Metrol. Meas. Syst., Vol. 31 (2024) No. 1, pp. 195–211
DOI: 10.24425/mms.2023.148542

PAN POLSKA AKADEMIA NAUK

METROLOGY AND MEASUREMENT SYSTEMS
Index 330930, ISSN 0860-8229
www.metrology.wat.edu.pl

M&MS

ATTITUDE ESTIMATION BASED ON MULTI-SCALE GROUPED SPATIO-TEMPORAL ATTENTION NEURAL NETWORKS

Hailong Rong, Xiaohui Wu, Hao Wang, Tianlei Jin, Ling Zou

Changzhou University, Changzhou 213164, China (✉ *rhle_16@163.com*)

Abstract

In recent years, due to the proliferation of inertial measurement units (IMUs) in mobile devices such as smartphones, attitude estimation using inertial and magnetic sensors has been the subject of considerable research. Traditional methods involve probabilistic and iterative state estimation; however, these approaches do not generalize well over continuously changing motion dynamics and environmental conditions. Therefore, this paper proposes a deep learning-based approach for attitude estimation. This approach segments data from sensors into different windows and estimates attitude by separately extracting local features and global features from sensor data using a residual network (ResNet18) and a *long short-term memory network* (LSTM). To improve the accuracy of attitude estimation, a multi-scale attention mechanism is designed within ResNet18 to capture finer temporal information in the sensor data. The experimental results indicate that the accuracy of attitude estimation using this method surpasses that of other methods proposed in recent years.

Keywords: MEMS, attitude estimation, deep learning, attention mechanism.

1. Introduction

As *micro-electro-mechanical systems* (MEMS) continue to advance, inertial sensors have also gained the benefits of being smaller in size, more cost-effective, and having lower power consumption. This makes inertial sensors ubiquitous in modern life and work, such as inertial navigation [1], autonomous driving [2], virtual reality [3], UAV attitude tracking [4] and human activity recognition [5].

An *inertial and magnetic measurement unit* (IMMU) is usually composed of a three-axis accelerometer, a gyroscope, and a magnetometer. Observations of the direction vector can be obtained from the accelerometer and magnetometer, while the gyroscope provides the angular velocity. Attitude information is obtained by integrating the angular velocity measured by the gyroscope, but the gyroscope measurement contains some bias and noise [6, 7], which will cause some errors in the measured values relative to the actual values, and these errors accumulate

Article history: received October 24, 2023; revised November 27, 2023; accepted December 4, 2023; available online February 14, 2024.

over time, resulting in errors in the final attitude of the object. Since the direct use of gyroscope measurements produces poor estimates, traditional estimation techniques use acceleration and magnetometer measurements to update the error calculations and compensate for drift. Thus, the general problem of attitude estimation is to combine these sensors to provide the best solution in the form of an optimal estimator.

The complexity of attitude estimation arises mainly from its nonlinearity, and therefore the solution of attitude estimation must take into account the nonlinear dynamics of the system. Early applications relied on the *extended Kalman filter* (EKF) to linearize the current best state estimate of a dynamic system, however, this process can yield poor performance, especially in highly dynamic situations, due to divergence and constant reinitialization. Although Kalman filter-based estimators have been iteratively improved over the years, they still rely on system modelling assumptions, and deviations from defined assumptions may lead to divergence or failure of the system [8]. Some researchers have utilized the *wavelet denoising* (WD) technique [9] to decompose the original *inertial measurement unit* (IMU) signal and improve the accuracy of attitude estimation by removing the noise from the original signal. the WD method is effective in removing the high-frequency part, however, it has a limited ability to remove low-frequency errors. In 2015 Huang used *auto regressive and moving average* (ARMA) [10] to model gyroscope random noise to reduce the error, but the ARMA modelling method requires a large number of samples and converges slowly. Also, as the output of IMU is temporally correlated, it is of great importance to study output models with nonlinear and temporal information dependence. In recent years, *deep learning* (DL) has gained momentum and performed well in various applications such as image processing, *natural language processing* (NLP) [13] and sequential signal processing. Therefore, it has also been introduced into the field of inertial navigation systems. In 2018 Chen [25] *et al.* proposed to use supervised learning to segment inertial data into independent windows to solve the drift problem of inertial positioning techniques. Wang [26] *et al.* proposed an inertial odometer solution in 2021 to accomplish 2D position estimation by feeding cell phone IMU data into a neural network. This all proves that deep networks are capable of modelling the complex nonlinear relationship between the raw IMU sequence data and position as well as attitude.

For predictive processing of non-smooth signals in time series, *recurrent neural networks* (RNNs) seem to perform well. RNN is a network with a memory function that retains previous temporal information and uses it in the current output computation. Specifically, the nodes between the hidden layers are connected, and the inputs to the hidden layers include not only the outputs of the input layers, but also the outputs of the hidden layers from a previous time period [11]. Theoretically, RNN is able to deal with time series problems of arbitrary length and is good at learning patterns among samples with certain sequence significance [12]. In recent years, RNN techniques have also been used to solve the problems of signal noise reduction and error compensation in MEMS sensors. Li [15] *et al.* proposed a new method for real-time estimation and compensation of random drifts in MEMS gyroscopes by combining *unscented Kalman filter* (UKF) with RNN. The results of experimental studies show that the method is effective and superior. Although RNN is cost-effective in time-series signal processing, it is prone to the problems of gradient explosion and gradient vanishing due to less information memorized [16]. Therefore, *long short-term memory* (LSTM) and *gated recurrent unit* (GRU) are two improved RNN algorithms developed to solve these problems [17], [18]. Jiang [19] *et al.* used LSTM to denoise the output signals of a MEMS gyroscope, and the results show that this method can effectively improve the accuracy of the device. However, the model test only uses two minutes of static data from the gyroscope, which lacks certain robustness. The GRU method not only solves the gradient vanishing and gradient explosion problems of RNN, but also uses fewer parameters than the LSTM method. As a result, the training time of GRU is greatly reduced and it is suitable for

dealing with time series problems. Jiang [20] *et al.* proposed a mixture of GRU and LSTM is used for noise suppression of MEMS gyroscope. However, only static data are used for training and prediction in the paper, which is not sufficiently quantified and analysed from the perspective of dynamic experiments. Esfahani [21] *et al.* proposed a OriNet deep learning framework to reduce the effect of the bias and measurement noise present in the raw IMU data on attitude estimation, and to realize the estimation of the 3D attitude of the UAV based on a single IMU. Huang [27] *et al.* proposed a deep learning framework based on time-series convolution. The original feature information is preserved by adding residual blocks to the time-series convolution, while the error feature is obtained from the past gyroscope data, and the error-compensated gyroscope data is used for attitude estimation. Narkhede [28] *et al.* combined incremental learning with an LSTM network to estimate the attitude of objects in 3D space. This is done by feeding inertial sensor data into the LSTM network and then incrementally updating it to incorporate changes in motion dynamics that occur during operation. The approach of Brossard [29] *et al.* is built on a neural network based on dilated convolution, which utilizes ground truth data for noise reduction of the gyroscope and real-time estimation of the robot's attitude.

