Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 1835798, 12 pages
https://doi.org/10.1155/2022/1835798

Research Article

Predictive Control of the Mobile Robot under the Deep Long-Short Term Memory Neural Network Model

Lan Zheng

The School of Civil Engineering, Harbin University, Harbin 150086, China

Correspondence should be addressed to Lan Zheng; zhenglan@hrbu.edu.cn

Received 23 July 2022; Revised 24 August 2022; Accepted 30 August 2022; Published 21 September 2022

Academic Editor: Ning Cao

At present, there is a phenomenon of network data packet loss in the trajectory tracking control system, which will degrade or even destabilize the system's performance. Therefore, this work first explains the theory of the deep long-short term memory (LSTM) neural network model, the kinematic model of mobile robots, and the trajectory tracking error model. The reasons for data packet loss in the control system are analyzed. Second, a prediction model based on the LSTM network is designed according to the theory mentioned above. Finally, the training effect of the LSTM model and the robot trajectory tracking effect based on the model are tested by setting up simulation experiments. The research results are as follows: (1) The pose test error of the mobile robot will eventually tend to zero through the simulation curve generated by the pose parameters (x, y, θ) of the mobile robot. (2) The trajectory tracking error of the deep LSTM neural network prediction and compensation method with the packet loss rate of 5% is less than that with the packet loss rate of 10%. (3) The linear velocity v of the mobile robot based on the prediction model of the LSTM network varies greatly but is always in the interval (−2, 2). Its angular velocity ω initially fluctuates greatly but gradually tends to zero after about 13 s. (4) When the prediction model tracks the trajectory of the robot, the horizontal position x, the vertical position y, and the angle θ coincide with the reference trajectory. The exploration is conducted to provide a reference for the research on data packet loss in the networked mobile robot trajectory tracking system.

1. Introduction

1.1. Research Background and Motivations. Mobile robots can be divided into wheeled, bipedal, crawler, and crawling types according to different moving methods, and wheeled mobile robots are the most widely used [1]. The wheeled mobile robot is a typical nonholonomic mechanical system with nonlinear, underactuated, and drift-free characteristics, which brings great challenges to the research on its motion control. In particular, the trajectory tracking problem of mobile robots has become one of the key technologies [2]. Its trajectory tracking error system is usually a nonlinear strong coupling system, which does not meet the necessary conditions of Brockett. Mobile robots inevitably come into contact with the external environment during the movement process, and there are problems such as model uncertainty and external disturbances [3]. In addition, some scholars have proposed solutions to the problem of data packet loss in the networked mobile robot trajectory tracking system. For example, the virtual polling algorithm implemented at the application layer improves the network performance by reducing the degree of data conflict. However, this method does not consider the situation that the wireless channel will produce packet loss. Using the method of setting the transmission protocol to reduce the error caused by the unreliable wireless channel effectively suppresses the influence of the wireless channel, but does not compensate for the data packet loss caused by medium access [4]. Besides, there is a method to modify the controller output to improve the control performance by setting a gain schedule according to the current network conditions. Although it considers the influence of the wireless channel and medium access control, its statistical estimation algorithm for network data packet loss needs to be improved [5]. In general, most researchers are committed to optimizing existing network communication protocols, reducing the occurrence of network data packet loss and developing new network protocols to solve the impact of data packet loss on the performance of networked mobile robots [6]. Although these two methods can improve the system performance, the changes or innovations for network protocols have an enormous workload and high work difficulty. It is effective to analyze the impact of data packet loss in the existing network communication on the performance of the control system, improve and optimize the controller and propose a good algorithm to predict and compensate for data packet loss [7].

1.2. Research Objectives. According to the content, the specific framework is given as shown in Figure 1.

This paper aims to use the LSTM neural network model to achieve high-precision prediction and control of mobile robot trajectories, thereby reducing some of the problems existing in mobile robots. This paper provides a reference value for further research on motion trajectory control.

2. Literature Review

The research on mobile robots began in the late 1960s, and Tianfu Innovation Institute was the first to begin research on mobile robots. At present, mobile robots have developed into an important branch of robotics [8]. In the aerospace field, there are lunar rovers and Mars rovers. The emergence of various reconnaissance robots and patrol cars in the military field, sweeping robots in the field of daily life, and hotel food delivery robots fully demonstrate the broad development prospects and application value of mobile robots [9]. The working conditions of the mobile robot have strong uncertainty according to the working field compared with the robot fixed in a position. Meanwhile, the mobile robot has the characteristics of unstructured, so its performance requirements are high. It needs to have the walking function, perception of the outside world, and some specific functions set by humans [10]. Researchers should conduct in-depth research on environmental perception, dynamic programming, and sensory information fusion [11].

Regarding the trajectory prediction research of mobile robots, Islam et al. combined differential flatness characteristics and integral sliding mode control and considered the dynamic characteristics of wheel actuators, which could realize trajectory tracking control of single-wheel mobile robots [12]. Barzegar-Kalashani et al. used a complete sliding mode controller for the first time in the research on trajectory tracking control of a two-wheeled mobile robot, which could effectively reduce the influence of uncertain factors and achieve a good tracking effect [13]. Yang et al. combined a neural network with a sliding mode control algorithm to ensure the stability of neural network adaptation, and they obtained appropriate equivalent control when the parameters of the robot model were unknown. This method could ensure that the output tracking error converged to zero [14]. Liang et al. designed an adaptive trajectory tracking control algorithm based on the kinematic error model of mobile robots according to the shortcomings of traditional trajectory tracking control laws. They verified that the algorithm could track the reference trajectory at a fast speed and has an excellent tracking control effect through experiments [15].

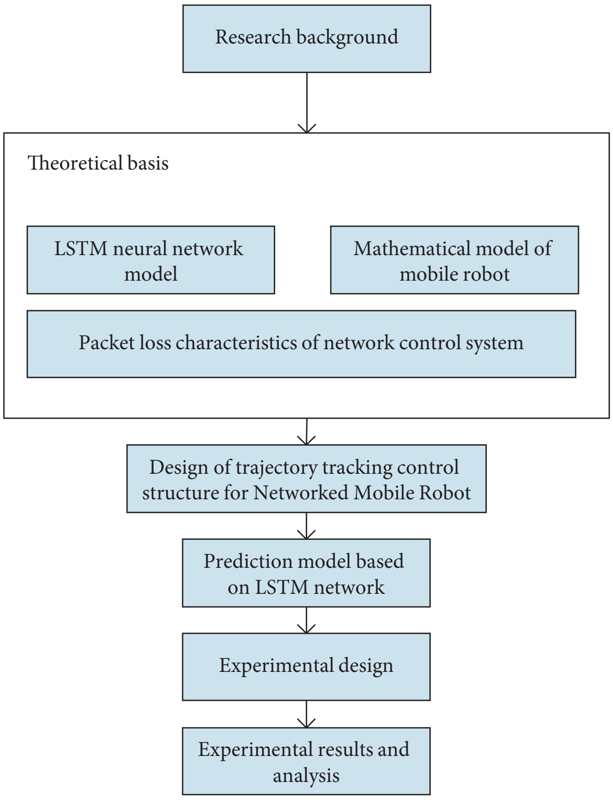

Research background
Theoretical basis
LSTM neural network model
Mathematical model of mobile robot
Packet loss characteristics of network control system
Design of trajectory tracking control structure for Networked Mobile Robot
Prediction model based on LSTM network
Experimental design
Experimental results and analysis

Figure 1: Structure of the work.

