



DATA MINING AND ANALYSIS OF FLIGHT CONTROL SYSTEM BASED ON TEMPORAL FEATURES AND IMPROVED LSTM ALGORITHM

Dingjun HE ¹ , Liping FAN ² , Lijun XU ^{3,*}

¹ School of Artificial Intelligence & Big Data, Luzhou Vocational & Technical College, Luzhou, 646000, China

² Department of Information Engineering, Heilongjiang Institute of Construction Technology, Harbin, 150025, China

³ Chengyi College, Jimei University, Xiamen, 361021, China

* Corresponding author, e-mail: tgyx202408@163.com

Abstract

Due to the structural characteristics of multi-redundancy and multi-closed loops in flight control systems, their fault propagation modes are complex, and the internal physical structure is closely coupled with system components, which poses challenges for analysis and modeling. To improve the accuracy and predictive ability of flight control system fault diagnosis, this study proposes a flight control system fault diagnosis method built on an improved bidirectional long short-term memory network. By integrating convolutional neural networks and bidirectional long short-term memory networks to extract local and temporal features of the data space, the classification and regression problems of flight control system state prediction have been solved. The results indicated that the proposed fault diagnosis algorithm had the highest recognition accuracy for the four modes. Compared with single convolutional neural networks and long short-term memory networks, the accuracy has increased by 2.11% and 1.32%, and the fault diagnosis accuracy has reached 99.49%, which could accurately identify various types of faults. The improved network proposed this time significantly improves the accuracy of flight control system fault diagnosis and reduces false alarm and missed alarm rates.

Keywords: temporal characteristics; LSTM; flight control system; data mining

1. INTRODUCTION

The Flight Control System (FCS) constitutes a vital component of contemporary civil aviation aircraft. It is responsible for controlling key parameters such as flight attitude, speed, and altitude of the aircraft, and uses flight control computers to issue control commands to various control surfaces, greatly improving the accuracy and reliability of control (1). To further enhance the reliability and safety of FCS, redundancy technology is often used to construct redundant FCS. Internal data exchange and voting mechanisms have been established between different flight control computers to significantly reduce system failure rates and ensure the safe operation of the aircraft (2-3). However, the application of this technology also brings challenges, especially in terms of the overall structure and control logic of the system. Due to the need to design and manage multiple backup components, the complexity of FCS significantly increases. This requires designers to carefully design the system architecture, ensure the coordinated work between various components, and be able to quickly switch to

the backup system in emergency situations, thereby avoiding safety issues caused by main system failures (4-5). There are two major problems with the existing health management methods for FCS: Firstly, the low accuracy of FCS operational status prediction leads to false alarms or omissions; Secondly, FCS involves multiple physical quantities and dynamic processes, and its predictive ability has significant uncertainty. During the operation of FCS, a massive amount of operational data is generated, and Deep Learning (DL) algorithms can process this large amount of data. By autonomously learning the feature information in the data, real-time monitoring and fault diagnosis of FCS machines can be achieved.

By real-time monitoring and analysis of multiple monitoring parameters of aircraft FCS, advanced algorithms and models can be used to predict possible system failures and take measures in advance to ensure flight safety. Han et al. proposed a system that can identify incorrect configurations or unreasonable parameter combinations to improve the physical stability of drone flight. The system evaluated the fitness of configurations through

Machine Learning (ML), optimized parameter combinations using multi-objective optimization algorithms, and successfully identified potential erroneous configurations (6). Yuksek et al. proposed a Reinforcement Learning (RL)-based FCS to enhance the flight operation performance of closed-loop reference model adaptive control systems. This method improved the transient response performance of all relevant indicators (7). Guo et al. proposed a health management system for FCS to achieve non-linear degradation signal extraction of FCS. It adopted a time-varying Markov autoregressive exogenous system with time delay to establish a system level degradation model for FCS, and detected FCS through health indicators. The method was found to successfully extract degradation information of FCS (8). Zhao et al. constructed a fault diagnosis deep residual convolution bidirectional Long Short-Term Memory (LSTM) attention network to achieve fixed time fault-tolerant control of hypersonic aircraft. The use of transfer learning techniques for fault diagnosis ensured the tracking performance of the system in the event of actuator damage (9). Kosova et al. designed a novel aircraft hydraulic system health monitoring system based on digital twins. Through the use of SVM and several ensemble learning algorithms (ML) for troubleshooting, the results showed that this method had high accuracy in early evaluation and diagnosis (10).

DL and RL methods can autonomously gain experience and gradually optimize decision-making strategies through trial and error and reward and punishment mechanisms, possessing the ability to balance exploration and utilization, and have enormous application prospects in the aerospace field. Guo et al. proposed a data-driven comprehensive safety risk warning model based on DL to apply real-time data from aircraft rapid access recorders to safety monitoring, risk prediction, and early warning. This model combined the characteristics of aircraft system faults and adopted methods such as fault mode and effect analysis, cause chain analysis, and LSTM (11). Djalel et al. proposed a data-driven aircraft fault prediction method that describes the availability of data during the degradation process and solves the data scarcity problem in remaining useful life prediction through LSTM. This method could improve the prediction accuracy of LSTM technology (12). Sun et al. developed an improved risk assessment algorithm grounded on LSTM and autoencoder to address the issue of inability to identify the quality of risk level labels in the field of flight safety. This algorithm utilized the time series characteristics of flight data to achieve an accuracy of 86.45% and had good risk assessment capabilities (13). Yildirim et al. used LSTM architecture and engine sensor data to predict the remaining life of aircraft engines in order to achieve predictive maintenance of aircraft. The classification accuracy of the LSTM model reached 98.916% (14). Bell et al. proposed a method

combined with LSTM-DL autoencoder to improve the safety of unmanned aerial vehicles and reduce operating costs. This method used dynamic threshold algorithm and weighted Loss Function (LF) for anomaly detection in unmanned aerial vehicle datasets, with high accuracy and fast fault detection speed (15).

In summary, in flight control environments, it often requires a significant amount of cost, simplicity, and the expenditure of a lot of manpower and resources. At the same time, in control systems, it is often not possible to directly use existing datasets for learning, and higher costs need to be paid to collect or construct datasets on their own. The application of deep RL in FCS often requires complex environments and unexpected situations. To ensure the operational efficiency and reliability of FCS, this paper proposes a fault prediction method on the basis of LSTM and Multi-Task Learning (MTL). Firstly, this study takes the historical data of FCS as the input of the model, and then extracts the temporal features through Convolutional Neural Networks (CNN). Finally, a neural network model combining LSTM and MTL is proposed to simultaneously solve the classification and regression problems of FCS state prediction.

2. METHODS AND MATERIALS

This study proposes an FCS fault diagnosis system based on deep RL. It uses LSTM as the shared layer for MTL load forecasting, simulates the coupling characteristics between multiple loads through the shared layer, and achieves the goal of improving prediction accuracy and reducing the incidence of accidents.

2.1. FCS fault diagnosis based on CNN-LSTM

LSTM is widely used to process time-series data. The unique structure of LSTM enables it to effectively capture and utilize long-term dependencies in sequences, resulting in significant achievements in tasks like Natural Language Processing (NLP), speech recognition, and time series analysis (16-17). In addition, the structure of LSTM allows the model to have depth, which means that more complex models can be constructed by stacking multiple LSTM layers, further improving the ability to handle complex tasks. The internal structure of the LSTM unit is shown in Figure 1. It introduces memory units and 3 "gate" structures (input gate, forget gate, and output gate) that control information flow, allowing the model to selectively remember or forget certain information, thereby better handling long-term dependency problems.

