provides 3D orientation in the prediction process through strapdown integration, and the accelerometer and magnetometer provide the attitude and heading reference through gravity acceleration and the geomagnetic field, respectively, to correct any drift errors due to integration. However, the accelerometer output is the sum of the gravitational acceleration and external acceleration, and the magnetometer output is the sum of the geomagnetic field and magnetic disturbance. These two components of each sensor output cannot be distinguished without orientation, which is the primary goal of estimation. Therefore, disturbance components such as external acceleration and magnetic disturbance are among the main factors that deteriorate the accuracy of orientation estimation.

Several methods have been developed in previous studies to overcome this problem. Most of these methods are rule-based sensor fusion algorithms, such as the Kalman filter (KF) and complementary filter (CF) [10], [11], [12], [13], [14]. To minimize the errors caused

The associate editor coordinating the review of this manuscript and approving it for publication was Henry Hess.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

by disturbance components, various compensation mechanisms have been proposed for filter-based estimation, such as Markov chain-based disturbance component modeling [10], [12], gradient descent method [13], and threshold setting [14]. However, the disturbance conditions still cause significant errors because the exact orientation is unknown.

Artificial neural networks have achieved great success in natural language processing, computer vision, and speech recognition owing to the rapid development of AI research and computer hardware [15], [16], [17], [18], [19]. Consequently, an end-to-end neural network can be trained using various raw IMMU datasets and ground-truth orientations to achieve accurate orientation estimation.

Until recently, neural networks were mostly used to improve the performance of conventional filters. In [20], a recurrent neural network (RNN) was used for motion state classification to select the appropriate filter for the current state. In [21] and [22], a feedforward neural network (FFNN) was used to correct the orientation estimated by using a filter algorithm. In [23] and [24], the RNN and convolutional neural network (CNN) were utilized to denoise the gyroscope signal, which was used as the input of the strapdown integration.

In [25], [26], and [27], neural networks were used to estimate the orientation through end-to-end training. Sun et al. used an RNN for orientation integration where the input vector was a nine-axis IMMU signal [25]. However, the outputs of the orientation estimation network were updated using an extended KF to ensure long-term stability. Similar to [25], Esfahani et al. used an RNN for orientation integration [26], instead of using only a gyroscope signal as the input vector for the network. Additionally, this method uses a genetic algorithm (GA) to correct the noise and bias of the gyroscope. That is, the methods developed in [25] and [26] require additional algorithms for long-term orientation estimation. Weber et al. [27] designed an end-to-end RNN to directly provide the orientation from the raw IMMU data without using additional algorithms. However, the method [27] was applied only for the attitude estimation without heading estimation. although the method is based on a unit quaternion capable of representing a 3D orientation. In this regard, to the best of our knowledge, there is no end-to-end network for 3D orientation estimation without using an additional algorithm.

In general, magnetic disturbance issue related to the heading accuracy is considered more problematic than the external acceleration issue related to the attitude accuracy. In order to secure practicality of an orientation estimation method, its heading estimation performance must be verified.

Therefore, this paper proposes an end-to-end neural network to estimate 3D orientation over time without using an additional conventional filter while providing both attitude and heading. The architecture of the proposed neural network comprises two separate RNNs in a parallel configuration, based on the filter algorithms that estimate attitude and heading in parallel [10], [12]. The proposed parallel network outputs two vectors which correspond to the attitude and heading. Subsequently, the two vectors are combined to form a direction cosine matrix (DCM) that represents a 3D orientation. When the quaternion is used as the orientation representation, each disturbance factor can affect the other due to the attitude-heading coupling issue. However, the proposed method constructs separate and independent network (i.e., attitude network and heading network), fundamentally preventing each disturbance from affecting the other side [14].

The proposed network architecture can minimize the attitude and heading errors through the training process and can estimate the two components separately. Lastly, the performance of the proposed neural network was determined based on a 3D orientation estimation task. It was observed that the performance was similar or even better than a neural network based on conventional filters under various disturbance conditions.

The remainder of this paper is organized as follows. In Section II, we introduce DCM-based orientation estimation, the architecture of the network for 3D orientation estimation, and the training algorithm. In Section III, the experimental process for acquiring training and test data, and data augmentation techniques applied to the training data are explained. In Section IV, we analyzed the performance of neural network according to the size of the network and compared the performance of the proposed network with those of the conventional filters and quaternion-based network.

II. METHODOLOGY

A. DCM-BASED ORIENTATION ESTIMATION

Consider an IMMU with a sensor coordinate system, S, which is firmly attached to the object of interest for a motion tracking application. For every motion performed by the object, we attempted to estimate the orientation of the sensor. The IMMU-based 3D orientation is expressed as the relative orientation between the sensor coordinate system frame, S, and inertial coordinate system, I. The vertical axis (z-axis) of inertial frame I is aligned with the Earth's gravitational axis, and the x-axis of I is aligned with the Earth's local magnetic field vector corrected for deep angle (also known as magnetic inclination [28]), and the y-axis of I is determined as the cross product of the z-axis and the x-axis. The relative orientation of frame, S, corresponding to frame, I, can be expressed through a DCM, ${}^{I}\mathbf{R}_{S}$, which contains three unit and orthogonal vectors, as follows:

$$ {}^{I}\mathbf{R}_{S} = \left[{}^{S}\mathbf{X}_{I} \quad {}^{S}\mathbf{Y}_{I} \quad {}^{S}\mathbf{Z}_{I} \right]^{T} \tag{1} $$

The nine-axis IMMU utilizes two unit vectors for the 3D orientation estimation. The first is ${}^{S}\mathbf{Z}_{I}$, which contains the attitude (i.e., roll and pitch) information, and the second is ${}^{S}\mathbf{X}_{I'}$, which contains the heading (i.e., yaw) information. The frame, I', is tilted by the dip angle from inertial frame, I. Following the estimation of the two reference vectors, the three unit vectors of the DCM can be determined based on the orthogonality between the unit vectors (${}^{S}\mathbf{Y}_{I} = {}^{S}\mathbf{Z}_{I} \times {}^{S}\mathbf{X}_{I'}$ and ${}^{S}\mathbf{X}_{I} = {}^{S}\mathbf{Y}_{I} \times {}^{S}\mathbf{Z}_{I}$).

