Journal of
Marine Science and Engineering

MDPI

Article

A Novel Short-Term Ship Motion Prediction Algorithm Based on EMD and Adaptive PSO–LSTM with the Sliding Window Approach

Xiaoyu Geng , Yibing Li and Qian Sun *

Key Laboratory of Advanced Marine Communication and Information Technology, College of Information and Communication Engineering, Ministry of Industry and Information Technology, Harbin Engineering University, Harbin 150001, China
* Correspondence: qsun@hrbeu.edu.cn

Abstract: Under the influence of variable sea conditions, a ship will have an oscillating motion comprising six degrees of freedom, all of which are connected to each other. Among these degrees of freedom, rolling and pitching motions have a severe impact on a ship's maritime operations. An accurate and effective ship motion attitude prediction method that makes the prediction in a short period of time is required to guarantee the safety and stability of the ship's maritime operations. Traditional methods are based on time domain analysis, such as the autoregressive moving average (ARMA) models. However, these models have limitations when it comes to predicting the nonlinear and nonstationary characteristics of real ship motion attitude data. Many intelligent algorithms continue to be applied in nonlinear and nonstationary ship attitude prediction, such as extreme learning machines (ELMs) and the long short-term memory (LSTM) neural network, as well as other deep learning methods, showing promising results. By using the sliding window approach, the time-varying dynamic characteristics of the ship's motion attitude can be preserved better. The simulation results demonstrate that the proposed model performs well in terms of predicting the nonlinear and nonstationary ship motion attitude.

Keywords: ship motion attitude prediction; deep learning; sliding window technique; parameter optimization algorithm

Citation: Geng, X.; Li, Y.; Sun, Q. A Novel Short-Term Ship Motion Prediction Algorithm Based on EMD and Adaptive PSO–LSTM with the Sliding Window Approach. *J. Mar. Sci. Eng.* **2023**, *11*, 466. https://doi.org/10.3390/jmse11030466

Academic Editor: Md Jahir Rizvi

Received: 19 January 2023
Revised: 12 February 2023
Accepted: 15 February 2023
Published: 21 February 2023

1. Introduction

Under the influence of wind [1], waves [2], and other environmental elements, the six degrees of freedom comprising the swaying motion of large ships can become complicated and uncertain [3]. These factors represent a threat to ships' offshore operations, especially in complex environments. Methods for the short-term prediction of a ship's motion attitude have been proposed to predict the offshore motion of ships in the next few seconds in real time. Such methods provide decision-making information for the accurate control of a ship's offshore operations and the selection of the optimal operation time to enhance the safety and efficiency of the offshore operations. In general, there are four types of models for predicting a ship's motion: physical models based on the ship's hydrodynamic coefficients and equations, statistical models based on historical data and future data, intelligent models based on time series analysis and intelligent algorithms [4,5], and hybrid models [6,7].

Methods for the short-term prediction of a ship's motion attitude that were based on linear hydrodynamic motion equations were extensively used in early research. In 1969, Kaplan [8,9] designed a Wiener filter based on the statistical parameters of a ship's motion power spectrum to predict the ship's short-term motion. However, this method has the disadvantages of calculation complexity and low accuracy. With the development of modern control theory, a method for the short-term prediction of a ship's motion attitude based on the Kalman filter was proposed to predict it in real time [10]. However, the accuracy and stability of the prediction method based on the Kalman filter cannot meet the necessary requirements, especially in severe sea environments.

The methods for the short-term prediction of a ship's motion attitude are based on the time domain analysis approach, which is a data-driven prediction method. It only requires the historical data of a ship's motion to establish a time series model to predict the ship's extremely short-term attitude. However, methods, such as AR and ARMA, require the ship's motion data to be stationary and linear, which is unsuitable for realizing good prediction results of the nonlinear or nonstationary characteristics of the ship's motion [11]. In the past few years, methods for the short-term attitude prediction of a ship's motion based on machine learning (ML) models have become very popular in the context of nonlinear or nonstationary characteristics [12–19]. These data-driven prediction methods have gained more attention due to their superior capabilities of learning and modeling complex nonlinear relationships, including artificial neural networks (ANNs) [20], recurrent neural networks (RNNs) [21], support vector machines (SVMs) [22,23], random forest (RF) [24], multi-layer perceptron (MLP) [25], feed-forward neural networks (FNNs), back-propagation neural networks (BPNNs), and extreme learning machines (ELMs) [26–28]. Of these, RNNs memorize the previously known information and pass this to the input, which ensures the relation among the input information and achieves good prediction results, especially for periodic sequences. However, relying solely on the above intelligent models may not meet the practical requirements for an accurate prediction. Furthermore, there are some limitations and considerations to be aware of when using a single machine learning model to predict a ship's motion:

1. Due to having less generalization ability, a single neural network encounters the problems of over-fitting, vanishing gradients, and network training explosions when faced with the complex patterns in a ship's motion dataset.
2. When dealing with huge datasets, simple neural network models may become unstable and have low efficiency.

To overcome the shortcomings of ML methods, researchers have been inspired to develop a promising methodology: deep learning (DL). DL, as a branch of ML, trains the data model by utilizing multiple processing layers at multiple levels [29]. The performance of DL models is improved by increasing the number of hidden layers, while the deeper architecture increases the number of parameters to be optimized, which further increases the training time of the model. The potential of DL in applications requiring predictions has been highlighted in the recent literature. For time series forecasting, DNN models, such as long short-term memory (LSTM), gated recurrent units (GRUs), and hybrid models have proven to be both powerful and accurate tools. Currently, researchers are considering hybrid models to enhance the performance of DL models. Qin et al. proposed AR-DWT-EMD to solve the problem of the prediction of the nonlinear and nonstationary motion of ships [30]. In [31], the decomposition and the Hilbert spectrum of the inputs (sea waves) were compared with the decomposition and the Hilbert spectrum of the outputs (ship movements, generated by the waves) to study the time–frequency characteristics of the ship's response. However, we mainly focused on the decomposition of the original nonstationary time series to decrease the effect of noise. The decomposition extracts the important features to improve the accuracy of the prediction. EMD can adaptively perform time–frequency localization analysis to effectively extract the characteristic information of the original signal. Wang et al. proposed the Bi-LSTM TPA hybrid model, which extracts the time features from both the forward and reverse roll angle time series to improve the prediction of the ship's roll angle [32]. Additionally, hyperparameter inference and optimization procedures in neural networks, such as ant colony optimization [33], PSO [34], and the genetic algorithm [35], are used to improve the prediction performance. Yin et al. proposed a scheme for predicting rolling based on an adaptive sliding window considering the characteristics of the ship's rolling motion. An online experiment on the prediction of the ship's rolling was conducted to verify the effectiveness of the adaptive sliding window [36].

In [37], the simulation results showed that the combination of LSTM and PSO improved the accuracy of the prediction of the ship's motion. Z. Nie et al. conducted a simulation experiment on the prediction results of support vector regression (SVR) algorithms based on four commonly used kernel functions and compared the effectiveness and practicality of these kernel functions in the prediction of the ship's motion [38]. Zeguo Zhang et al. proposed a prediction algorithm based on GPSO-ANFIS and applied it to the real-time prediction of the ship's rolling. The simulation results showed the advantage of the method regarding its accuracy, stability, and real-time performance [39]. In summary, the following issues need to be noted for the short-term prediction of a ship's motion:

1. To complete the short-term prediction, a few seconds of ship motion attitude data are derived by the proposed model.
2. The sliding window technique is introduced to turn time series predictions into supervised learning for ML methods. Each window is utilized to train and update the model. After each computation is completed, the window shifts to a new position by one step.
3. Because of the nonstationary characteristics of time-series data, the prediction accuracy is affected by the unstable mean and variance of datasets. Therefore, to obtain better prediction results, the present work needs to use a data pre-processing method to reduce the effect of nonstationary characteristics.
4. Considering the practicality of predictive models, the proposed model needs to guarantee high-accuracy results when faced with multi-step-ahead predictions.

In this paper, aiming to reduce the nonlinear and nonstationary ship motion characteristics and obtain the optimal parameter of the neural network, a hybrid multi-step prediction model is proposed by combining the LSTM model with EMD, and adaptive PSO is proposed to predict ship motion attitude in a few seconds. The EMD method is employed for dealing with nonlinear and nonstationary time series, and the LSTM approach is used for training and predicting the derived ship attitude, while the parameter optimization algorithm based on PSO is utilized to maximize the performance of the prediction model by adjusting the parameters of the LSTM neural network. Additionally, time series datasets can be framed into supervised learning for multi-step predictions by utilizing the sliding window technique.

The rest of this paper is organized as follows. Section 2 gives a brief description of EMD and the sliding window approach. In Section 3, the hybrid ship attitude prediction model is established by EMD and the LSTM neural network with the sliding window approach. In Section 4, the proposed model is implemented in a real-time ship motion attitude data experiment, and the model's performance is tested using at different datasets. Conclusions are derived in Section 5.

2. Basic Knowledge

2.1. Empirical Mode Decomposition Method

Affected by various random and uncertain factors, noise exists in the measured ship motion data and interferes with the valid signals. It overlays and obscures the best-expected results when processing the original observation data. Therefore, it is crucial to apply denoising techniques to the raw data prior to the ship motion attitude prediction processing in order to enhance the accuracy of the subsequent forecast. EMD decomposes the signal into a finite number of eigenmode functions (IMFs), discards the noisy IMFs, and reconstructs the remaining IMFs to denoise the raw data. Since EMD is based on the time-scale characteristics of the data itself and does not require a given basis function in advance, it has obvious advantages when dealing with nonsmooth and nonlinear time series. The process of decomposition involves the following steps:

- Find all maximum and minimum points of the time series $x(t)$ and then fit a curve with a cubic spline function to obtain the upper and lower envelope of $x(t)$, which can be represented, respectively, as $u(t)$ and $l(t)$.

- Calculate the average of $u(t)$ and $l(t)$ to obtain the mean envelope m_x, which is shown as Equation (1):

$$m_x = \frac{u(t) + l(t)}{2}. \tag{1}$$

- New time series h_x can be calculated as

$$h_x = x(t) - m_x. \tag{2}$$

- Judge if h_x is satisfied with the condition of IMFs. Repeat Steps (1), (2), and (3) if it is not satisfied until the mean envelope tends to zero. Then, the first intrinsic modal function imf_1 is obtained.
- By subtracting imf_1 from the original time series $x(t)$, the new time series r_{x1} without high frequency is derived.
- By repeating the above process, the intrinsic modal function $\{imf_2, imf_3, ..., imf_n\}$ is obtained. When the r_{xn} cannot be decomposed, it is represented as the residual of $x(t)$.

  After the above steps, the original time series $x(t)$ can be represented as

$$x(t) = \sum_{i=1}^{n} imf_i + r_{xn}, \tag{3}$$

where r_{xn} represents the trend of $x(t)$, and it is without a high-frequency component.

During the whole process of EMD, one of the cyclic processes is the shifting process in which the optimal time of shifting and decomposition needs to be considered. Additionally, the two termination criteria include the component termination condition and the decomposition termination condition.

The shifting is the process of obtaining the IMF component, and the basic method is to continuously find extreme points from the original signal and continue shifting according to the decomposition steps until a certain condition is met. The purpose of this process is to reduce the asymmetry of the signal so that the waveform tends to be symmetric around the zero mean line, so as to meet the basic characteristics of the component. Additionally, the instantaneous frequency can be calculated by transformation. In order to ensure that the components obtained by the decomposition have sufficient original physical significance of frequency modulation (FM) and amplitude modulation (AM), the number of shifting cycles should not be excessive. Too many cycles will over-smooth the component and make it a constant amplitude FM signal, which loses the original physical meaning. On the other hand, too few shifting cycles will make the resulting component not entirely meet the basic characteristics of the component and do not obtain accurate instantaneous and meaningful frequency.