LSTM generates a value of f_t from 0 to 1 based on the previous output h_{t-1} and the current input x_t , to determine whether the information C_{t-1} learned in the previous time was partially transmitted, as shown in equation (1).

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \quad (1)$$

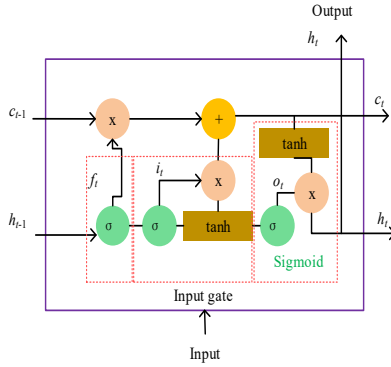


Fig. 1. LSTM internal structure

In equation (1), σ denotes the sigmoid function. W and b are corresponding weights and bias terms. Step 2 is to update and generate new message. Firstly, the "input gate" layer is updated using sigmoid. Secondly, a $\tanh$ layer is taken to generate novel candidate values $\tilde{C}_t$ and added to the cell state. Typically, the values obtained from these two parts are combined for updating, as shown in equation (2).

$$\begin{cases} i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \\ \tilde{C}_t = \tanh(W_C[h_{t-1}, x_t] + b_C) \end{cases} \quad (2)$$

In equation (2), $\tanh$ is the function. The old cell state is updated by first multiplying it by f_t to forget needless information, and then adding it to $i_t \tilde{C}_t$ to gain candidate values. The fusion of one and 2 steps is the period of discarding unnecessary data and adding new detail, as given by equation (3).

$$C_t = f_t C_{t-1} + i_t \tilde{C}_t \quad (3)$$

Step 3 is to decide the model's output, calculate the output of the model using sigmoid, scale the value of C_t to between -1 and 1 using $\tanh$, and

multiply it pairwise with the output of sigmoid. Thus, the model's output represented by equation (4) is obtained.

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (4)$$

Finally, the cell C_t value is updated using equation (5) to obtain the value of the next hidden layer.

$$h_t = o_t \tanh(C_t) \quad (5)$$

According to different functions, FCS can be segmented into longitudinal control system and lateral control system. The former is mainly responsible for controlling the pitch motion of the aircraft, that is, the altitude and speed changes of the aircraft. This control is achieved by adjusting the aircraft's elevators and other control surfaces, ensuring the stability and controllability of the aircraft in the vertical direction. The latter is mainly responsible for controlling the rolling and yaw movements of the aircraft, that is, the direction control of the aircraft. It controls the direction of the aircraft by adjusting control surfaces such as ailerons, ensuring the stability and controllability of the aircraft in the horizontal direction. This study focuses on the servo control in the directional control pathway. The most critical and prone to failure link is the servo control unit performing closed-loop control according to the commands of the Actuator Control Electronics (ACE). Figure 2 is its control block diagram.

In Figure 2, the system sends it to the Electrohydraulic Servo Valve (EHSV) located in the rudder Power Control Unit (PCU) through a set of proportional integral controllers and a set of rudder angle limiters. By driving the actuator through EHSV, the rudder deflects, achieving control over

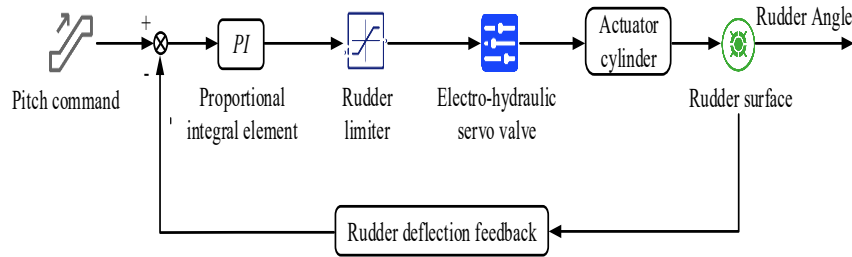


Fig. 2. B777 rudder control block diagram

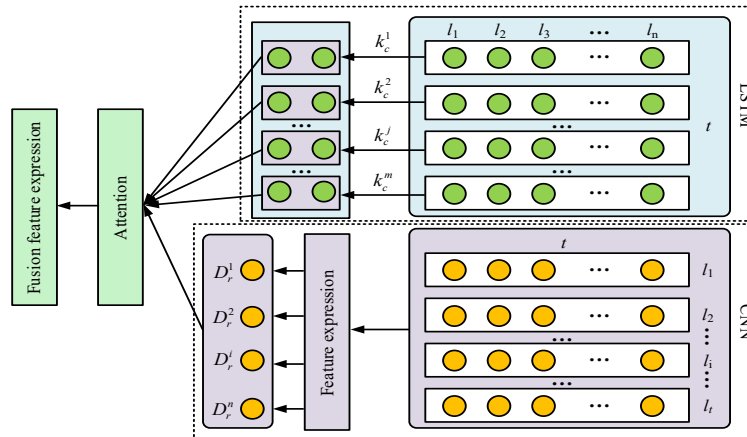


Fig. 3. CNN-LSTM model structure

the rudder. In FCS fault diagnosis, there is a direct correlation between multiple monitoring parameters, which can fully utilize the CNN's ability to capture its spatial local features (18). Meanwhile, due to the temporal characteristics and continuity of the collected system parameters, LSTM can be used to describe them. Therefore, this study constructs a multi-modal deep neural network model based on CNN-LSTM to effectively extract local spatial and temporal sequences of data. Figure 3 shows the framework of CNN-LSTM.

In order to achieve a balance between the feature extraction ability and model complexity of the CNN-LSTM model. The convolutional layer part adopts a two-layer one-dimensional convolutional network. One-dimensional convolution is chosen for the local correlation of multi-sensor time series data in the time dimension. The first layer uses 64×3 convolutional kernels to capture short-term and high-frequency transient features. The second layer uses a 128×5 convolution kernel to extract patterns on a slightly longer time scale. The LSTM layer part stacks two layers of LSTM, with the number of hidden units set to 128 and 64 respectively. The first-layer LSTM requires sufficient capacity (128 units) to learn the complex temporal dynamic characteristics of the input features (from CNN). The second-level LSTM (64 units) is used to abstract higher-level time dependencies.

