Remaining useful life estimation of aeroengine based on CNN-BiLSTM

Qing Zhang[1], Xiaojun Yang[2], Shihao Zhu[1], Fulin Liu[1], Zixuan Wu[1]
[1]College of Aeronautical Engineering, Civil Aviation University of China, Tianjin 300300, China
[2]Science and Technology Innovation Research Institute, Civil Aviation University of China, Tianjin 300300, China
Corresponding author: Qing Zhang (e-mail: 13820653856@163.com)

ABSTRACT As the core propulsion system of an aircraft, the safety and reliability of aeroengines are directly related to flight safety and operational efficiency. Modern health monitoring systems increasingly rely on electromagnetic sensing technologies and multi-source signal acquisition to provide reliable condition information under complex operating environments, making accurate remaining useful life prediction essential for predictive maintenance and intelligent decision-making. To improve the prediction accuracy of aeroengine remaining useful life, this paper proposes a hybrid model integrating convolutional neural networks and bidirectional long short-term memory networks. The convolutional neural network is first employed to automatically extract spatial features from multi-dimensional sensor data, after which the bidirectional long short-term memory network captures temporal dependencies and degradation trends throughout the engine operating process. The proposed framework enables end-to-end mapping from multi-state monitoring parameters to remaining useful life without requiring explicit degradation modeling or manual feature engineering. Validation on the CMAPSS benchmark dataset demonstrates that the proposed CNN-BiLSTM model consistently outperforms CNN, DCNN, RNN, and BiLSTM approaches in terms of RMSE and MAE, providing more accurate and robust prediction results for aeroengine systems operating under complex degradation conditions. The proposed method not only enhances predictive maintenance capability for intelligent propulsion systems but also offers valuable methodological support for multi-sensor information fusion and electromagnetic-enabled condition monitoring in advanced aerospace applications.

INDEX TERMS aeroengine, remaining useful life, cnn-bilstm, electromagnetic sensing, multi-sensor information fusion

I. INTRODUCTION

THE engine provides power for aircraft and is an important system to ensure the safe flight of aircraft, increasingly utilizing advanced composite materials and three-dimensionally woven ceramic fibers to achieve the high thermal resistance and structural integrity required for modern aviation. Its safety, reliability, economy and maintainability directly affect the flight safety of aircraft and the operating cost of airlines. Aero-engines possess complex structural architectures and operate under extreme environmental conditions. These systems are subjected to sustained high temperatures, pressures, rotational speeds, and mechanical stresses. Consequently, as operational cycles accumulate, internal components inevitably undergo varying degrees of performance degradation. Therefore, an accurate remaining life prediction method for aeroengine components can reduce maintenance frequency and costs while ensuring flight safety, thereby enhancing the operational reliability and economic benefits of advanced propulsion systems in aviation. The performance degradation of aero-engine is often reflected in the changes of efficiency and flow, which are difficult to measure directly, and need to be judged indirectly according to the performance parameters of the engine, such as speed, temperature, boost ratio, fuel flow, etc. With the continuous development of data acquisition technology, the performance parameters of massive airborne sensors can be obtained. How to use the performance data collected during engine operation to build a model to extract the remaining life is the current research focus. At present, the mainstream research methods can be divided into physical model-based methods and data-driven methods. The remaining life prediction method of Aeroengine Based on physical model needs to build mathematical equations according to the physical characteristics and failure modes of the engine, and rely on the mathematical model of aeroengine to simulate the degradation process of the engine. With the continuous development of machine learning technology and sensor technology, data-driven prediction method of aeroengine remaining life is applied more and more widely. The datadriven method does not rely on the specific physical characteristics of the engine, does not need to establish a complex mathematical model, and only needs a large number of monitoring data to get good prediction results. As an emerging field in data-driven research, deep learning has become a research hotspot in recent years. Wang B et al. [1] first applied convolutional neural network to Aero-engine remaining life prediction. Gerhardinger Det al. [2] proposed a deep convolutional network (DCNN) consisting of five two-dimensional convolutional layers to predict the performance of aeroengines, but the convolutional neural network CNN is limited by the size of convolutional kernel and cannot capture timing information. Dintén R et al. [3] proposed a fault diagnosis model of aero-engine gas path components based on bi-directional long shortterm memory neural network (BI LSTM). Shen L et al.[4] proposed using LSTM for Aeroengine EGT prediction. At present, most of the remaining life prediction methods based on neural network extract the performance degradation amount from the multi-state performance parameters, map the multi-state parameters into the performance degradation amount using various statistical methods, and then use the memory neural network to predict the degradation trend of the performance degradation amount, and indirectly obtain the remaining life by setting the degradation threshold [5].However, due to the unknown law of performance degradation, the prediction of performance degradation is not accurate. Taking performance degradation as an interme diate variable will lead to inaccurate prediction results of the model. Based on this, this paper proposes an aeroengine remaining life prediction method based on CNN bilstm hybrid model. By constructing a single hybrid model, the multi state parameters can be directly mapped to the remaining life. First, a large number of time series samples are obtained by sliding the time window, and the time series samples corresponding to the state parameters are used as the model input. After convolutional neural network, the local features in the data are effectively extracted, and then the extracted features are sent to the bilstm layer for time feature extraction, so that the model can directly output the remaining life label, thus avoiding the need for manual feature engineering or the calculation of intermediate performance degradation amounts, enabling the model to learn representative features directly from the multi-dimensional sensor suite and completing the end-to-end prediction. Finally, the validity of the model was verified in the c-mapss dataset.