In addition, most methods in the literature use only acceleration and gyroscope sensor data and do not include geomagnetic data that can be obtained from a nine-axis IMMU. In order to achieve high accuracy attitude estimation from IMUs, geomagnetic data needs to be utilized to assist in 3D attitude estimation. Similarly, attention mechanisms are a hot research topic for improving the performance of neural network models, which can tell us where and what to pay attention to [14]. The importance of attention was first proposed in natural language processing [22] and then widely used and improved in various applications such as multivariate time series prediction [23] and image segmentation [24].

Therefore, in this paper, a *multiscale grouped spatio-temporal attention neural network* (MGTA) is designed to realize 3D attitude estimation from IMMU raw data. Inertial systems are characterized by high noise and time-varying additive bias. The aim of this neural network is to simultaneously discover complex interrelationships between different time indicators and different modal input signals. Our main contributions can be summarized as follows:

1. We use a hybrid modelling framework of ResNet18 and LSTM for attitude estimation in order to identify local features at different scales and hierarchies in the time series using convolution kernels of different sizes in ResNet18, while LSTM is used to capture long-term dependencies in the sequence data.
2. We introduce a novel temporal attention mechanism that enables the network to adaptively focus on the most significant signal modalities and crucial temporal information across different fields of views.

The paper is organized as follows. Section 2 introduces the error models for the IMU and magnetic sensors, and analyses how gyroscopes can be used for attitude estimation, and how accelerometers and magnetic sensors can be used for attitude observations to assist in attitude estimation. Section 3 describes the MGTA-based approach for IMMU attitude estimation. Sections 4 and 5 present the experimental platform as well as experimental results and analysis. Section 6 describes the conclusions of this paper.

2. Problem formulation

This section introduces the error model of IMU and magnetic sensors, shows how to use gyroscopes for attitude estimation, introduces how to use accelerometers and magnetometers for attitude observations, and analyses the impact of noise on 3D attitude.

2.1. Inertial and magnetic measurement error modelling

The general measurement model applicable to inertial sensors is illustrated in Figure 1. Both environmental factors and random errors affect the measurements of inertial sensors. In (1), (U_{XT}, U_{YT}, U_{ZT}) is the true three-axis angular velocity or acceleration, (U_{XM}, U_{YM}, U_{ZM}) represents the three-axis angular velocity or acceleration measured by the inertial sensor corresponding to the true value, and the measured value contains noisy (η_x, η_y, η_z), Bias (b_x, b_y, b_z), scale factor (S_x, S_y, S_z) and misalignment (E_{xy}, E_{yz}, E_{zx}) errors, as presented in [6].

$$\begin{bmatrix} U_{XM} \\ U_{YM} \\ U_{ZM} \end{bmatrix} = \begin{bmatrix} S_x & E_{xy} & E_{xz} \\ E_{yx} & S_y & E_{yz} \\ E_{zx} & E_{zy} & S_z \end{bmatrix} \begin{bmatrix} U_{XT} \\ U_{YT} \\ U_{ZT} \end{bmatrix} + \begin{bmatrix} b_x \\ b_y \\ b_z \end{bmatrix} + \begin{bmatrix} \eta_x \\ \eta_y \\ \eta_z \end{bmatrix}, \tag{1}$$

$$\delta\omega_t = B_g a_x + B_{ae} a_y a_z. \tag{2}$$

The four errors above are the main errors of inertial sensors. The angular velocity error is also affected by acceleration. Equation (2) represents the error caused by three-axis acceleration (a_x, a_y, a_z) on the X-axis of the gyroscope. B_g is g-dependent bias coefficient, B_{ae} is anisoelastic coefficient.

Magnetic force measurements by the magnetic sensor also contain many different categories of noise [34], as shown in (3), where m_T and m_M are the true values and measured values of the three-axis magnetic force respectively; $I_{3\times3}$ is an identity matrix of size 3×3; C is an error matrix combining scale factor, nonorthogonality and soft iron effects; b_m is the combined bias vector; η_m is the noise vector.

$$m_M = (I_{3\times3} + C)m_T + b_m + \eta_m \tag{3}$$

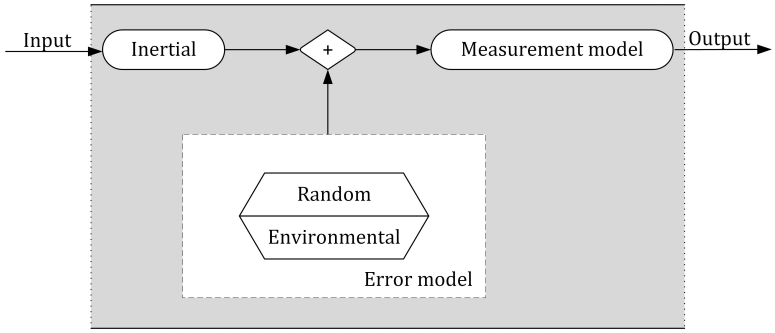


Fig. 1. Measurement models for inertial sensors.

2.2. Gyroscope-based attitude propagation

The IMU measures the angular velocity of the object and is used to estimate the attitude of the object. The angular increment is calculated from the angular velocity measured by the gyroscope, and then the attitude estimate is obtained through (4) and (5).

$$q_t = q_{t-1} + q_{t-1} \otimes (0 + \frac{1}{2}\omega_t dt), \tag{4}$$

$$\mathrm{q}_{t-1} \otimes \omega_t = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix}_t \begin{bmatrix} q_w \\ q_x \\ q_y \\ q_z \end{bmatrix}_{t-1}, \tag{5}$$

where $\otimes$ represents quaternion multiplication; $q_t = [q_w, q_x, q_y, q_z]_t \in R^4$ is the quaternion at time t representing the direction that maps the object coordinate system onto the navigation coordinate system; $\omega_t = [0, \omega_x, \omega_y, \omega_z]_t$ is a quaternion with a real part of zero, where ω_t represents the average angular velocity of the gyroscope within time interval dt at time t.