In the proposed method, two parallel neural networks are used to estimate ${}^{S}\mathbf{Z}_{I}$ and ${}^{S}\mathbf{X}_{I}$. Therefore, the estimation accuracy of the attitude and heading can be evaluated separately. If both the vector estimated by the orientation estimation algorithm and the ground-truth vector at an arbitrary time instance, t, are known, the angle between the two vectors, $\theta_{err}(t)$, can be expressed using the definition of the dot product as follows:

$$\theta_{err}(t) = \arccos(\hat{\mathbf{v}}(t) \bullet \mathbf{v}_{gt}(t)) \tag{2}$$

here, $\hat{\mathbf{v}}(t)$ represents the estimated unit vector and $\mathbf{v}_{gt}(t)$ represents the ground truth unit vector. By applying this to ${}^{S}\hat{\mathbf{Z}}_{I}$ and ${}^{S}\hat{\mathbf{X}}_{I}$, which are estimated using the parallel RNNs, the angle between the vector estimated by the neural network and the ground truth vector can be used as an error term of the loss function for the attitude and heading neural networks.

B. NEURAL NETWORK ARCHITECTURE

The proposed method utilizes two parallel neural networks to estimate the 3D orientation. Two neural networks estimate two three-dimensional unit vectors, ${}^{S}\mathbf{Z}_{I}$ and ${}^{S}\mathbf{X}_{I}$, that represent the attitude and heading, respectively. These two vectors form $\mathbf{R}$ based on the interaxial orthogonality of DCM.

We employ an RNN for the 3D orientation estimation. An RNN unit has repeated connections that store the state information which is used as the input for the current time step along a temporal sequence. This recurrent structure is computationally efficient and can theoretically store an infinite amount of past state information. However, in practical conditions, the number of time steps required to store the state information is limited due to the gradient vanishing problem. Various RNN architectures have been developed using long short-term memory (LSTM) or the recently proposed gate recurrent unit (GRU) to overcome this problem [29], [30].

Fig. 1 depicts the architecture of the proposed neural network. In [27], it was demonstrated that two-layer RNN was effective for attitude estimation. Therefore, the first and second layers of each parallel RNN architecture were developed using the RNN units. The input vectors of the two parallel RNNs are a six-dimensional vector containing accelerometer and gyroscope signals to determine the attitude and a nine-dimensional vector containing accelerometer, gyroscope, and magnetometer signals to determine the heading, respectively. At each time instance, t, two-layer RNN units with N neurons per layer transform the six or nine-dimensional input vectors into N-dimensional state vectors. The N-dimensional state vector is transformed into a three-dimensional vector with a Euclidean norm of 1 to ensure that the output of the neural network is a unit vector. Therefore, we utilized a linear layer for dimensional reduction and normalized it to a unit vector. Two unit vectors, ${}^{S}\hat{\mathbf{Z}}_{I}$ and ${}^{S}\hat{\mathbf{X}}_{I}$, are estimated using two parallel RNNs, and the matrix, $\hat{\mathbf{R}}$, is formed based on the orthogonality of the DCM.

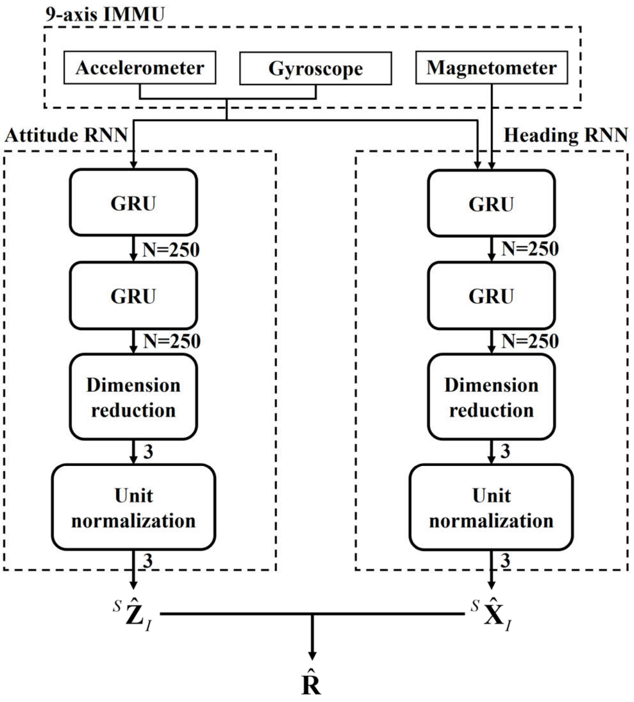

FIGURE 1. Architecture of the proposed parallel neural network for 3D orientation estimation.

A performance evaluation was performed based on two different RNN unit types (GRU and LSTM) and the number of neurons per layer. Consequently, the architecture of the proposed RNN was determined as two-layer GRU with 250 neurons per layer, as shown in Fig. 1. The network analysis is described in Section. VI.A.

In the heading estimation, the estimation performance is significantly reduced when only the gyroscope and magnetometer signals are used as inputs. In contrast, the performance is improved when all the nine-axis signals are used together. Conventional filter-based methods estimate the Earth's local magnetic field vector, ${}^{S}\mathbf{X}_{I'}$, including the dip angle, and then determine ${}^{S}\mathbf{X}_{I}$, which is corrected for dip angle through orthogonality using the attitude vector, ${}^{S}\mathbf{Z}_{I}$. Conversely, the proposed neural network is trained using the raw IMMU dataset with ground-truth orientation to estimate the orientation. The proposed heading RNN utilizes the ground truth vector, ${}^{S}\mathbf{X}_{I,gt}$, as the target data, i.e., the proposed neural network directly estimates ${}^{S}\mathbf{X}_{I}$. To correct the dip angle, ${}^{S}\mathbf{X}_{I'}$ must be projected onto a plane perpendicular to the attitude vector. Therefore, the accelerometer signal including the attitude information must be used together to directly estimate the heading vector when using the proposed RNN.