2.2. Sliding Window Approach

A sliding window is a fixed-size window that will circle the values around a point in the time series to obtain an interval that is used to calculate statistical indicators for data of a specified length. The sliding window width has an immediate effect on the model prediction. The window length is used to weigh the amount of input data and the length of historical information included. However, too much input data will lead to complex calculations and slow down the training of the neural network. Less input data tend to include less historical information, making it difficult to reflect the cycle pattern. The application of the sliding window technique in multi-steps is shown in Figure 1.

Datasets can be framed into supervised learning for prediction by sliding windows, and data from each window are utilized to train and update the model. After completing each computation, the window shifts to the next new position.

Since the selection of the key parameters of LSTM has a great influence on the accuracy of ship motion attitude forecasting, these parameters need to be selected reasonably.

The particle swarm optimization (PSO) algorithm has the advantages of simple structure, high precision, fast convergence speed, and ability to deal with nonlinear and multivariable problems, which are effective tools for the selection of LSTM model parameters.

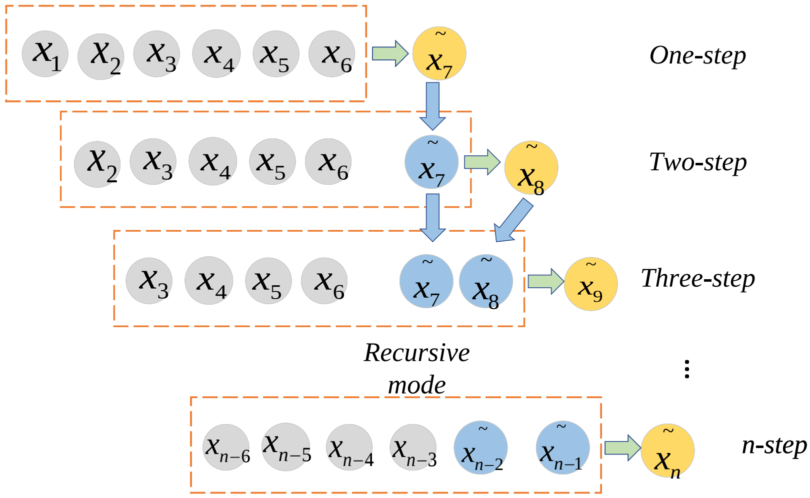


Figure 1. Sliding window approach for multi-step.

3. Short-Term Ship Motion Prediction Algorithm based on EMD and Adaptive PSO–LSTM

3.1. LSTM Neural Network Parameter Optimization for Ship Attitude Prediction

The long short-term memory (LSTM) neural network, which is a novel recurrent neural network, was proposed by Hochreiter and Schmidhuber [40]. As a deep learning model, LSTM learns the pattern from historical data accurately by utilizing the selective memory capability of machine learning and digging into the intrinsic patterns of known time series to achieve its short-term forecasting of time series. The advantage of LSTM is to solve the problem of gradient disappearance compared with other types of recurrent neural networks by introducing the concept of state units and gates into the neural network. As a result, it has better adaptability in data analysis compared with the RNN network. The structure of the LSTM unit is shown in Figure 2:

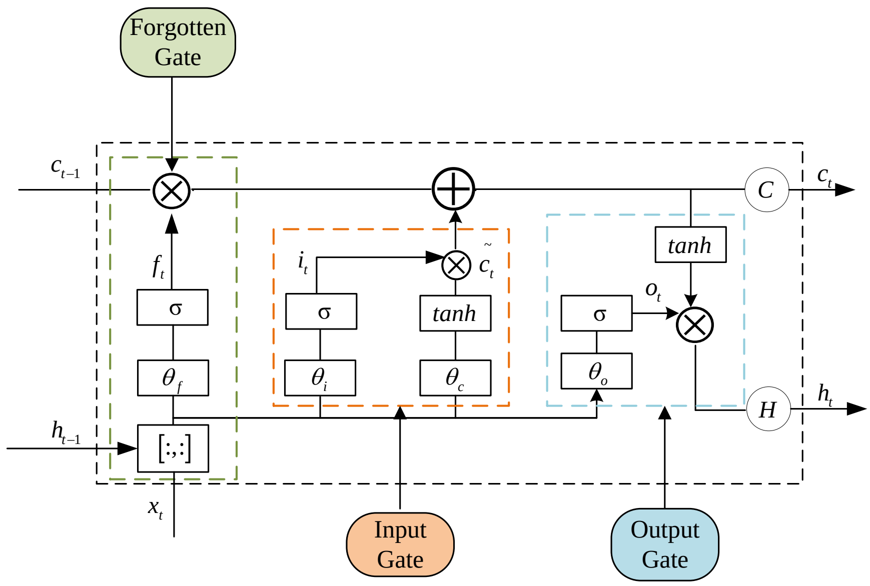


Figure 2. LSTM cell structure.

Each neural unit contains three gate control structures which are forgotten gate, input gate, and output gate. The selective transmission of data information is controlled by these gates, and the output of the sigmoid layer is a value between zero and one, which describes the ratio of information transmission. The valid information at the last epoch in f_t is discarded and retained by the forgotten gate. Current valid information i_t is stored in the input gate, which determines the update of the cell state. The output layer determines the information that needs to be output as o_t in the LSTM neural unit. The output value of the hidden layer h_t and the unit state c_t at the current epoch are determined by the output value of the hidden layer at the previous epoch h_{t-1}, the unit state c_{t-1}, and the input value at the current epoch x_t.

Since the input is determined by the output of the previous epoch h_{t-1}, the input of the network at the current epoch x_t and the activation function sigmoid control the information transfer ratio. The calculation process of the forget gate is shown as follows.

$$f_t = \sigma(\mathbf{W_f} h_{t-1} + \mathbf{U_f} x_t + b_f), \tag{4}$$

where $\mathbf{W_f}$ and $\mathbf{U_f}$ represent the weight matrix, b_f denotes the bias term, and σ is the activation function. Since the input of the sigmoid function is determined by the output at the previous epoch h_{t-1} and the input of the network at the current epoch x_t, the calculation process of the input gate is shown as follows.

$$i_t = \sigma(\mathbf{W_i} h_{t-1} + \mathbf{U_i} x_t + b_i), \tag{5}$$

where i_t represents output, $\mathbf{W_i}$, $\mathbf{U_i}$ represent the weight matrix, and b_i denotes the bias term. The tanh function is utilized to update the cell state and create a new candidate vector in the input gate. Then, the input of the memory cell $\tilde{c}_t$ is obtained as follows:

$$\widetilde{c_t} = \tanh(\mathbf{W_c} h_{t-1} + \mathbf{U_c} x_t + b_c), \tag{6}$$

where $\mathbf{W_c}$, $\mathbf{U_c}$ represent the weight matrix, and b_c is the bias term. Since the input is determined by the output of the previous moment h_{t-1} and the input of the network at the current moment x_t, the output of gate o_t is obtained from the activation function sigmoid. The output gate is calculated as follows.

$$o_t = \sigma(\mathbf{W_o} h_{t-1} + \mathbf{U_o} x_t + b_o), \tag{7}$$

where $\mathbf{W_o}$, $\mathbf{U_o}$ represent the weight matrix, and b_o denotes the bias term. We combine the partial information retained in the forget gate with the input gate to form a new cell unit c_t as follows.

$$c_t = f_t \odot c_{t-1} + i_t \odot c_t. \tag{8}$$

Next, c_t will be sent to the tanh function through the output gate to determine the output value of the hidden layer at the current epoch h_t, which can be derived by $h_t = o_t \odot \tanh(c_t)$, where $\odot$ denotes the element-wise vector product. In general, LSTM includes two kinds of hidden states: a slow state c_t to solve the vanishing gradient problem and a fast state h_t to make complex decisions over short periods of time.

The PSO algorithm is an appropriate method for parameter selection in the LSTM model, thanks to its benefits, such as straightforward structure, high accuracy, fast convergence rate, and the capacity to address nonlinear and multivariable problems. PSO describes the members of the group as particles, and the fitness of all particles is determined in space through a fitness function. In the early stage of evolution, both the position and speed of each particle are initialized randomly. The particles cooperate with each other during the flight and adjust their speed and positions in time according to the motion state of themselves and their companions in order to land on a better position. However, each particle is a solution of the solution space in PSO. Each particle knows its own position and the information of other particles. It adjusts its position and speed through its optimal position and area of the group, and then the global optimal solution is derived. The speed and position of the PSO are updated as follows:

$$v_{i,j}^{t+1} = wv_{i,j}^{t} + c_1 r_1 (\hat{y}_{i,j}^{t} - x_{i,j}^{t}) + c_2 r_2 (\overset{*}{y}_{j}^{t} - x_{i,j}^{t}), \tag{9}$$

$$x_{i,j}^{t+1} = x_{i,j}^{t} + v_{i,j}^{t+1}, \tag{10}$$

where w is the inertia weight, c_1 and c_2 are learning factors, r_1 and r_2 are independent random numbers distributed between zero and one, and $v_{i,j}^{t}$, $x_{i,j}^{t}$, $\hat{y}_{i,j}^{t}$, and $\overset{*}{y}_{j}^{t}$ are the velocity component, position component, individual optimal value, and group global optimal value of the i-th particle in the j-th dimension in the t-th iteration, respectively.

The value of w affects the model's optimization ability. In order to avoid premature convergence of the model, an adaptive PSO method is adopted in this paper. It adaptively adjusts the inertia weight as follows.

$$w = \begin{cases} w_{\min} - \frac{(w_{\max} - w_{\min})(f - f_{\min})}{f_{avg} - f_{\min}}, f \le f_{avg} \\ w_{\max}, f > f_{avg} \end{cases}, \tag{11}$$

where $w_{\max}$, $w_{\min}$ are maximum and minimum of w, respectively; f is the current fitness value of the particle; $f_{\min}$ and f_{avg} represent the current minimum fitness of all particles and the average fitness value, respectively.

Since ship motion data have the characteristics of instability, nonlinearity, and periodic uncertainty, an adaptive PSO–LSTM is proposed in this paper. In the novel model, an adaptive PSO is utilized to optimize the network hyperparameters. The whole process of the adaptive PSO–LSTM method is shown as follows.

- Preprocess the ship historical movement data.
- Initialize the particle swarm parameters, including the determination of the population size, number of iterations, learning factors, and limited intervals for particle position and velocity.
- Initialize the LSTM network structure, which refers to the determination of the number of neurons in each layer of the network and the number of hidden layers. It also divides the data into training samples, validation samples, and test samples.
- Determine the fitness function and select the optimal particle fitness value by calculating and comparing the fitness value of each particle. The fitness value fit_i of population individuals x_i with LSTM model parameters is defined as Equation (12).

$$fit_i = \frac{1}{2}\left(\frac{1}{M}\sum_{m=1}^{m}\frac{y_m - \hat{y}_m}{y_m} + \frac{1}{N}\sum_{n=1}^{n}\frac{y_n - \hat{y}_n}{y_n}\right), \tag{12}$$

  where M and N represent the number of training samples and verification samples, respectively; y_m and $\hat{y}_m$ represent the true value and the prediction value of the training sample, respectively; y_n and $\hat{y}_n$ represent the true value and the prediction value of verification samples, respectively.
- Calculate and evaluate the particle fitness value according to the difference of the particle fitness value. The global optimal position and the local optimal position of the particle are both determined.
- Update the velocity and position of the particles based on Equations (9) and (10).
- Determine whether the particles meet the conditions for the iteration termination. If the maximum number of iterations is reached, the optimal parameters are assigned to the LSTM, and the training is performed and outputs the short-term ship motion prediction value. Otherwise, it returns to Step 5 to continue execution until the termination condition is met.