2.3. Accelerometer and magnetometer-based attitude observation

The attitude estimation obtained from the gyroscope will diverge over time, so attitude observations using accelerometers and magnetic sensors are necessary to make real-time corrections to the attitude estimation above. The three-axis accelerometer is used to measure three-axis acceleration $a_M = [a_x, a_y, a_z]$ of the object, the three-axis magnetometer is usually used to measure the three-axis magnetic force $m_M = [m_x, m_y, m_z]$ of the Earth's magnetic field, and the heading angle can be estimated through the magnetometer [33]. Therefore, attitude observation is achieved by combining a_M and m_M.

$$q_t^a = \begin{bmatrix} \cos\left(a\tan 2\left(a_y, a_z\right)/2\right) * \cos\left(a\tan 2\left(-a_x, \sqrt{a_y^2 + a_z^2}\right)/2\right) \\ \sin\left(a\tan 2\left(a_y, a_z\right)/2\right) * \cos\left(a\tan 2\left(-a_x, \sqrt{a_y^2 + a_z^2}\right)/2\right) \\ \cos\left(a\tan 2\left(a_y, a_z\right)/2\right) * \sin\left(\cos\left(a\tan 2\left(-a_x, \sqrt{a_y^2 + a_z^2}\right)/2\right)\right) \\ -\sin\left(a\tan 2\left(a_y, a_z\right)/2\right) * \sin\left(\cos\left(a\tan 2\left(-a_x, \sqrt{a_y^2 + a_z^2}\right)/2\right)\right) \end{bmatrix}, \tag{6}$$

$$M_t = q_t^a \otimes m_{Mt}. \tag{7}$$

At time t, the object three-axis magnetic force m_{Mt} is turned into the magnetic force $M_t = [M_x, M_y M_{,z}]_t$ in the horizontal object coordinate system by (7), where q_t^a is a quaternion calculated from three-axis acceleration $(a_x, a_y, a_z)_t$ at time t by (6). M_t will then be utilized to calculate the horizontally rotated quaternion q_t^m as shown in (8), and finally the attitude observation q_t^o will be computed by multiplying q_t^a and q_t^m by (9). q_t^o will be used to correct the attitude estimation to improve the accuracy of the attitude estimation.

$$q_t^m = \begin{bmatrix} \cos\left(a\tan 2\left(M_y, M_x\right)\right) \\ 0 \\ 0 \\ \sin\left(a\tan 2\left(M_y, M_x\right)\right) \end{bmatrix}, \tag{8}$$

$$q_t^o = q_t^a \otimes q_t^m. \tag{9}$$

2.4. Motivation for introduction of MGTA

From (1), it can be seen that ω_t contains a large amount of nonlinear noise. Similarly, as revealed by (2), acceleration will also impact the accuracy of ω_t. The presence of these noise and formula nonlinearities leads to a decrease in the accuracy of attitude estimation. The error

presented in q_{t-1} is propagated in q_t through (4). When we substitute (3) into (6)-(9), the impact of various noises contained in acceleration and magnetic force on attitude estimation is also nonlinear. Therefore, the existing filtering methods based on mathematical models are not optimal, so we propose to use a deep neural network MGTA for attitude estimation.

3. Methodology

In this article, we propose an MGTA model to extract the features locally and globally from the cascaded ResNet18 and LSTM layers. MGTA is inspired by the need to consider both the correlation between different modes measured by IMMUs in neighbouring timestamps and the global time dependence in the attitude estimation task, whereas existing ResNet18 or LSTM-based frameworks can only focus on one of these two factors. Another advantage of the MGTA model is that the model simultaneously takes into account the importance of relations between different timestamps and multimodal measurement domains by embedding a new attentional mechanism to refine the features extracted by the hybrid neural network.

In this section, we will detail the designed MGTA model in our attitude estimation system. We also will give a brief introduction to the whole system, including the pre-processing of the raw IMMU measurements and training details of the proposed model.

3.1. Data pre-processing

In the IMMU, a given sensor measurement is an IMMU signal over continuous time, which are denoted as $\mathbf{U} = \{\mathbf{u_1}, \mathbf{u_2}, \cdots\cdots, \mathbf{u_n}\}$, where $u_t = [a, \omega, m]_t \in R^9$ represents the measured three-axis acceleration a, three-axis magnetic force m and three-axis angular velocity ω in the object coordinate system at the time index t. In our attitude estimation task, the entire signal is partitioned into separate windows for generating the attitude. Specifically, each split window contains $\Delta = 2H$ samples, where H is the sampling rate. The entire signal processing pipeline can be summarized by the following equation:

$$\{u_{t+1-\Delta}, u_{t+2-\Delta}, \cdots\cdots, u_t\} \rightarrow q_t, \tag{10}$$

where q_t is the quaternion estimated from the time index $t + 1 - \Delta$ to the time index t. We assume that the q_t has no relation with the IMMU measurement samples before time instants $t + 1 - \Delta$. The raw IMMU measurements are first divided by a sliding window and then imported into the MGTA network as shown in Fig. 2.

The inputs and outputs of the neural network model are the normalized IMMU measurements $[\widetilde{a}, \widetilde{\omega}, \widetilde{m}]$ and the quaternion $\hat{q}_t$ representing the attitude, respectively.

3.2. Neural network for attitude estimation

The IMMU measurements are normalized and then transmitted to the MGTA model for further processing. The MGTA model details are shown in Fig. 3. The purpose of using hybrid neural networks in the proposed model is twofold. First, we wish to capture the local time-invariant features and global long-time dependence in normalized IMMU measurements $[\widetilde{a}, \widetilde{\omega}, \widetilde{m}]$. Another reason is that in practice it is difficult for LSTM frameworks to capture correlations in long time series data due to gradient and training instability [23]. ResNet18 is mainly composed of an initial layer and four residual layers (each residual layer contains two BasicBlocks). The initial layer contains a 1-D convolution, a *batch normalization* (BN) layer and a *max-pooling* (MaxPool) layer.

Metrol. Meas. Syst., Vol. 31 (2024), No. 1, pp. 195–211
DOI: 10.24425/mms.2023.148542

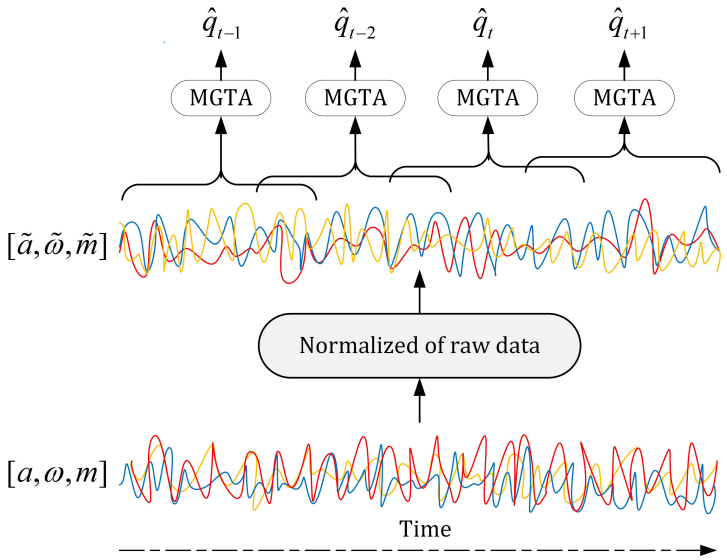


Fig. 2. Attitude estimation structure presented in this paper. This is a sliding window, sliding ten time indexes each time.