Assuming $L = \{l_1, l_2, \dots, l_i, \dots, l_n\} \in R^{m \times n}$ is a 2D matrix with $m \times n$ dimensions, used to record the changes of n state parameters l at different time points m . This data are input into the network through two channels, one above and one below. First, through the hierarchical structure and training process of CNN, spatial correlation features are extracted from the original input data. Finally, the extracted features are integrated and output as a fixed $1 \times n$ -dimensional feature vector D_r , as shown in equation (6).

$$D_r = [d_r^1, d_r^2, \dots, d_r^i, \dots, d_r^n] \quad (6)$$

In equation (6), after processing by CNN, the neuron (or filter) generates i specific output feature d_r^i . The input of LSTM is the transpose of the input data of the CNN layer. If the output dimension of the LSTM hidden layer is v and a fixed sequence length is specified as m , then it is transformed into a $m \times v$ -dimensional eigenvector K_c through LSTM, as shown in (7).

$$K_c = [k_c^1, k_c^2, \dots, k_c^j, \dots, k_c^m] \quad (7)$$

In equation (7), k_c^j is the hidden layer output of the j -th time step of the LSTM. This study uses attention mechanism to fuse the features obtained by the two methods. Firstly, the spatial feature D_r extracted by CNN is used as a query vector, and the hidden layer output k_c^j at each time step of LSTM is used as the key value and output value. The calculation of attention weights is based on equation (8).

$$\delta(k_c^j, D_r) = \tanh(k_c^j W D_r^T + b) \quad (8)$$

In equation (8), W is the $m \times n$ -dimensional weight matrix and b is the bias term. Both are obtained through learning during the training phase. D_r^T is the transpose of D_r , while equation (7) calculates the similarity function between the query vector and the key values. By using Softmax to standardize the attention weights, β_j as shown in equation (9) is obtained.

$$\beta_j = \frac{\exp(\delta(k_c^j, D_r))}{\sum_{i=1}^n \exp(\delta(k_c^i, D_r))} \quad (9)$$

Finally, based on the calculated attention weights, the original features are re-weighted to obtain a weighted feature table. The features in this table have been adjusted by the model based on their importance, which helps the model better understand and process input information, as shown in equation (10).

$$f = \sum_{j=1}^n \beta_j k_c^j \quad (10)$$

This context vector will be passed on to the next stage of the model for further processing. Through this process, the model can dynamically adjust weights based on the significance of various parts of the input, thereby capturing key information more accurately. The CNN-LSTM model utilizes the spatial features extracted by CNN and the time series characteristics of LSTM to weight attention, thereby adjusting the weights of LSTM temporal features for specific time enhancements. This enables the model to have more complete feature representation capabilities.

2.2. FCS state prediction based on MTL-LSTM

The integrity of FCS will directly affect the safety and quality of flight. Therefore, real-time monitoring of FCS is essential to quickly switch to the backup system in case of FCS failure, ensuring flight safety. This study takes the pitch channel in FCS as an example and proposes a dual closed-loop control method, as shown in Figure 4.

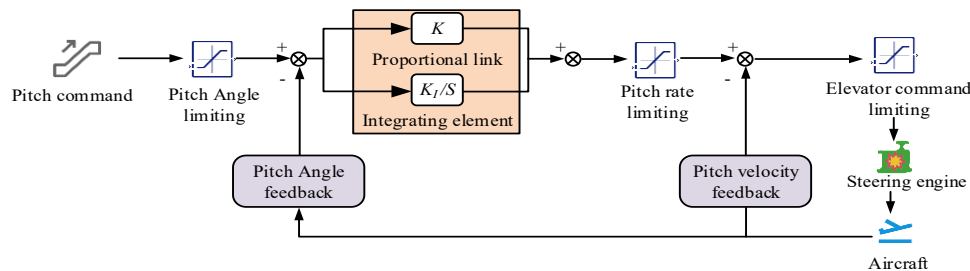


Fig. 4. Pitch loop control system

In Figure 4, FCS consists of multiple control loops, such as altitude control, speed control, etc. The coupling between various control loops makes it impossible to accurately model them physically, and data-driven methods are needed to model them. This study constructs a state set $H = \{h_1, h_2, \dots, h_n\}$ by selecting multiple state parameters related to the system. Then, using the predicted parameter h_i obtained after γ time steps as the predicted value $s_{i,d}$, a training set $\{H, s_{i,d}\}$ is constructed, and the mapping relationship of $H \rightarrow s_{i,d}$ is obtained during the training phase.

Predicting the future state of a parameter based solely on its historical data does not fully consider the correlation between various parameters. Traditional prediction methods based on multiple parameters cannot effectively extract information from sequences, and due to the inability to obtain unpredictable future data, they cannot be applied to iterative prediction. Meanwhile, in engineering practice, it is often necessary to make predictions for multiple state parameters with common characteristics. In response to the above issues, this study introduces the idea of MTL in flight control multi-variable state prediction. MTL is a recursive migration mechanism. The core idea is to achieve

parallel learning of multiple tasks with the same representation by mining the domain knowledge contained in multiple related tasks, in order to improve the generalization ability of the model. Figure 5 shows the state prediction process for multitasking.

Single task learning refers to independently processing multiple prediction tasks in the same state set and mapping them separately. Due to the lack of correlation information with future state parameters, this method can only predict a certain moment at a certain time. When the step size is too large, its prediction accuracy is not high. The multitasking mode integrates multiple tasks from the same state set, sharing the weights of most models, thereby improving the learning efficiency of the entire model and overcoming the disadvantage of not being able to make iterative predictions (19). On this basis, this study proposes a multi-layer adaptive neural network model MTL-LSTM based on multi-layer neural networks, whose structure is shown in Figure 6. In Figure 6, the MTL-LSTM model uses two layers of LSTM as shared layers, and the number of hidden units is set to 100 for both. This is because the shared layer needs to have a sufficiently powerful representation capability to learn the fundamental

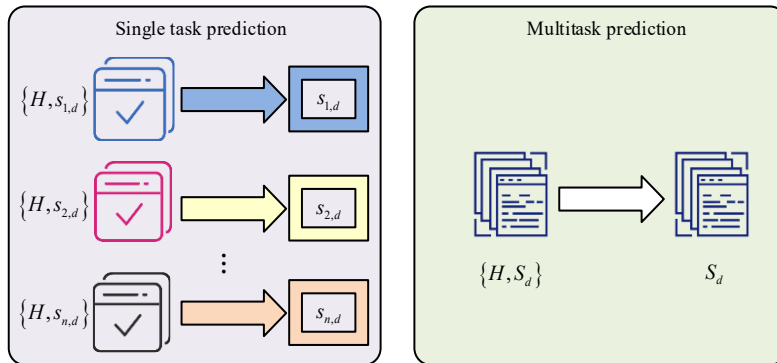


Fig. 5. State prediction process of multiple tasks

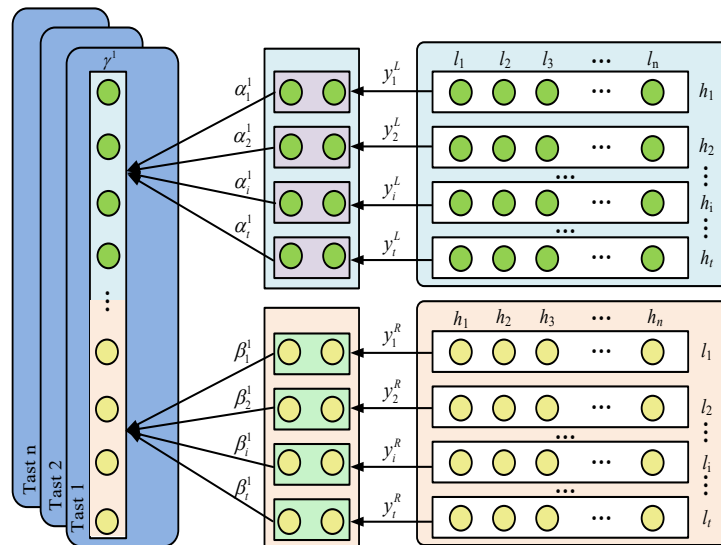


Fig. 6. MTL-LSTM network structure

spatiotemporal features that are common to all tasks and related to the dynamic characteristics of the flight control system. Secondly, the two-layer structure is adopted to construct a deeper time series model, enabling it to learn multi-level dependencies from low-level temporal patterns to high-level temporal contexts. Each task-specific output layer is designed as a lightweight fully connected network, with its input being a 100-dimensional feature vector output by the shared layer, and the output dimension is determined by the specific prediction task. This design paradigm of "emphasizing sharing over tasks" effectively promotes knowledge transfer among tasks while avoiding overfitting for individual tasks.