- The optimal results obtained are assigned to the connection weights of the LSTM network, and this prediction model is trained to output the optimal solution for time series prediction.

The flowchart of the adaptive PSO–LSTM method is shown in Figure 3.

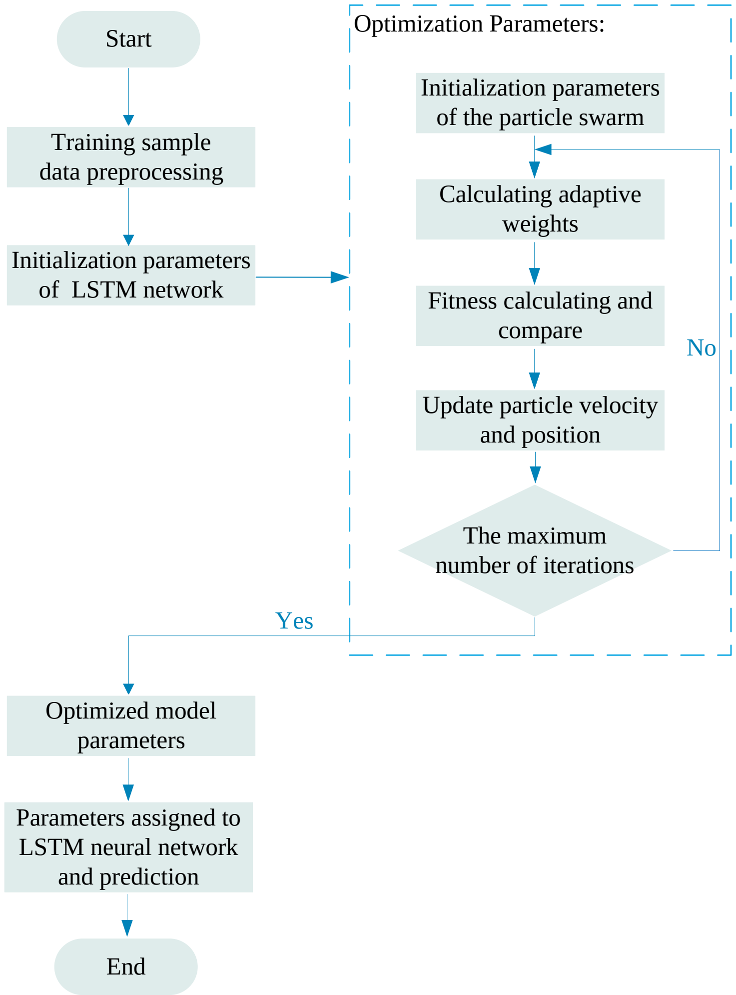


Figure 3. The flowchart of adaptive PSO–LSTM prediction method.

Most previous studies have only used the fitting error of the training sample as the fitness value and ignored the effect of the testing sample data, which results in over-fitting in the neural network. This leads to the model prediction result not being optimal. Therefore, the fitting error of the training sample and the verification error of the verification sample should be considered in the fitness function.

3.2. EMD–LSTM Model Based on Sliding Window Approach

The ship motion time series is regarded as a nonlinear signal sequence. EMD has certain advantages in dealing with the end effect and reducing the effect of system noise. Therefore, it was selected to decompose the original sequence before prediction, which improves the model's fitting performance. This part of the prediction is based on the LSTM neural network with the sliding window. The adaptive PSO is used to optimize the value of network hyperparameters in the LSTM model, which cannot be selected objectively and scientifically. As a result, the whole process is summarized as follows:

- Decompose the raw ship motion sequence into multiple specific subsequences by utilizing the EMD algorithm.

- Divide the dataset into the training set and the testing set and predict each sub-model component separately with a sliding window and optimized LSTM neural network.
- Weight and reconstruct the prediction of each sub-model to obtain the final prediction results.

The flowchart is shown in Figure 4:

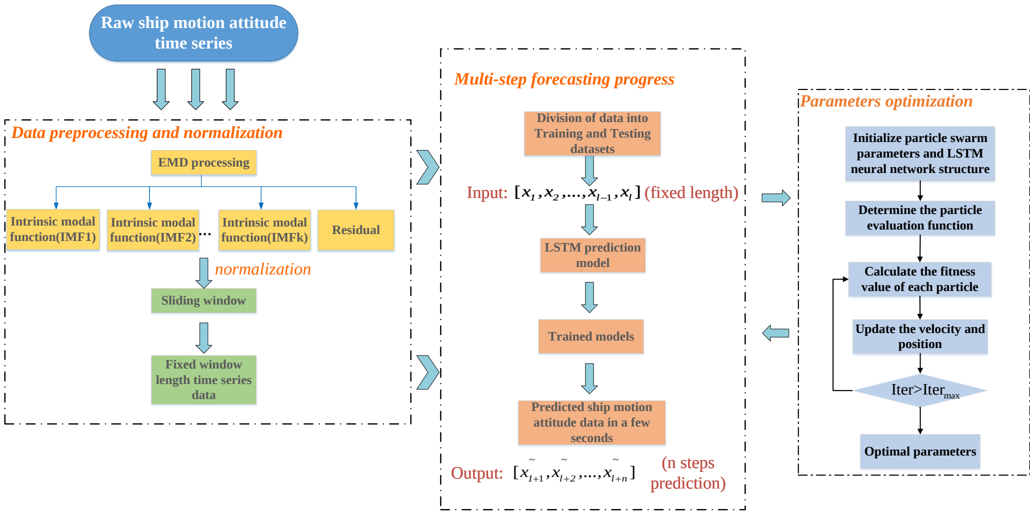


Figure 4. The overall structure of the proposed hybrid prediction model.

4. Experiment Results and Analysis

4.1. Experiment Design and Parameter Settings

Experiments were conducted to verify the effectiveness of the prediction model proposed in this paper. The experimental data used in this study were collected from an inertial measurement unit (IMU) installed on a large ship and divided into two categories in total: static state data set and motion state data set. The "static state" and "motion state" represent the state of the ship when we measured the experimental data of ship attitude. "Static state" means that the ship is docked at the port. "Motion state" means that the ship is sailing at a certain speed. To ensure the reliability and accuracy of the proposed algorithm, it was necessary to select the datasets in different states and different timestamps. The detailed description of the datasets is shown in Table 1. The sampling frequency of the IMU was 4 Hz. The scene of raw data acquisition is shown in Figure 5. The first 80% of the data was used for training and the rest was used for testing. Figures 6–10 show the different datasets of raw roll angle and pitch angle of the ship.

Table 1. The description of different raw datasets.

Dataset	The Start Time of Record Data	Duration (s)	Roll Angle (°)			Pitch Angle (°)		
			Average Value	Maximum Value	Minimum Value	Average Value	Maximum Value	Minimum Value
#1	20 December 2014 12:10:21.000	250	−0.543618911	−0.530798	−0.553233	−0.00710956	−0.004698	−0.010346
#2	20 December 2014 16:30:59:000	250	−0.560047567	−0.543898	−0.579896	−0.006100966	−0.004592	−0.007791

Table 1. *Cont.*

Dataset	The Start Time of Record Data	Duration (s)	Roll Angle (°)			Pitch Angle (°)		
			Average Value	Maximum Value	Minimum Value	Average Value	Maximum Value	Minimum Value
#3	20 December 2014 12:30:00:000	1500	−0.621413551	−0.534555	−0.636779	−0.006321536	−0.003541	−0.00883
#4	11 March 2015 12:48:22:250	300	0.729984617	8.2649	−6.5716	−0.04718685	0.8845	−0.9738
#5	19 April 2016 16:12:56:027	300	0.437897983	1.6867	−1.2942	−0.738008183	−0.3692	−1.1086

Figure 5. The scene of raw data acquisition.

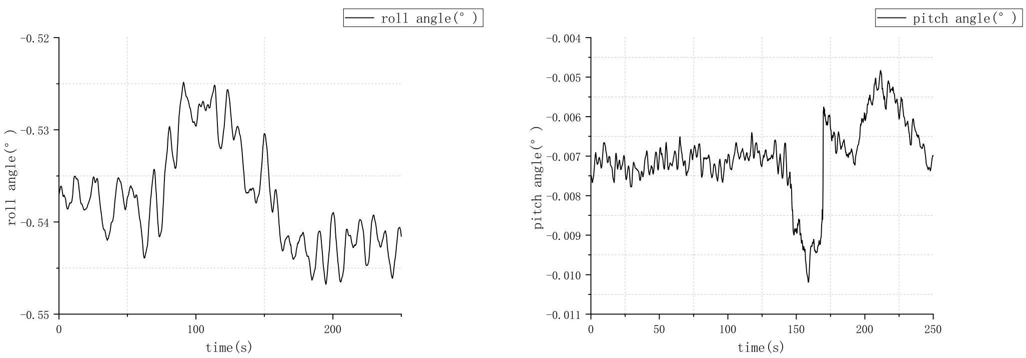


Figure 6. Dataset 1: The raw roll and pitch angle of the ship.

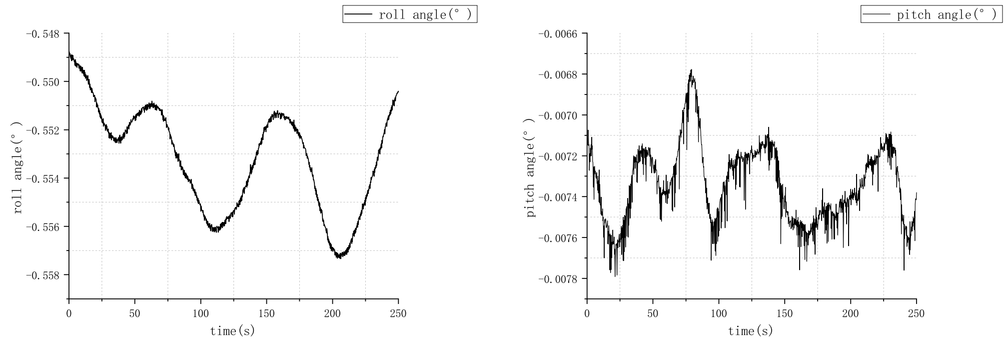


Figure 7. Dataset 2: The raw roll and pitch angle of the ship.

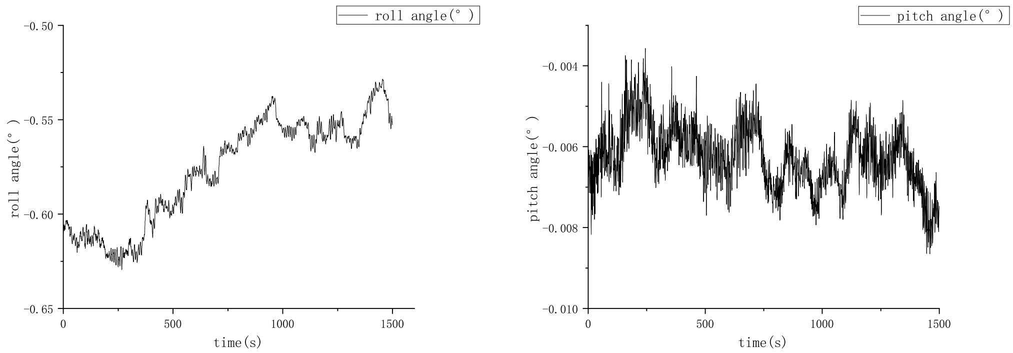


Figure 8. Dataset 3: The raw roll and pitch angle of the ship.