II. CNN-BILSTM MODEL PRINCIPLE

A. CNN PRINCIPLE

Convolutional neural network (CNN) is a feedforward neural network designed specifically for processing grid structured data (such as image, audio, and timing signals), which can effectively extract local features from data. Convolution layer is the core part, and features are extracted through convolution operation. Convo lution operation is that convolution kernel slides on data to conduct dot product and sum [6]. As shown in Figure 1.

$$X = [x_1, x_2, \ldots, x_n]. \tag{1}$$

Then the convolution calculation can be expressed as:

$$Z_i = \phi(\omega * X + b). \tag{2}$$

Where ϕ is the activation function to generate a nonlinear map, * is the convolution operation, and B is the offset.

According to different pooling functions, the pooling layer ignores unimportant details, reduces data dimen sions, reduces the amount of computation, and enhances the robustness of the network.

B. BILSTM PRINCIPLE

LSTM network is a special recurrent neural network (RNN). LSTM solves the gradient disappearance / explo sion problem of traditional RNNs by introducing gating mechanisms (input gate, forgetting gate, output gate) and cell state [7]. The specific structure is shown in Figure 2.

The LSTM network is divided into two parts, long-term memory C_T and short-term memory H_T, which regulate information through three control gates on the state path. The functions of the three control doors are as follows.

Forgetting gate: the function of forgetting gate is to determine which information in the long-term memory C_{T-1} needs to be discarded at the last moment. The implementation formula is:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f). \tag{3}$$

Among

W_f: forgetting gate weight matrix; b_f: offset vector; $[h_{t-1}, x_t]$; σ: Sigmoid activation function.

$$\sigma(x) = \frac{1}{1 + e^{-x}}. \tag{4}$$

Input gate: the function of the input gate is to determine the writing degree of the current long-term memory C_T at time t, input x_t and the last hidden state information h_{t-1}. The implementation formula is:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i). \tag{5}$$

$$Q_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c). \tag{6}$$

Among

W_i: enter the gate weight matrix; W_c: short term memory weight matrix; b_i, b_c: offset vector.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}. \tag{7}$$

Output gate: the function of the output gate is to determine which information in the current long-term memory C_T is output to the hidden state H_T. The implementation formula is:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o). \tag{8}$$

