IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 10, 2021, accepted January 24, 2021, date of publication January 28, 2021, date of current version February 23, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3055253

Research on Ship Motion Prediction Algorithm Based on Dual-Pass Long Short-Term Memory Neural Network

XIONG HU, BOYI ZHANG, AND GANG TANG
Logistics Engineering College, Shanghai Maritime University, Shanghai 201306, China
Corresponding authors: Xiong Hu (huxiong@shmtu.edu.cn) and Gang Tang (gtang@shmtu.edu.cn)

ABSTRACT When forecasting ship movements, the random errors of the inertial navigation system (INS) seriously affect the accuracy of general prediction methods. In actual measurement, the main causes of the random errors are electrostatic bias and micro-electric disturbance. In response to this problem, a novel type of dual-pass Long Short-Term Memory (LSTM) neural network architecture is developed, on the basis of regular LSTM neural network. In the designed dual-pass LSTM neural network, the random drift and the noise residual of the INS are regarded as a autoregressive moving average (ARMA) and generalized autoregressive conditional heteroskedasticity (GARCH) model. Through dual-pass layers, the prediction of drift and the correction of residual errors are realized respectively in the same time. The simulation of ship heave motion was carried out on the ship motion simulation platform, and the real-time datas which are measured by the INS are inputted to the trained dual-pass LSTM netural network. The experiment proved that, when training the same source datas offline, the average Root Mean Squared Error (RMSE) percentage of conventional LSTM network was 3.94%, but when training different source datas or training online, the prediction accuracy obvious decline. In contrast, the average RMSE percentage of the dual-pass LSTM neural network was 1.05% when training offline and 1.12% when training online. Compared with conventional LSTM networks, the dual-pass LSTM network is more targeted and has better adaptability in the field of ship-motion prediction, and this network restores the motion prediction to the actual trajectory of a ship more accurately.

INDEX TERMS Dual-pass LSTM neural network, generalized autoregressive conditional heteroskedasticity (GARCH) model, inertial navigation system (INS).

I. INTRODUCTION

Nowadays, Inertial navigation system (INS) has high application value in the field of ship-motion prediction, owing to its significant superiorities like low depend on external information and no energy radiation outside [1]–[3]. However, the long-term precision of INS is still unsatisfactory, which has restricted their further applications. Hence, analyzing the measurement errors of INS and compensating them as far as possible is of great significance to enhance the measurement precision. In the process of INS measurement, the main constraint factor for the improvement of uncertainty error accuracy. Uncertainty errors include drift error and noise error [4], [5]. Uncertainty errors are caused by the electrical characteristics and motion characteristics of the INS itself (such as electrostatic bias and electronic noise [4]), which cannot be eliminated by calibration. Therefore, in order to analyze and reduce random errors through software (without increasing hardware costs), various methods have been proposed, which can be divided into two categories, namely, conventional statistical methods and artificial intelligence (AI) methods.

Conventional statistical methods include Allan variance (AV) analysis [6], wavelet denoising (WD) [7], empirical mode decomposition (EMD) [8], Kalman filter (KF) [3], autoregressive movement mean value (ARMA) modeling [9]–[11] and autoregressive conditional heteroscedasticity (ARCH) modeling [11]–[13], etc. In particular, ARMA modeling is the use of traditional time series analysis methods to actually estimate and compensate the random drift time

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

of the gyroscope. M. Narasimhappa and J. Nayaka et al proposed a ARMA model to correct the random error of the fiber optic gyroscope [14]. TG. Zheng and H. Xiao proposed an improved GARMA model, which can predict two types of special non-Gaussian noise time series [15]. JM. Song and JW. Kang combine the generalized autoregressive conditional heteroscedasticity (GARCH) with the ARMA model for modeling, allowing the conditional variance to change over time. The method of describing this variance change is autoregressive, which can make the heteroscedastic sequence have a long-term memory [16].

With the rapid development of AI [17], multitude approaches have arisen to build more accurate nonlinear models of the random drift. The AI methods mainly include support vector machine modeling methods [18], neural networks modeling methods [19], [20], etc, which regard the processing of modeling as a problem of sequence prediction. Significantly, these methods aim to model and compensate the random drift as accurately as possible in offline conditions, there into the random drift is obtained after a preprocessing of removing white noise from the stochastic errors. The preprocessing is generally carried out by mean filtering [22], forward linear prediction filtering [23], WD [7], EMD [24], etc. aiming to reduce them pact of white noise on modeling. However, there are relatively few reports on real-time estimation and compensation of random drift based on more accurate models established by AI methods.

X. Li and F. Li et al proposed a kind of Markovian jump system (MJSs) recurrent neural networks, through the exponential stability problem in the meaning of network mean square, the stability of the time-delay system is discussed [21]. When actually measuring in a marine environment, it is found that the residual signal of random error is often not Gaussian white noise under normal distribution, and the residual change in random error should be estimated and compensated in real time [28]. Hence it is inadequate to stay at the stage of modeling and compensating offline, it is important to study further in applying the more accurate models into online compensation occasions.

In recent years, Long Short-Term Memory (LSTM, Long Short-Term Memory) neural network has been applied extensively in nonlinear fields [17], [25], not only because a LSTM neural network can approximate an arbitrary nonlinear mapping, but also because a LSTM neural network has Better performance than the time recurrent neural network (RNN) and the hidden Markov model (HMM) [22], [26]. In this paper, the random error characteristics of the INS are studied through the ARMA-GARCH model. According to the INS drift error and noise residual length and timing characteristics, a dual-transfer layer architecture LSTM neural network is designed. Using the unique structure of LSTM neural network, the error and residual error of INS can be separated and synchronized training to improve training efficiency. This improved architecture of the LSTM neural network should be the first to propose and apply to the INS system. The main contributions of this paper are summarized as follows.

1) A generalized nonlinear model of the random drift is built by a LSTM neural network, which is capable of capturing the dynamic characteristic of the random drift sufficiently.

2) On the basis of the original LSTM neural network, a transfer layer for residual calculation is added, and a network connection for residual correction is designed with reference to the GARCH model, so that the improved dual-pass LSTM neural network can predict drift errors while simultaneously Correction and compensation of noise residuals.

3) Experiments were conducted to verify the effectiveness and superiority of the improved dual-transfer layer LSTM neural network in predicting the random error of the INS system.

II. ARMA MODEL AND LSTM NEURAL NETWORK APPROXIMATION

This section introduces the ARMA model and its optimal predictor briefly, then gives the LSTM neural network which is able to approximate and realize the optimal predictor.