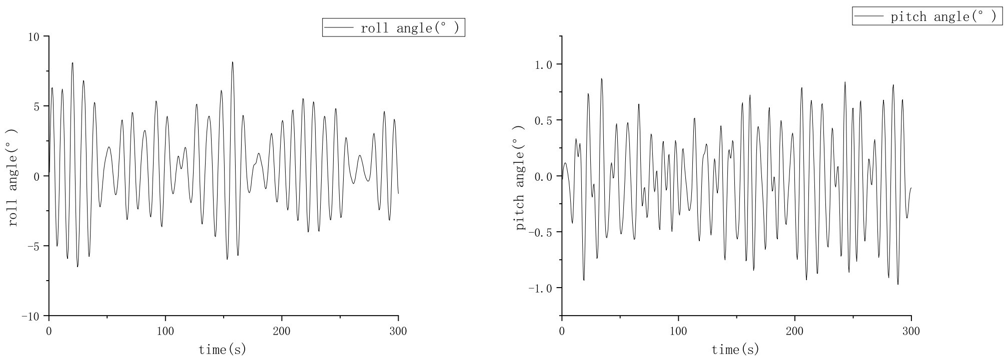


Figure 9. Dataset 4: The raw roll and pitch angle of the ship.

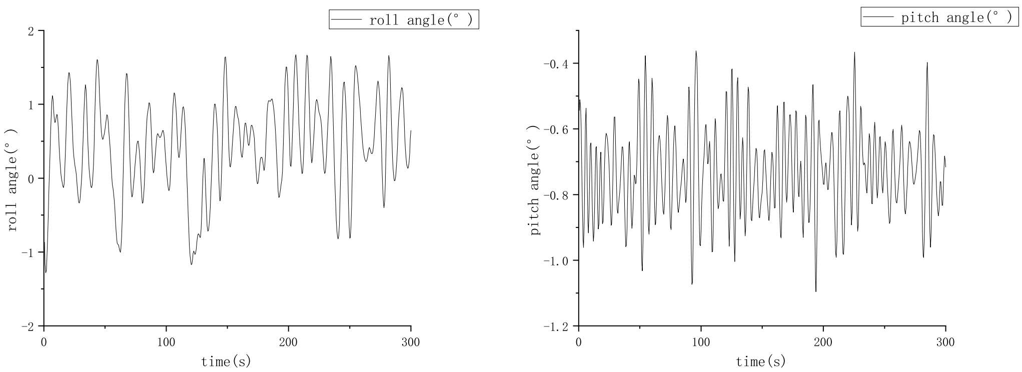


Figure 10. Dataset 5: The raw roll and pitch angle of the ship.

4.2. Prediction Performance Evaluation of Adaptive PSO Algorithm and Hybrid Model

To test the performance of the adaptive PSO algorithm, two standard test functions were chosen for both PSO and adaptive PSO. The two functions are expressed in Table 2.

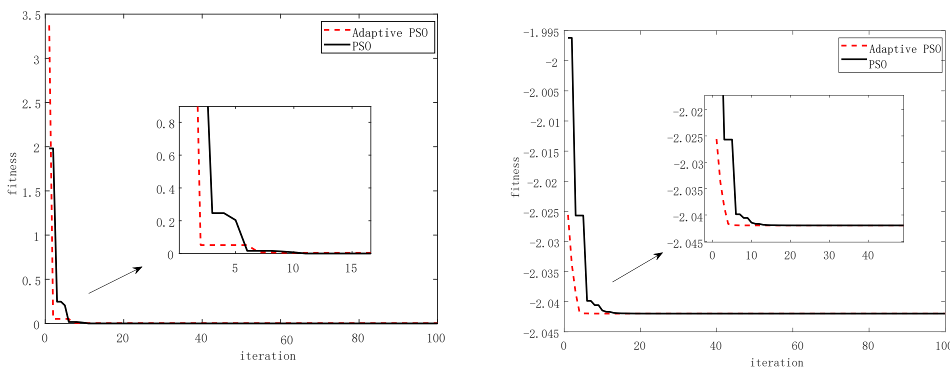
Table 2. Test functions to test the performance of the adaptive PSO algorithm.

Name	Test Function Expression
Rosenbrock	$f(x,y) = (1-x)^2 + 100(y-x^2)^2$
Griewank	$f(x_i) = \sum\limits_{i=1}^{N} \frac{x_i^2}{4000} - \prod\limits_{i=1}^{N} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$

The Griewank function has local minima, the number of which is related to the dimensionality of the problem, and the minimum is obtained at (0, 0 ... 0). It is a nonlinear multimodal function with a very wide search space and can be used to test particle swarm algorithms.

The Rosenbrock function is a nonconvex function used to test the performance of the optimization algorithm. Each contour of the Rosenbrock function is roughly parabolic in shape, and its full-domain minimum is also located in a parabolic-shaped valley (a banana-shaped valley). It is easy to find this valley, but it is quite difficult to find the minimum value of the full domain because the values within the valley do not vary much. Its full domain minimum is located at the point $(x,y) = (1,1)$ with the value $f(x,y) = 0$. Sometimes, the coefficient of the second term is different, but it does not affect the location of the full domain minimum.

The parameters in PSO and adaptive PSO were set as follows. The maximum population size was 100; the maximum number of iterations was 100; maximum particle velocity $V_{\max} = 5$; learning factors were set as $c_{\max} = 2.1, c_{\min} = 0.8$; and inertia weight was $w_{\max} = 0.9, w_{\min} = 0.4$, respectively. Figure 11 indicates the optimization results of the above test functions. It is obvious that the adaptive PSO algorithm showed better performance in finding the best individual fitness with less iteration and faster convergence.



(**a**) (**b**)

Figure 11. Optimization iteration results of different test functions: (**a**) Rosenbrock test function; (**b**) Griewank test function.

The ship motion varies greatly due to the environment at sea. Dense and highly fluctuating data may obscure the less fluctuating parts, resulting in missing details. Therefore, the data need to be normalized to eliminate the dimensional difference between different dataset inputs and scale them according to a certain ratio within the set interval. Both the sample set and target set should use the same normalization standard to ensure the consistency of neural network training and prediction. The formula of normalization is as follows:

$$X_N = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}}(b-a) + a, \tag{13}$$

where X_N is normalized data, $x_{\max}$ and $x_{\min}$ represent max and minimum of the data sequence x_i, respectively. Here, x_i can be set from a to b. It is set in the range from 0 to 1 in this paper.

Once the ship motion prediction is completed, it needs to be de-normalized, as indicated in Equation (14):

$$x_i = (x_{\max} - x_{\min}) * X_N + x_{\min}. \tag{14}$$

Mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean square error (RMSE) were utilized to evaluate the fitting performance of the established model between the true value and the prediction value, respectively. These three kinds of evaluation indexes are shown as

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |x_i - \hat{x}_i|, \tag{15}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n} \frac{|x_i - \hat{x}_i|}{x_i} \times 100\%, \tag{16}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (x_i - \hat{x}_i)^2}. \tag{17}$$

For the purpose of evaluating the prediction performance of this model, the difference between the target sequence and prediction sequence is presented to evaluate the performance of the hybrid model.

4.3. Roll Angle Prediction Results and Analysis

To ensure the fairness of the experiment, the initial parameters are set to remain the same, and the detailed model parameter settings are listed in Table 3. In order to prove the effectiveness of the hybrid prediction model proposed in this paper, adaptive PSO–LSTM and EMD–LSTM were selected as comparison models. Additionally, the BP neural network, ELM, LSTM, and its variant neural networks were selected to prove the effectiveness in short-term ship motion attitude prediction.

Table 3. Settings of different model parameters.

Model	Parameters	Values
BP	Number of Hidden Neurons	10
ELM	Transfer Function	sine function
	Number of Hidden Neurons	10
LSTM	Hidden Units	10
	MaxEpochs	250
	Initial learning rate	0.01
	numFeatures	1
	numResponse	1
	droprate	0.2
	activation function	sigmoid, tanh
	MiniBatchSize	128
	Optimizer	Adam
Bi-LSTM	Number of Hidden Neurons	10
	Activation function	sigmoid, tanh
SAE-LSTM	SparsityProportion	0.6
	Number of Hidden Neurons	10
Proposed	EMD decomposition	7Imfs + res.(take roll angle in dataset1 as an example)
	Sliding window length	10
	LSTM layer	the same as LSTM model

Taking Dataset 1 as an example, different benchmark models of roll angle prediction are shown in Figures 12 and 13. It is obvious that LSTM and its variant neural networks have better performance compared with the BP neural network and ELM, which proves the superiority of the deep learning method and that the LSTM neural network has higher accuracy. The errors between the predicted values and the recorded values for each method are shown in Figures 14 and 15. Additionally, the statistical results of errors in Dataset 1 are shown in Table 4, where LSTM, EMD–LSTM, and the adaptive PSO–LSTM method were selected to make comparisons with the proposed method to prove that EMD and adaptive PSO improved the LSTM neural network and had better adaptability. Although the input of the dataset is changed, the network structure can also be changed at the same time to achieve the best prediction performance. However, the above results are for one-step predictions. In practical engineering applications, short-term prediction of ship motion attitude generally requires multi-step-ahead predictions. This result means the prediction error ahead of different steps is based on the former observed value and current prediction value and has error accumulation, which is inevitable.

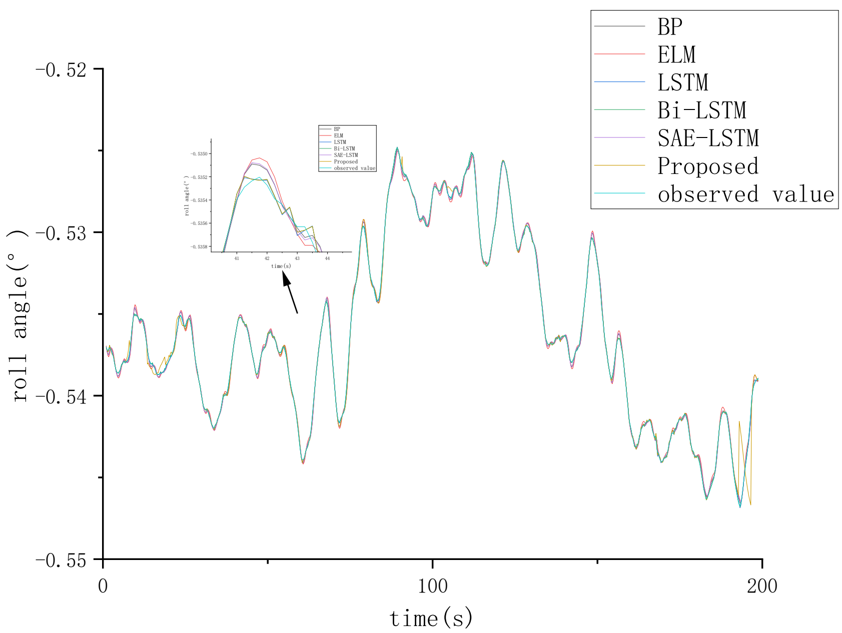


Figure 12. Roll angle prediction results in the training set of Dataset 1.

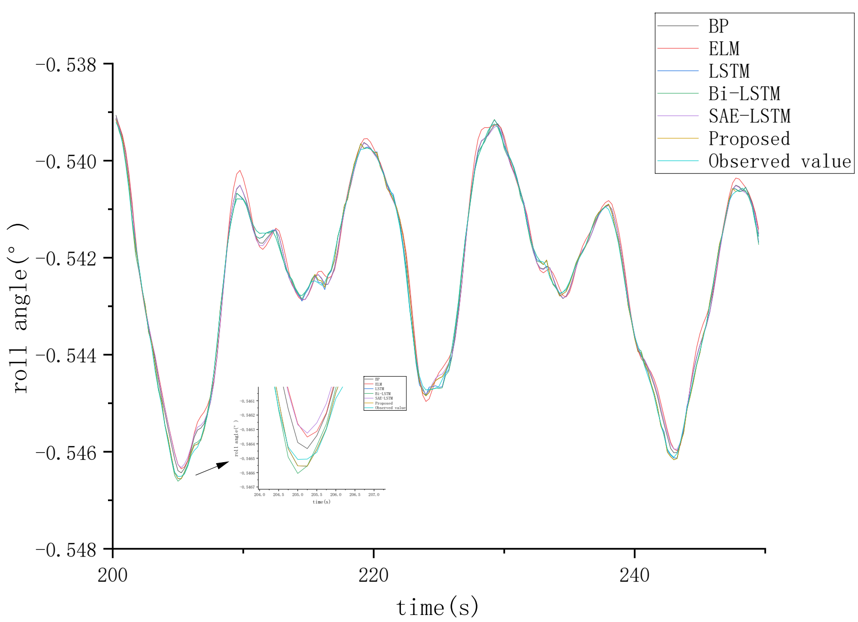


Figure 13. Roll angle prediction results in the testing set of Dataset 1.