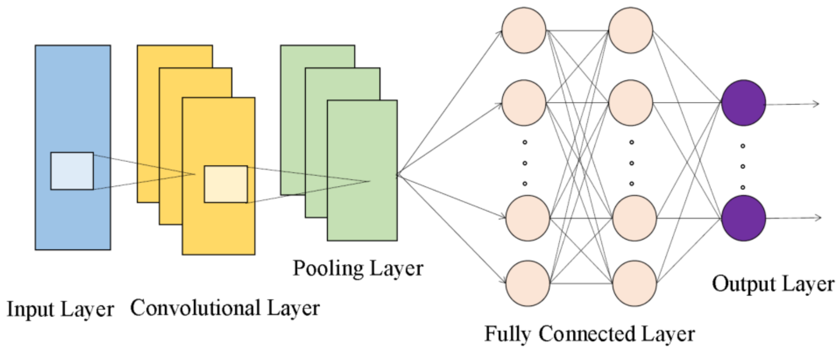

FIGURE 1. CNN Schematic Diagram

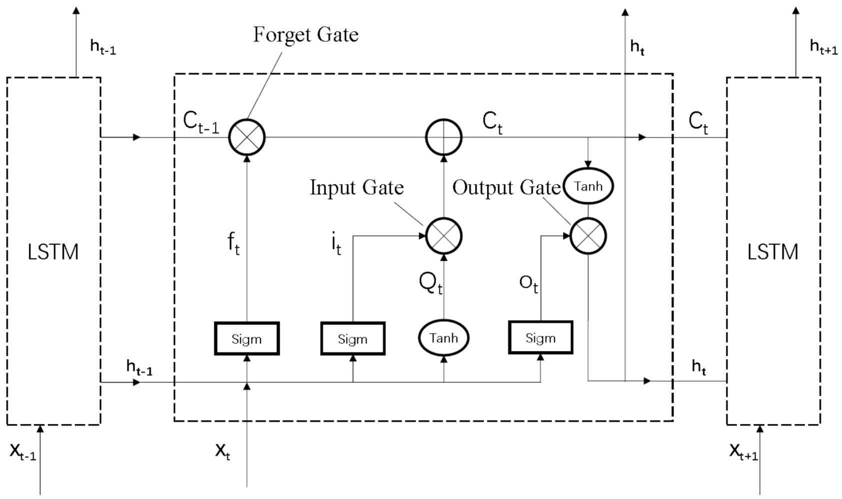

FIGURE 2. Long Short-Term Memory Neural Network Schematic Diagram

$$h_t = o_t \odot \tanh(C_t). \tag{9}$$

Among W_o is the output gate weight matrix, b_o is the bias vector, and $\odot$ multiply elements.

The long-term memory C_T can be updated by combining the forgetting gate and the input gate, and the formula is:

$$C_t = f_t \odot C_{t-1} + i_t \odot Q_t. \tag{10}$$

1) Bidirectional Long Short-Term Memory (BiLSTM) Neural Network

Bilstm is an improvement of LSTM, adding a reverse layer on the basis of the original LSTM forward layer. While the standard LSTM model processes time series data in a single forward direction, the BiLSTM model incorporates a reverse layer to capture dependencies from both directions within a given observation window. In the context of aeroengine RUL prediction, the 'future information' referred to does not imply data beyond the current monitoring timestamp; rather, it denotes the backward temporal context within the pre-defined sliding time window. By analyzing the sequence both forward (from t-n to t) and backward (from t to t-n), the model can more effectively identify complex degradation patterns and subtle sensor fluctuations that a unidirectional model might overlook.

Its structure is shown in Figure 3.

The mathematical expressions of each hidden layer are as follows

$$h_t^R = f^R(w_1 x_t + w_2 h_{t-1}^R). \tag{11}$$

$$h_t^L = f^L(w_3 x_t + w_5 h_{t+1}^L). \tag{12}$$

$$h_t = f(w_4 h_t^R + w_6 h_t^L). \tag{13}$$

The hierarchical structure of CNN bilstm is shown in Table 1.

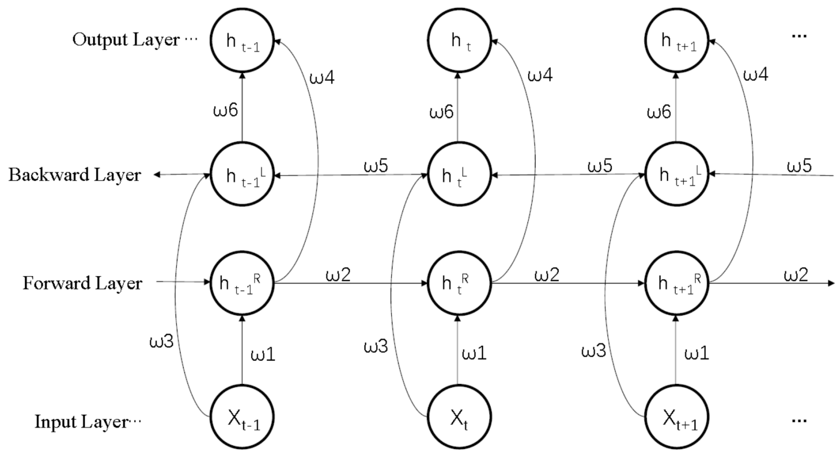

FIGURE 3. BiLSTM Model Schematic Diagram

TABLE 1. CNN-BiLSTM Model Parameters

Network Layer	Layer Parameters
Sequence Folding Layer	N/A
Convolutional Layer 1	Filters: 32, Kernel size: [1, 5], Stride: 1
Activation Layer 1	Activation function ReLU
Max Pooling Layer 1	Kernel size: [1,2], stride 2
Convolutional Layer 2	Filters: 64, Kernel size: [1, 3], Stride: 1
Activation Layer 2	Activation function ReLU
Max Pooling Layer 2	Kernel size: [1,2], stride 2
Flatten Layer	N/A
BiLSTM Layer	Number of units 64
Dropout Layer	Dropout rate 0.3
Fully Connected Layer	one
Fully Connected Layer	one