A. ARMA MODEL AND ITS OPTIMAL PREDICTOR

The ARMA model is described as [11]

$$x_k = c_n + \sum_{i=1}^{p} \varphi_i x_{t-i} + \sum_{j=1}^{q} \theta_j e_{t-j} + e_t \tag{1}$$

where x_k is the observation value of the time series at time k, $h(\cdot)$ is an unknown smooth nonlinear function, p and q represent the order of the autoregressive (AR) component The number and moving average (MA) components, φ_i and θ_i represent the autoregressive parameters and moving average parameters, respectively. C_n and e_t represent regression residuals and sliding errors, respectively, e_k represents zero mean, independent and uniformly distributed Gaussian noise, it is also independent of the infinite past of the observation, that is, $E[e_k \mid x_{k-1}, x_{k-2}, \cdots] = 0$. Based on the infinite past of observations, the best predictor of the ARMA model is the conditional mean $E[h(x_{k-1}, x_{k-2}, \cdots, x_{k-p}, e_{k-1}, e_{k-2}, \cdots, e_{k-q}) \mid x_{k-1}, x_{k-2}, \cdots]$ (minimum mean square error). Assuming that the ARMA model is reversible, the best predictor variable is given by [11]

$$\hat{x} \sim_k = h(x_{k-1}, x_{k-2}, \cdots, x_{k-p}, e_{k-1}, e_{k-2}, \cdots, e_{k-q}) \tag{2}$$

among them, when the parameters of $\hat{x} \sim_k$ take different values, they can represent different single-step prediction models, such as the more commonly used moving average (MA) models and autoregressive (AR) models. However, the actual observations are limited, and the ARMA model with fixed parameters has limited prediction accuracy.

Therefore, an LSTM neural network can be used to train the parameters of $\hat{x} \sim_k$ to achieve the best prediction model.

B. LSTM NEURAL NETWORK APPROXIMATION

In Eq.(2), the ARMA model constructs a type of time series predictor, which can evaluate and predict future data through the past time series and error series. In the process of real-time online prediction, the best predictor of the ARMA model can be realized through the recurrent neural network.

Compared with the conventional recurrent neural network, the internal structure of LSTM adds three gates, input gate, forget gate and output gate, and an internal memory unit c_t. The input gate controls how much the new state of the current calculation is updated to the memory unit; the forgetting gate controls how much information in the memory unit of the previous step is forgotten; the output gate controls how much the current output depends on the current memory unit. which is shown in Fig.1.

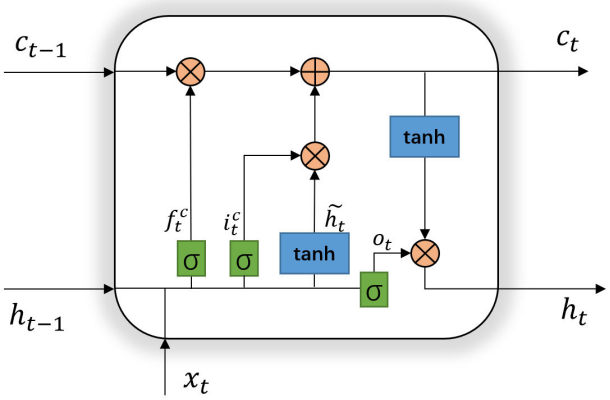


FIGURE 1. LSTM neural network unit.

In this network model, each network unit contains three logic gates, which are input gate i_t, forget gate f_t and output gate o_t.

$$\begin{aligned} i_t &= \sigma\left(W_i \cdot [h_{t-1}, x_t] + b_i\right) \\ f_t &= \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \\ o_t &= \sigma\left(W_o \cdot [h_{t-1}, x_t] + b_o\right) \end{aligned} \tag{3}$$

In the formula, W_i, b_i, W_f, b_f, W_o, b_o are the network weights to be trained. (Sigmoid function operation) is a linear unit function, and the independent variable can be mapped to the interval [0,1]. The transfer coefficient of the previous basic unit c_{t-1} is calculated by the above formula $\tilde{c}_t$ for linear superposition, That is, the update formula of memory layer c_t is

$$\begin{aligned} \tilde{c}_t &= \tanh\left(W_c \cdot h_{t-1}\right) \\ c_t &= f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \end{aligned} \tag{4}$$

W_c is the weight to be trained. The tanh unit is similar to the σ unit, which means that the value is mapped to the interval [-1,1]. The formula includes the design of the "forgotten gate", and the weight of the input information and the forgetting ratio of the previous information are determined by linear superposition. The design of the forget gate control unit is mainly used to control the flow of long-term dependence information of data, which is similar to the idea of moving average. If f_t is close to 1, the information can be transmitted for a long time in memory, and is close to 0, which means complete forgetting. The gradient disappears due to long-term dependence during training. The calculation formula of output layer h_t is:

$$h_t = o_t \odot \tanh\left(\tilde{c}_t\right) \tag{5}$$

The main part of the network can be constructed by connecting the basic units according to the scale of the problem and the length of the predicted output into a topology structure. In order to verify the prediction effect of LSTM neural network on absolute sensor data and relative sensor data, the ship motion simulation platform is used to simulate the ship deck motion, the absolute sensor system (inertial integrator) is used to measure the platform motion [27], and the relative Type sensor system (laser rangefinder) to verify the measurement results. Build 10-layer and 30-layer LSTM neural networks, select 10 sets of offline measurement data with a time sequence length of 500 as training samples, and use the training network to perform real-time prediction and verification on real-time collected unverified data and verified data, and calculate the root mean square Error (RMSE, Root Mean Square Error). The effect of training prediction is shown in Fig. 2:

It can be seen that when predicting data without relative verification, as the time series is shortened, the predictive ability of the LSTM neural network will be greatly reduced. The main reason for this phenomenon is that when the electrical sensor is working, the absolute sensor system will have irregular ultra-low frequency DC signal bias interference, which causes the drift characteristics and residual characteristics of the INS signal to be unstable. When the external disturbance changes, the network weights of the offline trained LSTM will have a significant decrease in the prediction accuracy of the same time series under the actual working conditions without reference correction. Excessive amplification of the time series will affect the learning efficiency and timeliness of the network. Therefore, the traditional LSTM neural network has greater limitations and greater interference under harsh working conditions.