Thus, the ResNet18 layer preceding the LSTM layer not only captures local invariant features, but also refines the timestamps to make the long time series inputting to the LSTM layer more concise.

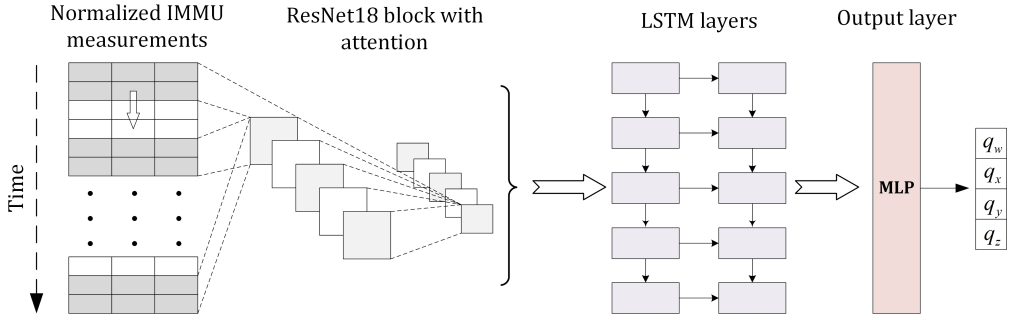


Fig. 3. Schematic representations of the proposed MGTA model. The entire MGTA model slides over the normalized IMMU measurements along the time axis and generates attitude estimates separately for each input window.

Specifically, the ResNet18 module takes a window of $[\widetilde{a}, \widetilde{\omega}, \widetilde{m}]$ as the input and first processes the signals by a cascaded 1-D convolutional and the BN layer, where the stride and padding of the convolution operation are 2 and 3, respectively. The following is an ReLU activation function and a MaxPool layer with the kernel size of 3. Thus, the refined time window now becomes 0.25Δ, *i.e.*, $0.5H$. The feature map F generated by the max-pooling layer is of size $64 \times 0.5H$ and can be summarized as

$$F = \text{MaxPool}\left(\text{ReLU}(\text{BN}(f_{64}^{7}([\widetilde{a}, \widetilde{\omega}, \widetilde{m}])))\right), \tag{11}$$

where f_{64}^{7} stands for the 1-D convolution with kernel size of 7, the output channel of 64, and no adding biases.

Next, we will use feature map F as input, and first pass it to the first BasicBlock of the first residual layer to get the intermediate feature map F_a. Then, pass F_a to the second BasicBlock to obtain the output of the first residual layer, F_b. Subsequently, using F_b as input, we will continue to repeat this process until we have traversed all the residual layers. In each BasicBlock, the initial features that come in are first passed through a convolutional network to receive a residual F_r. This residual is then combined with the initial features that are input to this BasicBlock to get the final feature F_a. In each BasicBlock, the filter slides over the feature maps and treats these feature maps constructed from each modal signal equally.

However, the extent to which these spatial features affect the final result should be different. Therefore, we fused the multi-scale convolutional attention blocks to distil the spatial features, which consist of the cascaded channel-type transform $G_c \in R^{C\times W}$, as shown in Fig. 4. The channel transform G_c is an attentional weight that recognizes the correlation between channels in the residual F_r obtained by the convolutional network. It emphasizes important channels and suppresses unnecessary channels to produce a channel-refined feature map A. The element-wise transformation G_e tries to identify the relations among all element features. The convolutional attention operation can be summarized as:

$$A = G_c(F_r) \times F_r, \tag{12}$$

$$F_a = A + F_{Id}, \tag{13}$$

where $\times$ is element-wise multiplication. Finally, the generated channel-refined feature map A is combined with the original feature F_{Id} inputted to the BasicBlock to obtain the final feature map F_a as shown in (13). After that, F_a will be used as input for the next BasicBlock until the entire ResNet18 is traversed and the result L is output.

Specifically, the channel attention module focuses more on the relationships between channels. Each channel can be considered as a separate feature detector. These features have different effects on attitude estimation. To efficiently compute channel attention, we use two ways to describe each channel. One way to characterize each channel is to use its average value. By pooling the average over the entire feature map, the network can obtain contextual and global information F_r^g. This helps to improve the perceptual capabilities of the model. Another approach is to use convolution to obtain local feature information F_r^l for each channel. Local features are computed to extract local structures and patterns from the input data to better understand the detailed information in the data. By combining these two types of information, the model can synthesize multiple levels of information to improve the understanding and characterization of the input data.

In addition, the fusion of global and local information can improve the robustness of the model. Global information helps the model to have better resistance to overall changes in the data, while local information can help the model to fight against noise in the local area. Group convolution is then employed to enable communication among channel features within each group while reducing the number of channels to decrease the model's parameters. Immediately after that, in order to realize inter-group feature communication, a convolutional layer with a convolutional kernel size of 1×1 is used and the number of channels is restored to the size before reduction. The entire channel attention module can be summarized as follows:

$$F_r^g = \mathrm{Conv}\left(\mathrm{GpConv}\left((\mathrm{AvgPool}(F_r))\right)\right), \tag{14}$$

$$F_r^l = \mathrm{Conv}\left(\mathrm{GpConv}\left(F_r\right)\right), \tag{15}$$

$$G_c(F_r) = \sigma(F_r^g \oplus F_r^l), \tag{16}$$

where σ denotes the sigmoid function and $\oplus$ denotes the broadcasting addition.

Metrol. Meas. Syst., Vol. 31 (2024), No. 1, pp. 195–211
DOI: 10.24425/mms.2023.148542