Each task shares 2 LSTMs, where $L = \{l_1, l_2, \dots, l_t\} \in R^{n \times t}$ and $H = \{h_1, h_2, \dots, h_t\} \in R^{t \times n}$ are inputs to the upper and lower LSTMs. l_i is the state vector of the feature parameter set H at time i . The relationship between L and H can be represented by equation (11).

$$L = H^T \quad (11)$$

At each time point, the upper input l_i is passed through an LSTM network with a m -dimensional vector to obtain y_i^L , which is then output through the hidden layer at each time point as a $m \times t$ -dimensional feature vector Y^L . The input data h_j on the lower side can obtain a $c \times n$ -dimensional feature vector Y^R through LSTM with an output dimension of c in the hidden layer. The expressions for Y^L and Y^R are shown in equation (12).

$$\begin{cases} Y^L = [y_1^L, y_2^L, \dots, y_t^L] \\ Y^R = [y_1^R, y_2^R, \dots, y_n^R] \end{cases} \quad (12)$$

Each task in MTL-LSTM has two attributes, Y^L and Y^R , and attention mechanisms are established based on the impact of different time and feature parameters on each task, as shown in equation (13).

$$\begin{cases} \alpha^z = \text{soft max}(W_\alpha Y^L + b_\alpha) \\ \beta^z = \text{soft max}(W_\beta Y^R + b_\beta) \\ \gamma^z = [\sum_{i=1}^t \alpha_i^k y_i^L, \sum_{j=1}^n \beta_j^k y_j^R]^T \end{cases} \quad (13)$$

In equation (13), z refers to the z -th task. W_α and W_β are the weight matrices of α^z and β^z . b_α and b_β are utilized to adjust the output of the activation function to minimize the LF. α^z and β^z are the weight sets of different state parameters in the z -th task. In the LSTM model, the output vector of each time step goes through a fully connected layer, which is taken to weight and sum the output vectors of the time dimension and feature dimension to obtain the final output. Specifically, the output vector for each time step has a weight matrix. Each row of this matrix corresponds to a feature, and each column corresponds to a time step. By multiplying the output vectors γ^z of the time dimension and feature dimension by their respective weights, and then adding the results, the merged input data can be obtained. This process can be expressed mathematically as equation (14).

$$s^z = \text{sigmoid}(W\gamma^z + b) \quad (14)$$

In equation (14), W and b are the weight matrix and bias term. In practical applications, the prediction of key state parameters also involves more complex data analysis and model construction, requiring higher accuracy and real-time requirements. Therefore, the formula for constructing a weight LF for the entire network is shown in equation (15).

$$\text{Loss} = \sum_{z=1}^n \lambda_z \sum_{u=1}^U \text{loss}(s_u^z - \hat{s}_u^z) \quad (15)$$

In equation (15), s_u^z and $\hat{s}_u^z$ are the true and predicted values of the u -th sample in the z -th task. $\text{loss}()$ is a function of mean square error. U is the total number of samples taken. λ_z is a punitive factor. Among them, for the key state parameters expected to be predicted, the larger the penalty factor, the more severe the penalty when the prediction deviates.

3. RESULTS

This study selected three common faults for simulation and used them as the target of fault diagnosis to verify the performance of CNN-LSTM structure and MTL-LSTM, respectively.

3.1. Performance analysis of fault diagnosis based on CNN-LSTM

This study chose AMESim as the system model and sampled every 30 seconds at 100 Hz to obtain 3,000 time points of sampling data. According to the above method, 11,408 samples were obtained from 4 different models. Then, these samples were divided into 8,213 training datasets, 2,054 validation datasets, and 1,141 testing datasets. This study set up a software experimental environment of Python 3.5 and Tensorflow 1.15.0 on a computer with hardware configuration of Intel(R) Core(TM) i5-3210M CPU @2.50GHz and 8GB memory. Finally, a CNN-LSTM model was constructed using the Keras DL network framework.

In order to verify the applicability of the model in complex fault scenarios and real systems, this study further designed synchronous fault mode experiments, and introduced the Boeing 777 rudder control system as a real case for analysis. The system structure of this case is complex, and the sensor and actuator are closely coupled, which has typical flight control system characteristics. The following two synchronous fault situations are simulated: synchronous fault 1: hydraulic source leakage+actuator hydraulic leakage; Synchronous fault 2: hydraulic oil air mixture+control surface control circuit intermittently fails. At the same time, the research has extracted real operation data containing multiple fault modes from a certain type of civil aviation flight data recorder, and constructed a real system data set containing 12000 groups of samples, covering normal state, single fault and synchronous fault.

The loss curve and accuracy curve of CNN-LSTM model on training set and test set are shown

in Figure 7 In Figure 7 (a), the absolute value of the slope of the loss curve of both training samples and verification samples is large and very smooth in the first 10 single training iterations, and basically keeps the horizontal synchronous change, which indicates that the model is converging rapidly: in the 10th to 25th single training iteration interval, the absolute value of the slope of the loss curve of both training samples and verification samples gradually decreases and approaches 0, and the loss value of both training samples and verification samples decreases from 0.025 to about 0.010 after 15 single training iterations, indicating that the model is further learning and will complete convergence: the loss curve of both training samples and verification samples basically coincides in the last 5 single training iterations, the absolute value of the slope of the curve is infinitely close to 0, and the loss value will no longer change, indicating that the model has completed training and converged successfully. In Figure 7 (b), except for the first five single training iterations, the accuracy change curve of training samples and verification samples is very smooth, and there is no fluctuating broken line, indicating that the model is stable.