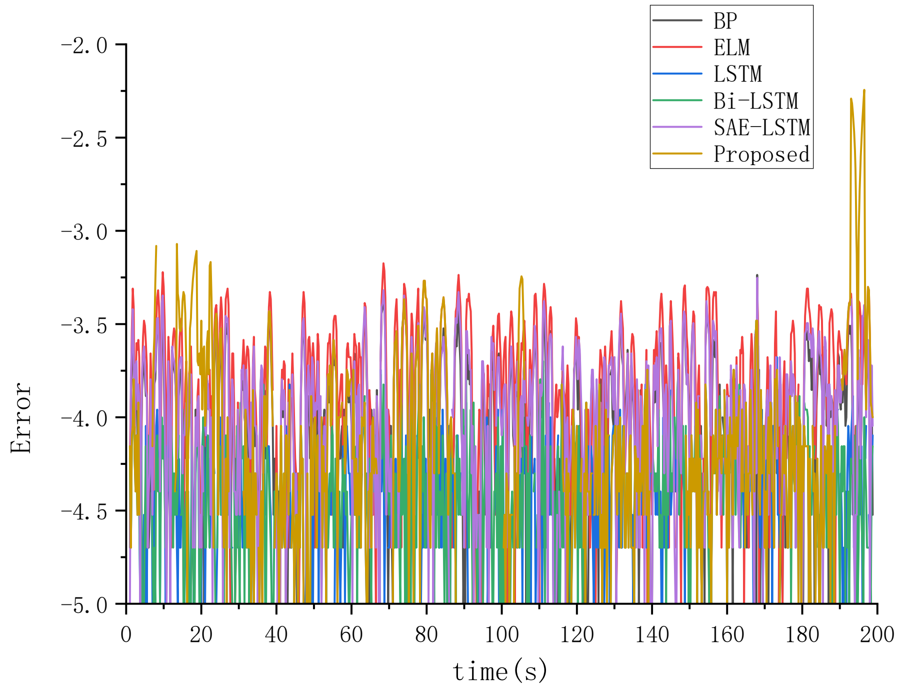


Figure 14. Roll angle prediction error results in the training set of Dataset 1.

In this paper, one-step , two-step, and three-step-ahead predictions were set aiming at all the datasets. The roll angle prediction errors of different models are shown in Table 5. As for dataset 1, ELM and the proposed method indicate the obvious error accumulation phenomenon, and the error result shows that almost all the datasets using the selected methods showed error accumulation as the number of predicted steps increases. Theoretically speaking, error accumulation is inevitable, but in practical prediction, not all neural networks demonstrate such situations, especially when using different datasets.

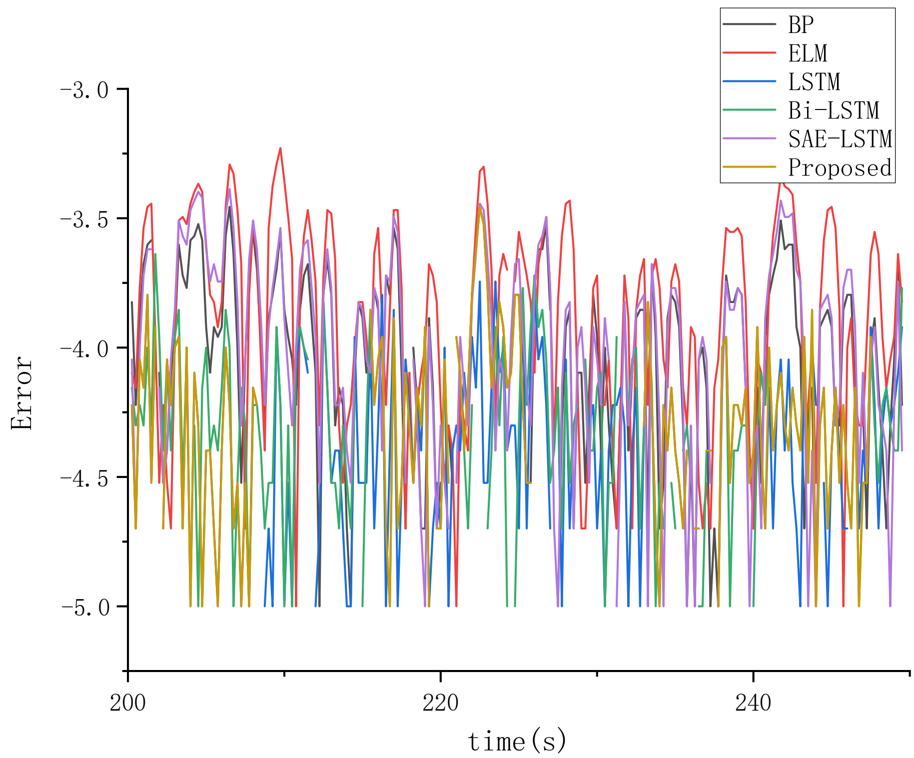


Figure 15. Roll angle prediction error results in the testing set of Dataset 1.

Table 4. Statistical results of roll angle prediction errors in Dataset 1.

Method	**MAE ($\times 10^{-5}$)**	**MAPE ($\times 10^{-6}$)**	RMSE ($\times 10^{-5}$)
LSTM	2.85	6.42	4.63
adaptive PSO–LSTM	2.51	4.60	2.79
EMD–LSTM	2.74	5.00	3.06
proposed	**1.40**	**2.60**	**1.62**

Table 5. The roll angle prediction errors of different models.

Dataset	**Model**	**RMSE ($\times 10^{-5}$)**			**MAE ($\times 10^{-5}$)**			**MAPE ($\times 10^{-5}$)**		
		1 Step	**2 Steps**	**3 Steps**	**1 Step**	**2 Steps**	**3 Steps**	**1 Step**	**2 Steps**	**3 Steps**
#1	BP	13.9865	10.9504	17.7120	11.4120	8.1201	14.2360	0.1063	0.0768	0.1345
	ELM	21.9596	30.6676	32.0620	17.7876	24.4303	25.5310	0.1657	0.2311	0.2414
	LSTM	6.5631	6.5619	6.8207	5.2209	5.2104	5.5277	0.0486	0.0492	0.0522
	Bi-LSTM	6.8805	18.9087	7.8066	5.4891	15.2301	6.0338	0.0511	0.1441	0.0570
	SAE-LSTM	16.2097	15.4307	17.7260	12.9579	12.5685	14.3710	0.1208	0.1189	0.1360
	Proposed	**1.6200**	**2.2500**	**2.8930**	**1.4000**	**3.8520**	**3.9050**	**0.0262**	**0.0254**	**0.0295**
#2	BP	24.1841	24.6001	28.3410	18.0727	19.9550	23.5170	0.1635	0.1831	0.4260
	ELM	21.0534	42.2065	77.0902	27.0664	33.9358	46.4172	0.17	0.3110	0.0753
	LSTM	24.0533	24.1604	**15.7080**	14.8452	17.3944	18.2150	0.1346	0.1573	0.2160
	Bi-LSTM	22.5563	25.7633	29.266	15.9178	17.0951	22.981	0.144	0.1597	0.2109
	SAE-LSTM	55.6759	26.9501	44.3218	18.7665	21.5600	35.3644	0.1638	0.1979	0.3240
	Proposed	**15.5211**	**18.426**	19.7256	**13.2542**	**14.9812**	**17.3158**	**0.02845**	**0.03249**	**0.03674**
#3	BP	16.2574	16.5112	17.2154	7.8379	8.4329	9.1245	1.9280	2.6220	2.6270
	ELM	15.3113	16.2548	18.7512	8.1729	8.2146	8.7512	1.9220	3.5140	3.2140
	LSTM	8.3712	9.2140	9.7451	6.1535	6.8421	7.1500	0.1006	0.1920	0.5320
	Bi-LSTM	11.3180	12.0541	12.9542	6.1678	6.9874	7.5413	0.1230	0.3610	0.6210
	SAE-LSTM	18.7051	19.0124	21.0244	6.6167	7.1593	8.1736	0.1830	0.6270	0.8240
	Proposed	**6.2451**	**7.6214**	**9.0215**	**5.2154**	**5.9241**	**6.8314**	**0.0980**	**0.1670**	**0.3450**
#4	BP	55.1250	55.8125	56.0124	21.2154	22.4375	23.4571	2.9853	3.0427	3.6125
	ELM	50.5120	51.0321	51.8742	23.6451	23.9461	24.6518	2.4127	2.4672	2.6523
	LSTM	37.0120	37.6200	38.3124	15.2243	16.0214	16.9572	1.1245	1.2451	1.3526
	Bi-LSTM	40.1520	42.0158	42.6547	16.2751	16.9542	17.9512	1.4267	1.5134	1.5142
	SAE-LSTM	47.0420	48.3219	47.0124	16.9585	17.9561	18.6412	1.3421	1.4000	1.4873
	Proposed	**25.0450**	**26.0127**	**27.6541**	**14.3214**	**14.9546**	**15.3125**	**0.8713**	**0.8971**	**1.0821**
#5	BP	51.017	52.4275	52.9781	22.8124	23.1245	23.9971	2.6452	2.9451	3.8451
	ELM	50.423	51.4379	53.1245	21.4648	22.9875	23.7815	2.1252	2.3215	2.4025
	LSTM	38.042	38.4512	39.4124	17.5145	17.4516	18.9155	1.4652	1.4824	1.6715
	Bi-LSTM	42.024	42.3721	43.7512	17.6891	17.9815	18.6785	1.3425	1.6541	1.7512
	SAE-LSTM	43.127	44.1289	45.0124	18.7512	18.7912	19.1544	0.9842	1.0245	1.6522
	Proposed	**32.045**	**32.7818**	**33.1587**	**15.4215**	**15.9754**	**16.3242**	**0.9421**	**1.0134**	**1.6452**

Figures 16 and 17 show the error distribution of one-step-ahead, two-step-ahead, and three-step-ahead predictions with different models in a more visual way. The ELM method and BP neural network had a higher error in the three-step-ahead prediction, which may be due to the simple network structures.