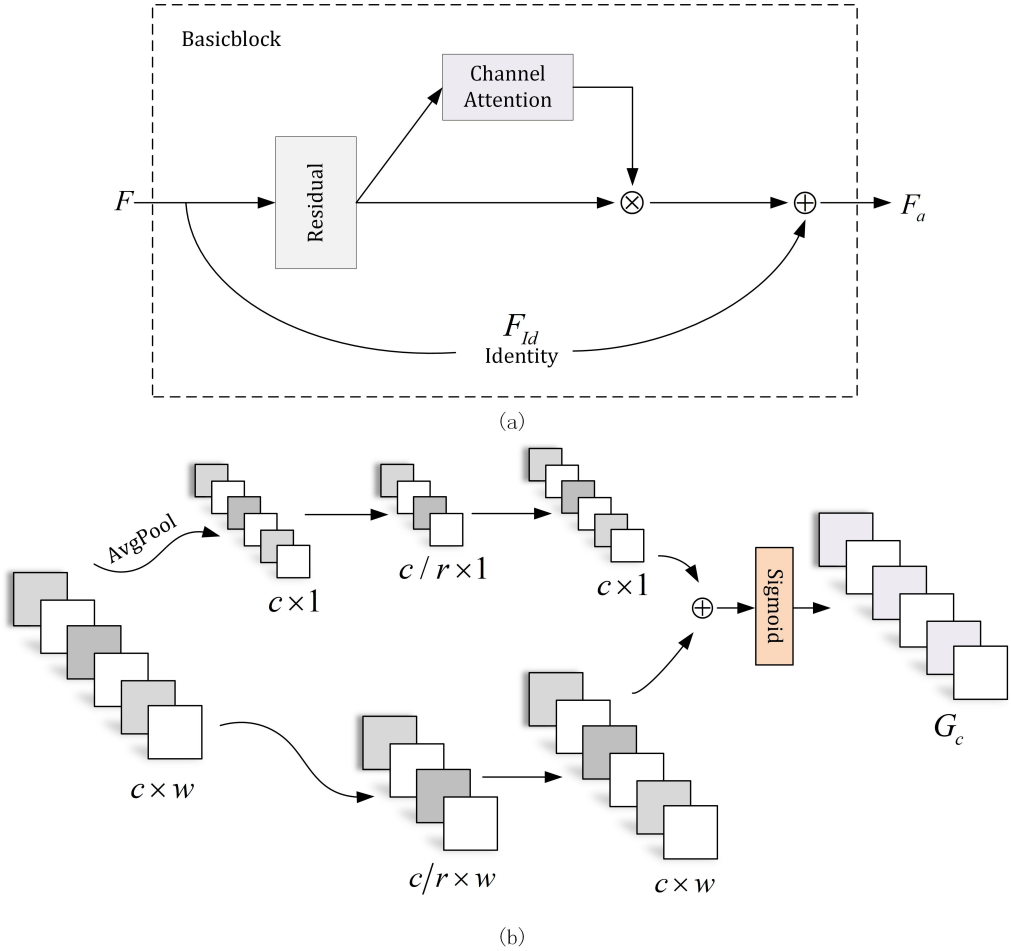


Fig. 4. (a) A BasicBlock and (b) Multiscale channel attention module in the Convolutional Attention Module, where c is the number of channels, w is the feature map size, and r is a scaling parameter of size 4.

Since in a convolutional block, the filter slides over the IMMU measurements along the time axis, global features may not be captured. Thus, an LSTM block is followed to extract the global features from the refined feature map L. We propose to utilize a 2-layer LSTM to output the final feature map of the IMMU attitude estimation system, and there are 256 hidden nodes in each LSTM layer. In addition to the normal input gate and output gate, the LSTM module also contains a forgetting gate for training long-term temporal dependencies between memorized input sequences. The 2-layer LSTM takes $P = L^T$ as input. The final step of the MGTA model is to estimate the attitude using the hidden state of the last time step in the hidden layer output with the LSTM network, which is achieved by a separate *multilayer perceptron* (MLP) as shown in (17)

$$h = \text{LSTM}(P), \quad q = \text{MLP}(h_{-1}), \tag{17}$$

where h_{-1} is the hidden state of the last time step in h.

4. Experiments

The data used for the experiments were from the *Oxford Inertial Ranging Dataset* (OxIOD) [30] and the *Robust Neural Inertial Navigation* (RoNIN) dataset [31], respectively.

OxIOD provides inertial and magnetic data recorded by a phone at a sampling rate of 100 Hz as it moved around the environment under different conditions. It also contains accurate and synchronized ground truth 3D attitudes. This dataset contains 158 sequences with a total distance of over 42 kilometres, which is much larger than previous inertial datasets. Another remarkable feature of this dataset is its diversity, which can reflect the complex movements of cell phone-based IMMUs in a variety of daily uses. Measurements were collected using four different attachments (handheld, pocket, handbag and trolley), four movement modes (slow walking, normal walking and running), *etc*. We chose all different device placement positions and different motion modes to evaluate our work. For different motion modes, we chose "slow walking" and "running", as the dataset did not provide data for the other two motion modes.

In RoNIN dataset, both ground truth and inertial data are measured independently, *i.e.*, a stationary 3D tracking cell phone (Asus Zenfone AR Tango phone) measures the body's motion state, while another cell phone, freely carried by the subject, is used to measure inertial and magnetic data.

The MGTA model was trained using the ADAM optimizer with an initial learning rate of 0.005 and a total of 200 epochs. During training, samples are randomly shuffled during each epoch. If the verification quaternion error does not decrease within 10 epochs, the learning rate will be decreased by a factor of 0.2. During training, we set the batch size to 128. Dropout regularization with a retention probability of 0.5 was applied to the recurrent connections in the LSTM layer. The networks in this paper were implemented using PyTorch, and an NVIDIA RTX 2080TI GPU was used to run our experiments.

5. Results and discussion

This section describes the evaluation metrics and some state-of-the-art frameworks for attitude estimation. We then compare the results of the MGTA model with the other competing methods and qualitatively and quantitatively analyse the attitude estimation performance of the model proposed in this paper on the OxIOD dataset and the RoNIN dataset, respectively.

5.1. Evaluation metric

In order to quantitatively evaluate the performance of the proposed method for test sequences of length N, the following indicator metrics are defined.

1. *Root Mean Quaternion Error* (RMQE):

$$\mathrm{RMQE} = \sqrt{\frac{1}{N}\sum_{i}^{N} \|(q_i - \hat{q}_i)\|^2}. \tag{18}$$

RMQE expresses the error between the estimated quaternion and the ground truth for each component in terms of the *root mean square error* (RMSE), where q denotes quaternions and N denotes the number of samples.

2. *Unit Quaternion Distance Metric* (UQD):

$$\mathrm{UQD} = \frac{1}{N}\sum_{i=1}^{N} 1 - \|\langle q_i, \hat{q}_i \rangle\|. \tag{19}$$

To characterize the difference between the estimated quaternion and the true quaternion, we measure the distance between two quaternions by calculating the inner product UQD of these two quaternions according to [32], as shown in (19).

5.2. Competing methods

We trained our proposed structure using publicly available OxIOD data and RoNIN data and compared our attitude estimation results with three other methods:

1. *Dilated Convolution Network* (DCN): A learning method for IMU noise reduction using ground truth data is proposed to improve the accuracy of 3D attitude estimation [29]. The authors use extended convolution to model the IMU error and use a suitable orientational incremental loss function to achieve noise reduction.
2. *Incremental Learning of LSTM* (IL–LSTM): The authors propose a 3D object attitude estimation method based on incremental learning of *Long Short-Term Memory* (LSTM) networks [28]. In this paper, data from inertial sensors and magnetic sensors is fed to an LSTM network and then incrementally updated to incorporate dynamic motion changes that occur during operation. Specifically, in the prediction phase, after every fixed time interval, the weights of the training model are updated again to learn new features from the inputs and further predict the attitude.
3. *Temporal Convolutional Network with Residual* (TCN): In this paper, the authors propose a deep learning-based method for MEMS IMU error compensation and use the compensated data for attitude estimation [27]. The method utilizes a temporal convolutional network to construct an output model of the MEMS gyroscope. By analyzing the IMU error model, different types of errors are compensated separately, while residual blocks are introduced in the network structure to obtain stronger feature performance.