C. TRAINING PROCESS

The neural network updated its weights during the training process to minimize the error between the unit vector output to the neural network and the ground truth unit vector.

The proposed neural network has two parallel RNNs which estimate the heading and attitude. Therefore, both the error components can be individually minimized by defining each loss function.

The loss function for the attitude neural network is the mean square error (MSE) for a sequence of M samples starting at a sampling instant, k, where the error component is the angle, $\theta_{att}(t)$, between the estimated attitude vector, ${}^{S}\hat{\mathbf{Z}}_{I}(t)$, and ground truth attitude vector, ${}^{S}\mathbf{Z}_{I,\,gt}(t)$, as defined in (2). Similarly, the loss function for the heading neural network is MSE, where the error component is the angle, $\theta_{head}(t)$, between ${}^{S}\hat{\mathbf{X}}_{I}(t)$ and ${}^{S}\mathbf{X}_{I,\,gt}(t)$. Therefore, each loss function is expressed as:

$$\mathrm{MSE}_{att} = \frac{1}{M}\sum_{t=k}^{M+k-1}\theta_{att}(t)^2 \tag{3}$$

$$\mathrm{MSE}_{head} = \frac{1}{M}\sum_{t=k}^{M+k-1}\theta_{head}(t)^2 \tag{4}$$

However, the argument of the arccos function, x, converges to a target value of 1 to minimize the loss function during the training process. Thus, the gradient of the loss function diverges to infinity, presenting a gradient explosion problem [27].

$$\lim_{x\to 1}\frac{d\ \mathrm{arccos}}{d\,x} = \frac{-1}{\sqrt{1-x^2}} = -\infty \tag{5}$$

We avoided such a numerical problem by utilizing the method proposed in [27], which involves increasing the floating-point precision to calculate $\theta_{err}(t)$ to 64 bits and truncating the values which are too close to 1.

The RNNs are trained using limited time steps in each mini-batch to avoid the inherent gradient vanishing or exploding problems. To handle this problem, we used a truncated backpropagation through time algorithm. This approach divides the long sequence into concatenations of short sequences and sequentially passes the last hidden state between each mini-batch without zero initialization. Thus, the RNN can process longer sequences during the training process. The input vectors were standardized to a standard normal distribution to improve the training stability and remove the scale-related bias of the input signals. A state-of-the-art optimizer called Ranger, which exhibited good performance in several tasks, was used in the training process [31]. This optimizer is a combination of the RAdam and Lookahead optimizers. We use the one-cycle learning rate policy [32] to improve the convergence speed and generalization performance. Furthermore, the optimal maximum learning rate was determined using the learning rate finder [33].

The neural network was implemented and trained in the Google Colab environment using the Fastai v2 API based on PyTorch (v. 1.11.0). For fast training process of the deep learning models NVIDIA Tesla P100 PCIe 16GB GPU were used in Google Colab environment.

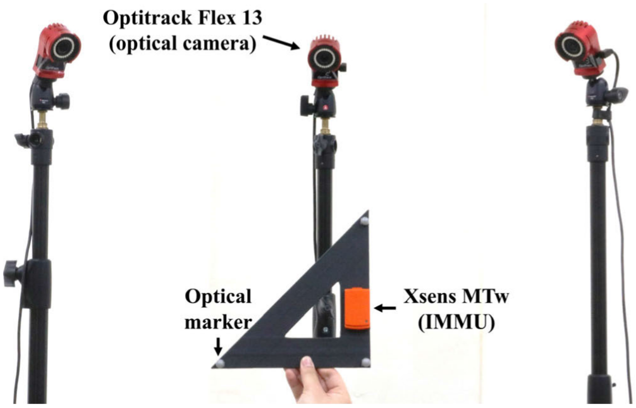

FIGURE 2. Experimental setup.

III. EXPERIMENTS AND DATA PROCESSING

A. EXPERIMENTS

The IMMU module MTw (Xsens Technologies B.V., Enschede, The Netherlands) and Optitrack Flex 13 optical motion capture system (NaturalPoint, Inc., Corvallis, OR, USA) were used to obtain various raw IMMU datasets with ground truth measurements for the training and validation of the neural network. The sampling rate of both the systems was set to 100 Hz. The IMMU was rigidly attached to a triangular ruler using three optical markers which provide a ground-truth orientation for the triangular ruler (see Fig. 2). Any misalignment between the local frame of the sensor and triangular ruler can significantly reduce the training performance of the neural network and the estimation accuracy of the filter algorithm since it presents an incorrect ground truth orientation. A quaternion-based local frame alignment method [34] was used to align the two coordinate systems.

All the experiments were conducted for approximately three minutes including an initial static period of 20 s. The sensor may not be able to cover various movements if only one person is conducting the experiment because humans unconsciously move in similar patterns. Therefore, three participants were employed in the experiment to prevent the sensor from moving in a fixed pattern. The experimental dataset comprised 202 trial sequences. A dataset with a wide spectrum of disturbance characteristics is required for the neural network to achieve a robust estimation performance. The trial sequences are divided into two types of conditions: a dynamic condition which limits the movement, and a magnetic disturbance condition which applies magnetic disturbance to the sensor. Dynamic condition sequences are divided into three types of movements: (1) rotation sequences which perform rotation while maintaining the same position, (2) translation sequences which perform translation while maintaining almost the same orientation, and (3) combined sequences which perform both rotation and translation. The magnetic disturbance sequences are divided into three types: (1) static magnetic sequences which move an IMMU around a fixed magnetic material; (2) moving magnetic sequences which move a magnetic material around a moving IMMU; and (3) attached magnetic sequences which directly attach a magnetic material to a triangular ruler. In magnetic disturbance sequences, the IMMU performs combined movements. Additionally, to increase the diversity of the motion characteristics, some sequences include short break terms (20 s) during motion, which present some time to recover from the estimation errors.