III. GARCH MODEL AND DUAL-PASS LSTM NEURAL NETWORK APPROXIMATION

Generalized conditional heteroscedasticity (GARCH) is an autoregressive conditional heteroscedasticity model for the variance of time series changes. The errors in the GARCH model are not independent, and the conditional variance is not constant, which is essentially different from the ARMA model or the martingale difference sequence with constant conditional variance. It is widely used to predict variance and

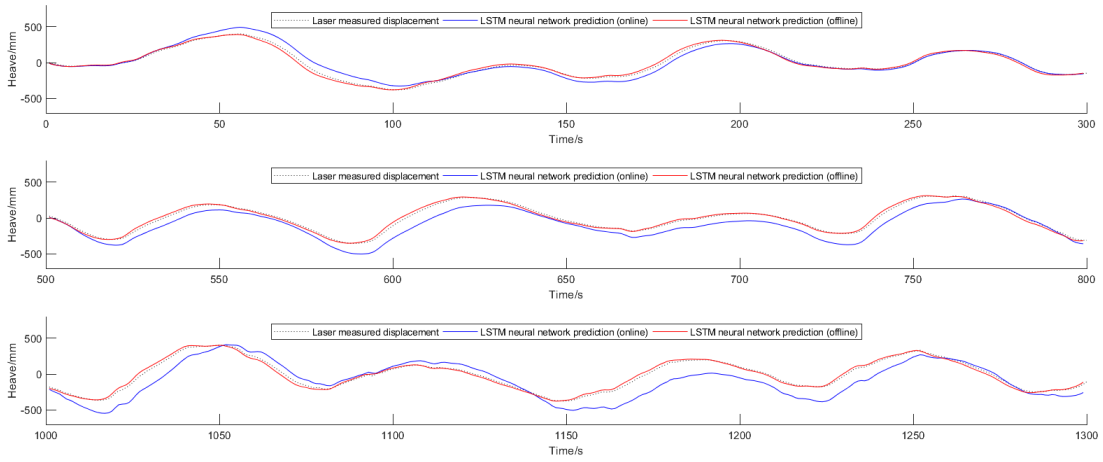


FIGURE 2. LSTM neural network prediction comparison of absolute sensor data and relative sensor data.

accurately test the changes in returns in financial market stock models.

The LSTM neural network constructed based on the ARMA model can better deal with the random errors and noise caused by the INS electrical signal. However, in the actual measurement, it is found that due to the limitation of the sampling conditions, some ultra-low frequency or slow linear actual motion may be affected. Mistakes as the residual item is lost. In this chapter, through the GARCH model analysis and prediction of the INS signal noise residual, the compensation algorithm for the lost ultra-low frequency motion signal is given, and combined with the previous LSTM neural network, an improved architecture scheme is given, so that it can control drift Separate training and correction of value and residual value.

A. GARCH MODEL

In the laboratory environment, the INS measured the vibration table movement x_t and the actual vibration table movement y_t, and calculated the covariance sequence and the autovariance sequence of x_t:

$$R_{xy}(t) = \mathrm{cov}(x_t, y_t) = \frac{\sum_{i=1}^{t}(x_i - \bar{x})(y_i - \bar{y})}{t-1}$$
$$R_x(t) = \mathrm{var}(x_t) = \frac{\sum_{i=1}^{t}(x_i - \bar{x})^2}{t-1} \quad (6)$$

Calibrate x_t and y_t to synchronize the zero position. For the variance of the time series, the GARCH-ARMA model is established for the difference, The GARCH model is described as:

$$s_t = y_t - x_t = c_s + \sum_{i=1}^{p}\psi_i x_{t-i} + \sum_{j=1}^{q}\vartheta_j \varepsilon_{t-j} + \varepsilon_t$$
$$\varepsilon_t = \eta_t\sqrt{B_t}$$
$$B_t = \alpha_t + \beta_{1t}\varepsilon_{t-1}^2 + \beta_{2t}\varepsilon_{t-2}^2 + \cdots + \beta_{qt}\varepsilon_{t-q}^2 \quad (7)$$

where ε_t is the INS signal error term, ψ_i and ϑ_j, $i = 1, 2, \cdots, q$ are unknown coefficients, $\alpha_t > 0$, $\beta_{it} \geq 0$ is a series of independent and identically distributed random variables with zero mean and same variance, and is independent for all t, η, and $\{\varepsilon_{t-k}, k \geq 1\}$.

B. CONSTRUCTION OF DUAL-PASS LSTM NETWORK

In the actual measurement process, the noise interference of the motion signal measured by the INS is often not the standard Gaussian white noise, and its fluctuation variance will change with different side sea conditions. The traditional LSTM neural network will significantly reduce the prediction accuracy of such signals. In order to solve this problem, this paper refers to the GARCH model and improves the traditional LSTM neural network. The difference is used as an additional training item for the network element. Through the training of the covariance sequence, the original output is compensated and modified. Its structure is shown in the figure 3:

In unit t, S_t and C_t represent the residual memory layer and drift memory layer, respectively, h_{t-1} is the transfer output of the displacement variable of the previous unit, x_t and y_t represent the current INS signal input and laser ranging calibration signal, respectively enter. f_t^c, i_t^c, o_t and respectively represent the forget gate, memory gate, and output gate of the drift memory layer.

$$i_t^c = \sigma\left(W_i^c \cdot [h_{t-1}, x_t] + b_i^c\right)$$
$$f_t^c = \sigma\left(W_f^c \cdot [h_{t-1}, x_t] + b_f^c\right)$$
$$o_t^c = \sigma\left(W_o^c \cdot [h_{t-1}, x_t] + b_o^c\right) \quad (8)$$

In the formula, W_i^c, b_i^c, W_f^c, b_f^c, W_o^c, b_o^c are the training weights of the network, and the drift memory layer is updated through the training results of the drift layer memory

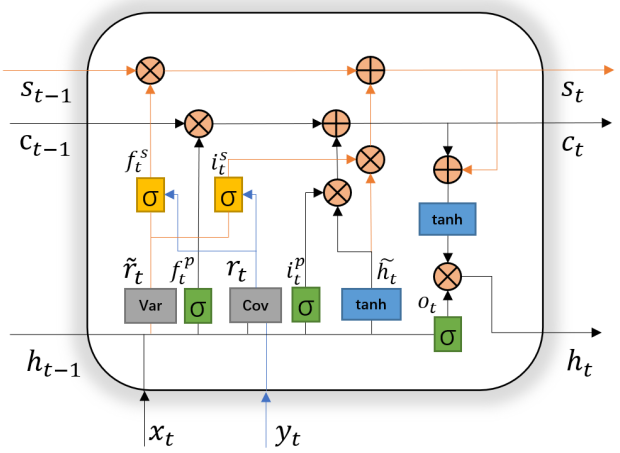


FIGURE 3. Inertial coordinate system and shipboard coordinate system.