5.3. Evaluations

In order to verify the validity of the proposed method, we calculated the RMSE of the quaternion estimated with each method with the true quaternion on the four components, and then compared the RMSE of each method, as shown in Table 1.

According to Table 1, for the test sequence Trolley, the attitude estimation error obtained by our proposed method is larger than that of DCN, but is smaller than that of TCN and IL_LSTM. For the test sequence Pocket, the attitude estimation error obtained by our proposed method is larger than that of IL_LSTM, but is smaller than that of DCN and TCN. The attitude estimation errors of the proposed method in this paper are smaller than the other methods on the four test sequences, i.e., Walk, Run, Handheld, and Handbag. The minimum RMSE of our proposed method on each component is reduced by 7.24%, 13.23%, 7.93% and 12.73%, respectively, compared with the other methods.

The quaternion estimates and errors of the different methods on the test sequence are shown in Fig. 5 and Fig. 6, respectively. In Fig. 6 for the error of each component of the quaternion, we can see that in the $q(x)$ and $q(y)$ components of the quaternion the fluctuation of the error range of our method is more obvious, and the advantage is not particularly significant, but on the $q(w)$ and

Table 1. RMQE values for each component on each test sequence for different methods.

OxIOD test sequence	DCN				TCN			
	q_x	q_y	q_z	q_w	q_x	q_y	q_z	q_w
Walk	0.0285	0.0286	0.1385	0.0861	0.0224	0.0236	0.1326	0.0788
Run	0.0321	0.0285	0.1098	0.0838	0.0318	0.0273	0.1264	0.0756
Trolley	**0.0047**	**0.0049**	**0.1048**	**0.0662**	0.0063	0.0057	0.1145	0.0769
Handheld	0.0482	0.0370	0.1236	0.0864	0.0511	0.0368	0.1392	0.0892
Handbag	0.0472	0.0325	0.1087	0.0797	0.0445	0.0331	0.1201	0.0791
Pocket	0.0703	0.0614	0.1253	0.0972	0.0687	0.0592	0.1249	0.0945
Average	0.0385	0.0321	0.1184	0.0832	0.0374	0.0309	0.1262	0.0823
OxIOD test sequence	IL_LSTM				MGTA			
	q_x	q_y	q_z	q_w	q_x	q_y	q_z	q_w
Walk	0.0207	0.0213	0.1138	0.0733	**0.0197**	**0.0206**	**0.0958**	**0.0610**
Run	0.0309	0.0297	0.1465	0.0975	**0.0227**	**0.0203**	**0.1046**	**0.0675**
Trolley	0.0061	0.0054	0.1082	0.0695	0.0054	0.0053	0.1065	0.0669
Handheld	0.0516	0.0389	0.1475	0.0968	**0.0497**	**0.0339**	**0.1123**	**0.0752**
Handbag	0.0502	0.0377	0.1104	0.0762	**0.0463**	**0.0311**	**0.1188**	**0.0774**
Pocket	**0.0629**	**0.0518**	**0.1194**	**0.0898**	0.0633	0.0520	0.1203	0.0901
Average	0.0370	0.0308	0.1243	0.0838	**0.0345**	**0.0272**	**0.1097**	**0.0730**

$q(z)$ components of the quaternion, it can be clearly seen that the quaternion error obtained by our proposed method only fluctuates a small range of fluctuations in the whole test sequence, while the other methods still have a large range of error fluctuations.

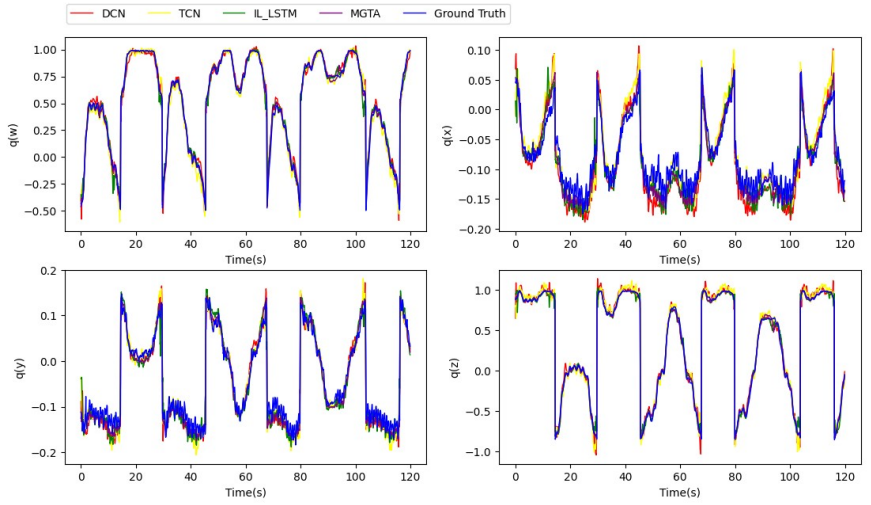


Fig. 5. True quaternions on the test dataset and the quaternions estimated by the three methods and the proposed method for comparison.

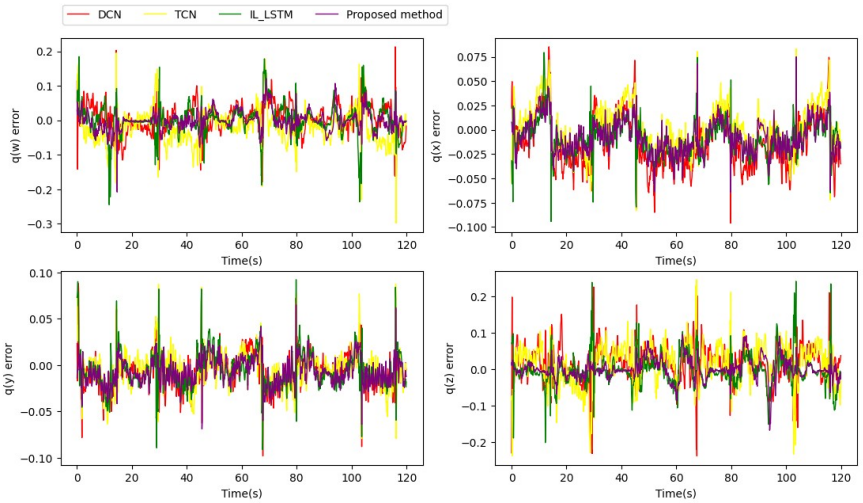


Fig. 6. Error of each component of the quaternion estimated by various methods on the test dataset.

This is likely attributed to the fact that DCN and TCN only focus on the local features of the time series and do not take into account the effect of global information on the quaternion estimation.

We calculate the average of the sum of the RMQE values of each component of the quaternion to obtain a comprehensive performance index, and the results are listed in Table 2. According to Table 2, we can see that in most cases, our proposed method is better than the other methods.