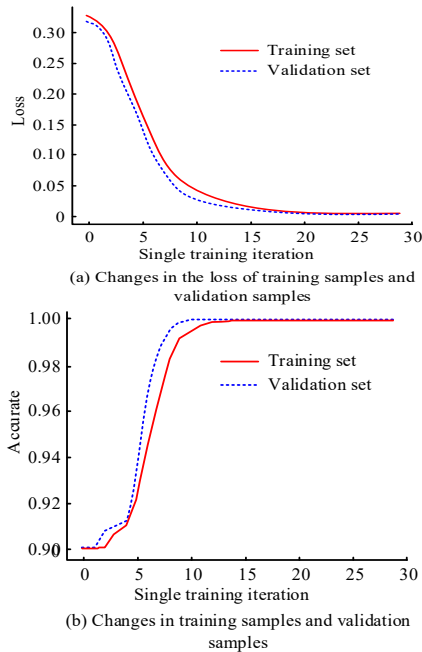


Fig. 7. Convergence performance analysis

In order to further evaluate the stability and generalization ability of the proposed CNN-LSTM model and avoid the accidental impact of a single data partition on the performance evaluation, a 50 fold cross validation was conducted. As shown in Table 1, the CNN-LSTM model shows a high degree of stability and robustness in the five fold cross validation. The accuracy of 50% verification has a very small fluctuation range, ranging from 99.35% to 99.63%, with an average accuracy of 99.50%. More importantly, the standard deviation of each performance index is very small, which indicates that the model is not sensitive to the division of different data subsets, and its excellent performance is not caused by specific training test set division. The results of cross validation strongly prove that the CNN-LSTM model has excellent and consistent generalization capability. The research takes accuracy as the evaluation criterion, conducts fault classification diagnosis experiments, and evaluates the performance of five methods: CNN, LSTM, CNN-GRU, CNN-Transformer, and CNN-LSTM. As shown in Table 2, CNN-LSTM ranks first with an overall accuracy rate of 99.49% and an overall F1 score of 99.48%, significantly outperforming the existing baseline and frontier fusion architectures. Compared with traditional networks, the accuracy rate is 2.13 percentage points higher than that of CNN and 1.18 percentage points higher than that of LSTM. Compared with the fusion schemes of the same period, it leads CNN-GRU and CNN-Transformer by 1.04 and 0.61 percentage points respectively. The fine-grained results show that this method achieves the best results in three out of the four types of faults: the recognition rate of hydraulic oil mixed gas faults is 99.21%, which is the highest in the entire field; The most challenging actuator hydraulic leakage task reached 98.42%, leading the suboptimal model by 1.35 percentage points. The leakage of the hydraulic source achieves 100% zero misjudgment. Although it is slightly lower than CNN-Transformer under normal working conditions (99.44% vs. 99.25%), the comprehensive spatio-temporal feature extraction ability still maintains an absolute advantage, fully verifying the effectiveness and robustness of the CNN-LSTM collaborative architecture in the fault diagnosis of flight control systems.

Table 1. Performance results of 5-fold cross-validation of the CNN-LSTM model

Fold times	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
1	99.52	99.5	99.55	99.52
2	99.41	99.38	99.45	99.41
3	99.63	99.61	99.66	99.63
4	99.35	99.32	99.39	99.35
5	99.58	99.56	99.61	99.58
Average value	99.5	99.47	99.53	99.5
Standard deviation	0.11	0.12	0.1	0.11

Table 2. Fault diagnosis accuracy (%)

Model	Overall accuracy rate	Overall F1-Score	Normal mode	Hydraulic oil gas mixing fault	Actuator hydraulic leakage fault	Hydraulic source leakage fault
CNN	97.36	97.30	97.82	99.03	97.46	95.12
LSTM	98.31	98.28	98.63	97.44	97.48	97.52
CNN-GRU	98.45	98.41	98.80	97.85	97.91	97.25
CNN-Transformer	98.88	98.85	99.25	98.50	98.23	98.55
CNN-LSTM	99.49	99.48	99.44	99.21	98.42	98.67

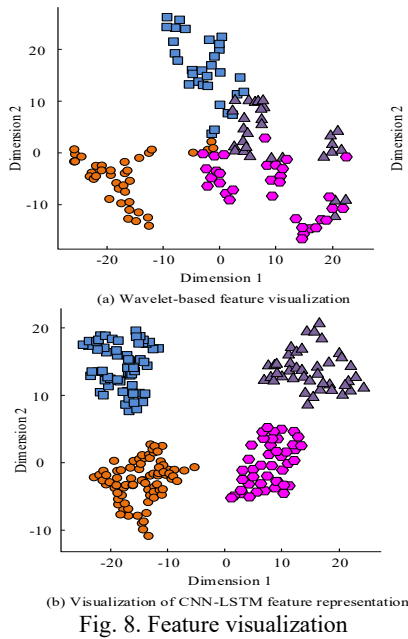


Fig. 8. Feature visualization

This study uses the T-SEN (20) dimensionality reduction technique to perform wavelet-based

feature dimensionality reduction on CNN-LSTM and Bior3.7, resulting in 128 connected levels of features and achieving feature visualization. Figure 8 shows the visualization effect. Figure 8 (a) shows the wavelet transform characteristics after wavelet transformation. In the process of wavelet transform, there may be overlapping between sampling points, and it is difficult to determine the category surface. Figure 8 (b) is a visualization view of CNN-LSTM features. The features extracted by CNN-LSTM have significant clusters and can clearly determine the classification plane, once again demonstrating the powerful ability of the model in feature expression.

Table 3 shows the diagnosis performance of CNN-GRU, CNN-Transformer and CNN-LSTM models under various fault modes in a real system case. Table 3 shows that in the real system case, the three models all show good fault diagnosis ability, but there are gradient differences in performance. CNN-LSTM consistently leads in various indicators, especially in the most challenging synchronous failure scenarios. In synchronous fault 2, the

Table 3. Fault diagnosis performance analysis of each model under real system case

Failure mode	Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Normal mode	CNN-GRU	98.72	98.85	98.63	98.74
	CNN-Transformer	99.15	99.23	99.08	99.15
	CNN-LSTM	99.36	99.41	99.32	99.36
Hydraulic oil gas mixing fault	CNN-GRU	98.45	98.32	98.61	98.46
	CNN-Transformer	98.93	98.87	99.02	98.94
	CNN-LSTM	99.28	99.25	99.34	99.29
Actuator hydraulic leakage fault	CNN-GRU	96.83	96.95	96.74	96.84
	CNN-Transformer	97.52	97.68	97.41	97.54
	CNN-LSTM	98.15	98.27	98.06	98.16
Hydraulic source leakage fault	CNN-GRU	99.28	99.35	99.22	99.28
	CNN-Transformer	99.51	99.58	99.45	99.51
	CNN-LSTM	99.67	99.72	99.63	99.67
Synchronous fault 1	CNN-GRU	95.64	95.52	95.78	95.65
	CNN-Transformer	96.88	96.75	97.03	96.89
	CNN-LSTM	97.95	97.83	98.09	97.96
Synchronous fault 2	CNN-GRU	94.27	94.13	94.45	94.29
	CNN-Transformer	95.86	95.72	96.03	95.87
	CNN-LSTM	96.73	96.61	96.88	96.74

accuracy of CNN-LSTM reached 96.73%, 0.87 and 2.46 percentage points higher than that of CNN-Transformer and CNN-GRU respectively. In relatively simple fault types such as hydraulic source leakage, the three models can achieve excellent performance of more than 99%, but CNN-LSTM still maintains a weak advantage with an accuracy of 99.67%. It is worth noting that the accuracy of 98.15% of CNN-LSTM is significantly better than that of 97.52% of CNN-Transformer in such medium difficult faults as actuator hydraulic leakage, which shows its special advantages in dealing with complex fault modes. These results fully prove that the CNN-LSTM architecture has the best robustness and reliability in fault diagnosis of real flight control systems.

The results of the CNN-LSTM model architecture ablation experiment are shown in Table 4. As can be seen from the table, when the CNN or LSTM modules are removed respectively, the model performance drops significantly, with the accuracy rates decreasing by 2.61% and 1.97% respectively. Secondly, replacing the double-layer LSTM structure with a single-layer LSTM led to a 1.28% decrease in accuracy, verifying the necessity of deeper temporal modeling for capturing complex fault dynamics. Furthermore, reducing the number of convolutional kernels in the CNN or removing the Dropout layer both led to varying degrees of performance degradation (by 0.54% and 0.73% respectively), indicating that the model capacity and regularization strategy we selected are effective and moderate.