It is found that although the research on the trajectory tracking control of networked mobile robots with data packet loss has been developed, there are still many deficiencies by sorting out the literature. On the one hand, in the process of trajectory tracking control of mobile robots, the two sub-problems of motion planning and tracking control are usually solved separately. This results in that the external constraints are not considered when the reference trajectory of the mobile robot is given. The robot cannot reach the given input at some sampling moments, so it cannot achieve effective trajectory tracking. On the other hand, most of the current methods for improving the tracking performance of networked mobile robots are the development or optimization of network communication protocols with a heavy workload and high difficulty. It is urgent to propose an effective packet loss prediction compensation algorithm. The innovation lies in proposing a new prediction model and setting simulation experiments based on the original research on robot trajectory control. The new model is based on a long short-term memory (LSTM) neural network, which is optimized and integrated into robot trajectory prediction control. This work has a reference value in the research of robot trajectory prediction control.

3. Research Methodology

3.1. LSTM Neural Network Model

3.1.1. The Concept of the LSTM Model. With the continuous development of deep learning, the types of network architectures are also increasing, which are mainly divided into two categories: deep discriminative models and deep generative models [16]. This work uses deep learning for network data packet loss prediction compensation, so the deep LSTM neural network in the deep discriminant model is adopted. The LSTM unit is a variant of the most widely used recurrent neural network (RNN). It inherits the characteristics of most of the RNN models and solves the problem that RNN is difficult to train due to long-term dependencies [17]. LSTM units realistically represent or simulate the cognitive processes of human behavior, logical development, and neural organization. It is suitable for handling problems that are highly related to time series, such as machine translation, dialogue generation, encoding, and decoding [18]. The LSTM unit structure is shown in Figure 2.

Figure 2 shows the overall research framework. For LSTM, the model cell adds a cell state to store the long-term state, so it can effectively solve the long-term dependency problem. The key to the LSTM unit is how to effectively control the cell state. Three gates are added: input gate, forget gate, and output gate. The essence of the gate is a fully connected layer, the input is a vector, and the output is a real vector between zero and one. The elements in the gate's output vector are in turn multiplied by the vector to be controlled. When the output of the gate equals zero, its product with any vector is zero. At this point, no information can pass through. When the output of the gate is equal to one, its product with any vector is unchanged. At this point, any information can pass [19]. LSTM cells control the cell state through input gates and forget gates. The input gate determines how much of the current input is stored in the cell. The forget gate determines how much of the cell state at the previous moment is retained in the current cell state. The output gate determines how many cell states are passed to the current output of the network [20].

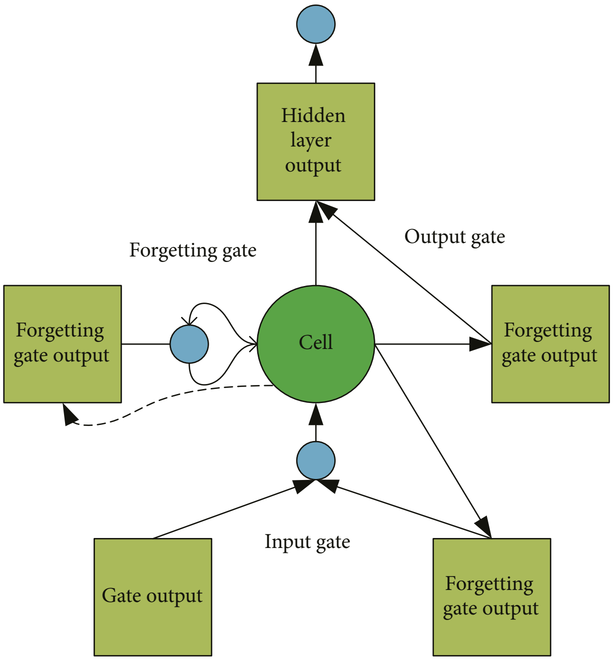


Figure 2: Schematic diagram of the LSTM unit structure.

3.1.2. Features of the LSTM Model.

In the process of training the network model using the gradient descent method, the traditional RNN's weight update strategy is the correct direction based on the weight at the end of the output sequence. The weight changes only depend on the sequence input at the most recent moments, and the input data from a long time ago has little effect. Besides, the network has poor long-term memory function, and the training results are inclined to new input information [21].

During the training process of the LSTM model, the stochastic gradient descent method is used to optimize the network parameters. This is the standard optimization method recently adopted to train deep neural networks, and it is easy to implement [22]. At each iteration, the objective function is run on a subset of the training set, which is called the mini-batch. The gradient of the mini-batch objective function is obtained by back-propagating the neural network parameters. The step size is determined by the average value of the past step size and the current gradient, and the weight of the past step size is determined by the momentum hyperparameter. The parameters (weights and biases) are updated according to the new step size and scaled by the learning rate. The new mini-batch is randomly determined from the training set and used for the next iteration. The learning rate is decreased during optimization [23].

3.2. Mathematical Model of the Mobile Robot

3.2.1. Kinematics Model of the Mobile Robot.

The kinematic model of a mobile robot directly reflects the relationship between its pose state and control input, and it is the most intuitive mathematical model [24]. The research object is a typical two-wheel differential mobile robot. This kind of mobile robot is composed of a balance wheel and two driving wheels, and the mobile robot realizes the rotation of the car body through the differential speed of the two driving wheels. The balance wheel only has a supporting function and cannot provide power for the mobile robot. Its specific structure is displayed in Figure 3.

According to the analysis of the collected literature, the balance wheel does not provide power, so the balance wheel can be ignored when the kinematic model equation of the mobile robot is derived. Besides, its number will not affect the kinematic equation form. The kinematic model of the two-wheel differential mobile robot can be expressed as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}, \tag{1}$$

(x, y) is the center position coordinate of the mobile robot. θ is the direction angle, and it is also the angle between the forward direction and the x-axis. v and ω are control input signals in the kinematic model, and their physical meanings are the linear and angular velocities of the mobile robot [25].

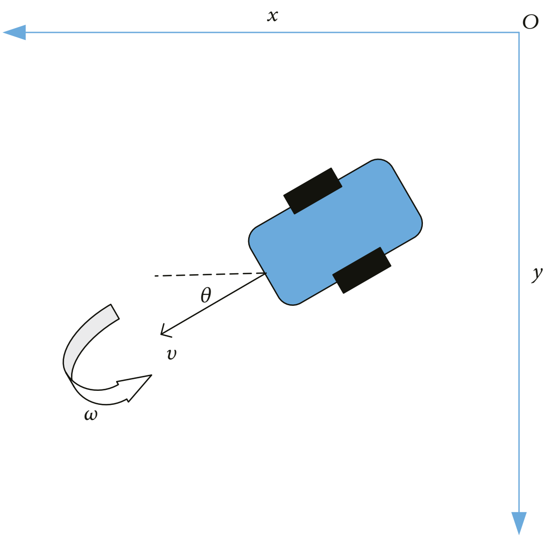


FIGURE 3: Schematic diagram of a two-wheel differential mobile robot.