Convolutional neural network and bidirectional long-term and short-term memory neural network have two different network structures, and their input data formats are also different. Therefore, the difficulty in constructing hybrid models lies in how to deal with the data formats betwe en different network layers [8]. The specific operations ar e as follows:

Firstly, the aeroengine data is normalized through data preprocessing, and then the data is slid according to the fixed time window to generate the time series data with fixed time steps. After sliding the time window, each sample has multiple time steps and multiple features, including both temporal and spatial features. The generated time series data, initially structured as [Batch Size, Time Steps, Features], is processed through a sequence folding layer to be reshaped into a format compatible with 2D convolutional layers, specifically [(Batch Size × Time Steps), 1, Features, 1]. This folding operation allows the CNN to treat each time step as an individual spatial observation while preserving the original feature alignment. By shifting the temporal dimension into the batch dimension, the model can apply identical convolutional filters across all time steps simultaneously, ensuring consistent spatial feature extraction without losing the underlying data structure. Next, multi-scale 2D convolution kernels (specifically [1, 5] for the first layer and [1, 3] for the second layer) are used to convolute the samples. The use of a 1 × 5 kernel in the initial layer allows the model to capture broader local spatial correlations between sensors, while the subsequent 1 × 3 kernel refines these features before they are passed to the BiLSTM layer [9]. The significance of 2D convolution is to simultaneously consider the relationship between multiple features to meet the possible mutual coupling problem between actual data. At the same time, sliding along the time axis enables the convolution operation to also extract the temporal features of features. Following spatial feature extraction, the high-dimensional feature maps are restored to their temporal sequence via a sequence unfolding or specialized flattening process. The tensor is reshaped back to [Batch Size, Time Steps, Flattened Features], effectively mapping the extracted spatial characteristics back onto their original chronological axis. This step is critical for maintaining temporal integrity, as it allows the subsequent BiLSTM layer to capture the long-term dependencies and degradation trends across the restored time series. Finally, the BiLSTM hidden states are passed to a fully connected layer to map these spatio-temporal features to the specific Remaining Useful Life (RUL) labels [10]. In this study, a dropout rate of 0.3 was selected after a grid search analysis, as it provided the optimal balance between preventing co-adaptation of neurons and maintaining the learning capacity of the BiLSTM layer. The specific process is shown in Figure 4.

C. HYPERPARAMETER OPTIMIZATION AND SENSITIVITY ANALYSIS

To ensure the robustness of the CNN-BiLSTM architecture, a sensitivity analysis was conducted on the key hyperparameters, including the number of convolutional filters (unit counts) and BiLSTM hidden units. The configuration of 32 filters for the first CNN layer and 64 units for the BiLSTM layer was determined to be the optimal 'sweet spot' for the FD001 dataset.

Experimental trials indicated that increasing the unit count beyond 64 led to marginal improvements in RMSE but significantly increased the training time and the risk

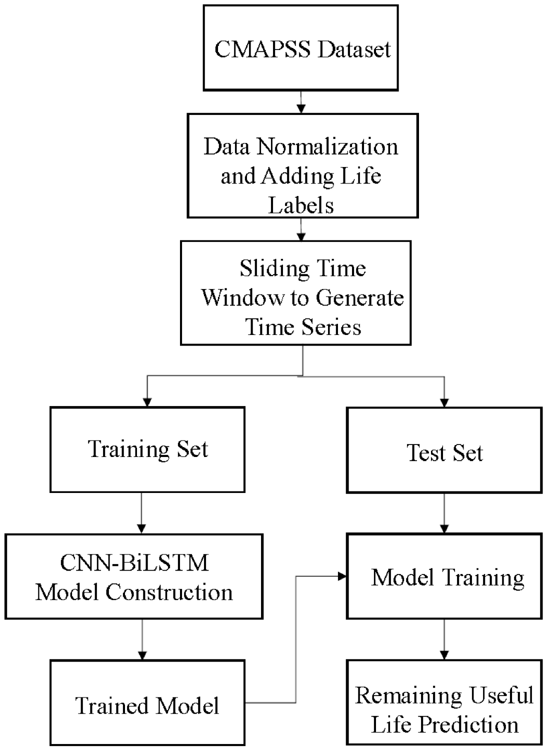

FIGURE 4. Flowchart of the Proposed Model

of overfitting on smaller subsets like FD001 and FD003. Conversely, reducing the unit count below 32 resulted in a failure to capture complex spatial features. The selection of a 0.3 dropout rate was also validated through sensitivity testing; values higher than 0.4 caused underfitting, while values lower than 0.2 were insufficient to stabilize the loss curve during the late stages of engine degradation.