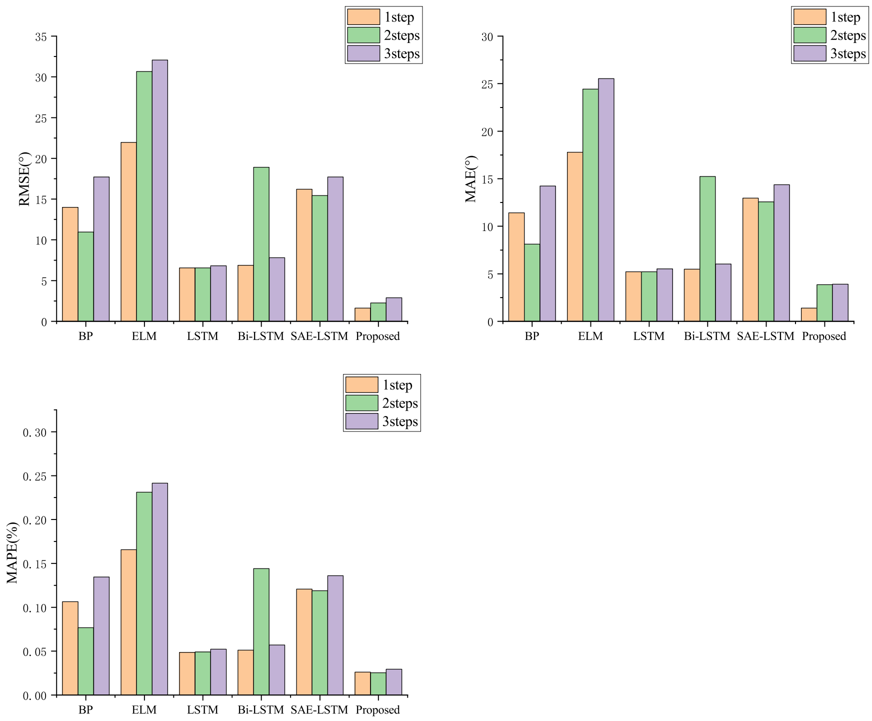


Figure 16. Roll angle prediction error distribution of Dataset 1.

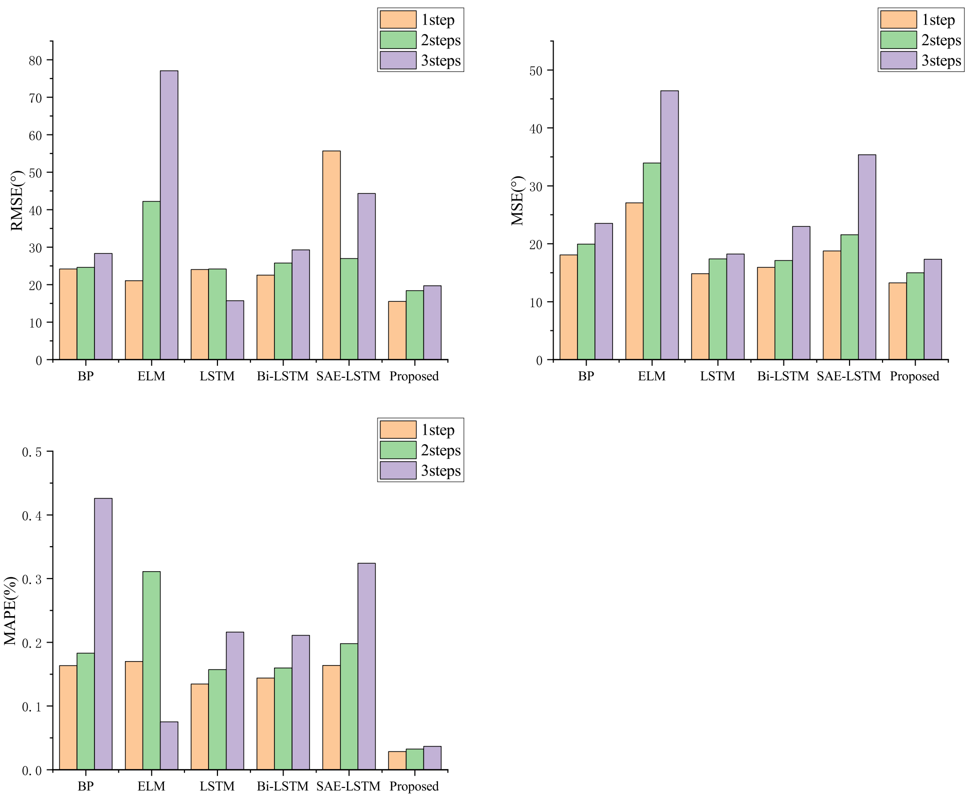


Figure 17. Roll angle prediction error distribution of Dataset 2.

4.4. Pitching Angle Prediction Results and Analysis

Taking Dataset 1 as an example, different benchmark models for pitching angle prediction results are shown in Figures 18 and 19. Figure 18 indicates prediction results in 200 s, and Figure 19 shows the testing data in 50 s. It can be seen that LSTM and BiLSTM had better fitting performance. In order to prove that EMD and adaptive PSO are both effective methods in improving LSTM for short-term ship attitude prediction, EMD–LSTM and adaptive PSO–LSTM were chosen for comparison, and the statistical results of errors are shown in Table 6. As a whole, adaptive PSO–LSTM had lower MAE and RMSE compared with EMD–LSTM, which means that adaptive PSO improves adaptability by adjusting network parameters dynamically and has more accurate results. Although the MAPE result is different from the MAE and RMSE results, the two method prediction results are similar and can be ignored. EMD is mainly used to reduce the nonstationary characteristic before dividing fixed window length time series data into training and testing datasets. The proposed model had the lowest error results and can further prove the validity of short-term prediction. The error results between the prediction and the recorded values for each method are shown in Figures 20 and 21. Similar to roll angle prediction experiments, pitching angle prediction of all the datasets is shown in Table 7, which shows that the proposed model outperforms the other methods in terms of one-step-ahead, two-step-ahead, and three-step-ahead prediction with the lowest error results. Still, the error accumulation is obvious.

Both Figures 22 and 23 show the error distribution of the one-step-ahead, two-step-ahead, and three-step-ahead predictions of pitching angle with different models in a more visual way, and MAPE distribution shows an increasing trend. Compared with the previous model, the BP neural network still had the higher error results in multi-step-ahead predictions. It is worth noting that the SAELSTM method also had poor prediction performance, which demonstrates that complex networks do not always result in better predictions. On the contrary, this may cause a long overall prediction time.

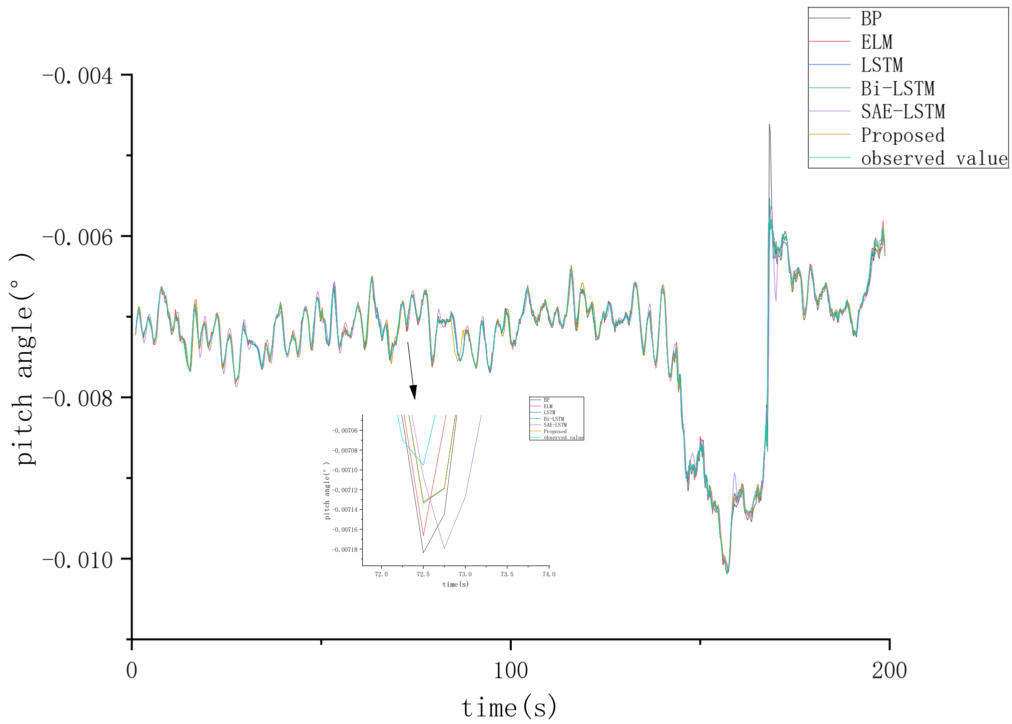


Figure 18. Pitch angle prediction results in the training set of Dataset 1.

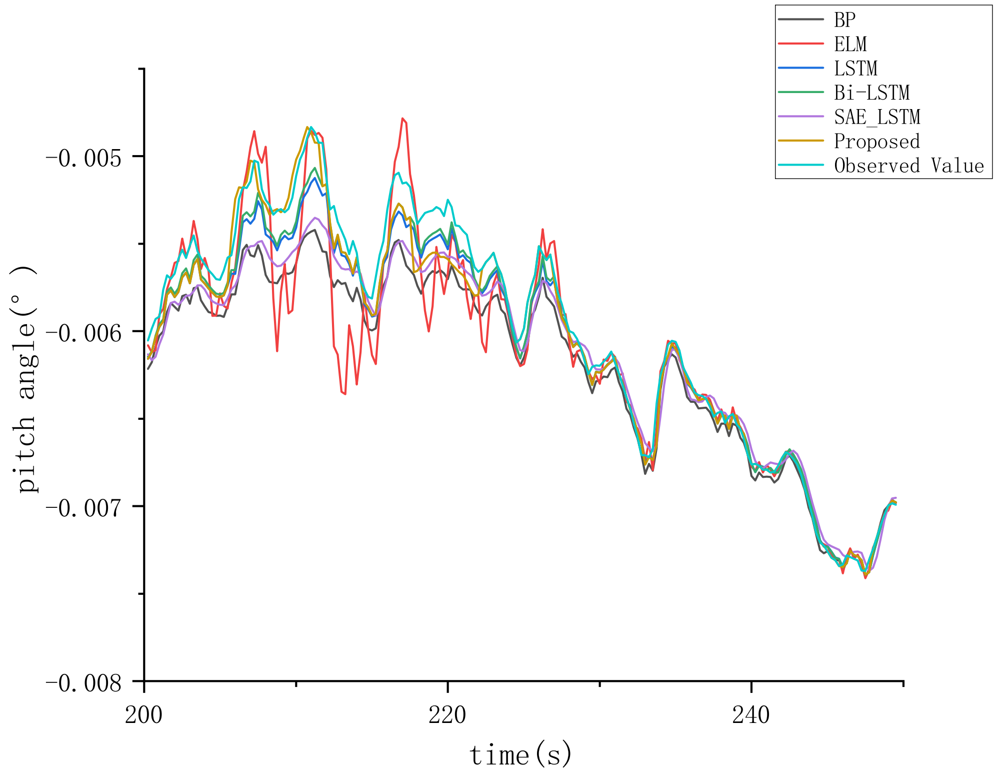


Figure 19. Pitch angle prediction results in the testing set of Dataset 1.

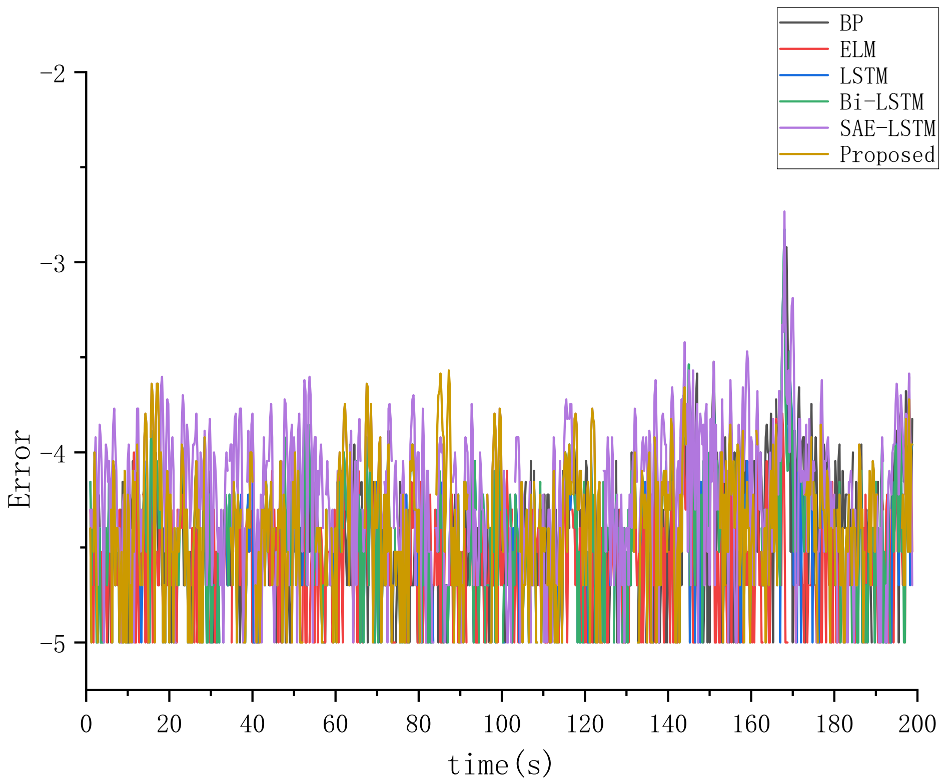


Figure 20. Pitch angle prediction error results in the training set of Dataset 1.