3.2.2. Trajectory Tracking Error Model of the Mobile Robot.

The mobile robot task trajectory tracking error model established here is demonstrated in Figure 4.

In Figure 4, point A (x_a, y_a) is the center position of the target mobile robot. θ_a is the direction angle. They together constitute the pose state vector of the target mobile robot $[x_a, y_a, \theta_a]^T$. Point B (x_b, y_b) is the center position of the reference mobile robot. θ_b is the direction angle. The pose state vector is $[x_b, y_b, \theta_b]^T$. At this time, the pose error between the target robot and the reference robot can be expressed as

$$\begin{bmatrix} x_e \\ y_e \\ \theta_e \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_b - x_a \\ y_b - y_a \\ \theta_b - \theta_a \end{bmatrix}. \quad (2)$$

Then, equation (3) can be acquired as

$$\begin{cases} x_e = (x_b - x_a)\cos\theta + (y_b - y_a)\sin\theta, \\ y_e = -(x_b - x_a)\sin\theta + (y_b - y_a)\cos\theta, \\ \theta_e = (\theta_b - \theta_a). \end{cases} \quad (3)$$

The trajectory tracking error model of the mobile robot can be obtained by derivation of (3).

$$\begin{bmatrix} \dot{x}_e \\ \dot{y}_e \\ \dot{\theta}_e \end{bmatrix} = \begin{bmatrix} v_b \cos\theta_e - v + y_e\omega \\ v_b \sin\theta_e - x_e\omega \\ \omega_b - \omega \end{bmatrix}. \quad (4)$$

The trajectory tracking control problem of the mobile robot can be transformed into the stabilization problem of the trajectory tracking error model shown in (4) based on the new pose error state vector $[x_e, y_e, \theta_e]^T$. All the items in the trajectory tracking error vector $[x_e, y_e, \theta_e]^T$ tend to be zero by designing appropriate v and ω control laws. When the system reaches $[x_e, y_e, \theta_e] = [0, 0, 0]$, $x_b = x_a$, $y_b = y_a$, and $\theta_b = \theta_a$, the mobile robot trajectory tracking control task is completed [26].

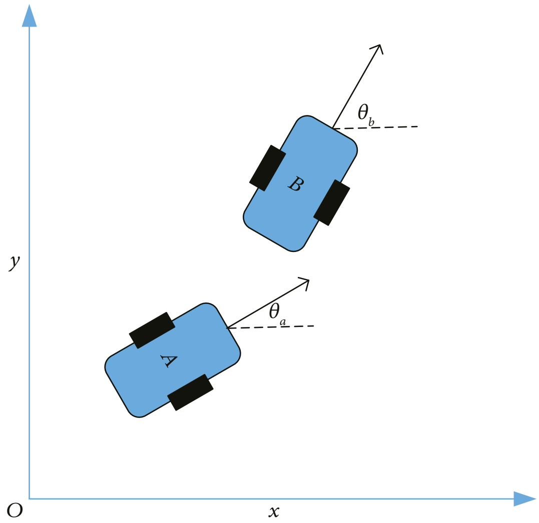


FIGURE 4: Schematic diagram of pose coordinate error.

3.3. Packet Loss Characteristics of the Network Control System

3.3.1. Reasons for packet loss.

In a network control system, each network node is frequently exchanging information. During network transmission, data collision or competition between nodes may occur, which will result in data loss during transmission. The network protocol has a retransmission mechanism to solve the problem of packet loss, but packet retransmission timeout may also occur when the system load is too heavy and there are many nodes in the network for data exchange. Packet loss can still happen. However, the phenomenon of packet loss sometimes occurs in the practical application of network control systems, which will reduce the system performance. Different network control systems will set a range of packet loss thresholds. Once the packet loss rate exceeds the threshold, the system will oscillate erratically, resulting in packet loss during network transmission [27]. There are three main reasons for packet loss.

A network node in the system has a communication failure. When the processor of a node in the network fails, the buffer where the data packet is located will be emptied, resulting in packet loss.

The packet transmission task in the system is too frequent. When the system communicates frequently, data conflicts inevitably occur. At this time, each node in the network competes for the right to use the network bandwidth, and the loss of data packets will occur. Although there is a retransmission mechanism, the destination node usually directly discards the packets that have not been transmitted beyond the retransmission time threshold.

There is channel interference in the network. The external environmental factors that the actual system is exposed to affect the transmission quality of the data packets in the channel. Channel interference in the network may cause disorder or loss of physical signals, resulting in distortion of data packets after reaching the destination node. At this time, valid data cannot be recovered through corresponding algorithms, and data packets are lost [28].

3.3.2. Description and Analysis of Network Packet Loss Characteristics. At present, the following three methods are mainly used to describe the data packet loss characteristics of the network control system. The first is statistical methods. For example, the probability distribution and the packet loss rate of data packets are assumed by using random system theory and switching system theory. The second is the theory of variable-delay systems. The total number of data packet losses between two sampling times needs to be given. The third is the theory of switched systems or predictive control theory. The sampling moment of packet loss in the system is regarded as the disconnection of the network transmission channel. A switching system with dynamic switches is used to represent the network control system structure of packet loss [29].

In the network control system, serial communication is used for information transmission among the controller, the controlled object, and the sensor. Each node device in the system shares the network channel bandwidth. In the case of frequent system communication, errors occur in the process of data packet transmission or reception due to the competition of each node for the right to use the network bandwidth. When the data packet does not reach the receiver within the specified time, it is called packet loss [30]. When data loss occurs in the system, the system information transmission channel is temporarily disconnected. Valid information at some sampling moments is not transmitted to the receiver in time. This will directly affect the structure and parameters of the system, causing system performance degradation. For the problem of data packet loss, although most network control systems have robustness, it is only limited to the number of data packet losses within the allowable threshold range. If this threshold is exceeded, it will seriously affect the control performance of the system and even make the system unstable [31]. Therefore, Figure 5 shows the structure obtained by simplifying the network control system with packet loss.

The switch can be opened and closed, and its state indicates whether there is a packet loss. When the switch is closed (S_2, S_4), the data packet at the sampling moment can be transmitted smoothly. When the switch is off (S_1, S_3), the data packet is lost at the sampling moment. At this time, the transmission size at the previous moment is regarded as the packet loss data at this moment or the data at this moment is directly set to zero, which are called the keep-input strategy and the zero-input strategy, respectively. The keep-input strategy is widely used, and its mathematical expression is as follows.