III. EXPERIMENTAL VERIFICATION

To verify the effectiveness of the remaining life prediction method proposed in this article, the algorithm in this paper runs on MATLAB r2023b, and the model is built on a computer equipped with Intel Core i7-12700h processor, NVIDIA geforce rtx3060gpu, and 16g random access memory (RAM).

A. INTRODUCTION OF EXPERIMENTAL DATA

The CMAPSS dataset is a degradation dataset for turbofan engine simulation released by NASA, and is a classic benchmark dataset used for predicting the remaining service life of aircraft engines [11]. The dataset contains four sub datasets (fd001 to fd004), each of which contains a different number of engines, failure modes and operating conditions, as shown in Table 2.

Each sub data set is divided into a training set and a test set, and the training set contains the complete time series sensor data from the initial state of engine operation to failure. The dataset contains 26 kinds of information, including engine number, cycle number, 3 operating conditions and 21 kinds of sensor data.

B. DATA PREPROCESSING

In addition to the engine number and the number of engine cycles, the cmpass dataset has three operating conditions and the monitoring data sent back by 21 sensors, but not all of the data are valid information. For example, columns 1, 2, 5, 6, 10, 16, 18, and 19 in the FD001 dataset remain unchanged throughout the engine operation period and do not provide useful degradation information. In addition to the constant data, columns 3, 4, 7, 8, 9, 11, 12, 13, 14, 15, 17, 20, and 21 are selected as valid performance parameters. These include aircraft altitude, flight Mach number, low pressure compressor inlet temperature, high pressure compressor outlet temperature, low pressure turbine outlet temperature, high pressure compressor outlet total pressure, uncorrected fan speed, uncorrected core turbine speed, HP compressor outlet static pressure, fuel flow, fan correction speed, core machine correction speed, and bypass ratio [12]. Because the data returned by each sensor has different dimensions, the selected performance parameters are standardized by Z-score, which is changed into a normal distribution with a mean of 0 and a standard deviation of 1. The specific formula is:

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i. \tag{14}$$

TABLE 2. CMAPSS Dataset

Sub-dataset	Training Set	Test Set	Number of Operating Conditions	Number of Fault Modes
FD001	one hundred	one hundred	one	one
FD002	two hundred and sixty	two hundred and fifty-nine	six	one
FD003	one hundred	one hundred	one	two
FD004	two hundred and forty-eight	two hundred and forty-nine	six	two

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2}. \tag{15}$$

The normalized value is:

$$z_i = \frac{x_i - \mu}{\sigma}. \tag{16}$$

After standardization, the changes of performance parameters with the number of flight cycles can be obtained, as shown in Figure 5.

There is no clear remaining life label in the training set, only the number of cycles of the engine, so it is neces sary to convert the number of cycles into the remaining life. In order to fully capture the data characteristics of the engine when it is near fault, this paper will segment the remaining life label of the engine.

In order to mine the deep-seated rules of time series samples of performance degradation data, this paper uses time sliding window to generate time series samples, uses fixed time window sliding repeated segmentation for the same engine, and divides the continuous time series data into multiple subsequences of fixed length [13]. In this paper, the repeated segmentation method can not only ensure the correlation between adjacent samples and retain their time characteristics, but also increase the number of samples on the basis of the original data and improve the robustness of the model. The sample generation process is shown in Figure 7.

C. PERFORMANCE EVALUATION INDEX

In order to verify the accuracy of the model, this paper uses root mean square error (RMSE) and mean absolute error (MAE) to evaluate the remaining life prediction performance of the model. The calculation process is:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\breve{y}_i - y_i)^2}. \tag{17}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\breve{y}_i - y_i|. \tag{18}$$

Where n is the total number of engines, $\breve{y}_i$ is the predicted value of the remaining life of the engine, and y_i is the true value of the remaining life of the engine.

D. PREDICTION RESULTS AND ANALYSIS

In order to accurately evaluate the prediction performance of the CNN bilstm hybrid model, this paper uses the fd001 data set for experimental verification, and selects CNN, DCNN, RNN and bilstm models for comparative experiments. The comparison results of RMSE are shown in Table 3.

TABLE 3. RMSE Comparison of Different Life Prediction Methods

Method	RMSE (Root Mean Square Error)
CNN [4]	twenty-three point two one
DCNN [4]	twenty-two point eight six
RNN [4]	twenty-two point three seven
BiLSTM	eighteen point one six
CNN-BiLSTM	fifteen point six five

The comparison results of MAE are shown in Table 4.