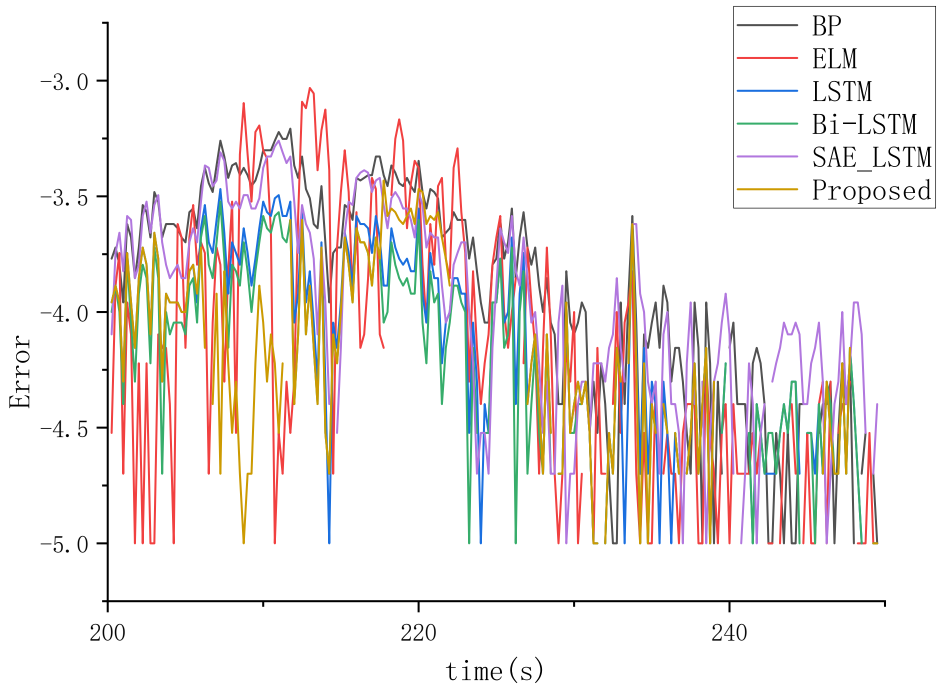


Figure 21. Pitch angle prediction error results in the testing set of Dataset 1.

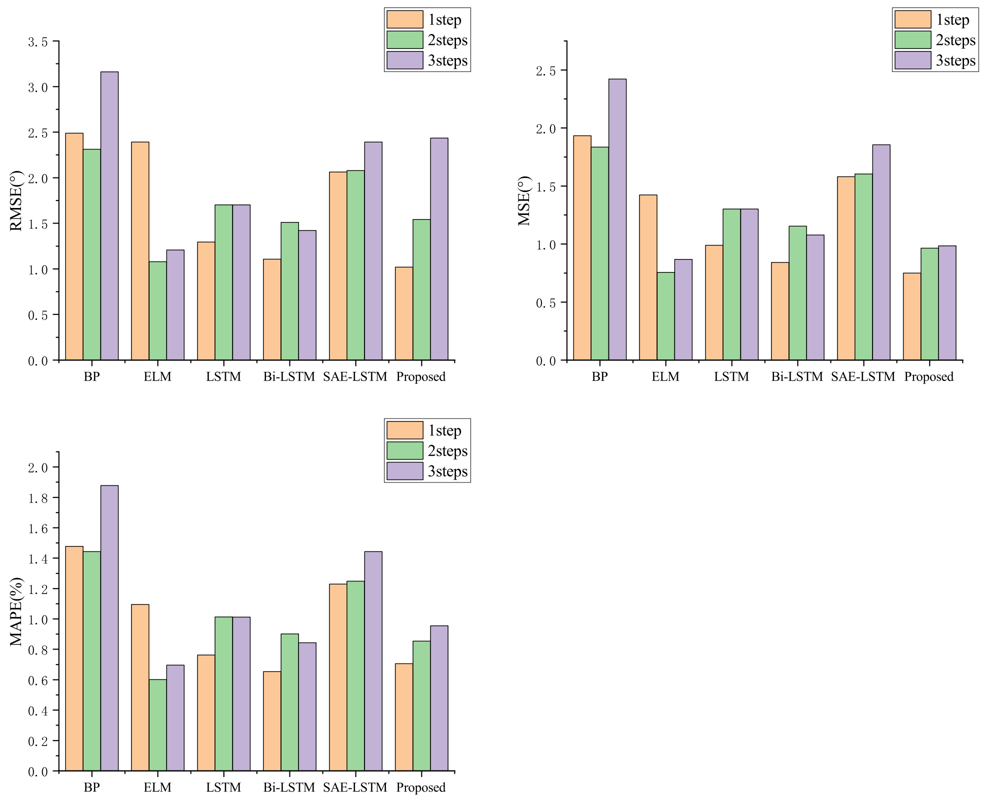


Figure 22. Pitch angle prediction error distribution of Dataset 1.

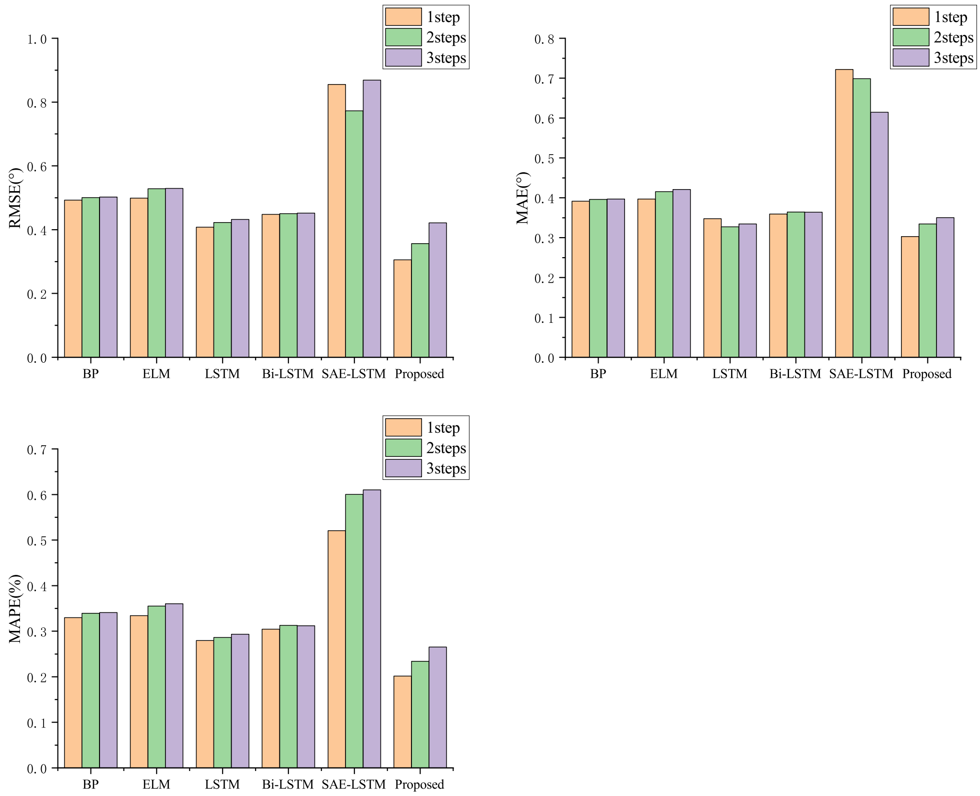


Figure 23. Pitch angle prediction error distribution of Dataset 2.

Table 6. Statistical results of pitch angle prediction errors in Dataset 1.

Method	MAE ($\times 10^{-5}$)	MAPE ($\times 10^{-5}$)	RMSE ($\times 10^{-5}$)
LSTM	2.36	1.54	3.76
adaptive PSO–LSTM	0.59	0.81	0.76
EMD–LSTM	1.44	**0.20**	1.82
proposed	**0.366**	0.49	**0.46**

Table 7. The pitch angle prediction errors of different models.

Dataset	Model	RMSE ($\times 10^{-4}$)			MAE ($\times 10^{-4}$)			MAPE ($\times 10^{-4}$)		
		1 Step	2 Steps	3 Steps	1 Step	2 Steps	3 Steps	1 Step	2 Steps	3 Steps
#1	BP	2.4880	2.3117	3.1614	1.9340	1.8354	2.4225	1.4779	1.4434	1.8784
	ELM	2.3914	**1.0785**	**1.2083**	1.4225	**0.7563**	**0.8680**	1.0949	**0.6007**	**0.6957**
	LSTM	1.2950	1.7018	1.7027	0.9889	1.3026	1.3012	0.7623	1.0137	1.0121
	Bi-LSTM	1.1073	1.5101	1.4217	0.8423	1.1547	1.0781	**0.6527**	0.9016	0.8431
	SAE-LSTM	2.0635	2.0784	2.3906	1.5800	1.6044	1.8553	1.2290	1.2487	1.4431
	Proposed	**1.0205**	1.5426	2.4352	**0.7502**	0.9642	0.9852	0.7052	0.8543	0.9548
#2	BP	0.4926	0.5004	0.5021	0.3916	0.396	0.3970	0.3299	0.3394	0.3409
	ELM	0.4986	0.5285	0.5295	0.3967	0.4153	0.4210	0.3342	0.3551	0.3604
	LSTM	0.4079	0.4224	0.4321	0.3474	**0.3272**	**0.3346**	0.2797	0.2861	0.2932
	Bi-LSTM	0.4478	0.4501	0.4521	0.3595	0.3644	0.3638	0.3043	0.3128	0.3118
	SAE-LSTM	0.8550	0.7729	0.869	0.7217	0.6985	0.6143	0.5207	0.6002	0.6103