When the switch is located at S_1 and S_3, there i

$$\begin{cases} \overline{u}(k) = u(k-1), \\ \overline{x}(k) = x(k-1). \end{cases} \tag{5}$$

When the switch is located at S_1 and S_4, there is

$$\begin{cases} \overline{u}(k) = u(k-1), \\ \overline{x}(k) = x(k). \end{cases} \tag{6}$$

When the switch is located at S_2 and S_3, there is

$$\begin{cases} \overline{u}(k) = u(k), \\ \overline{x}(k) = x(k-1). \end{cases} \tag{7}$$

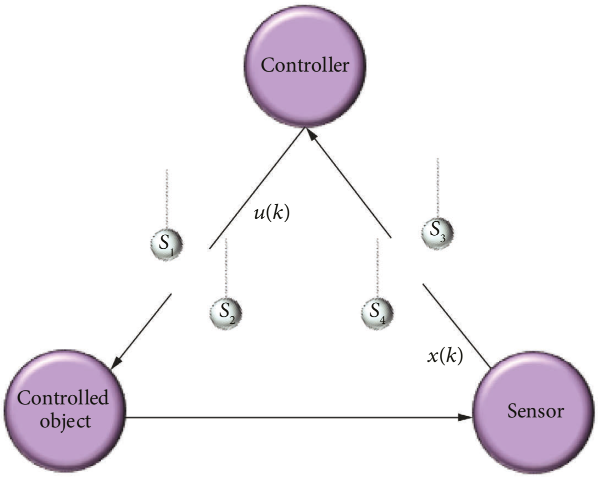


Figure 5: Simplified structure of the network control system.

When the switch is located at S_2 and S_4, there is

$$\begin{cases} \overline{u}(k) = u(k), \\ \overline{x}(k) = x(k), \end{cases} \tag{8}$$

$\overline{x}(k)$ represents the input value, and $\overline{u}(k)$ represents the output value. Besides, a new variable$z(k)$ is defined as

$$z(k) = \begin{bmatrix} x(k) \\ \overline{x}(k) \\ \overline{u}(k) \end{bmatrix}. \tag{9}$$

The network control system model with data packet loss can be expressed as

$$z(k+1) = \Phi_s z(k). \tag{10}$$

However, only adopting the keep-input strategy and the zero-input strategy to compensate for the data packet loss of the network control system cannot solve the problem of system performance degradation or even system instability caused by data packet loss. Reasonable methods need to be proposed to compensate for the negative impact of packet loss. Some scholars have proposed the method of predictive control to realize the prediction of missing data, and they use the predicted value obtained by this method to replace the lost control input value. The main problem with the predictive control method is the need to define extended vectors to assist in the design of the closed-loop system controller. The introduction of the extended vector increases the conservatism of the entire closed-loop control system, so it remains to find a good predictive compensation method to overcome this adverse effect.

3.4. Trajectory Tracking Control of the Networked Mobile Robot

3.4.1. Problem Description. The model structure of the networked mobile robot control system with the data packet loss problem studied here is shown in Figure 6.

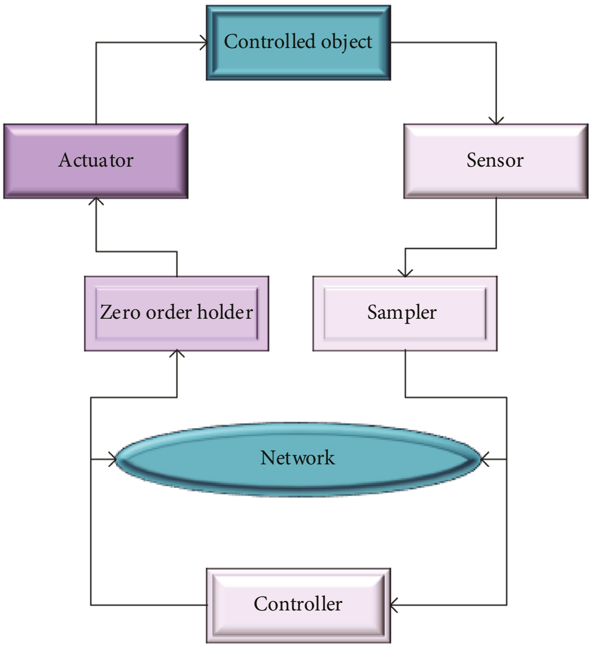


Figure 6: Network control system model with packet loss.

In this model, only the impact of packet loss on the system performance is considered during network transmission, and the sensor and controller are assumed to be time-driven and event-driven, respectively.

3.4.2. System Model. Based on the above analysis of the system packet loss characteristics and causes, the following assumptions are made to ensure the enforceability of the packet loss prediction and compensation method proposed here.

In the control loop, the sensor sends data packets to the controller with a constant sampling period. In practical industrial design and implementation, a time-driven approach is usually used to ensure a constant sampling period [32–37].

The sensors, controllers, and controlled objects in the control loop are time-synchronized through time-stamping technology to deal with the problem of possible out-of-order data packets [38–42].

In each sampling period, the deep neural network makes predictions for a single data packet obtained by the sensor and sends this data packet to the controller [43–48].

Based on the above assumptions, the structure of the networked mobile robot trajectory tracking control system under the deep neural network model reported here is revealed in Figure 7.

In Figure 7, $r(k)$ represents the reference pose of the mobile robot obtained by the partial point-taking method. $x(k)$ represents the actual motion pose of the mobile robot measured by the sensor. $e(k)$ represents the deviation between the actual pose and the reference pose of the mobile robot. This deviation serves as an input to the controller. $u(k)$ is the control amount of the mobile robot, including the linear velocity v and the angular velocity ω. $W(k)$ is the interference noise of the external environment. If no data packet loss occurs at the current sampling time, S_1 and S_2 are closed, and S_3 is disconnected. If data packet loss occurs at the current sampling time, S_1 and S_2 are disconnected, and S_3 is closed [49–54].

3.5. Prediction Model Based on the LSTM Network. According to the above LSTM model, the mathematical model of the mobile robot, and the analysis of the data packet loss characteristics, the prediction model based on the LSTM network proposed here is shown in Figure 8.

The mathematical description of the LSTM unit is as follows:

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right), \tag{11}$$

$$i_t = \sigma\left(W_i x_t + U_i h_{t-1} + b_i\right), \tag{12}$$

$$o_t = \sigma\left(W_o x_t + U_o h_{t-1} + b_o\right), \tag{13}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tan h\left(W_c x_t + U_c h_{t-1} + b_c\right), \tag{14}$$

$$h_t = o_t \odot \tan h\ \left(c_t\right). \tag{15}$$

In Equations (11)–(15), x_t is the input vector at time t, f_t, i_t, o_t, c_t, and h_t represent forget gate, input gate, output gate, cell state, and hidden layer output, respectively. W_f, W_i, W_o, and W_c are input weights. U_f, U_i, U_o, and U_c are cycle weights. b_f, b_i, b_o, and b_c are biased values. σ and tanh are the *Sigmoid* and tanh activation functions, respectively. $\odot$ represents the dot multiplication operation.

4. Experimental Design and Performance Evaluation

4.1. Datasets Collection. This work will use the deep LSTM neural network toolbox is used to calculate the unmodeled parameters of the model, and 1100 groups of data with two inputs and three outputs are randomly generated. The first 1000 groups are used to train the neural network, and the last 100 groups are used to test the modeling error. The whole dataset comes from the output of the deep LSTM neural network toolbox.

4.2. Experimental Environment. To verify the prediction model of robot trajectory tracking error based on the LSTM neural network without different packet loss rates, this paper builds and simulates the model by using the toolbox in MatLab application.

4.3. Parameters Setting. In the process of modeling, the training goal of the model is set to 0.001, and other parameters are the default values.

4.4. Performance Evaluation

4.4.1. LSTM Network Model Training Effect. The prediction model based on the LSTM network is trained by setting up simulation experiments. Figure 9 reveals the results.