gate and the forget gate.

$$\begin{aligned} \tilde{h}_t &= \tanh\left(W_h \cdot h_{t-1}\right) \\ c_t &= f_t^c \odot c_{t-1} + i_t^c \odot \tilde{h}_t \end{aligned} \tag{9}$$

r_t and $\tilde{r}_t$ are the autovariance sequence of the INS signal and the covariance sequence of the laser ranging signal:

$$\begin{aligned} r_t &= \operatorname{cov}\left(x_t, y_t\right) \\ \tilde{r}_t &= \operatorname{var}\left(x_t\right) \end{aligned} \tag{10}$$

i_t^s and f_t^s represent the memory gate and forget gate of the residual memory layer, respectively.

$$\begin{aligned} i_t^s &= \begin{cases} \xi\sigma\left(W_i^s \tilde{r}_t + b_i^s\right), & \text{(Online)} \\ \xi\sigma\left(V_i^s r_t + b_i^s\right), & \text{(Offline)} \end{cases} \\ f_t^s &= \begin{cases} \xi\sigma\left(W_f^s \tilde{r}_t + b_f^s\right), & \text{(Online)} \\ \xi\sigma\left(V_f^s r_t + b_f^s\right), & \text{(Offline)} \end{cases} \end{aligned} \tag{11}$$

$W_i^s, V_i^s, b_i^s, \quad W_f^s, \quad V_f^s, b_f^s$ is the training weight of the network, ξ is the network switch weight, a bool variable. update the residual memory layer correction formula.

$$s_t = f_t^s \odot s_{t-1} + i_t^s \odot \tilde{h}_t \tag{12}$$

The final output layer h_t calculation formula is:

$$h_t = o_t \odot \tanh\left(c_t + s_t\right) \tag{13}$$

IV. TRAINING PROCESS

Since the structure of the dual-pass LSTM neural network is more complicated, in order to more accurately predict the actual movement of the inertial platform, the prediction and correction of the INS signal through the dual-pass LSTM neural network can be completed through distributed training. As shown in Fig. 4, the network training and the verification is completed in the following 5 steps.

Step 1: Preprocessing and data splitting. Collect the static outputs of INS signal data in groups, for each group record,as $\{z_k, k = 1, 2, \cdots, N\}$, record the actual movement of the vibrating platform through the laser sensor as $\{y_k, k = 1, 2, \cdots, k\}$, where N denotes the length of the dataset. Then preprocess the $\{z_k\}$ through mean filtering and generate a new dataset labelled as $\{x_k\}$, to reduce the impact of the white noise on modeling the random drift. Split the $\{x_k\}$ into four parts: an offline training set with a length of N_1 employed to train and optimize the parameters of the drift layer in the dual-pass LSTM neural network, including the weights and thresholds; an offline validation set with a length of N_2 used to guide the selection of the hyperparameters of the drift layer, namely, to determine a proper structure of the dual-pass LSTM neural network; an online training set with a length of N_3 employed to train and optimize the parameters of the residual layer in the dual-pass LSTM neural network; and a test set with a length of N_4 used to estimate the generalization performance of the determined LSTM neural network. The proportion between the four parts is set as $N_1 : N_2 : N_3 : N_4 = 50\% : 10\% : 30\% : 10\%$.

Step 2: Drift layer offline training. Drop off the residual layer, set $\xi = 0$. Train parameters of the drift layer with the given structure to realize the one-step optimal predictor of the random drift. Based on the training set, the training matrix is designed as:

$$X^{(1)} = \begin{bmatrix} x_{p+1} & x_{p+2} & \cdots & x_{N_1} \\ x_p & x_{p+1} & \cdots & x_{N_1-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_1 & x_2 & \cdots & x_{N_1-p} \end{bmatrix} \tag{14}$$

where each column represents a training sample, including an input vector (the second row to the last row) and the corresponding target (the first row). After training, the parameters $W_i^c, b_i^c, W_f^c, b_f^c, W_o^c, b_o^c$ of the drift layer can be acquired.

Step 3: Drift layer evaluation. Evaluate the generalization performance of the trained Drift layer on the validation set. Based on the validation set, the validation matrix is:

$$X^{(2)} = \begin{bmatrix} x_{N_1+p+1} & x_{N_1+p+2} & \cdots & x_{N_1+N_2} \\ x_{N_1+p} & x_{N_1+p+1} & \cdots & x_{N_1+N_2-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N_1+1} & x_{N_1+2} & \cdots & x_{N_1+N_2-p} \end{bmatrix} \tag{15}$$

where each column represents a validation sample, including an input vector (the second row to the last row) and a target (the first row). For each validation sample, the generalization error between the actual output and the corresponding target can be calculated. Obviously, the mean square value of the generalization errors for all validation samples, i.e. the root mean square error (RMSE) [28], can be used to quantify the generalization performance. The smaller the RMSE, the better the performance.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{16}$$