To ensure that the proposed fault diagnosis model has the potential to be deployed on the embedded flight control computer, the study analyzed its computational complexity and real-time performance. Flight control systems typically require the completion of a control cycle within a few milliseconds to tens of milliseconds, so the reasoning time of the fault diagnosis algorithm must be much shorter than this period. The research transformed the model into the TensorFlow Lite format and deployed it on an embedded development board with an ARM Cortex-A53 CPU (simulating the computing power level of a typical flight control computer) for inference speed tests. The test results are shown in Table 5. The test results show that the CNN-LSTM model proposed in the study only requires approximately 8.7ms in one forward inference. Considering that the main cycle period of a typical flight control system is usually between 20 and 50ms, this reasoning time fully meets the real-time requirements, leaving sufficient time margin for online fault diagnosis.

3.2. Performance analysis of prediction based on MTL-LSTM

A significant advantage of multitasking mode is the ability to perform iterative prediction. This method is commonly used for processing time series data, where each data point depends on the value of the previous data point or points. The core of iterative prediction lies in using historical data to predict future data points, continuously updating the

Table 4. Results of the ablation experiment for the CNN-LSTM model architecture

Model variant	Overall accuracy rate (%)	Overall F1-Score (%)	Gap with full model accuracy
Complete CNN-LSTM model (benchmark)	99.49	99.48	-
Remove the CNN module (use only LSTM)	97.52	97.48	-1.97
Remove the LSTM module (use only CNN)	96.88	96.82	-2.61
Replace the double-layer LSTM with a single-layer LSTM (128 units)	98.21	98.17	-1.28
Reduce the number of CNN convolution kernels (32→16)	98.95	98.93	-0.54
Remove all Dropout layers	98.76	98.74	-0.73
Replace LSTM units with GRU units	99.02	99	-0.47

Table 5. Inference performance of the model on the embedded platform (Based on single inference, input sequence length T=50)

Performance indicators	CNN-LSTM	CNN-Transformer	CNN-GRU	LSTM
Parameter quantity	185220	210580	168500	154300
Model file size	0.74 MB	0.84 MB	0.68 MB	0.62 MB
Single inference FLOPs	12.5 M	15.1 M	10.8 M	9.5 M
Reasoning time (mean)	8.7 ms	11.2 ms	7.5 ms	6.8 ms
Peak memory usage (at runtime)	~2.1 MB	~2.5 MB	~1.8 MB	~1.6 MB

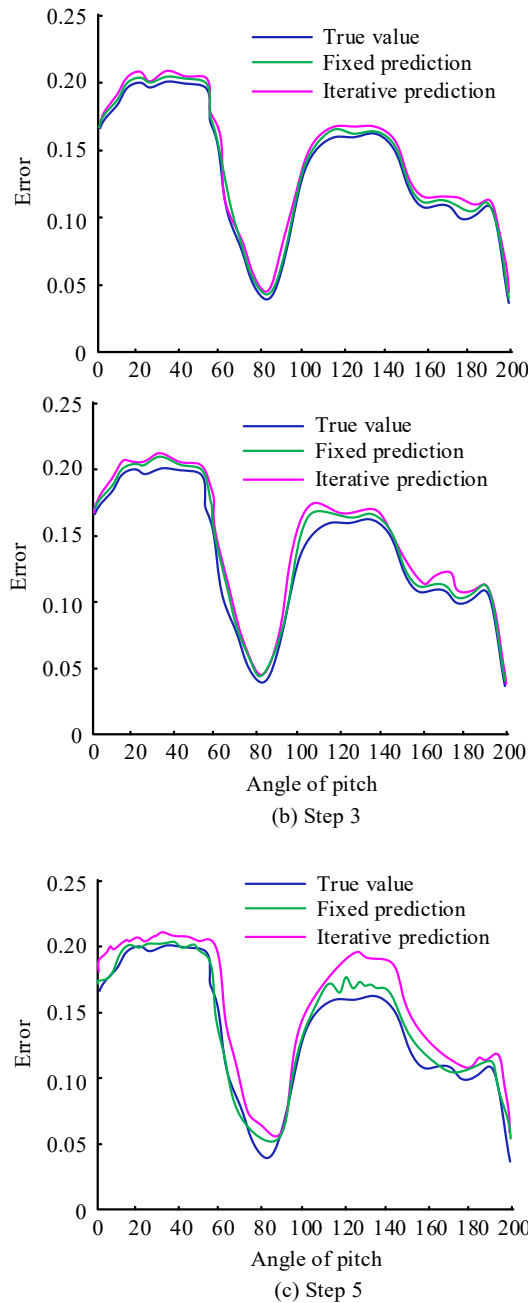


Fig. 9. Iteration prediction results of MTL-LSTM pitch angle

model with predicted values as new inputs to achieve dynamic data prediction. Compared with fixed step prediction, iterative method has stronger adaptability and does not require training for each step. Figure 9 shows the results of iterative prediction of tilt angle using MTL-LSTM model. The error of iterative method is relatively large, because when the prediction step size increases, the prediction error of the last step will be carried over to the next step, resulting in the accumulation of errors. As the forecast step size increases, the forecast error will also increase. However, iterative prediction only requires training one model, thus having greater flexibility, and at a certain level of accuracy, iterative prediction is more realistic.

In the case of a single operating condition and single failure mode, the average attention weight of the model in the time dimension is shown in Figure 10. The model pays more attention to the data at the end of the sequence, and the weights of each time step have been increased compared to the FD001 data, with relatively reduced temporal differences.

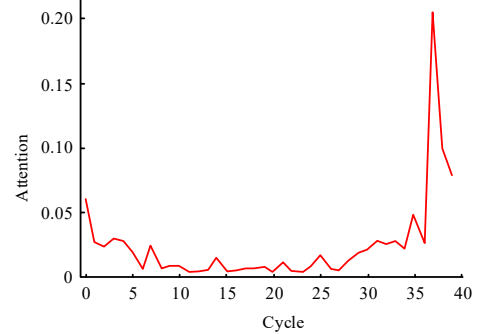


Fig. 10. Changes in attention weight of MTL-LSTM model

In MTL, the importance of each task varies. In the multi-variable state prediction of FCS, the role of system key parameters is much greater than that of non key parameters. Therefore, it becomes very important to apply different penalty factors to different flight missions. Punishment factors are commonly used in optimization problems to adjust the weights of different parameters or constraints, thereby affecting the final optimization results. To test the influence of penalty factors on forecast accuracy, this study takes pitch angle as the main parameter and analyzes other parameters as non critical parameters. Assuming that the importance of key parameters relative to non key parameters is represented by the ratio of the two, the result of the change in pitch angle error when the ratio changes is shown in Figure 11. As the ratio increases, the prediction error of pitch angle becomes smaller, and as the weight increases, the decreasing trend also becomes smaller and tends to stabilize.