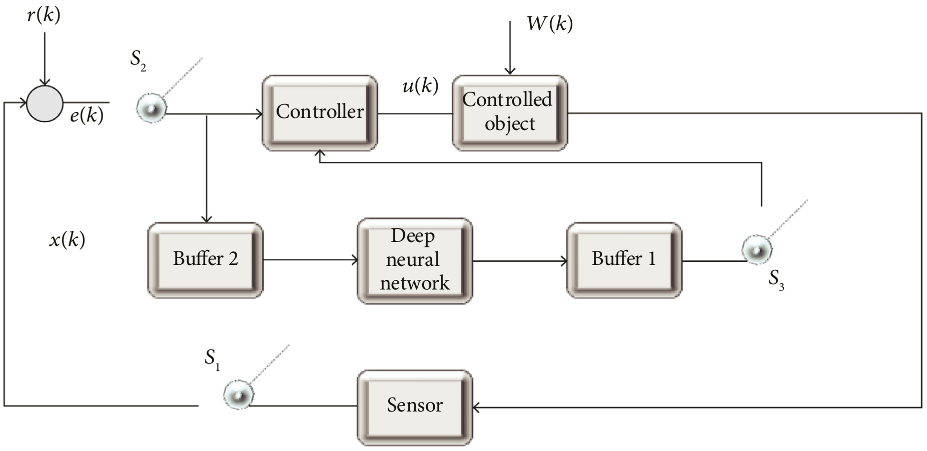


Figure 7: Structure of the mobile robot trajectory tracking control system based on the deep neural network.

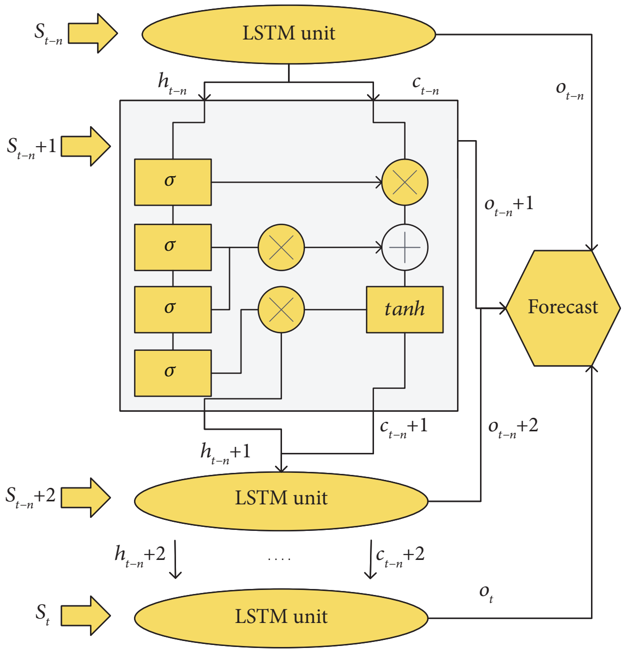


Figure 8: Mobile robot trajectory prediction model based on the LSTM network.

The meaning of the upper coordinate axis in Figure 9 refers to the serial number of the training samples collected in the experiment. In Figure 9, three lines with different colors represent the pose parameters (x, y, θ) of the mobile robot. The simulation curves generated by these three variables show that the pose test error of the mobile robot will eventually tend to zero. Therefore, it can be determined that the trained LSTM neural network can approximate the mobile robot model infinitely. The accuracy and efficiency are also high, which lays a solid foundation for the trajectory tracking controller to have a good tracking performance.

4.4.2. Robot Trajectory Tracking Control Effect Based on the LSTM Model. (1) Robot Trajectory Tracking Error Under Different Packet Loss Rates. The trajectory tracking error results of the prediction model based on the LSTM model under different packet loss rates are shown in Figure 10.

Figure 10 indicates that the trajectory tracking error of the deep LSTM neural network prediction compensation method when the packet loss rate is 5% is smaller than that of the deep LSTM neural network prediction compensation method when the packet loss rate is 10%. Generally, this method can complete the corresponding trajectory tracking task.

(2) Predictive Control Effect Of the Robot Based on the LSTM Network. Figure 11 shows the result of the change of control amount when the robot based on the LSTM network moves through the simulation experiment.

Figure 11 reveals that the linear velocity v of the mobile robot based on the prediction model of the LSTM network varies greatly but is always in the interval $(-2, 2)$. The angular velocity ω initially fluctuates greatly but gradually tends to zero after about 13 s. In conclusion, the control increment changes of the mobile robot under this model are stable. The prediction effect of the model is further verified.

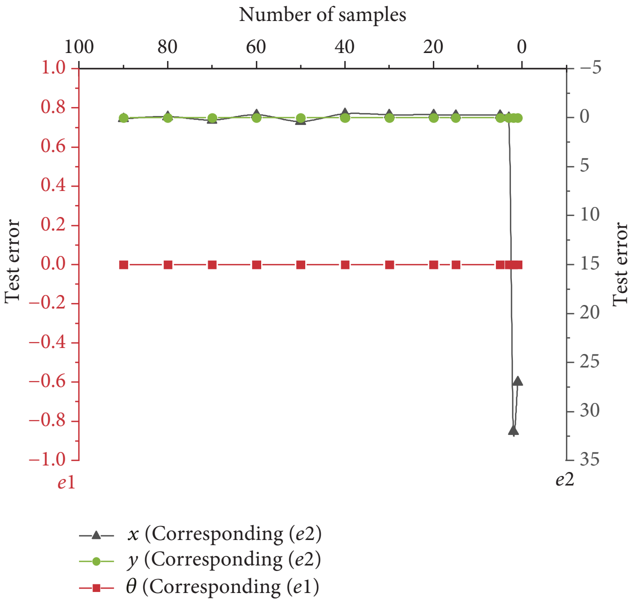


Figure 9: Prediction model training results based on the LSTM network.

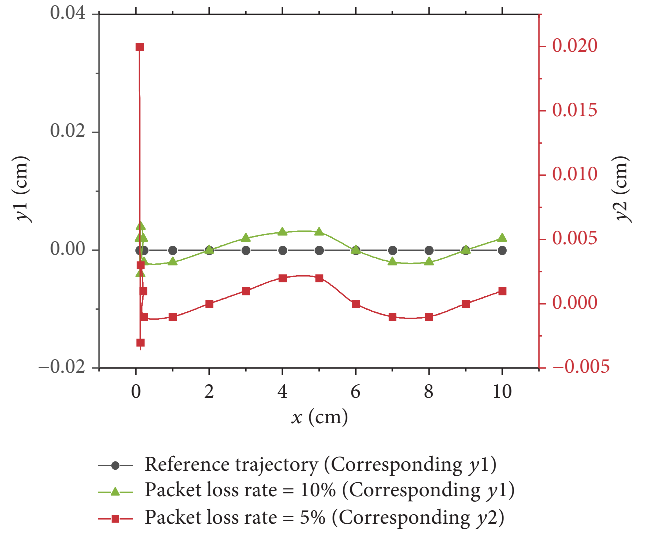


Figure 10: Trajectory tracking error results under different packet loss rates.