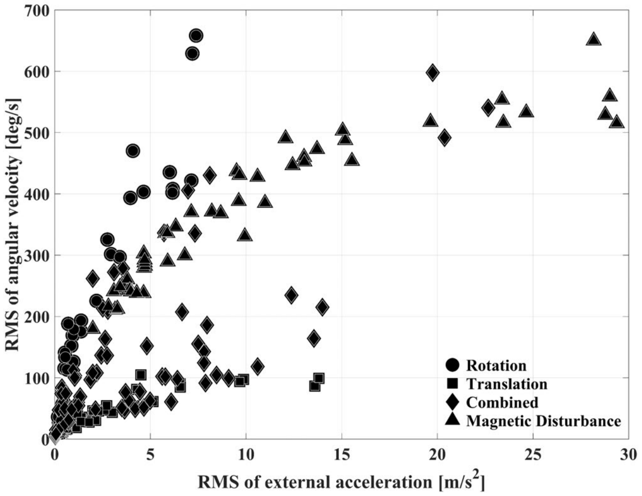

FIGURE 3. RMS distribution of external accelerations and angular velocities over all experimental sequences.

The experiments were conducted at various speeds for each condition to obtain a wide range of motion characteristics. Fig. 3 illustrates the RMS of the magnitude of the external acceleration and angular velocity for each trial sequence. A dataset with a wide dynamic range of external acceleration (up to 29 m/s^2) and angular velocity (up to 658 °/s) is obtained, as shown in the graph.

The dataset comprising 202 trial sequences was randomly divided into training and validation data for each condition for the training and validation of the neural networks. A total of 105 sequences from the dataset were used as the training dataset, and the remaining 97 were used as the test dataset for the neural network performance evaluation.

B. DATA AUGMENTATION

The generalization performance of a neural network can be improved by increasing the limited number of datasets through data augmentation based on specific domain knowledge. The effectiveness of this approach has been demonstrated in various fields, such as images, text, and audio data [35], [36], [37]. In this study, two data augmentation techniques were used to increase the size of the orientation data.

The first augmentation technique is rotation augmentation. this technique virtually rotates the 9-axis IMMU signals (i.e., ${}^{S}\mathbf{y}_{\mathrm{A}}$, ${}^{S}\mathbf{y}_{\mathrm{G}}$, and ${}^{S}\mathbf{y}_{\mathrm{M}}$), which are used as input vectors for the network, and the reference orientation ${}^{I}\mathbf{R}_{S,gt}$ used as the target value. For virtual rotation, we randomly generated an arbitrary constant DCM, ${}^{S'}\mathbf{R}_{S}$, which is the relative orientation of the actual sensor coordinate system S with respect to the virtual sensor coordinate system S'. The virtually rotated 9-axis sensor signals are ${}^{S'}\mathbf{y}_{\mathrm{A}}$, ${}^{S'}\mathbf{y}_{\mathrm{G}}$ and ${}^{S'}\mathbf{y}_{\mathrm{M}}$. And the virtually rotated ground truth orientation is ${}^{I}\mathbf{R}_{S',gt}$. This technique is equivalent to tilting the original sensor frame by an arbitrary constant angle. Therefore, the virtual data generated by rotational augmentation has different sensor movements from the original data.

The bias of the gyroscope signal is a major factor which deteriorates the orientation estimation performance based on drift error. Therefore, the second augmentation is performed to add a random bias to the gyroscope signal. An arbitrary constant bias value, $\mathbf{b}$, is generated from a normal distribution with a mean of 0 and standard deviation of 0.5 deg/s. It was then applied to the three-axis gyroscope signal (i.e., ${}^{S}\mathbf{y}_{\mathrm{G},biased} = {}^{S}\mathbf{y}_{\mathrm{G}} + \mathbf{b}$). Therefore, the orientation estimation of the neural network exhibits robustness against drift errors due to the gyroscope bias.

The two augmentation techniques were applied individually or in combination to the training data set. The number of training data increased by a factor of 6.

IV. RESULTS AND DISCUSSION

A. NETWORK ANALYSIS

In this section, we analyze the effect of the neural network size and type on the estimation accuracy and determine the optimal neural network architecture for 3D orientation estimation. To this end, each neural network architecture was developed using two-layer GRU or LSTM ranging from 10 to 300 neurons per layer. Each neural network was then trained five times using the training dataset, and its performance was evaluated over all the test dataset sequences.

TABLE 1. Estimation performance (Mean±STD of RMSEs) depending on the number of neurons (unit: °).

Number of neurons	GRU	LSTM
10	10.10±0.61	9.89±0.30
50	6.88±0.21	6.53±0.11
100	6.77±0.11	6.39±0.08
150	6.49±0.09	6.24±0.17
200	6.39±0.16	6.18±0.13
250	6.18±0.06	6.23±0.14
300	6.14±0.08	6.08±0.07

Table 1 shows the mean and standard deviation (STD) of the root mean square errors (RMSEs) depending on the two types of RNN (LSTM and GRU) and network size over all the test sequences. And table 2 shows the number of parameters depending on the two types of RNN and network size. Both results (table 1 and 2) were validated over all test sequences. Increasing the number of neurons improves the performance both types of RNN. Additionally, a decrease in

TABLE 2. The number of parameters depending on the number of neurons.

Number of neurons	GRU	LSTM
10	2,556	3,386
50	48,756	64,906
100	187,506	249,806
150	416,256	554,706
200	735,006	979,606
250	1,143,756	1,524,506
300	1,642,506	2,189,406

TABLE 3. Estimation performance (Mean±STD of RMSEs) (unit: °).