Table 2. Average of the RMQE values of each component of the quaternion.

OxIOD test sequence	DCN	TCN	IL_LSTM	MGTA
Walk	0.0704	0.0643	0.0572	**0.0492**
Run	0.0635	0.0652	0.0761	**0.0537**
Trolley	**0.0451**	0.0508	0.0473	0.0460
Handheld	0.0738	0.0790	0.0837	**0.0677**
Handbag	0.0670	0.0692	0.0686	**0.0684**
Pocket	0.0885	0.0868	**0.0809**	0.0814
Average	0.0680	0.0692	0.0689	**0.0610**

Our approach first utilizes ResNet18 to capture temporal features, and then increases the depth of the network through residual linking to obtain deeper and more complex nonlinear relationships contained in the data. At the same time, we introduce a multi-scale channel attention mechanism to obtain global and local channel scale relationships to improve the model's ability to perceive useful information and reduce the influence of noise on the results. The LSTM network is then utilized to obtain global time information. In IL_LSTM, past information is conveyed in time through the memorability of hidden units, but it may cause the model to have difficulty in capturing long-term dependencies when the time sequence is long. We resolve the problem of the LSTM model's difficulty in capturing long-term dependencies by refining the timestamps and reducing the sequence length through convolution.

Table 3 shows the UQD values of various methods on the RONIN test sequence. We can see from Table 3 that the UQD values of our method are all smaller than those of all the methods on the Seen test sequence, while on the Unseen test sequence, although the UQD values of our method are slightly higher than those of TCN, they are still lower than those of the other methods. So, in general the attitude estimation of our method is more reliable compared to the current methods.

Table 3. UQD value for each method.

RoNIN test sequence	DCN	TCN	IL_LSTM	MGTA
Seen	0.2696	0.3418	0.2788	0.2473
Unseen	0.4407	0.4101	0.4807	0.4261

6. Conclusions

In view of the noise and bias contained in the raw MEMS IMU signals, this paper proposes a new network structure based on deep learning for attitude estimation. In this model we consider correlations and global time dependencies between different patterns measured by the IMU at adjacent timestamps. The spatial features are first extracted from the normalized IMMU measurements using ResNet, then the spatial features are refined using the convolutional attention module (which combines local and global features through multi-scale channel attention), and finally the LSTM network is used to further capture and emphasize the temporal features. We validate the method on two public datasets and compare it with existing methods. The results show that the method can effectively reduce the error of the attitude estimation and make the attitude estimation more reliable. In the future, it will be necessary to study the generalization capabilities of the proposed network structure when using new IMMUs, as well as the variations in IMMU frequency and noise models, and to further develop the framework into an inertial odometer.

References

[1] Groves, P. D. (2015). Navigation using inertial sensors. *IEEE Aerospace and Electronic Systems Magazine, 30*(9), 42–69. https://doi.org/10.1109/MAES.2014.130191

[2] Brossard, M., Barrau, A. & Bonnabel, S. (2020). AI-IMU Dead-Reckoning. *IEEE Transactions on Intelligent Vehicles, 5*(4), 585–595. https://doi.org/10.1109/TIV.2020.2980758

[3] Liu, W., Caruso, D., Ilg, E., Dong, J., Mourikis, A. I., Daniilidis, K., Kumar, V. & Engel, J. (2020). TLIO: tight learned inertial odometry. *IEEE Robotics and Automation Letters, 5*(4), 5653–5660. https://doi.org/10.1109/LRA.2020.3007421

[4] Koksal, N., Jalalmaab, M. & Fidan, B. (2019). Adaptive linear quadratic attitude tracking control of a quadrotor UAV based on IMU sensor data fusion. *Sensors, 19*(1), 46. https://doi.org/10.3390/s19010046

[5] Alo, U. R., Nweke, H. F., Teh, Y. W. & Murtaza, G. (2020). Smartphone Motion Sensor-Based Complex Human Activity Identification Using Deep Stacked Autoencoder Algorithm for Enhanced Smart Healthcare System. *Sensors, 20*(21), 6300. https://doi.org/10.3390/s20216300

[6] Bhardwaj, R., Kumar, N. & Kumar, V. (2018). Errors in micro-electro-mechanical systems inertial measurement and a review on present practices of error modelling. *Transactions of the Institute of Measurement and Control, 40*(9), 2843–2854. https://doi.org/10.1177/0142331217708237

[7] Han, S., Meng, Z., Omisore, O., Akinyemi, T. & Yan, Y. (2020). Random error reduction algorithms for MEMS inertial sensor accuracy improvement – a review. *Micromachines, 11*(11), 1021. https://doi.org/10.3390/mi11111021

[8] Brotchie, J., Li, W., Kealy, A. & Moran, B. (2021). Evaluating tracking rotations using maximal entropy distributions for smartphone applications. *IEEE Access, 9*, 168806–168815. https://doi.org/10.1109/ACCESS.2021.3135012

[9] Quinchia, A. G., Falco, G., Falletti, E., Dovis, F. & Ferrer, C. (2013). A comparison between different error modeling of MEMS applied to GPS/INS integrated systems. *Sensors, 13*(8), 9549–9588. https://doi.org/10.3390/s130809549

[10] Huang, L. (2015). Auto regressive moving average (ARMA) modeling method for gyro random noise using a robust Kalman filter. *Sensors, 15*(10), 25277–25286. https://doi.org/10.3390/s151025277

[11] LeCun, Y., Bengio, Y. & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444. https://doi.org/10.1038/ nature14539

[12] Bilski, J., Rutkowski, L., Smoląg, J. & Tao, D. (2021). A novel method for speed training acceleration of recurrent neural networks. *Information Sciences, 553*, 266–279. https://doi.org/10.1016/j.ins.2020.10.025

[13] Masumura, R., Asami, T., Oba, T., Sakauchi, S. & Ito, A. (2019). Latent words recurrent neural network language models for automatic speech recognition. *IEICE Transactions on Information and Systems, E102D*(12), 2557–2567. https://doi.org/10.1587/transinf.2018EDP7242

[14] Cui, Q., Wu, S., Huang, Y. & Wang, L. (2019). A hierarchical contextual attention-based network for sequential recommendation. *Neurocomputing, 358*, 141–149. https://doi.org/10.1016/j.neucom.2019.04.073

[15] Li, D., Zhou, J. & Liu, Y. (2021). Recurrent-neural-network-based unscented Kalman filter for estimating and compensating the random drift of MEMS gyroscopes in real time. *Mechanical Systems and Signal Processing, 147*, 107057. https://doi.org/10.1016/j.ymssp.2020.107057

[16] Yang, D., Gu, C., Zhu, Y., Dai, B., Zhang, K., Zhang, Z. & Li, B. (2020). A concrete dam deformation prediction method based on LSTM with attention mechanism. *IEEE Access, 8*, 185177–185186. https://doi.org/10.1109/ACCESS.2020.3029562