(3) Robot Linear Trajectory Tracking Effect Based on the LSTM Network. Figure 12 shows the effect of robot linear trajectory tracking based on the LSTM network through simulation experiments.

Figure 12 shows that each variable of the predicted trajectory has good prediction accuracy when the prediction model is used to track the robot's moving trajectory. It suggests that the lateral position x, longitudinal position y, and angle θ predicted by the LSTM model basically coincide with the reference trajectory provided. The above analysis shows that the model has a good prediction effect on the trajectory of the robot, and the effect of the prediction model is finally verified.

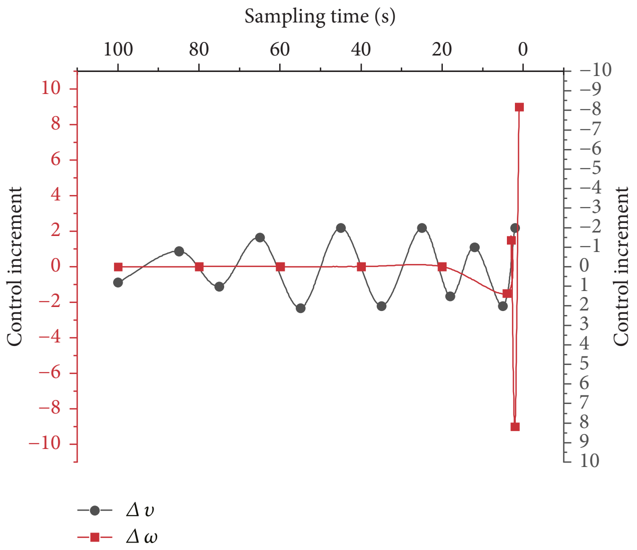
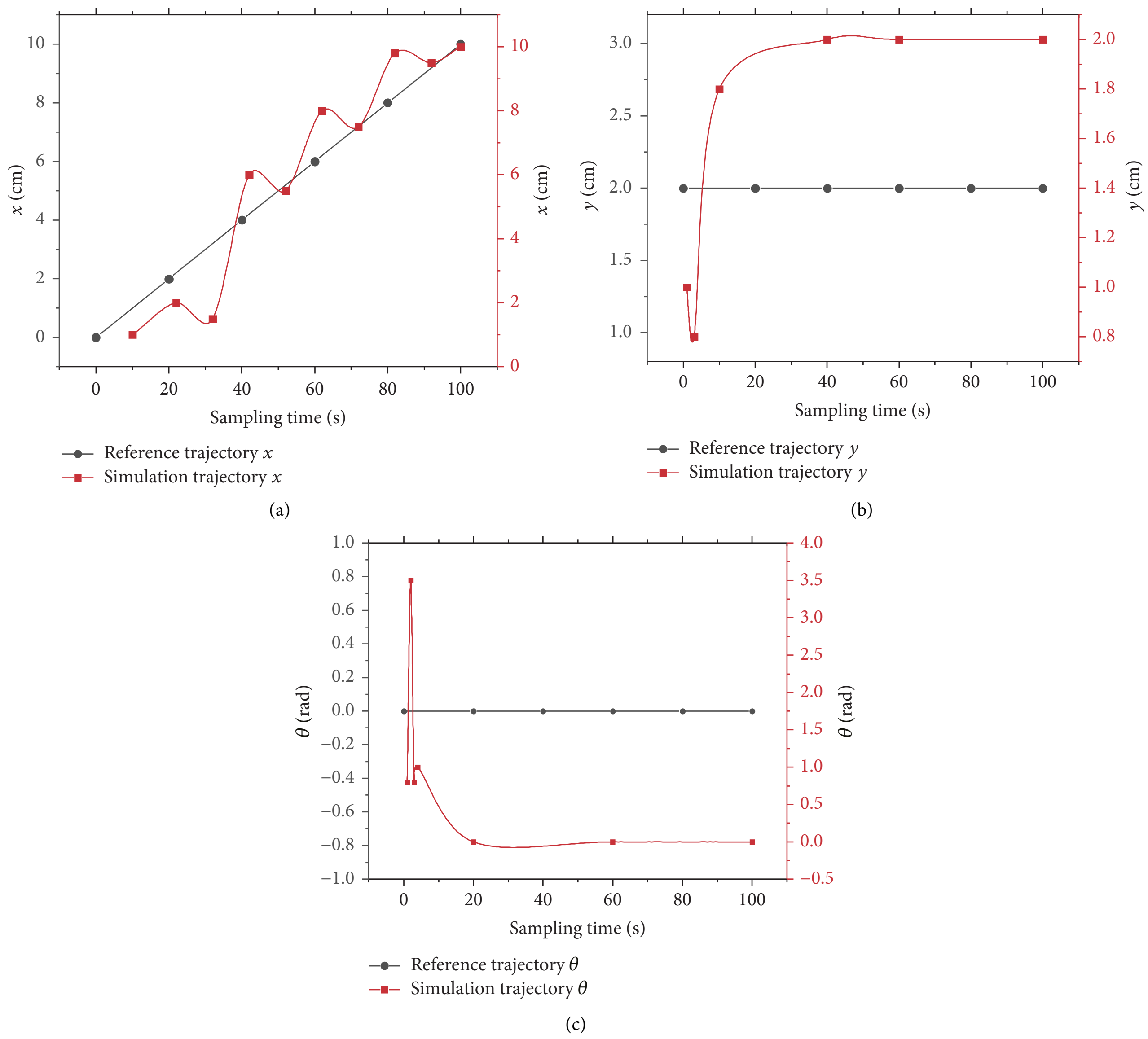


Figure 11: Control increment changes for mobile robots.

5. Discussion

On the one hand, the results show that the LSTM model is different from other network models. The training time of the model is less because of the particularity of its structure, so it has higher accuracy, which provides a reliable basis for the high-precision prediction of the robot trajectory. In addition, the model can simulate and predict the motion trajectory of the robot with high precision by reducing the packet loss rate in the network control system. On the other hand, based on the previous research on mobile robot control, this work uses the deep learning algorithm to design its control model. The selected deep learning algorithm is the LSTM algorithm, which has a good performance for



(a)

(b)

(c)

Figure 12: Mobile robot trajectory tracking status. (a) The lateral displacement tracking the state of the mobile robot; (b) the longitudinal displacement tracking the state of the mobile robot; (c) rotation angle tracking the state of the mobile robot.

predictive control. Therefore, this algorithm is selected as the theoretical support to complete the prediction of mobile robot trajectory. The basis is the existing problem of trajectory predictive control of mobile robots.

6. Conclusion

6.1. Research Contribution. This paper presents a robot trajectory prediction model based on an LSTM neural network combined with the mathematical model of mobile robots and the phenomenon of data packet loss. The research object is the existing problems of trajectory prediction control of mobile robots. The author presents a robot trajectory prediction model based on the LSTM neural network combined with the mathematical model of mobile robots and the phenomenon of data packet loss. After the model is tested by setting up simulation experiments, the following conclusions are drawn. First, the trained LSTM neural network can approximate the mobile robot model infinitely, not only with high accuracy but also with high efficiency. Second, in the case of different packet loss rates, the model can still complete the corresponding curve tracking task well. Third, the control increment changes of the mobile robot under this prediction model are stable. Fourth, when the prediction model tracks the movement trajectory of the robot, the specific variables in it are coincident with the reference trajectory.