TABLE 4. MAE Comparison of Different Life Prediction Methods

Method	MAE (Mean Absolute Error)
CNN [4]	twenty-four point six five
DCNN [4]	twenty-two point seven one
RNN [4]	twenty-two point zero five
BiLSTM	thirteen point five one
CNN-BiLSTM	eleven point eight one

According to the result comparison chart, the proposed method is significantly better than other models. Compared with CNN, DCNN and RNN, the RMSE of the proposed model is reduced by 32.57%, 31.54%, 30.04% respectively, and compared with the bilstm model, the RMSE is reduced by 13.82%. In terms of MAE evalu ation index, the proposed method decreased by 52.09%, 47.99%, 46.44% and 12.58% respectively compared with other models. In contrast, the RMSE and Mae of CNN, DCNN and RNN models are higher, indicating that these models have poor ability to process complex multidimensional time series data, while the RMSE and Mae of bilstm network are higher than those of the model proposed in this paper, indicating that the single bidirectional long-term and short-term memory neural network is slightly insufficient in spatial feature extraction ability [14].

Four engines were randomly selected from fd001, and their predicted remaining life and actual remaining life curves were drawn. The comparison results are shown in Figure 8.

It can be seen from the figure that in the middle of the prediction of the remaining life of the engine, that is, at the beginning of entering the performance decline stage,

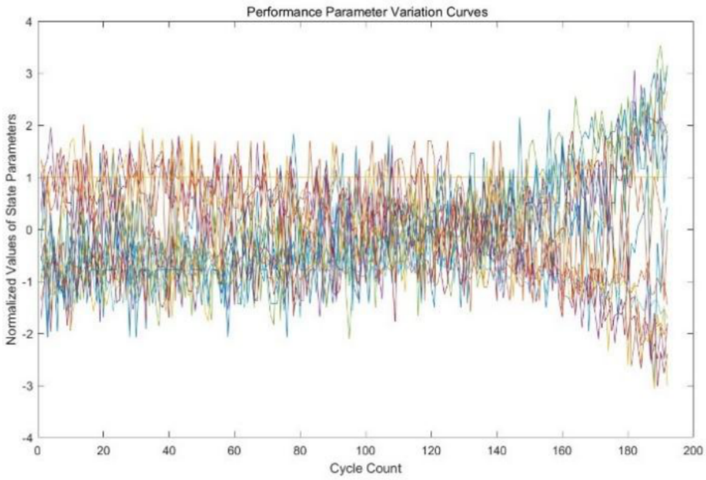

FIGURE 5. Performance Parameter Variation Curves

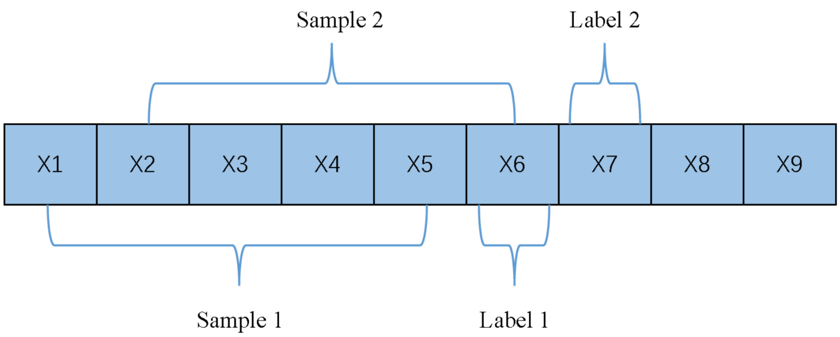

FIGURE 6. Remaining Useful Life Truncation (for a specific engine)

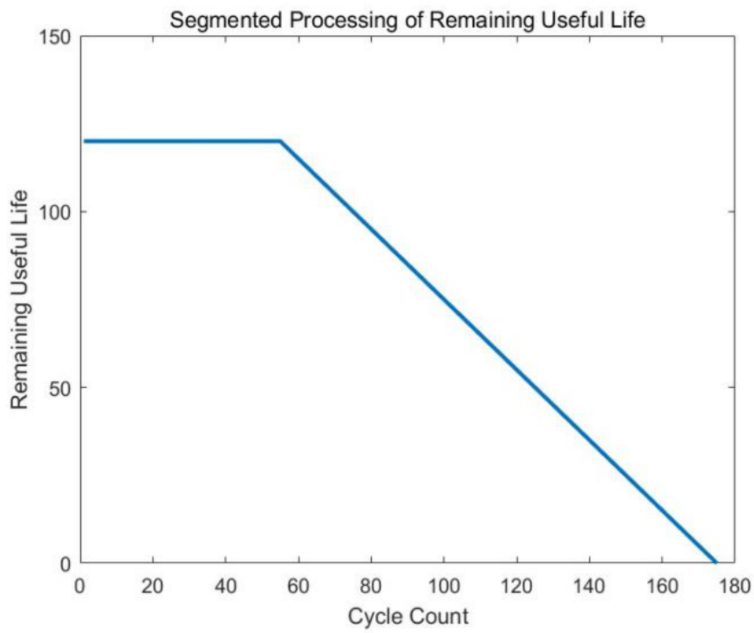

FIGURE 7. Sample Generation by Sliding Time Window

the prediction at this stage fluctuates greatly due to the different reasons for the initial decline of different engines,