Slow		Attitude	Heading	Avg
Translation	M1	1.46±0.83	6.64±4.06	4.05
	M2	2.15±0.57	7.56±4.77	4.85
	M3	1.21±0.86	4.17±2.06	2.69
Rotation	M1	1.38±0.51	6.61±3.53	4.00
	M2	1.57±0.45	7.13±2.35	4.35
	M3	1.17±0.31	3.89±1.75	2.53
Combined	M1	2.02±1.26	6.29±4.97	4.16
	M2	2.19±1.30	4.80±3.89	3.50
	M3	1.47±0.69	4.10±2.15	2.79
Magnetic disturbance	M1	3.95±0.95	9.25±3.67	6.60
	M2	4.84±1.39	9.04±3.49	6.94
	M3	2.82±0.96	6.86±2.34	4.84
Fast		**Attitude**	**Heading**	**Avg**
Translation	M1	2.72±0.89	6.70±3.55	4.71
	M2	5.73±2.78	7.85±4.14	4.97
	M3	1.58±0.63	4.14±1.85	2.86
Rotation	M1	4.92±1.54	13.02±3.45	8.97
	M2	4.03±2.76	7.15±2.26	5.59
	M3	2.06±0.62	4.49±1.38	3.27
Combined	M1	4.54±3.13	7.94±5.17	6.24
	M2	4.68±2.14	4.56±3.14	4.62
	M3	2.24±1.64	4.95±1.76	3.60
Magnetic disturbance	M1	10.53±4.92	12.89±4.71	11.71
	M2	11.30±6.98	10.90±5.15	11.10
	M3	4.02±1.76	8.08±2.88	6.05

the STD indicates an improved estimation stability. However, the performance did not vary significantly when the number of neurons was greater than 250. Despite the similarity in the performance between the LSTM and GRU, the LSTM has approximately 33% more parameters than the GRU in the proposed neural network structure (see Table 2). The number of parameters increased exponentially with the size of the network, and the difference in the number of parameters between the two RNN types also increased. Therefore, the LSTM has a lower computational efficiency in real-time orientation estimation than the GRU. Consequently, the architecture of the parallel RNNs for 3D orientation estimation included two-layer GRU with 250 neurons per layer.

B. COMPARISON WITH CONVENTIONAL FILTERS

To verify the effectiveness of the proposed neural network, we compared the orientation estimation performance using two filter algorithms: a complementary filter (CF) [13] and a Kalman filter (KF) [12]. The parameter, β, in the CF was set to $\beta = 0.33$ and the parameters, c_a and c_b, in the KF were set to $c_a = 0.1$ and $c_b = 0.15$ for attitude and heading, respectively. The parameters of both the filters were tuned to obtain the best estimation performance. To analyze the effect of disturbance components on the orientation estimation, the performance was compared by dividing the conditions based on motion speed (slow and fast) and magnetic disturbance. The estimated $\hat{\mathbf{R}}$ is converted into a quaternion, $\hat{\mathbf{q}}$, to calculate the attitude and heading errors. Subsequently, the quaternion-based orientation evaluation method proposed in [38] was used.

Table 3 presents the mean and STD of the RMSEs of the attitude and heading for all the test dataset sequences. M1 corresponds to the KF, M2 corresponds to the CF, and M3 corresponds to the proposed neural network.

The estimation performance of the proposed neural network was superior under all conditions apart from one case (heading of combined fast). Additionally, a low STD indicates a robust estimation performance. M3 also outperformed the conventional filters in terms of robustness.

While moving from slow to fast conditions in non-magnetic disturbances, the attitude estimation accuracy tended to decrease with the increase in the external acceleration. In the slow condition, there was no significant difference in the attitude estimation performance. In the fast condition, the attitude accuracy of M1 and M2 significantly deteriorated by over 4° on average, but M3 exhibited an outstanding performance of less than 2.3°. This indicates that the proposed RNN is more effective in compensating for external acceleration than conventional filters. M3 outperformed the other two methods in terms of heading estimation under all conditions except for the combined fast condition.

Magnetic disturbance is a major factor which deteriorates the estimation performance. Therefore, all three methods exhibited an increase in the estimation error under the magnetic disturbance condition. Particularly, both the attitude and heading errors of M1 and M2 increased by more than 10 ° under fast conditions. Conversely, M3 exhibited the best estimation performance with an average RMSE of 6 °, even under severe conditions.

Fig. 4 presents the orientation estimation results of the three methods when a magnetic disturbance (the averaged magnitude of magnetic disturbance of 0.42 a.u. (arbitrary unit) with maximum of 1.46 a.u. is applied under the slow