Table 7. *Cont.*

Dataset	Model	RMSE ($\times 10^{-4}$)			MAE ($\times 10^{-4}$)			MAPE ($\times 10^{-4}$)		
		1 Step	2 Steps	3 Steps	1 Step	2 Steps	3 Steps	1 Step	2 Steps	3 Steps
	Proposed	**0.3052**	**0.3564**	**0.4215**	**0.3025**	0.3345	0.3501	**0.2015**	**0.2341**	**0.2654**
#3	BP	3.1247	3.4521	4.0421	2.3210	2.3954	2.9525	1.6421	1.7512	1.7834
	ELM	3.4541	3.6422	3.9875	2.0423	2.3212	2.6215	1.0455	1.4245	1.4625
	LSTM	2.3124	2.4215	2.5124	1.0425	1.3251	1.4236	0.6421	0.6678	0.8451
	Bi-LSTM	2.6452	2.7812	2.9815	1.3451	1.3642	1.4203	0.8451	0.8945	0.9451
	SAE-LSTM	2.4215	2.6451	2.9875	1.6254	1.6355	1.8533	0.8342	0.8643	0.8965
	Proposed	**1.4521**	**1.6751**	**1.7235**	**0.9345**	**1.0421**	**1.3425**	**0.6421**	**0.6512**	**0.7345**
#4	BP	7.1515	7.6421	8.4512	4.0451	4.3125	4.5421	2.6425	2.7562	2.7865
	ELM	7.1245	7.5245	7.6154	3.4512	3.6427	3.9854	2.4514	2.6452	2.7635
	LSTM	4.4512	4.6457	5.0421	2.4512	2.5634	2.8754	1.4516	1.4634	1.6552
	Bi-LSTM	4.6124	4.7815	4.9871	2.9451	3.0124	3.0125	1.6421	1.7542	1.7954
	SAE-LSTM	5.1544	5.3425	5.2154	2.8754	2.9845	3.0124	1.6784	1.9780	2.0451
	Proposed	**3.2451**	**3.6452**	**3.9875**	**1.9458**	**2.0143**	**2.1345**	**1.3451**	**1.3649**	**1.4535**
#5	BP	7.9146	8.1246	8.9578	4.0414	4.3215	4.9485	2.5861	2.6784	2.7625
	ELM	7.8124	8.0451	8.6524	4.6152	4.9781	5.0145	2.5463	2.6458	2.8451
	LSTM	5.6412	5.7215	5.9841	3.0154	3.4614	3.8452	1.6535	1.6854	1.7058
	Bi-LSTM	5.9842	6.2451	6.3758	3.1245	3.2481	3.4585	1.6784	1.6524	1.7815
	SAE-LSTM	6.5167	7.0124	7.6255	3.4625	3.6421	4.0458	1.6736	1.7544	1.8065
	Proposed	**4.5671**	**4.6875**	**4.9587**	**2.8451**	**2.9845**	**3.0142**	**1.1643**	**1.4268**	**1.3815**

5. Conclusions

To improve the predicted performance of roll and pitch angles in the short term, a novel hybrid EMD-adaptive PSO–LSTM model with the sliding window approach was proposed in this paper. Firstly, in order to solve the problem of nonlinear and nonstationary data affecting the prediction accuracy, an EMD-based denoising method was proposed to smooth the original ship motion time series data. Secondly, an adaptive PSO algorithm was proposed to optimize the number of hidden units in the LSTM neural network to improve the accuracy of ship motion attitude prediction. To properly evaluate the performance of the proposed model, five sets of sea trial data were used for verification. The experimental results show the superiority of the proposed algorithm.

Author Contributions: X.G.; Conceptualization, methodology, software, writing—original draft, Y.L.; supervision, formal analysis, mathematical check, writing—review and editing, Q.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the Foundation of National Defense Key Laboratory(2021-JCJQ-LB-066-12), National Natural Science Foundation of China (No.52271311), Heilongjiang Touyan Innovation Team Program.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author due to privacy restrictions.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Tang, G.; Wu, Y.; Li, C.; Wong, P.K.; Xiao, Z.; An, X. A novel wind speed interval prediction based on error prediction method. *IEEE Trans. Ind. Inform.* **2020**, *16*, 6806–6815. [CrossRef]
2. Ali, M.; Prasad, R. Significant wave height forecasting via an extreme learning machine model integrated with improved complete ensemble empirical mode decomposition. *Renew. Sustain. Energy Rev.* **2019**, *104*, 281–295. [CrossRef]

3. Wang, Z.; Zou, Z.; Soares, C.G. Identification of ship manoeuvring motion based on nu-support vector machine. *Ocean Eng.* **2019**, *183*, 270–281. [CrossRef]
4. Wei, W.W.S. Time series analysis. In *The Oxford Handbook of Quantitative Methods in Psychology*; Oxford Library: Oxford, UK, 2006; Volume 2.
5. Majnarić, D.; Šegota, S.B.; Lorencin, I.; Car, Z. Prediction of main particulars of container ships using artificial intelligence algorithms. *Ocean Eng.* **2022**, *265*, 112571. [CrossRef]
6. Kumari, P.; Toshniwal, D. Long short term memory–Convolutional neural network based deep hybrid approach for solar irradiance forecasting. *Appl. Energy* **2021**, *295*, 117061. [CrossRef]
7. Sarıca, B.; Eğrioxgxlu, E.; Barış, A. A new hybrid method for time series forecasting: Ar–anfis. *Neural Comput. Appl.* **2018**, *29*, 749–760. [CrossRef]
8. Khan, A.; Bil, C.; Marion, K.E. Ship motion prediction for launch and recovery of air vehicles. In Proceedings of the OCEANS 2005 MTS/IEEE, Washington, DC, USA, 17–23 September 2005; pp. 2795–2801.
9. Kaplan, P. A study of prediction techniques for aircraft carrier motions at sea. *J. Hydronautics* **1969**, *3*, 121–131. [CrossRef]
10. Fossen, T.I.; Perez, T. Kalman filtering for positioning and heading control of ships and offshore rigs. *Control. Syst. IEEE* **2009**, *29*, 32–46.
11. Wang, W.; Chau, K.; Xu, D.; Chen, X. Improving forecasting accuracy of annual runoff time series using arima based on eemd decomposition. *Water Resour. Manag.* **2015**, *29*, 2655–2675. [CrossRef]
12. ElMoaqet, H.; Tilbury, D.M.; Ramachandran, S.K. Multi-step ahead predictions for critical levels in physiological time series. *IEEE Trans. Cybern.* **2016**, *46*, 1704–1714. [CrossRef]
13. Liu, X.M.; Zhang, Y.Y.; Feng, X.Y.; Wang, Q.X. A novel method for hull's three dimensional deformation measurement. *Appl. Mech. Mater. Trans. Technol. Publ.* **2013**, *344*, 93–98. [CrossRef]
14. Xu, D.; Wang, Y.; Peng, P.; Beilun, S.; Deng, Z.; Guo, H. Real-time road traffic state prediction based on kernel-knn. *Transportmetrica* **2020**, *16*, 104–118. [CrossRef]
15. Xiao, X.; Duan, H.; Wen, J. A novel car-following inertia grey model and its application in forecasting short-term traffic flow. *Appl. Math.l Modell.* **2020**, *87*, 546–570. [CrossRef]
16. Yang, Y.; Tu, H.; Song, L.; Chen, L.; Xie, D.; Sun, J. Research on accurate prediction of the container ship resistance by rbfnn and other machine learning algorithms. *J. Mar. Sci. Eng.* **2021**, *9*, 376. [CrossRef]
17. Sun, Q.; Tian, Y.; Diao, M. Cooperative localization algorithm based on hybrid topology architecture for multiple mobile robot system. *IEEE Internet Things J.* **2018**, *2018*, 1. [CrossRef]
18. Sun, Q.; Tang, Z.; Gao, J.; Zhang, G. Short-term ship motion attitude prediction based on lstm and gpr. *Appl. Ocean Res.* **2022**, *118*, 118. [CrossRef]
19. Ye, F.; Chen, J.; Tian, Y.; Jiang, T. Cooperative multiple task assignment of heterogeneous uavs using a modified genetic algorithm with multi-type-gene chromosome encoding strategy. *J. Intell. Robot. Syst.* **2020**, *100*, 1–13. [CrossRef]
20. Babu, C.N.; Reddy, B.E. A moving-average filter based hybrid arima–ann model for forecasting time series data. *Appl. Soft Comput.* **2014**, *23*, 27–38. [CrossRef]
21. Moreira, L.; Soares, C.G.; Simulating ship manoeuvrability with artificial neural networks trained by a short noisy data set. *J. Mar. Sci. Eng.* **2023**, *11*, 1. [CrossRef]
22. Li, J.; Dai, Q.; Ye, R. A novel double incremental learning algorithm for time series prediction. *Neural Comput. Appl.* **2019**, *31*, 6055–6077. [CrossRef]
23. Dai, X.; Sheng, K.; Shu, F. Ship power load forecasting based on pso-svm. *Math. Biosci. Eng.* **2022**, *19*, 4547–4567. [CrossRef] [PubMed]
24. Saha, A.; Basu, S.; Datta, A. Random forests for spatially dependent data. *J. Am. Stat. Assoc.* **2021**, 1–19. [CrossRef]
25. Liu, H.; Tian, H.; Chen, C.; Li, Y. An experimental investigation of two wavelet-mlp hybrid frameworks for wind speed prediction using ga and pso optimization. *Int. J. Electr. Power Energy Syst.* **2013**, *52*, 161–173. [CrossRef]
26. Atiquzzaman, M.; Kandasamy, J. Robustness of extreme learning machine in the prediction of hydrological flow series. *Comput. Geosci.* **2018**, *120*, 105–114. [CrossRef]
27. Wang, H.; Liu, X.; Song, P.; Tu, X. Sensitive time series prediction using extreme learning machine. *Int. J. Mach. Learn. Cybern.* **2019**, *10*, 3371–3386. [CrossRef]
28. Huang, G.; Zhu, Q.; Siew, C. Extreme learning machine: Theory and applications. *Neurocomputing* **2006**, *70*, 489–501. [CrossRef]
29. Mahdi, K.; Wang, J. Spatio-temporal graph deep neural network for short-term wind speed forecasting. *IEEE Trans. Sustain. Energy* **2018**, *2*, 1.
30. Qin, S.-Q.; De, J.; Wu, W. A hybrid ar-dwt-emd model for the short-term prediction of nonlinear and nonstationary ship motion. In Proceedings of the IEEE 2016 Chinese Control and Decision Conference (CCDC), Yinchuan, China, 28–30 May 2016; pp. 4042–4047.
31. Veltcheva, A.; Soares, C.G. Analysis of wave-induced vertical ship responses by hilbert-huang transform method. *Ocean Eng.* **2023**, *269*, 113533. [CrossRef]
32. Wang, Y.; Wang, H.; Zou, D.; Fu, H. Ship roll prediction algorithm based on bi-lstm-tpa combined model. *J. Mar. Sci. Eng.* **2021**, *9*, 387. [CrossRef]

33. El Said, A.; El Jamiy, F.; Higgins, J.; Wild, B.; Desell, T. Optimizing long short-term memory recurrent neural networks using ant colony optimization to predict turbine engine vibration. *Appl. Soft Comput.* **2018**, *73*, 969–991. [CrossRef]
34. Peng, X.; Zhang, B.; Zhou, H. An improved particle swarm optimization algorithm applied to long short-term memory neural network for ship motion attitude prediction. *Trans. Inst. Meas. Control.* **2019**, *41*, 4462–4471. [CrossRef]
35. Qian, L.; Wang, W.; Chen, G.; Yu, M. A fetal electrocardiogram signal extraction method based on long short term memory network optimized by genetic algorithm. *J. Biomed. Eng.* **2021**, *38*, 257–267.
36. Yin, J.; Wang, N.; Perakis, A.N. A real-time sequential ship roll prediction scheme based on adaptive sliding data window. *IEEE Trans. Syst. Man Cybern. Syst.* **2017**, *48*, 2115–2125. [CrossRef]
37. Yao, Y.; Han, L.; Wang, J. Lstm-pso: Long short-term memory ship motion prediction based on particle swarm optimization. In Proceedings of the 2018 IEEE CSAA Guidance, Navigation and Control Conference (CGNCC), Xiamen, China, 10–12 August 2018; pp. 1–5.
38. Nie, Z.; Yuan, Y.; Xu, D.; Shen, F. Research on support vector regression model based on different kernels for short-term prediction of ship motion. In Proceedings of the IEEE 2019 12th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China, 14–15 December 2019; Volume 1, pp. 61–64.
39. Zhang, Z.; Yin, J.; Hu, J.; Liu, C. Ship rolling motion prediction and analysis based on grey pso-anfis model. *Sci. Technol. Eng.* **2016**, *16*, 124–129.
40. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]