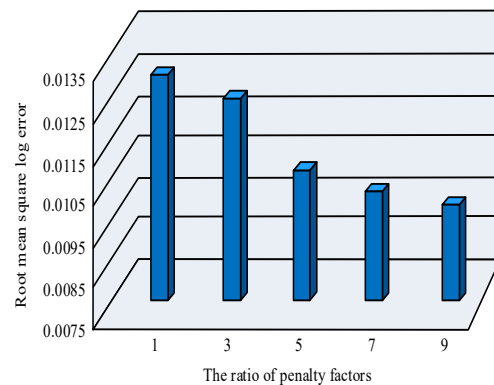
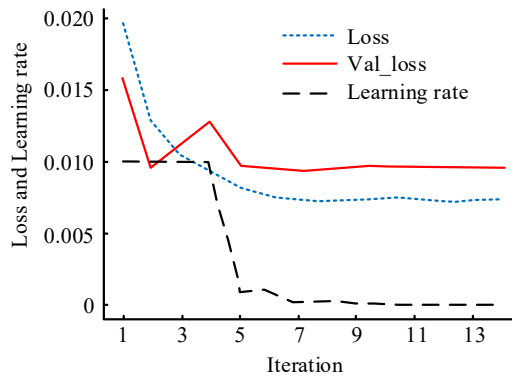


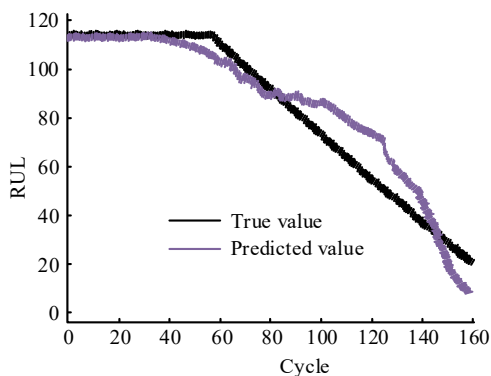
Fig. 11. Variation trend of pitch angle error

The change curves of the LF and learning rate on the two sets for the FD001 single working state in the case of a single machine failure are shown in Figure 12 (a). An engine is randomly selected from the FD001 test set for complete testing, and the results are shown in Figure 12 (b). On the 24

engines in the FD001 dataset, the predicted values are mostly smaller than the actual values, which can provide early warning in the late stage of engine operation and prevent accidents from occurring.



(a) Change curve of loss and learning rate of MTL-LSTM model



(b) Prediction results of MTPL-LSTM model

Fig. 12. Comparison of prediction of MTL-LSTM model

4. DISCUSSION AND CONCLUSION

In response to the problem of traditional intelligent diagnostic methods relying on signal processing and fault diagnosis experience to extract fault features and poor model generalization ability, this study proposed to combine CNN algorithm with LSTM based on DL theory. Aiming at the problem of imbalanced dataset, model optimization techniques like weighted LF, regularization, and batch normalization were introduced to build an MTL-LSTM model suitable for FCS fault diagnosis. The model learned layer by layer from the state parameters of real FCS to achieve feature extraction and target classification. The results showed that the proposed CNN-LSTM method had high accuracy in four modes, with improvements of 1.80% and 0.99% compared to CNN and LSTM. This method has been used for fault diagnosis of FCS, with an accuracy rate of 99.49%. It could accurately identify various types of faults and had good application prospects. The CNN-LSTM method yielded a large number of feature expression clusters and could effectively identify class surfaces, demonstrating strong feature expression ability. Compared with FD001, the MTL-

LSTM model focused more on the data at the end of the sequence, and the weights of each time step were increased, while the time difference was correspondingly reduced. The forecast results of this model were mostly smaller than the measured values, which could provide timely warnings in the late stage of engine operation to prevent accidents from occurring. The research model relies on massive labeled data and lacks effective collection and storage of FCS fault data, resulting in insufficient research on FCS fault analysis. In the future, it is necessary to further build a comprehensive FCS fault database and conduct in-depth research on failure mechanisms.

Source of funding: The research is supported by: Research project of Sichuan Landscape and Recreation Research Center, Key Research Base of Humanities and Social Sciences of Sichuan Provincial Department of Education, on the application of VR in the restoration of historical sites - taking Longshi Mountain Villa in Pingshan County as an example, project number: JGYQ2020023. (Completed in 2023); Research project of Luzhou Vocational and Technical College in 2022, "Research on the Key Path of Virtual Campus Construction" (project number: KB-2206). (Completed in 2023); Key Laboratory Project of Data Intelligence Analysis and Processing in Luzhou City, Intelligent Analysis and Application Research on the Correlation between Online Learning Behavior Data and Grades, Project Number: SZ200306. (Under research); Research on the Application of Deep Learning Algorithms in Retinal Vessel Segmentation at the Key Laboratory of Data Intelligence Analysis and Processing in Luzhou City in 2024.

Author contributions: Research concept and design, D.H.; Collection and/or assembly of data, L.F.; Data analysis and interpretation, L.X.; Writing the article, L.X.; Critical revision of the article, L.F.; Final approval of the article, D.H.

Declaration of competing interest: The author declares no conflict of interest.

REFERENCES

1. Kosacki K, Tomczyk A. Application of analytical redundancy of measurements to increase the reliability of aircraft attitude control. *Aviation*. 2022; 26(3):138-144. <https://doi.org/10.3846/aviation.2022.17555>.
2. Kennedy IR, Hodzic M, Crossan AN, Crossan N, Acharige N, Runcie JW. Estimating maximum power from wind turbines with a simple newtonian approach. *Archives of Advanced Engineering Science*. 2023;1(1):38-54. <https://doi.org/10.47852/bonviewAAES32021330>.
3. Wang HP, Duan FH, Ma J, Wang XL, He YL. Research on redundancy design of a certain aircraft hydraulic system based on goal-oriented methodology. *International Journal of System Assurance Engineering and Management*. 2023;14(1):343-352. <https://doi.org/10.1007/s13198-022-01800-4>.
4. Aryavalli SNG, Kumar GH. Futuristic vigilance: empowering chipko movement with cyber-savvy IoT to safeguard forests. *Archives of Advanced*