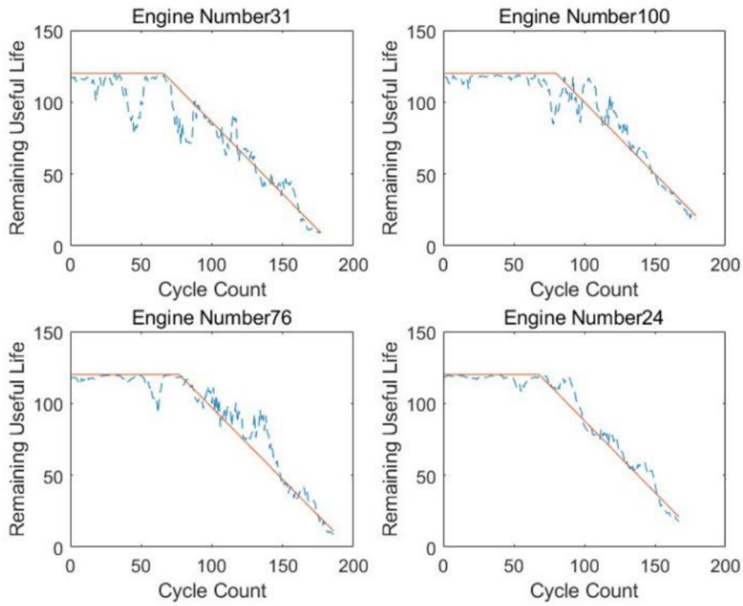

FIGURE 8. Comparison Curve of Actual Life and Predicted Life

while in the later service period of the engine, the decline degree of the internal parts of the engine gradually tends to be the same, so the prediction at this stage is more accurate. Aeroengine health managers are more concerned about the later stage of the engine service period. The accurate prediction of this stage plays a guiding role in the aeroengine maintenance plan. Therefore, the remaining life prediction model proposed in this paper has certain industrial value [15].

IV. CONCLUSION

Aiming at the low accuracy of remaining life prediction for traditional turbofan engines, this paper proposes a CNN-BiLSTM hybrid model specifically designed to capture the complex degradation patterns of advanced composite components and high-performance woven turbine structures. Compared with other traditional models, the method in this paper does not require knowledge of the performance degradation law of the engine, does not need to solve the performance degradation amount, and does not need expert prior knowledge. The CNN-BiLSTM hybrid model enables end-to-end mapping from multi-state performance parameters to RUL. By utilizing a bidirectional recurrent structure, the model extracts comprehensive temporal features from fixed-length historical buffer sequences, making it logically robust for real-time streaming applications where decision-making relies on the most recent trajectory of sensor data. This approach significantly improves the stability of life prediction without requiring future data points beyond the inference moment. The cmpass data set of NASA is used to verify, which significantly improves the accuracy and stability of life prediction. Furthermore, the sensitivity analysis confirms that the selected hyperparameters provide a stable and optimized architecture across varying operational conditions, ensuring that the high-precision end-to-end prediction is not merely a result of specific parameter tuning.

AUTHOR CONTRIBUTIONS

Conceptualization –Qing ZHANG, Xiaojun YANG, Shihao ZHU, Fulin LIU and Zixuan WU; methodology – Qing ZHANG, Shihao ZHU, Fulin LIU and Zixuan WU; investigation – Qing ZHANG, Xiaojun YANG, Shihao ZHU, Fulin LIU and Zixuan WU; writing-original draft preparation – Qing ZHANG, Xiaojun YANG, Shihao ZHU, Fulin LIU and Zixuan WU. All authors have read and agreed to the published version of the manuscript.

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

FUNDING

This research was supported by, This paper is part of the 2025 Civil Aviation University of China Undergraduate Innovation and entrepreneurship training program "Research on the prediction method of a certain type of engine in wing life" (No.: iecauc2025032); The research results of the 2022-2024 educational science research topic "reform and practice of the teaching mode of" new engineering "operation and maintenance courses based on the construction of smart civil aviation" of China Communications Education Research Association (No.: jt2022yb326).