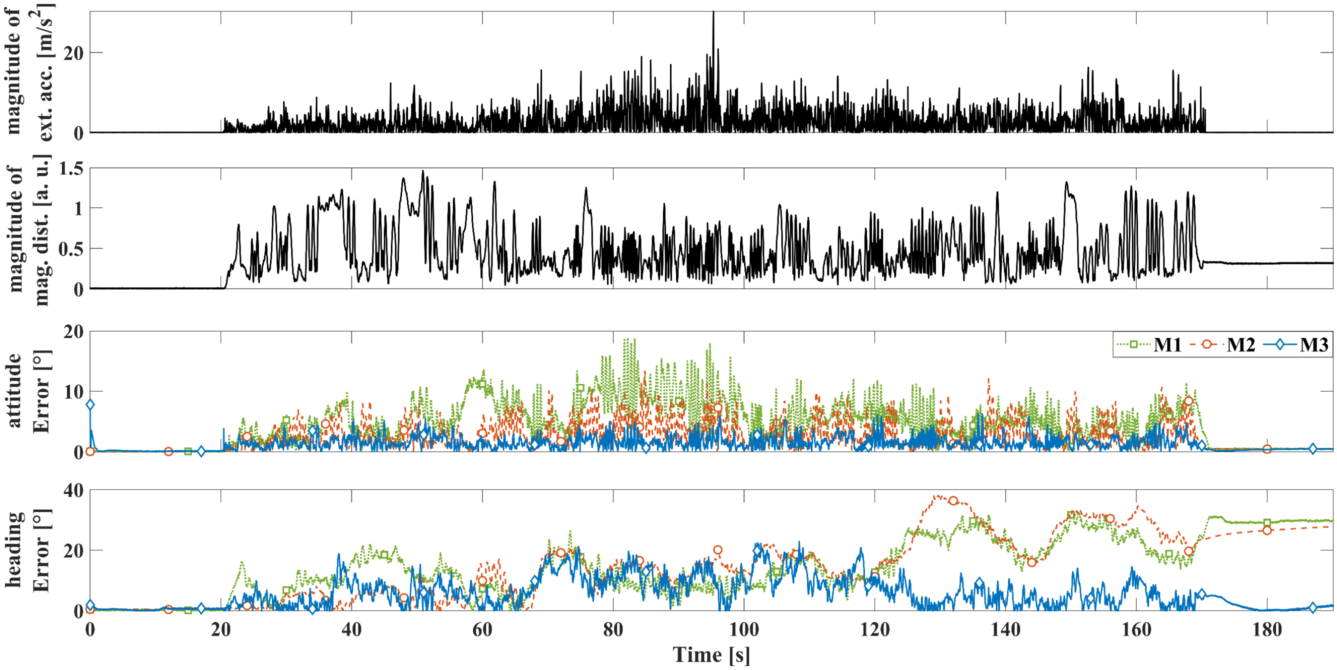

FIGURE 4. Example results of estimation errors in accelerated and magnetically disturbed conditions.

condition (the averaged magnitude of external acceleration of 2.33 m/s^2 with maximum of 74.4 m/s^2). Although the two filter algorithms have a compensation mechanism for disturbances, the estimation performance of the attitude and heading is severely deteriorated due to the continuous magnetic disturbance under dynamic conditions. The proposed neural network effectively reduced the estimation error caused by continuous disturbances. This indicates that the proposed neural network presents a better compensation effect than conventional filters due to training using the disturbance data. The proposed neural network rapidly converged to zero after a significant increase in the error at the beginning of the estimation, as shown in the graph. This is due to absence of an initial static assumption, such as a filter algorithm, to determine the initial orientation assuming that the external acceleration and magnetic disturbance are zero in the initial state from which the estimation starts.

Conventional filters, such as M1 and M2, use only the accelerometer and gyroscope signals to estimate the attitude. However, the above results demonstrate that when a magnetic disturbance is applied, both the heading estimation performance utilizing the magnetometer signal and the attitude estimation performance are deteriorated. Therefore, the magnetic disturbance affects the attitude estimation, indicating that a compensation mechanism is required for the magnetic disturbance, even when estimating only the attitude.

Although conventional filters also have a compensation mechanism for the disturbance components, M3 exhibited the best performance. This indicates that the compensation presented by a neural network using the proposed method for the disturbance component surpasses the compensation mechanism of conventional filters. Additionally, this indicates that conventional filters can be adequately replaced by a neural network for 3D orientation estimation tasks.

C. COMPARISON WITH NEURAL NETWORK

The proposed neural network can be trained to minimize the error of the two components since it can separately estimate the two ground truth vectors of DCM, i.e., the attitude and heading information. The effectiveness of the proposed method on the estimation performance was determined by comparing it with a neural network which simultaneously estimates 3D orientation. A unit quaternion, which is a four-dimensional vector mixed with the attitude and heading information, is another notation for expressing a 3D orientation. Therefore, we compared the performance with that of a quaternion-based neural network.

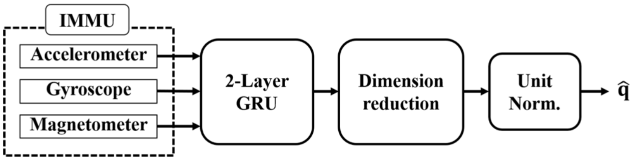

FIGURE 5. Architecture of the quaternion-based RNN for 3D orientation estimation.

Fig. 5 illustrates the architecture of a quaternion-based neural network. Through the network analysis, A neural network was developed with two-layer GRU with 250 neurons per layer. The output of the network is a unit quaternion $\hat{\mathbf{q}}$. The loss function is the MSE, and the error term of the loss function is set to the angle, $\theta_{quat}(t)$, between the estimated and ground-truth quaternions as follows:

$$\theta_{quat}(t) = 2\arccos\left|q_{err,w}(t)\right| \tag{6}$$

here, $q_{err,w}(t)$ represents the scalar part of the error quaternion, $\mathbf{q}_{err}(t)$, which can be obtained as $\mathbf{q}_{err}(t) = \hat{\mathbf{q}}(t) \otimes \mathbf{q}_{ref}(t)$ (here, $\otimes$ denotes the quaternion product). The training process and dataset were identical to those of the proposed neural network.

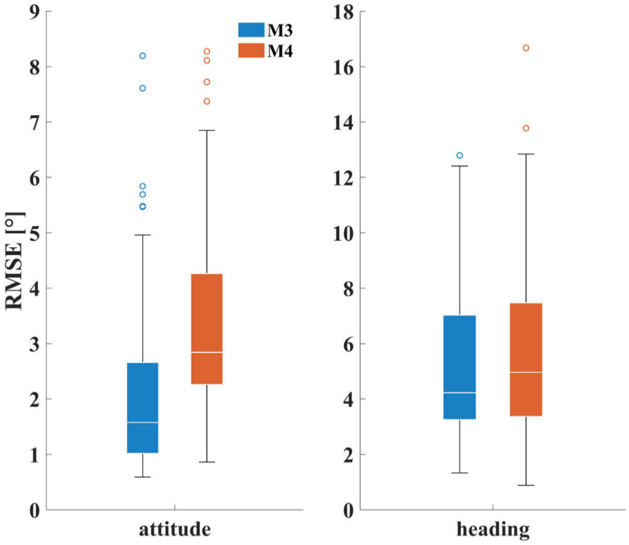

FIGURE 6. Comparison of estimation performance between the proposed RNN and the quaternion based RNN.

Fig. 6 presents the attitude and heading estimation performance of the two neural networks. M4 corresponds to a quaternion-based neural network. M3 presented better performance in terms of both the attitude and heading. Particularly, a significant difference was observed in the attitude estimation performance. M3 is trained to minimize the attitude and heading errors, while M4 is trained to minimize the 3D orientation errors in which the attitude and heading components are mixed. Therefore, the attitude and heading errors of M4 are observed to be larger than those of M3 because M4 cannot focus on one error component and update the parameter of the network. However, the difference in the heading was not significant. This indicates that training performed to minimize only the heading component does not significantly affect the reduction of the heading error.