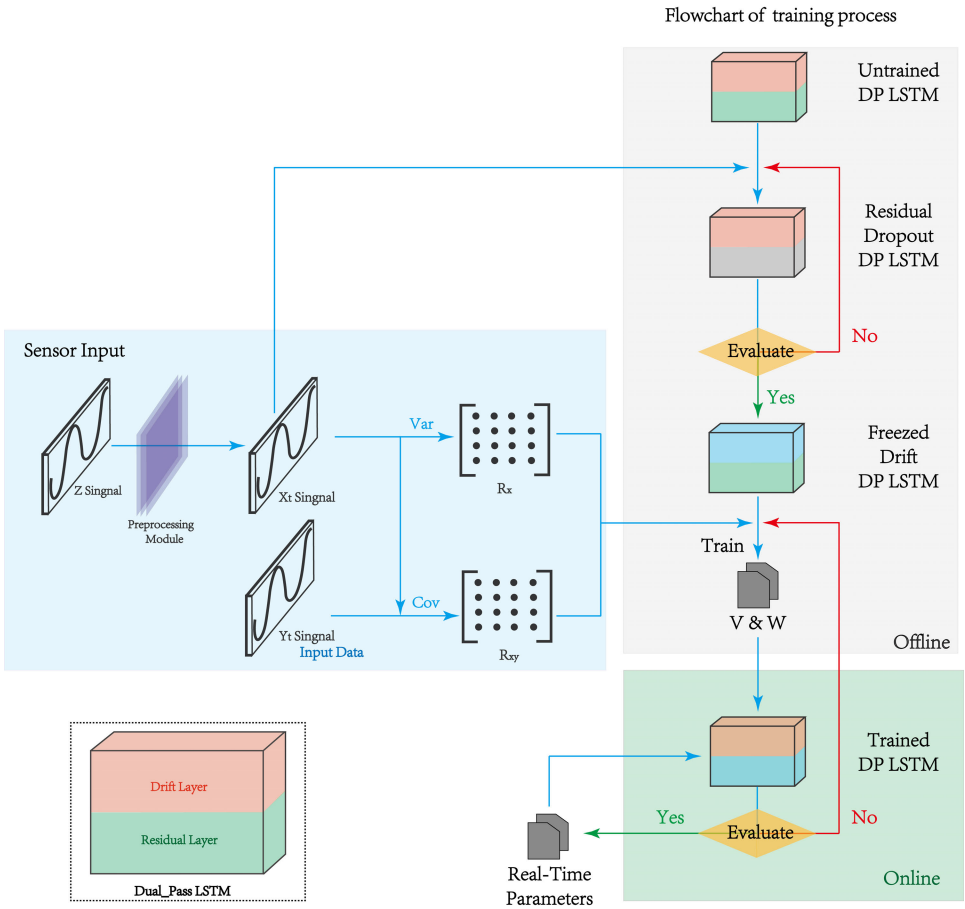


FIGURE 4. Flowchart for framework.

Step 4: Residual layer offline training. Turn on the residual layer, set $\xi = 1$. Build the training matrix is designed from:

$$X^{(3)} = \begin{bmatrix} x_{N_1+N_2+p+1} & x_{N_1+N_2+p+2} & \cdots & x_{N_1+N_2+N_3} \\ x_{N_1+N_2+p} & x_{N_1+N_2+p+1} & \cdots & x_{N_1+N_2+N_3-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N_1+N_2+1} & x_{N_1+N_2+2} & \cdots & x_{N_1+N_2+N_3-p} \end{bmatrix} \tag{17}$$

$$Y^{(3)} = \begin{bmatrix} y_{N_1+N_2+p+1} & y_{N_1+N_2+p+2} & \cdots & y_{N_1+N_2+N_3} \\ y_{N_1+N_2+p} & y_{N_1+N_2+p+1} & \cdots & y_{N_1+N_2+N_3-1} \\ \vdots & \vdots & \ddots & \vdots \\ y_{N_1+N_2+1} & y_{N_1+N_2+2} & \cdots & y_{N_1+N_2+N_3-p} \end{bmatrix} \tag{18}$$

Record the autovariance sequence of $\{x_k\}$ as $R_x^{(3)}$ and the covariance sequence of $\{x_k\}$ and $\{y_k\}$ respectively as $R_x^{(3)}$.

$$\begin{aligned} R_x^{(3)} &= \begin{bmatrix} r^{(p+1)} & r^{(p)} & \cdots & r^{(1)} \end{bmatrix}' \\ R_{xy}^{(3)} &= \begin{bmatrix} \tilde{r}^{(p+1)} & \tilde{r}^{(p)} & \cdots & \tilde{r}^{(1)} \end{bmatrix}^T \end{aligned} \tag{19}$$

where,

$$\begin{aligned} r^{(t)} &= \operatorname{var}\left(x_{N_1+N_2+t:N_1+N_2+N_3+t-p}\right) \\ &= \frac{\sum_{N_1+N_2+t}^{N_1+N_2+N_3+t-p}(x_i-\bar{x})^2}{N_3-p-1} \\ \tilde{r}^{(t)} &= \operatorname{cov}\left(x_{N_1+N_2+t:N_1+N_2+N_3+t-p}\right) \\ &= \frac{\sum_{N_1+N_2+t}^{N_1+N_2+N_3+t-p}(x_i-\bar{x})(y_i-\bar{y})}{N_3-p-1} \end{aligned} \tag{20}$$

Training the parameters $W_i^s, b_i^s, W_f^s, b_f^s$ of the residual layer in offline status with $R_{xy}^{(3)}$, Then calculate the parameters of the residual layer in offline status with $R_x^{(3)}$ as:

$$\begin{aligned} V_i^s &= W_i^s \tilde{r}_t r_t^{-1} \\ V_f^s &= W_f^s \tilde{r}_t r_t^{-1} \end{aligned} \tag{21}$$

Step 5: Testing. Estimate the generalization performance of the determined dual-pass LSTM neural network on the test set. Based on the test set, the test matrix is:

$$X^{(4)} = \begin{bmatrix} x_{N_1+N_2+N_3+p+1} & x_{N_1+N_2+N_3+p+2} & \cdots & x_N \\ x_{N_1+N_2+N_3+p} & x_{N_1+N_2+N_3+p+1} & \cdots & x_{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N_1+N_2+N_3+1} & x_{N_1+N_2+N_3+2} & \cdots & x_{N-p} \end{bmatrix} \tag{22}$$

$$Y^{(4)} = \begin{bmatrix} y_{N_1+N_2+N_3+p+1} & y_{N_1+N_2+N_3+p+2} & \cdots & y_N \\ y_{N_1+N_2+N_3+p} & y_{N_1+N_2+N_3+p+1} & \cdots & y_{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ y_{N_1+N_2+N_3+1} & y_{N_1+N_2+N_3+2} & \cdots & y_{N-p} \end{bmatrix} \tag{23}$$

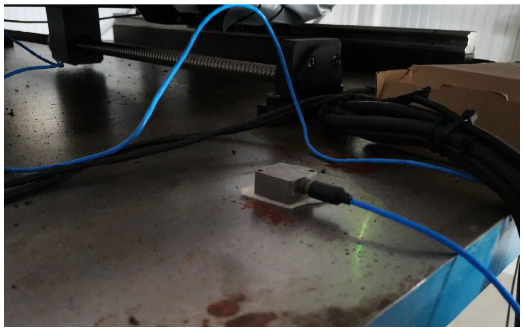
where each column represents a test sample, including an input vector (the second row to the last row) and a target (the first row). After the above procedures, an appropriate dual-pass LSTM neural network model can be obtained, then used to estimate and compensate the random drift in real time.

V. RESULTS AND DISCUSSION

This section will show the verification experiments clearly, to prove the feasibility and superiorities of the proposed dual-pass LSTM neural network.