[17] Sherstinsky, A. (2020). Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D: Nonlinear Phenomena, 404*, 132306. https://doi.org/10.1016/j.physd.2019.132306

[18] Pan, E., Mei, X., Wang, Q., Ma, Y. & Ma, J. (2020). Spectral-spatial classification for hyperspectral image based on a single GRU. *Neurocomputing, 387*, 150–160. https://doi.org/10.1016/j.neucom.2020.01.029

[19] Jiang, C., Chen, S., Chen, Y., Zhang, B., Feng, Z., Zhou, H. & Bo, Y. (2018). A MEMS IMU de-noising method using long short term memory recurrent neural networks (LSTM-RNN). *Sensors, 18*(10), 3470. https://doi.org/10.3390/s18103470

[20] Jiang, C., Chen, Y., Chen, S., Bo, Y., Li, W., Tian, W. & Guo, J. (2019). A mixed deep recurrent neural network for MEMS gyroscope noise suppressing. *Electronics, 8*(2), 181. https://doi.org/10.3390/electronics8020181

[21] Esfahani, M. A., Wang, H., Wu, K. & Yuan, S. (2020). OriNet: Robust 3-D orientation estimation with a single particular IMU. *IEEE Robotics and Automation Letters, 5*(2), 399–406. https://doi.org/10.1109/LRA.2019.2959507

[22] Bahdanau, D., Cho, K. & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv.Org*. https://arxiv.org/abs/1409.0473v7

[23] Shih, S.-Y., Sun, F.-K. & Lee, H. (2019). Temporal pattern attention for multivariate time series forecasting. *Machine Learning, 108*(8–9), 1421–1441. https://doi.org/10.1007/s10994-019-05815-0

[24] Wang, Y., Zhang, J., Kan, M., Shan, S. & Chen, X. (2020). Self-Supervised Equivariant Attention Mechanism for Weakly Supervised Semantic Segmentation. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 12272–12281. https://doi.org/10.1109/CVPR42600.2020.01229

[25] Chen, C., Lu, X., Markham, A. & Trigoni, N. (2018). IONet: Learning to cure the curse of drift in inertial odometry. *Thirty-Second AAAI Conference on Artificial Intelligence*, America, 6468–6476. https://ojs.aaai.org/index.php/AAAI/article/view/12102

[26] Wang, Y., Cheng, H., Wang, C. & Meng, M. Q.-H. (2021). Pose-invariant inertial odometry for pedestrian localization. *IEEE Transactions on Instrumentation and Measurement, 70*, 8503512. https://doi.org/10.1109/TIM.2021.3093922

[27] Huang, F., Wang, Z., Xing, L. & Gao, C. (2022). A MEMS IMU gyroscope calibration method based on deep learning. *IEEE Transactions on Instrumentation and Measurement, 71*, 1003009. https://doi.org/10.1109/TIM.2022.3160538

[28] Narkhede, P., Walambe, R., Poddar, S. & Kotecha, K. (2021). Incremental learning of LSTM framework for sensor fusion in attitude estimation. *Peerj Computer Science, 7*, e662. https://doi.org/10.7717/peerj-cs.662

[29] Brossard, M., Bonnabel, S. & Barrau, A. (2020). Denoising IMU gyroscopes with deep learning for open-loop attitude estimation. *IEEE Robotics and Automation Letters, 5*(3), 4796–4803. https://doi.org/10.1109/LRA.2020.3003256

[30] Chen, C., Zhao, P., Lu, C., Wang, W., Markham, A. &Trigoni, N. (2020). Deep-learning-based pedestrian inertial navigation: methods, data set, and on-device inference. *IEEE Internet of Things Journal, 7*(5), 4431–4441. https://doi.org/10.1109/JIOT.2020.2966773

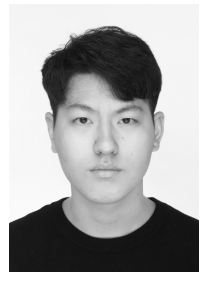
[31] Herath, S., Yan, H. & Furukawa, Y. (2020). RoNIN: Robust Neural Inertial Navigation in the Wild: Benchmark, Evaluations, & New Methods. IEEE International Conference on Robotics and Automation (ICRA), 3146–3152. https://doi.org/10.1109/icra40945.2020.9196860

[32] Huynh, D. Q. (2009). Metrics for 3D Rotations: comparison and analysis. *Journal of Mathematical Imaging and Vision, 35*(2), 155–164. https://doi.org/10.1007/s10851-009-0161-2

[33] Seong, J.-H., Lee, S.-H., Yoon, K.-K. & Seo, D.-H. (2019). Ellipse coefficient map-based geomagnetic fingerprint considering azimuth angles. *Symmetry-Basel, 11*(5), 708. https://doi.org/10.3390/sym11050708

[34] Ousaloo, H. S., Sharifi, G., Mahdian, J. & Nodeh, M. T. (2017). Complete Calibration of three-axis strapdown magnetometer in mounting frame. *IEEE Sensors Journal, 17*(23), 7886–7893. https://doi.org/10.1109/JSEN.2017.2766200

Hailong Rong received the B.E. degree in automation and the M.E. degree in pattern recognition and intelligent systems from Northeastern University, Shenyang, China, in 2003 and 2006, respectively, and the Ph.D. degree in control theory and engineering from Southeast University, Nanjing, China, in 2010. He is currently with the School of Mechanical Engineering and Rail Transit, Changzhou University, Changzhou, China. His research interests are attitude tracking and pattern recognition based on magnetic and inertial measurement units.



Tianlei Jin received his bachelor's degree in electrical engineering and automation from Linyi University in 2021. He is currently studying for a Master's degree at the School of Mechanical Engineering and Rail Transportation, Changzhou University, China. His research interest is gyroscopic noise reduction.

Metrol. Meas. Syst., Vol. 31 (2024), No. 1, pp. 195–211
DOI: 10.24425/mms.2023.148542

Xiaohui Wu received his Bachelor's degree in electrical engineering and automation from Anhui University of Technology in 2021. He is currently pursuing his Master's degree at the School of Mechanical Engineering and Railway Transportation, Changzhou University, China. His research interests are deep learning-based IMU attitude estimation.

Hao Wang obtained a Bachelor's degree in Electrical Engineering and Automation from Changzhou University in 2021. He is currently pursuing a Master's degree in Mechanical and Electronic Engineering from the School of Intelligent Manufacturing Industry at Changzhou University. He mainly engages in research related to inertial sensors.

Ling Zou received the Ph.D. degree in control science and control engineering from Zhejiang University, Hangzhou, China, in 2004. She is currently a professor with the School of Microelectronics and Control Engineering, Changzhou University, Changzhou, China. Her research interests are control engineering, biomedical signal processing, and pattern recognition.