V. CONCLUSION

In this study, we utilize an RNN for robust IMMU-based 3D orientation estimation to overcome the drawbacks of conventional filters. The proposed RNN 3D orientation estimation technique involves training two parallel RNNs using two different loss functions to improve the accuracy by estimating the attitude and heading. Each parallel RNN estimates two reference vectors of the DCM and then completes the DCM using orthogonality. The effectiveness of the proposed neural network was determined by analyzing the estimation performance was using a novel test dataset. The effectiveness of the proposed neural network was determined based on the performance of a 3D orientation estimation task. This performance was then compared to that of conventional filters and a quaternion-based neural network which simultaneously estimates 3D orientation under various disturbance conditions. The proposed neural network outperformed the conventional filter algorithms in terms of both attitude and heading under most conditions. The proposed parallel neural network also outperformed the quaternion-based neural network since it estimated the attitude and heading separately. Particularly, a significant improvement was observed in the attitude performance.

In future works, we aim to integrate an initial static assumption into a neural network. We also aim to incorporate neural networks into IMMU-based motion tracking and analysis applications.

REFERENCES

[1] V. Malyavej, W. Kumkeaw, and M. Aorpimai, "Indoor robot localization by RSSI/IMU sensor fusion," in *Proc. 10th Int. Conf. Electr. Eng./Electron., Comput., Telecommun. Inf. Technol.*, Krabi, Thailand, May 2013, pp. 1–6.

[2] M. Alatise and G. Hancke, "Pose estimation of a mobile robot based on fusion of IMU data and vision data using an extended Kalman filter," *Sensors*, vol. 17, no. 10, p. 2164, Sep. 2017, doi: 10.3390/s17102164.

[3] N. Koksal, M. Jalalmaab, and B. Fidan, "Adaptive linear quadratic attitude tracking control of a quadrotor UAV based on IMU sensor data fusion," *Sensors*, vol. 19, no. 1, p. 46, Dec. 2018, doi: 10.3390/s19010046.

[4] R. C. Avram, X. Zhang, J. Campbell, and J. Muse, "IMU sensor fault diagnosis and estimation for quadrotor UAVs," *IFAC-PapersOnLine*, vol. 48, no. 21, pp. 380–385, 2015, doi: 10.1016/j.ifacol.2015.09.556.

[5] Y. N. Bezkorovainyi and O. A. Sushchenko, "Improvement of UAV positioning by information of inertial sensors," in *Proc. IEEE 5th Int. Conf. Methods Syst. Navigat. Motion Control (MSNMC)*, Kiev, Ukraine, Oct.2018, pp. 123–126.

[6] C. Caramia, D. Torricelli, M. Schmid, A. Muñoz-Gonzalez, J. Gonzalez-Vargas, F. Grandas, and J. L. Pons, "IMU-based classification of Parkinson's disease from gait: A sensitivity analysis on sensor location and feature selection," *IEEE J. Biomed. Health Informat.*, vol. 22, no. 6, pp. 1765–1774, Nov. 2018, doi: 10.1109/JBHI.2018.2865218.

[7] T. Seel, J. Raisch, and T. Schauer, "IMU-based joint angle measurement for gait analysis," *Sensors*, vol. 14, no. 4, pp. 6891–6909, Apr. 2014, doi: 10.3390/s140406891.

[8] Y. C. Han, K. I. Wong, and I. Murray, "Gait phase detection for normal and abnormal gaits using IMU," *IEEE Sensors J.*, vol. 19, no. 9, pp. 3439–3448, May 2019, doi: 10.1109/JSEN.2019.2894143.

[9] O. Dehzangi, M. Taherisadr, and R. ChangalVala, "IMU-based gait recognition using convolutional neural networks and multi-sensor fusion," *Sensors*, vol. 17, no. 12, p. 2735, Nov. 2017, doi: 10.3390/s17122735.

[10] J. K. Lee, "A parallel attitude-heading Kalman filter without state-augmentation of model-based disturbance components," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 7, pp. 2668–2670, Jul. 2019, doi: 10.1109/TIM.2019.2906417.

[11] A. M. Sabatini, "Quaternion-based extended Kalman filter for determining orientation by inertial and magnetic sensing," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 7, pp. 1346–1356, Jul. 2006, doi: 10.1109/TBME.2006.875664.

[12] G. Ligorio and A. M. Sabatini, "A novel Kalman filter for human motion tracking with an inertial-based dynamic inclinometer," *IEEE Trans. Biomed. Eng.*, vol. 62, no. 8, pp. 2033–2043, Aug. 2015, doi: 10.1109/TBME.2015.2411431.

[13] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," in *Proc. IEEE Int. Conf. Rehabil. Robot.*, Jun. 2011, pp. 1–7.

[14] J. K. Lee and E. J. Park, "A fast quaternion-based orientation optimizer via virtual rotation for human motion tracking," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 5, pp. 1574–1582, May 2009, doi: 10.1109/TBME.2008.2001285.

[15] A. Galassi, M. Lippi, and P. Torroni, "Attention in natural language processing," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 10, pp. 4291–4308, Oct. 2021, doi: 10.1109/TNNLS.2020.3019893.

[16] T. Wolf et al., "Transformers: State-of-the-art natural language processing," in *Proc. Conf. Empirical Methods Natural Lang. Processing: Syst. Demonstrations*, 2020, pp. 38–45. [Online]. Available: https://www.aclweb.org/anthology/2020.emnlp-demos.6

[17] S. S. Yadav and S. M. Jadhav, "Deep convolutional neural network based medical image classification for disease diagnosis," *J. Big Data*, vol. 6, no. 1, pp. 1–18, Dec. 2019.

[18] X. Yang, Y. Ye, X. Li, R. Y. K. Lau, X. Zhang, and X. Huang, "Hyperspectral image classification with deep learning models," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 9, pp. 5408–5423, Sep. 2018, doi: 10.1109/TGRS.2018.2815613.