A. EXPERIMENTAL SETUP

The equipment used in the experiment includes self-developed INS-FPGA circuit board and supporting acceleration sensor PCB-3711b (Fig. 5); relative laser sensor Panasonic HG-C1400 (Fig. 5) and Host computer, acquisition board, etc.



(a) Acceleration sensor.

(b) Laser sensor.

FIGURE 5. Experimental setup.

B. OFFLINE TRAINING

The trend undetermined weight can be set by referring to the weighted moving average method. For time series input x_k, use the following formula to update the weighted trend

$$c_t = (1-\beta)\dot{h}_{t-1} + \beta c_{t-1} \tag{24}$$

The shock energy density of conventional transport barges is mainly distributed in 0.05 0.25hz. The actual measurement comparison shows that the trend interference main frequency of acceleration integration is lower than 3mHz, so the forgetting correction coefficient is 0.98. Respectively, to build 200 Layer dual-pass LSTM neural network. Set the initial parameters of the drift layer as

$$W^c_{\text{initial}} = \begin{bmatrix} \beta^p(1-\beta)^{N-p} & \beta^{p+1}(1-\beta)^{N-p-1} & \cdots & \beta^N(1-\beta)^0 \\ \beta^{p-1}(1-\beta)^{N-p+1} & \beta^p(1-\beta)^{N-p} & \cdots & \beta^{N-1}(1-\beta)^1 \\ \vdots & \vdots & \ddots & \vdots \\ \beta^0(1-\beta)^N & \beta^1(1-\beta)^{N-1} & \cdots & \beta^{N-p}(1-\beta)^p \end{bmatrix} \tag{25}$$

Then follow steps 1-3 to conduct multiple offline prediction experiments on the dual-pass LSTM. Fig. 6 presents the training results of the drift layer and residual layer of the daul-pass LSTM neural network under partial offline conditions and the corrected output results, And compared with the laser positioner data, and calculated the error distribution of the final output result compared with the laser data through the box-plot. Experiments have found that in the offline training process, the drift training layer has higher training accuracy for the same group, but for different groups of test data, due to inconsistent vibration amplitude-frequency characteristics, the training accuracy decreases. The residual layer can compensate and modify the residual noise according to the change, which significantly improves the training accuracy.The comparison results are shown in Table. 1.

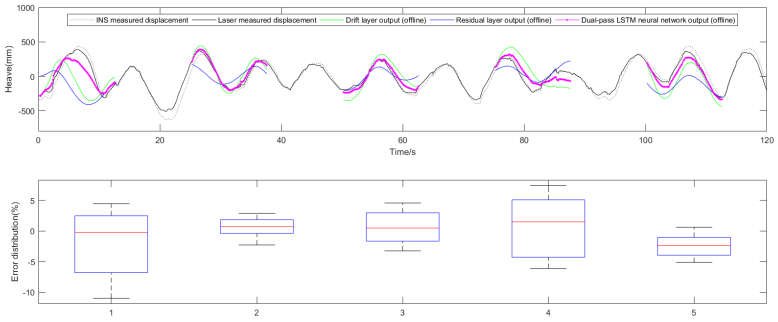
TABLE 1. Comparison of offline training results between LSTM network and dual-pass LSTM network (RMSE%).

Group No.	LSTM			Dual-pass LSTM		
	Same	Different		Same	Different	
1	4.11	11.11	9.25	1.16	0.92	0.95
2	4.19	13.74	7.36	1.14	1.44	1.33
3	3.67	9.35	7.12	0.85	1.17	1.14
4	4.11	10.34	13.87	0.88	0.93	0.77
5	4.97	8.78	12.21	1.42	1.34	0.85
6	4.05	9.40	13.49	0.78	1.13	1.41
7	3.95	9.88	12.41	0.86	1.07	1.41
8	3.12	9.60	12.80	1.00	1.35	1.27
9	4.89	12.78	9.11	0.92	0.76	1.10
10	2.74	13.34	10.71	0.81	0.90	1.18

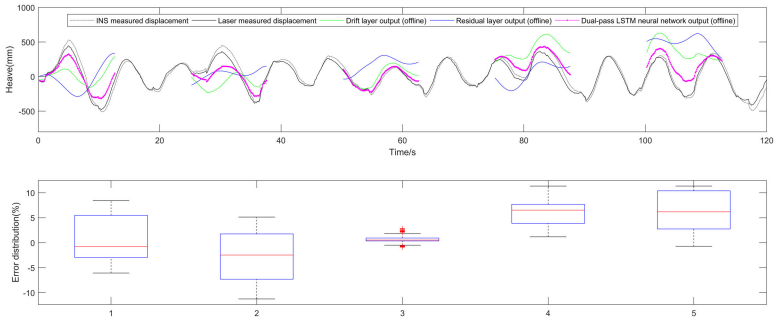
It can be seen from Table. 1, when training the same source datas offline, the average RMSE percentage of conventional LSTM network was 3.94%, but when training different source datas, the prediction accuracy obvious decline. In contrast, the average RMSE percentage of the dual-pass LSTM neural network was 0.98% for same source datas and 1.21% for different source datas.

C. ONLINE TRAINING

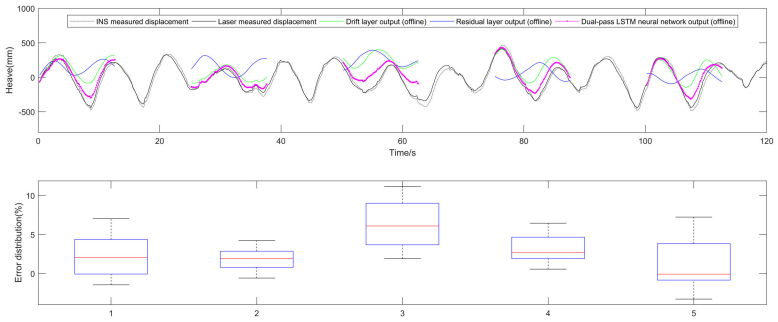
Combining the results of offline training, calculate the online prediction parameters according to the above steps 4-5, freeze the autovariance training weights of the residual layer,



(a) Same group.



(b) Different group(1).



(c) Different group(2).

FIGURE 6. Offline predicted results.

and verify the prediction effect in an environment without laser signal y_t verification. And selected a conventional LSTM neural network for comparison. The results show in Fig. 7. Through the above-mentioned networks, a statistical comparison of the results of multiple sets of data is carried out. The comparison results are shown in Table. 2.