6.2. Future Works and Research Limitations. The deficiency lies in that, on the one hand, only the straight-line form is selected for the reference trajectory when testing the model, which does not reflect the model testing effect under other reference trajectory forms. The purpose is to design a trajectory prediction model by combining neural network technology and the kinematic model to improve the accuracy and quality of trajectory prediction. On the other hand, the effect of the LSTM prediction model proposed here has only been reflected in the simulation experiment, but it has not been applied to the mobile robot, and its effectiveness needs to be further studied. Subsequently, the reference trajectory in the form of the sine curve will be set to further verify the effect of the model. Additionally, it will be introduced into a specific mobile robot example to further verify the effect of the model.

Data Availability

The data are available from the corresponding author upon request.

Ethical Approval

This article does not contain any studies with human participants or animals performed by any of the authors.

Consent

Informed consent was obtained from all individual participants included in the study.

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

[1] D. V. Shabanov, A. V. Kozlovich, R. R. Valiev, and O. V. Kochneva, "Autonomous wheeled mobile robot maneuvering in constraint environment. trajectory tracking quality criteria," *Journal of Physics: Conference Series*, vol. 1753, no. 1, Article ID 012051, 2021.

[2] C. Zhao, D. Wang, J. Hu, and Q. Pan, "Nonlinear model predictive control-based guidance algorithm for quadrotor trajectory tracking with obstacle avoidance," *Journal of Systems Science and Complexity*, vol. 34, no. 4, pp. 1379–1400, 2021.

[3] M. A. Miroliubov, A. K. Samusev, I. D. Toftul, and M. I. Petrov, "Spectral characteristics and time dynamics of tunable acoustic resonators in the strong coupling regime," *JETP Letters*, vol. 113, no. 8, pp. 547–553, 2021.

[4] V. Seedha Devi, T. Ravi, and S. B. Priya, "Cluster based data aggregation scheme for latency and packet loss reduction in WSN," *Computer Communications*, vol. 149, pp. 36–43, 2020.

[5] X. Lei, X. Chen, and S. H. Rhee, "A hybrid access method for broadcasting of safety messages in ieee 802.11p VANETs," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, pp. 71–18, 2021.

[6] J. V. Sorribes, L. Peñalver, and J. Lloret, "A spontaneous wireless ad hoc trusted neighbor network creation protocol," *Wireless Communications and Mobile Computing*, vol. 2021, no. 2, 20 pages, Article ID 5531923, 2021.

[7] C. H. Lin and T. Y. Wang, "A novel convolutional neural network architecture of multispectral remote sensing images for automatic material classification," *Signal Processing: Image Communication*, vol. 97, no. 105, Article ID 116329, 2021.

[8] A. B. Filimonov and N. B. Filimonov, "Constructive aspects of the method of potential fields in mobile robotics," Автометрия, vol. 57, no. 4, pp. 45–53, 2021.

[9] S. V. Ivanova and O. B. Ivanov, "Education in the era of the fourth industrial revolution: development vector, prospects and challenges for Russia," *Space and Culture India*, vol. 7, no. 5, pp. 70–79, 2020.

[10] B. Debnath, A. Ghosh, and S. Deb, "Designing a rapid optical response trigger(RORT) for self-navigating and path explorer robot," *Procedia Computer Science*, vol. 167, pp. 1543–1552, 2020.

[11] M. A. Ali, M. S. Radzak, M. Mailah et al., "A novel inertia moment estimation algorithm collaborated with active force control scheme for wheeled mobile robot control in constrained environments," *Expert Systems with Applications*, vol. 183, no. 6, Article ID 115454, 2021.

[12] Y. Islam, I. Ahmad, M. Zubair, and K. Shahzad, "Double integral sliding mode control of leukemia therapy," *Biomedical Signal Processing and Control*, vol. 61, Article ID 102046, 2020.

[13] M. Barzegar-Kalashani, M. A. Mahmud, M. A. Barik, and A. M. T. Oo, "Control of arc suppression devices in compensated power distribution systems using an integral sliding mode controller for mitigating powerline bushfires," *International Journal of Electrical Power & Energy Systems*, vol. 134, no. 5, Article ID 107481, 2022.

[14] Z. Yang, D. Wang, X. Sun, and J. Wu, "Speed sensorless control of a bearingless induction motor with combined neural network and fractional sliding mode," *Mechatronics*, vol. 82, Article ID 102721, 2022.

[15] Z. Zhang, Q. Wu, X. Li, and C. Liang, "Barrier Lyapunov function-based robot control with an augmented neural network approximator," *Industrial Robot: The International Journal of Robotics Research and Application*, vol. 49, no. 2, pp. 359–367, 2022.

[16] L. Chen, A. Fan, H. Shi, and G. Chen, "Region-action LSTM for mouse interaction sequence based search satisfaction evaluation," *Information Processing & Management*, vol. 57, no. 6, Article ID 102349, 2020.

[17] Z. Chen, C. Yang, and J. Qiao, "The optimal design and application of LSTM neural network based on the hybrid coding PSO algorithm," *The Journal of Supercomputing*, vol. 78, no. 5, pp. 7227–7259, 2021.

[18] H. Zhu, C. He, Y. Fang, B. Ge, M. Xing, and W. Xiao, "Patent automatic classification based on symmetric hierarchical convolution neural network," *Symmetry*, vol. 12, no. 2, 186 pages, 2020.

[19] A. N. Moustafa and W. Gomaa, "Gate and common pathway detection in crowd scenes and anomaly detection using motion units and LSTM predictive models," *Multimedia Tools and Applications*, vol. 79, no. 29-30, pp. 20689–20728, 2020.

[20] Y. Sun and X. Xu, "Calibration of MEMS triaxial accelerometers based on the maximum likelihood estimation method," *Mathematical Problems in Engineering*, vol. 2020, no. 2, pp. 1–10, 2020.

[21] J. S. Lin and B. H. She, "A bci system with motor imagery based on bidirectional long-short term memory," *IOP Conference Series: Materials Science and Engineering*, vol. 719, no. 1, Article ID 012026, 2020.

[22] X. Han, Z. Wang, and H. J. Xu, "Time-weighted collaborative filtering algorithm based on improved mini batch k-means clustering," *Advances in Science and Technology*, vol. 105, pp. 309–317, 2021.

[23] V. N. Sizykh, S. B. Antoshkin, R. A. Daneev, M. V. Bakanov, A. V. Livshits, and A. A. Aleksandrov, "Analytical design of control system mathematical models for mobile robots based on the methods of inverse problems of dynamics and modal PID controllers," *IOP Conference Series: Materials Science and Engineering*, vol. 760, no. 1, Article ID 012053, 2020.

[24] S. Glowinski, T. Krzyzynski, A. Bryndal, and I. Maciejewski, "A kinematic model of a humanoid lower limb exoskeleton with hydraulic actuators," *Sensors*, vol. 20, no. 21, 6116 pages, 2020.

[25] U. Zangina, S. Buyamin, M. Abidin, S. Azimi, and H. S. Hasan, "Non-linear PID controller for trajectory tracking of a differential drive mobile robot," *Journal of Mechanical Engineering Research and Developments*, vol. 43, no. 1, pp. 255–270, 2020.