[19] Z. Zhang, J. Geiger, J. Pohjalainen, A. E.-D. Mousa, W. Jin, and B. Schuller, "Deep learning for environmentally robust speech recognition: An overview of recent developments," *ACM Trans. Intell. Syst. Technol.*, vol. 9, no. 5, pp. 1–28, Apr. 2018, doi: 10.1145/3178115.

[20] M. Brossard, A. Barrau, and S. Bonnabel, "RINS-W: Robust inertial navigation system on wheels," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Nov. 2019, pp. 2068–2075.

[21] K.-W. Chiang, H.-W. Chang, C.-Y. Li, and Y.-W. Huang, "An artificial neural network embedded position and orientation determination algorithm for low cost MEMS INS/GPS integrated sensors," *Sensors*, vol. 9, no. 4, pp. 2586–2610, Apr. 2009, doi: 10.3390/s90402586.

[22] M. K. Al-Sharman, Y. Zweiri, M. A. K. Jaradat, R. Al-Husari, D. Gan, and L. D. Seneviratne, "Deep-Learning-Based neural network training for state estimation enhancement: Application to attitude estimation," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 1, pp. 24–34, Jan. 2020, doi: 10.1109/TIM.2019.2895495.

[23] C. Jiang, S. Chen, Y. Chen, B. Zhang, Z. Feng, H. Zhou, and Y. Bo, "A MEMS IMU de-noising method using long short term memory recurrent neural networks (LSTM-RNN)," *Sensors*, vol. 18, no. 10, p. 3470, Oct. 2018, doi: 10.3390/s18103470.

[24] R. Li, C. Fu, W. Yi, and X. Yi, "Calib-Net: Calibrating the low-cost IMU via deep convolutional neural network," *Frontiers Robot. AI*, vol. 8, Jan. 2022, Art. no. 772583, doi: 10.3389/frobt.2021.772583.

[25] S. Sun, D. Melamed, and K. Kitani, "IDOL: Inertial deep orientation-estimation and localization," in *Proc. Conf. AAAI. Artif. Intell.*, May 2021, pp. 6128–6137.

[26] M. A. Esfahani, H. Wang, K. Wu, and S. Yuan, "OriNet: Robust 3-D orientation estimation with a single particular IMU," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 399–406, Apr. 2020, doi: 10.1109/LRA.2019.2959507.

[27] D. Weber, C. Gühmann, and T. Seel, "RIANN—A robust neural network outperforms attitude estimation filters," *AI*, vol. 2, no. 3, pp. 444–463, Sep. 2021, doi: 10.3390/ai2030028.

[28] D. Roetenberg, H. J. Luinge, C. T. M. Baten, and P. H. Veltink, "Compensation of magnetic disturbances improves inertial and magnetic sensing of human body segment orientation," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 13, no. 3, pp. 395–405, Sep. 2005, doi: 10.1109/TNSRE.2005.847353.

[29] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

[30] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," 2014, *arXiv:1406.1078*.

[31] L. Wright and N. Demeure, "Ranger21: A synergistic deep learning optimizer," 2021, *arXiv:2106.13731*.

[32] L. N. Smith and N. Topin, "Super-convergence: Very fast training of neural networks using large learning rates," *Proc. SPIE*, vol. 11006, May 2019, Art. no. 1100612.

[33] L. N. Smith, "Cyclical learning rates for training neural networks," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2017, pp. 464–472.

[34] J. K. Lee and W. C. Jung, "Quaternion-based local frame alignment between an inertial measurement unit and a motion capture system," *Sensors*, vol. 18, no. 11, p. 4003, Nov. 2018, doi: 10.3390/s18114003.

[35] C. Shorten, T. M. Khoshgoftaar, and B. Furht, "Text data augmentation for deep learning," *J. Big Data*, vol. 8, no. 1, pp. 1–34, Jul. 2021, doi: 10.1186/s40537-021-00492-0.

[36] L. Nanni, G. Maguolo, and M. Paci, "Data augmentation approaches for improving animal audio classification," *Ecol. Informat.*, vol. 57, May 2020, Art. no. 101084, doi: 10.1016/j.ecoinf.2020.101084.

[37] L. Nanni, M. Paci, S. Brahnam, and A. Lumini, "Comparison of different image data augmentation approaches," *J. Imag.*, vol. 7, no. 12, p. 254, Nov. 2021, doi: 10.3390/jimaging7120254.

[38] D. Laidig, M. Caruso, A. Cereatti, and T. Seel, "BROAD—A benchmark for robust inertial orientation estimation," *Data*, vol. 6, no. 7, p. 72, Jun. 2021, doi: 10.3390/data6070072.

JI SEOK CHOI received the B.S. and M.S. degrees in mechanical engineering from the School of ICT, Robotics and Mechanical Engineering, Hankyong National University, Anseong, South Korea, in 2021 and 2023, respectively.

His research interests include wearable inertial sensing-based human motion tracking and data-driven motion tracking as well as biomechatronics.

CHANG JUNE LEE received the B.S. and M.S. degrees in mechanical engineering from Hankyong National University, Anseong, South Korea, in 2019 and 2022, respectively, where he is currently pursuing the Ph.D. degree with the Department of Integrated Systems Engineering.

His research interests include inertial sensing-based human motion tracking and joint torque estimation as well as wearable robotics.

JUNG KEUN LEE (Member, IEEE) received the B.S. and M.S. degrees in mechanical engineering from Hanyang University, Seoul, South Korea, in 1997 and 1999, respectively, and the Ph.D. degree in mechatronic systems engineering from the School of Engineering Science, Simon Fraser University (SFU), Surrey, BC, Canada, in 2010.

Prior to the Ph.D. study, he was a Naval Officer with the Republic of Korea Navy and then was a Research Engineer with the Powertrain Research and Development Centre, Hyundai Motor Company. After the Ph.D. study, he was a Postdoctoral Fellow with the School of Engineering Science, SFU. Since 2012, he has been with Hankyong National University, Anseong, South Korea, where he is currently a Professor with the School of ICT, Robotics and Mechanical Engineering. His research interests include inertial sensing-based human motion tracking, biomechatronics, wearable sensing and robotics, and data-driven estimation.