- Engineering Science. 2023;1(8):1-16. <https://doi.org/10.47852/bonviewAAES32021480>.
5. Fornaro E, Cardone M, Terzo M, Strano S, Tordella C. Experimentally validated neural networks for sensors redundancy purposes in spark ignition engines. SAE International Journal of Engines 2023; 17(2): 203-220. <https://doi.org/10.4271/03-17-02-0012>.
 6. Han R, Ma S, Li J, Nepal S, Lo D, Ma Z. Range specification bug detection in flight control system through fuzzing. IEEE Transactions on Software Engineering. 2024;50(3):461-473. <https://doi.org/10.1109/TSE.2024.3354739>.
 7. Yuksek B, Inalhan G. Reinforcement learning based closed-loop reference model adaptive flight control system design. International Journal of Adaptive Control and Signal Processing. 2021;35(3):420-440. <https://doi.org/10.1002/acs.3181>.
 8. Guo Y, Ma C, Jing Z. A hybrid health monitoring approach for aircraft flight control systems with system-level degradation. IEEE Transactions on Industrial Electronics. 2022;70(7):7438-7448. <https://doi.org/10.1109/TIE.2022.3201317>.
 9. Zhao J, Lu P, Du C, Cao F. Active fault-tolerant strategy for flight vehicles: Transfer learning-based fault diagnosis and fixed-time fault-tolerant control. IEEE Transactions on Aerospace and Electronic Systems. 2024;60(1):1047-1059. <https://doi.org/10.1109/TAES.2023.3333763>.
 10. Kosova F, Unver HO. A digital twin framework for aircraft hydraulic systems failure detection using machine learning techniques. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science. 2023;237(7): 1563-1580. <https://doi.org/10.1177/09544062221132697>.
 11. Guo Y, Sun Y, He Y, Du F, Su S, Peng C. A data-driven integrated safety risk warning model based on deep learning for civil aircraft. IEEE Transactions on Aerospace and Electronic Systems. 2022;59(2): 1707-1719. <https://doi.org/10.1109/TAES.2022.3204224>.
 12. Djalel D, Yahia K, Mohamed TM, Dimitri L. A new approach for remaining useful life estimation using deep learning. Automatic Control and Computer Sciences. 2023;57(1):93-102. <https://doi.org/10.3103/s0146411623010030>.
 13. Sun H, Yang F, Zhang P, Jiao Y, Zhao Y. An innovative deep architecture for flight safety risk assessment based on time series data. CMES-Computer Modeling in Engineering & Sciences. 2024;138(3):2549-2569. <https://doi.org/10.32604/cmescs.2023.030131>.
 14. Yildirim S, Rana ZA. Enhancing aircraft safety through advanced engine health monitoring with long short-term memory. Sensors. 2024;24(2):518-529. <https://doi.org/10.3390/s24020518>.
 15. Bell V, Moral Arce I, Mase JM, Figueredo GP. Anomaly detection for unmanned aerial vehicle sensor data using a stacked recurrent autoencoder method with dynamic thresholding. SAE International Journal of Aerospace. 2022;15(2):219-229. <https://doi.org/10.48550/arXiv.2203.04734>.
 16. Kant R, Saini P, Kumari J. Long short-term memory auto-encoder-based position prediction model for fixed-wing UAV during communication failure. IEEE Transactions on Artificial Intelligence. 2022;4(1):173-181. <https://doi.org/10.1109/TAI.2022.3153763>.
 17. Garai S, Paul RK, Kumar M, Choudhury A. Intra-annual national statistical accounts based on machine learning algorithm. Journal of Data Science and Intelligent Systems. 2023;2(2):12-15. <https://doi.org/10.47852/bonviewJDSIS3202870>.
 18. Pandiyan V, Akeddar M, Prost J, Vorlauffer G, Varga M, Wasmer K. Long short-term memory based semi-supervised encoder-Decoder for early prediction of failures in self-lubricating bearings. Friction. 2023; 11(1):109-124. <https://doi.org/10.1007/s40544-021-0584-3>.
 19. Chinthamu N, Karukuri M. Data science and applications. Journal of Data Science and Intelligent Systems. 2023;1(1):83-91. <https://doi.org/10.47852/bonviewJDSIS3202837>.
 20. Li J, Jia Y, Niu M, Zhu W, Meng F. Remaining useful life prediction of turbofan engines using CNN-LSTM-SAM approach. IEEE Sensors Journal. 2023;23(9):10241-10251. <https://doi.org/10.1109/JSEN.2023.3261874>.



Dingjun HE, male, Han ethnicity, born September 1975, from Pingshan County, Sichuan Province. He obtained a Master's degree in Software Engineering from University of Electronic Science and Technology of China (UESTC) in 2011. His research focuses primarily on artificial intelligence and big data.

Work Experience: 1994-2008: Pingshan County Education System, Sichuan Province, engaged in educational work. 2008-2012: IT-related enterprises in Chengdu, involved in enterprise information system development. 2012-Present: Luzhou Vocational & Technical College, engaged in teaching and research related to computer science.

Academic Achievements:

Published an EI paper titled Risk Assessment Model of Information Base Based on Machine Learning in Big Data Environment and nearly 20 Chinese academic papers.

Applied for and obtained software copyrights for five systems, including AI Recruitment Information Push System.

Led provincial-level research projects such as Research on the Application of VR in Cultural Heritage Restoration-Taking Longsh Villa in Pingshan County as an Example, as well as 7 municipal and university-level projects. Additionally, served as a core researcher for nearly 10 projects at various levels.

Co-authored two textbooks, including Django Web Development Project-Based Tutorial.

Awarded the second prize for university-level teaching achievements and the third prize in the university-level teaching competition.

e-mail: dingjunhe0821@126.com



Liping FAN, born in February 1980, female, Harbin City, Heilongjiang Province, Han ethnicity, obtained a Bachelor's degree in Computer Science and Application from Harbin University of Science and Technology in 2002 and a Master's degree in Control Engineering from Harbin University of Science and Technology in 2016. Her research focuses on databases and programming.

Work experience: From July 2002 to September 2003, I worked as a teaching intern in the Department of Mechanical and Electrical Engineering at Heilongjiang Institute of Construction Technology. From September 2003 to September 2007, I served as a teaching assistant in the Department of Mechanical and Electrical Engineering at Heilongjiang Institute of Construction Technology. From September 2007 to July 2012, I served as a lecturer in the Department of Mechanical and Electrical Engineering at Heilongjiang Institute of Construction Technology. From July 2012 to April 2018, Associate Professor of Teaching at the School of Computer and Communication Engineering, Heilongjiang Institute of Construction Technology. From April 2018 to present, Associate Professor of Teaching at the Department of Information Engineering, Heilongjiang Institute of Construction Technology.

Academic situation:

Published 10 national and provincial-level academic papers in the past five years.

Publication status of academic works and textbooks

The research status of the scientific research project has been completed. It is a key project of the Education Planning Province of Heilongjiang Province, titled "Research on the Mixed and Diversified Teaching Mode of Database Courses under the Big Data Driven Mode". I have participated in 5 projects and led 2 ongoing projects.

Patent situation:

Apply for 1 international patent, 2 utility model patents, and 2 software copyrights.

Academic award situation

One second prize for excellent papers in the housing and construction industry and vocational education in the first Heilongjiang Province.

e-mail: flp8882024@126.com



Lijun XU, born on December 1983, female, native place (Datong City, Shanxi Province), Han ethnicity, obtained a Bachelor's degree in Mechanical Design, Manufacturing, and Automation from Jimei University from 2001 to 2005, and a Master's degree in Engineering Mechanics from Dalian University of Technology from 2005 to 2008.

Work experience: From 2008 to 2015, worked as a Strength Engineer at the Engineering Research and Development Center of AVIC Shenyang Aircraft Co., Ltd. From 2015 to 2020, worked as a Senior System Strength Engineer at General Electric (China) Co., Ltd. From 2020 to 2021, worked as a Senior Simulation Specialist at the BMW Research and Development Center in Shenyang Huachen. From 2021 to present, worked as a full-time teacher in the Department of Mechanical Engineering at Chengyi College, Jimei University.

Academic situation:

Publication of academic papers, publication of academic works and textbooks, research projects, patents, and academic awards

Exploration of Damage Tolerance Analysis Method for Reinforced Wall Panels of Civil Aircraft Skin, Civil Aircraft Design and Research, 2015, Issue 2, First Author. A method for determining aircraft fatigue analysis coefficients, Mechanical Design, May 2015, first author. Research on Bolt Bending Analysis Method Based on FEM, Journal of Physics: Conference Series, Volume: 2383, in 2022, the first author. Accurate Measurement of

Fatigue Life of Aircraft Engines after Laser Shock Enhancement, Laser Journal, August 2024, First Author.

Improved Ant Colony Algorithm in Aviation Finite Element Analysis and Path Planning Simulation, Innovative. Computing 2025, Volume 1, in 2025, the first author.

Invention patent: A method for analyzing the damage tolerance characteristics of reinforced wall panels, ranked second, November 2024.

e-mail: tgyx202408@163.com