[26] E. Wiyadi, R. N. Setiadi, and L. Umar, "Effect of vegetation profile and air data rate on packet loss performance of lora e32-30dbm 433 MHz as a wireless data transmission," *Journal of Physics: Conference Series*, vol. 1655, no. 1, Article ID 012015, 2020.

[27] J. Guo, J. Cheng, and J. D. Diao, "System identification with binary-valued output observations under either-or communication and data packet dropout," *Systems & Control Letters*, vol. 156, no. 2, Article ID 105010, 2021.

[28] H. Wang, Y. Li, and J. Qian, "Self-adaptive resource allocation in underwater acoustic interference channel: a reinforcement learning approach," *IEEE Internet of Things Journal*, vol. 7, no. 4, pp. 2816–2827, 2020.

[29] S. Nurkholiq, R. K. Putra, A. Khumaidi, F. Fatimah, and E. Kusuma, "Redesign integrated control system GTG and HRSG to reduce loss of electrical production at combined cycle power plant muara karang," *IOP Conference Series: Materials Science and Engineering*, vol. 1096, no. 1, Article ID 012087, 2021.

[30] Y. Zhou, J. Zhao, and X. Liu, "Loss minimization control of dual three-phase linear induction motor with system-level loss model," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 16, no. 2, pp. 324–331, 2021.

[31] A. Shan, X. Fan, C. Wu, X. Zhang, and S. Fan, "Quantitative study on the impact of energy consumption based dynamic selfishness in MANETs," *Sensors*, vol. 21, no. 3, 716 pages, 2021.

[32] L. Cheng, L. Zhang, J. Li, and Y. Ke, "Measurement, identification, and compensation of pose errors for six-axis gantry automated fiber placement machine," *International Journal of Advanced Manufacturing Technology*, vol. 120, no. 3-4, pp. 2259–2276, 2022.

[33] W. Hu, S. Zhang, and L. Niu, "Self-adaptive PID control based on RBF network for trajectory tracking of dual-mass servo system," *Journal of Physics: Conference Series*, vol. 1871, no. 1, Article ID 012113, 2021.

[34] G. Sun, Y. Cong, Q. Wang, B. Zhong, and Y. Fu, "Representative task self-selection for flexible clustered lifelong learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 4, 1481 pages, 2020.

[35] F. Liu, G. Zhang, and J. Lu, "Multisource heterogeneous unsupervised domain adaptation via fuzzy relation neural networks," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 11, pp. 3308–3322, 2021.

[36] D. Li, S. S. Ge, and T. H. Lee, "Simultaneous arrival to origin convergence: sliding-mode control through the norm-normalized sign function," *IEEE Transactions on Automatic Control*, vol. 67, no. 4, pp. 1966–1972, 2022.

[37] D. Li, S. S. Ge, and T. H. Lee, "Fixed-time-synchronized consensus control of multiagent systems," *IEEE Transactions on Control of Network Systems*, vol. 8, no. 1, pp. 89–98, 2021.

[38] L. Zhang, T. Gao, G. Cai, and K. L. Hai, "Research on electric vehicle charging safety warning model based on back propagation neural network optimized by improved gray wolf algorithm," *Journal of Energy Storage*, vol. 49, Article ID 104092, 2022.

[39] W. Yang, X. Chen, Z. Xiong, Z. Xu, G. Liu, and X. Zhang, "A privacy-preserving aggregation scheme based on negative survey for vehicle fuel consumption data," *Information Sciences*, vol. 570, pp. 526–544, 2021.

[40] R. Sun, J. Wang, Q. Cheng, Y. Mao, and W. Y. Ochieng, "A new IMU-aided multiple GNSS fault detection and exclusion algorithm for integrated navigation in urban environments," *GPS Solutions*, vol. 25, no. 4, pp. 147–217, 2021.

[41] R. Sun, Z. Zhang, Q. Cheng, and W. Y. Ochieng, "Pseudorange error prediction for adaptive tightly coupled GNSS/IMU navigation in urban areas," *GPS Solutions*, vol. 26, no. 1, pp. 28–13, 2022.

[42] Y. Wang, H. Wang, B. Zhou, and H. Fu, "Multi-dimensional prediction method based on Bi-LSTMC for ship roll," *Ocean Engineering*, vol. 242, Article ID 110106, 2021.

[43] T. Ni, D. Liu, Q. Xu, Z. Huang, H. Liang, and A. Yan, "Architecture of cobweb-based redundant TSV for clustered faults," *IEEE Transactions on Very Large Scale Integration Systems*, vol. 28, no. 7, pp. 1736–1739, 2020.

[44] Z. Wang, R. Ramamoorthy, X. Xi, and H. Namazi, "Synchronization of the neurons coupled with sequential developing electrical and chemical synapses," *Mathematical Biosciences and Engineering*, vol. 19, no. 2, pp. 1877–1890, 2021.

[45] Z. Zhang, J. Tian, W. Huang, L. Yin, W. Zheng, and S. Liu, "A haze prediction method based on one-dimensional convolutional neural network," *Atmosphere*, vol. 12, no. 10, Article ID 1327, 2021.

[46] K. Shang, Z. Chen, Z. Liu et al., "Haze prediction model using deep recurrent neural network," *Atmosphere*, vol. 12, no. 12, Article ID 1625, 2021.

[47] L. Yin, L. Wang, W. Huang et al., "Haze grading using the convolutional neural networks," *Atmosphere*, vol. 13, no. 4, Article ID 522, 2022.

[48] Y. Liu, Z. Zhang, X. Liu, L. Wang, and X. Xia, "Efficient image segmentation based on deep learning for mineral image classification," *Advanced Powder Technology*, vol. 32, no. 10, pp. 3885–3903, 2021.

[49] Y. Liu, Z. Zhang, X. Liu, L. Wang, and X. Xia, "Ore image classification based on small deep learning model: evaluation and optimization of model depth, model structure and data size," *Minerals Engineering*, vol. 172, Article ID 107020, 2021.

[50] W. Lei, Z. Hui, L. Xiang, Z. Zelin, X. Xu-Hui, and S. Evans, "Optimal remanufacturing service resource allocation for generalized growth of retired mechanical products: maximizing matching efficiency," *IEEE Access*, vol. 9, pp. 89655–89674, 2021.

[51] L. Guo, C. Ye, Y. Ding, and P. Wang, "Allocation of centrally switched fault current limiters enabled by 5G in transmission system," *IEEE Transactions on Power Delivery*, vol. 36, no. 5, pp. 3231–3241, 2021.

[52] J. Zhang, C. Zhu, L. Zheng, and K. Xu, "ROSEFusion: random optimization for online dense reconstruction under fast camera motion," *ACM Transactions on Graphics*, vol. 40, no. 4, pp. 1–17, 2021.

[53] S. Zhao, F. Li, H. Li et al., "Smart and practical privacy-preserving data aggregation for fog-based smart grids," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 521–536, 2021.

[54] J. Li, K. Xu, S. Chaudhuri, E. Yumer, H. Zhang, and L. Guibas, "Grass: generative recursive autoencoders for shape structures," *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 1–14, 2017.