It can be seen from Table. 2 that the average RMSE percentage of the traditional LSTM is about 8.75% when online training is performed on the INS signal of the same source, and the prediction accuracy will fluctuate significantly with the change of the signal source. The average RMSE percentage of the online training of the Dual-pass LSTM neural network constructed in this paper is about 1.12%, and the fluctuation range is significantly smaller, and the accuracy and reliability of its prediction are

TABLE 2. Comparison of online training results between LSTM network and dual-pass LSTM network (RMSE%).

No.	LSTM	Dual-pass LSTM
1	6.82	1.22
2	7.98	1.17
3	7.54	1.16
4	12.08	0.76
5	9.32	1.31

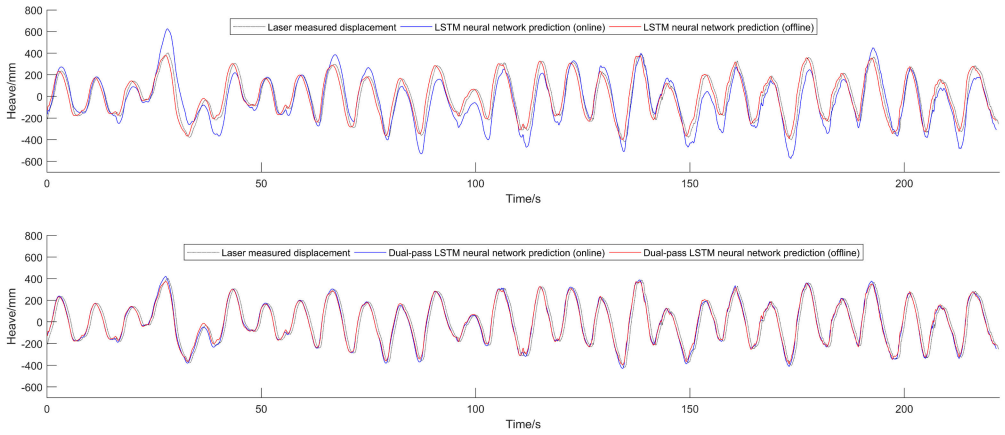


FIGURE 7. Comparison of the predicted results online.

significantly improved. It can be seen that under the same training scale, the dual-pass LSTM neural network has better accuracy in fitting and predicting INS signal data than the conventional LSTM neural network, especially when the time series is short. However, in the online prediction process, due to frequent fluctuations of the INS, the adaptability of the correction coefficient and input correction coefficient has fluctuated in the past. Of course, the reason for the decrease in accuracy is still due to insufficient training samples and parameter settings. However, by observing Fig. 6 and Fig. 7, it can be seen that the prediction still guarantees a considerable fit for the noise trend, and this result can still be used as a colored noise range to optimize the design of the filtering algorithm.

VI. CONLUSION

In this paper, based on the characteristics of the INS signal, a new scheme for real-time prediction and correction of the drift error and noise residual of the INS signal is proposed. In this method, firstly, the random drift is regarded as the ARMA model, and on the basis of the comparison of its predicted residual variables, a GARCH model is established for correction and compensation. Furthermore, according to the GARCH-ARMA model, a Daul-pass LSTM neural network model was built and designed on the basis of the conventional LSTM neural network, and the random drift and noise residual of the INS signal were carried out through the real-time sequence and the self-variance variance sequence. The training has realized the best predictor of the GARCH-ARMA model, which can fully capture the dynamic characteristics of the INS signal, and has carried out a verification experiment. Experimental results show that in the real-time prediction process, the Daul-pass LSTM neural network model can correct the residuals more accurately than the conventional LSTM neural network model, especially in the online state prediction accuracy has been significantly improved, thus verifying the Daul-pass The effectiveness and superiority of LSTM neural network model. By applying this method, it is expected to improve the measurement accuracy of INS. In addition, the design idea of the Daul-pass LSTM neural network model proposed in this article may provide a useful reference for the construction of other cyclic neural network control models (especially time series analysis). One of the future work will be to develop a hyperparameter optimization algorithm that can automatically and optimally select hyperparameters, with better generalization performance.

REFERENCES

[1] W. Sun, D. Wang, L. Xu, and L. Xu, "MEMS-based rotary strapdown inertial navigation system," *Measurement*, vol. 46, no. 8, pp. 2585–2596, Oct. 2013.

[2] Q. Q. Zhang, L. Zhao, J. H. Zhou, B. H. Wang, and Y. P. Zhang, "A real-time airborne terrain aided inertial navigation system and its performance analysis," *Adv. Space Res.*, vol. 60, no. 12, pp. 2751–2762, Dec. 2017.

[3] Y. Liu, X. Fan, C. Lv, J. Wu, L. Li, and D. Ding, "An innovative information fusion method with adaptive Kalman filter for integrated INS/GPS navigation of autonomous vehicles," *Mech. Syst. Signal Process.*, vol. 100, pp. 605–616, Feb. 2018.

[4] G. Hu, B. Gao, Y. Zhong, and C. Gu, "Unscented Kalman filter with process noise covariance estimation for vehicular INS/GPS integration system," *Inf. Fusion*, vol. 64, pp. 194–204, Dec. 2020.

[5] J. C. Springmann and J. W. Cutler, "Flight results of a low-cost attitude determination system," *Acta Astronautica*, vol. 99, pp. 201–214, Jun. 2014.

[6] T. Jianghe, F. Zhenxian, and D. Zhenglong, "Identification method for RLG random errors based on allan variance and equivalent theorem," *Chin. J. Aeronaut.*, vol. 22, no. 3, pp. 273–278, Jun. 2009.

[7] C. Huimin, Z. Ruimei, and H. Yanli, "Improved threshold denoising method based on wavelet transform," *Phys. Procedia*, vol. 33, pp. 1354–1359, 2012.

[8] P. K. Shaw, D. Saha, S. Ghosh, M. S. Janaki, and A. N. S. Iyengar, "Investigation of coherent modes in the chaotic time series using empirical mode decomposition and discrete wavelet transform analysis," *Chaos, Solitons Fractals*, vol. 78, pp. 285–296, Sep. 2015.

[9] K. Wang, Y. Wu, Y. Gao, and Y. Li, "New methods to estimate the observed noise variance for an ARMA model," *Measurement*, vol. 99, pp. 164–170, Mar. 2017.

[10] H. Jiang, S. Duan, L. Huang, Y. Han, H. Yang, and Q. Ma, "Scale effects in AR model real-time ship motion prediction," *Ocean Eng.*, vol. 203, May 2020, Art. no. 107202.

[11] Q. Liu, G. Zhang, S. Ali, X. Wang, G. Wang, Z. Pan, and J. Zhang, "SPI-based drought simulation and prediction using ARMA-GARCH model," *Appl. Math. Comput.*, vol. 355, pp. 96–107, Aug. 2019.
[12] T. Nakatsuma, "Bayesian analysis of ARMA–GARCH models: A Markov chain sampling approach," *J. Econometrics*, vol. 95, no. 1, pp. 57–69, Mar. 2000.
[13] M. Escobar-Anel, J. Rastegari, and L. Stentoft, "Option pricing with conditional GARCH models," *Eur. J. Oper. Res.*, vol. 289, no. 1, pp. 350–363, Feb. 2021.
[14] M. Narasimhappa, J. Nayak, M. H. Terra, and S. L. Sabat, "ARMA model based adaptive unscented fading Kalman filter for reducing drift of fiber optic gyroscope," *Sens. Actuators A, Phys.*, vol. 251, pp. 42–51, Nov. 2016.
[15] T. Zheng, H. Xiao, and R. Chen, "Generalized ARMA models with martingale difference errors," *J. Econometrics*, vol. 189, no. 2, pp. 492–506, Dec. 2015.
[16] J. Song and J. Kang, "Parameter change tests for ARMA–GARCH models," *Comput. Statist. Data Anal.*, vol. 121, pp. 41–56, May 2018.
[17] M. Sangiorgio and F. Dercole, "Robustness of LSTM neural networks for multi-step forecasting of chaotic time series," *Chaos, Solitons Fractals*, vol. 139, Oct. 2020, Art. no. 110045.
[18] S. Kim, Y. Choi, and M. Lee, "Deep learning with support vector data description," *Neurocomputing*, vol. 165, pp. 111–117, Oct. 2015.
[19] D. Li, J. Zhou, and Y. Liu, "Recurrent-neural-network-based unscented Kalman filter for estimating and compensating the random drift of MEMS gyroscopes in real time," *Mech. Syst. Signal Process.*, vol. 147, Jan. 2021, Art. no. 107057.
[20] J. T. Connor, R. D. Martin, and L. E. Atlas, "Recurrent neural networks and robust time series prediction," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 240–254, Mar. 1994.
[21] X. Li, F. Li, X. Zhang, C. Yang, and W. Gui, "Exponential stability analysis for delayed semi-Markovian recurrent neural networks: A homogeneous polynomial approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6374–6384, Dec. 2018.
[22] J. Tulensalo, J. Seppänen, and A. Ilin, "An LSTM model for power grid loss prediction," *Electric Power Syst. Res.*, vol. 189, Dec. 2020, Art. no. 106823.
[23] S. Li, X. Luo, Y. Li, L. Zeng, and Z. He, "Harmonic current forecasting method for hybrid active power filter based on optimal linear prediction theory," in *Proc. Int. Conf. Electr. Control Eng.*, Sep. 2011, pp. 4806–4809.
[24] G. Xu, W. Tian, and L. Qian, "EMD- and SVM-based temperature drift modeling and compensation for a dynamically tuned gyroscope (DTG)," *Mech. Syst. Signal Process.*, vol. 21, no. 8, pp. 3182–3188, Nov. 2007.
[25] Y. Hu, J. Ni, and L. Wen, "A hybrid deep learning approach by integrating LSTM-ANN networks with GARCH model for copper price volatility prediction," *Phys. A, Stat. Mech. Appl.*, vol. 557, Nov. 2020, Art. no. 124907.
[26] S. Chakraborty, J. Banik, S. Addhya, and D. Chatterjee, "Study of dependency on number of LSTM units for character based text generation models," in *Proc. Int. Conf. Comput. Sci., Eng. Appl. (ICCSEA)*, Mar. 2020, pp. 1–5.
[27] X. Chen, L. Qi, Y. Yang, Q. Luo, O. Postolache, J. Tang, and H. Wu, "Video-based detection infrastructure enhancement for automated ship recognition and behavior analysis," *J. Adv. Transp.*, vol. 2020, pp. 1–12, Jan. 2020.
[28] X. Chen, X. Xu, Y. Yang, H. Wu, J. Tang, and J. Zhao, "Augmented ship tracking under occlusion conditions from maritime surveillance videos," *IEEE Access*, vol. 8, pp. 42884–42897, 2020.

XIONG HU received the bachelor's and Ph.D. degrees from Shanghai Jiao Tong University. He was a Visiting Professor with the Queensland University of Technology, Australia. He is currently a Professor and a Ph.D. Supervisor with Shanghai Maritime University (SMU), the Dean of Logistics Engineering College, SMU, and the Dean of the Sino-Dutch Mechatronics Engineering College, SMU. He was responsible for or participated in almost 100 research projects, which were sponsored by national funding programs, government, and both here and abroad enterprises, such as NNSF, 863 Program, Dubai Aluminium Company Ltd., Port of Damman, Port of Shanghai, Shanghai Bao Steel Company, ZPMC, and BV. His current research interests include research and teaching works in health condition monitoring, remote control, condition assessment, maintenance management of large equipment and its structure, and intelligent processing and prediction of health condition data and signals of machines. In recent years, his major research works are emphasized on the technique development and its application of health condition management, fault diagnosis, safety assessment, and integrated online system of the large hoisting appliances, such as cranes and logistics equipment. He serves as the Executive Director for the Shanghai Mechanical Engineering Society, the Chairman for the Committee of Equipment Condition Monitoring, Assessment, and Strategic Decision of the Shanghai Mechanical Engineering Society, and the Deputy Director for the Branch of Health Condition Monitoring of Shanghai Equipment Management Association.

BOYI ZHANG received the bachelor's degree in mechanical and electronic engineering and the master's degree in mechanical engineering from Shanghai Maritime University, where he is currently pursuing the Ph.D. degree in logistics equipment safety engineering. His research interest includes Ocean Engineering.

GANG TANG received the master's degree in mechanical engineering from the Harbin Institute of Technology and the Ph.D. degree in mechanical engineering from Shanghai Jiao Tong University. He joined the faculty at Shanghai Maritime University, where he is currently an Associate Professor with the Department of Mechanical Engineering. He has been in charge of more than ten programs and participated in more than 15 programs. He has published 36 articles and holds 16 patents awarded. His research interest includes Ocean Engineering.