FUNDING (CONTINUED)

The key project of civil aviation university of china Experimental Technology Innovation Fund (2025CXJJ07) and 2025 University-Level Education and Teaching Reform and Research Flight Vehicle Propulsion Engineering (Emerging Engineering Discipline) / Aero engine Fleet Management (Civil Aviation University of China) (CAUC-2025-B1-30)

ACKNOWLEDGEMENTS

Not applicable.

REFERENCES

[1] B. Wang, P. Baraldi, A. Shokry, et al., "A novel two-stage heterogeneous transfer learning framework for the estimation of the remaining useful life of industrial components," Reliability Engineering and System Safety, vol. 267, no. PB, pp. 111968-111968, 2026, doi: 10.1016/J.RESS.2025.111968.

[2] D. Gerhardinger, K. Nikolić K, and A. Domitrović, "Quantifying Operational Uncertainty in Landing Gear Fatigue: A Hybrid Physics–Data Framework for Probabilistic Remaining Useful Life Estimation of the Cessna 172 Main Gear," Applied Sciences, vol. 15, no. 20, pp. 11049-11049, 2025, doi: 10.3390/APP152011049 .

[3] R. Dintén and M. Zorrilla, "Using Time Series Foundation Models for Few-Shot Remaining Useful Life Prediction of Aircraft Engines," Computer Modeling in Engineering & Sciences, vol. 144, no. 1, pp. 239-265, 2025, doi: 10.32604/CMES.2025.065461.

[4] L. Shen, Y. Wang, B. Du, et al., "Remaining Useful Life Prediction of Aero-Engine Based on Improved GWO and 1DCNN," Machines, vol. 13, no. 7, pp. 583-583, 2025, doi: 10.3390/MACHINES13070583.

[5] I. Barry and M. Hafsi, "Advanced multi-model prediction of aircraft engine remaining useful life with random samplingbased class balancing and voting-based features selection," Engineering Applications of Artificial Intelligence, vol. 156, no. PC, pp. 111201-111201, 2025, doi: 10.1016/J.ENGAPPAI.2025.111201.

[6] D. Li H, J. Tu L, H. Liu, et al., "Remaining useful life prediction of a small sample of aero-engine based on an improved gray Markov model," Results in Engineering, vol. 26, pp. 105486-105486, 2025, doi: 10.1016/J.RINENG.2025.105486.

[7] M. Huang, L. Yang, G. Jiang, et al., "ReScConv-xLSTM: An improved xLSTM model with spatiotemporal feature extraction capability for remaining useful life prediction of Aero-engine," Results in Engineering, vol. 26, pp. 105513-105513, 2025, doi: 10.1016/J.RINENG.2025.105513.

[8] Q. Xuan L, M. Munderloh, and J. Ostermann, "Self-supervised domain adaptation for machinery remaining useful life prediction," Reliability Engineering and System Safety, vol. 250, pp. 110296-110296, 2024, doi: 10.1016/J.RESS.2024.110296.

[9] A. Sara, K. Ali, W. Ali, et al., "Aero engines remaining useful life prediction based on enhanced adaptive guided differential evolution," Evolutionary Intelligence, vol. 17, no. 2, pp. 1209-1220, 2022, doi: 10.1007/S12065-022-00805-Z.

[10] H. Wang, D. Li, D. Li, et al., "Remaining Useful Life Prediction of Aircraft Turbofan Engine Based on Random Forest Feature Selection and Multi-Layer Perceptron," Applied Sciences, pp. 13(12), 2023, doi: 10.3390/APP13127186.

[11] G, "M G D, N," M, A. M, et al. A data-driven approach for health status assessment and remaining useful life prediction of aero-engine. Journal of Physics: Conference Series. 2023; 2526(1):012071. doi: 10.1088/1742-6596/2526/1/012071.

[12] T. Unnati and C. Hicham, "Remaining Useful Life Prediction of an Aircraft Turbofan Engine Using Deep Layer Recurrent Neural Networks," Actuators, vol. 11, no. 3, pp. 67-67, 2022, doi: 10.3390/ACT11030067.

[13] D. Azevedo, B. Ribeiro, and A. Cardoso, "Prediction of the Remaining Useful Life of Aircraft Systems via Web Interface," International Journal of Online and Biomedical Engineering (iJOE), vol. 16, no. 04, pp. 23-32, 2020, doi: 10.3991/ijoe.v16i04.11873.

[14] T. Berghout, L. Mouss, O. Kadri, et al., "Aircraft Engines Remaining Useful Life Prediction with an Improved Online Sequential Extreme Learning Machine," Applied Sciences, pp. 10(3), 2020, doi: 10.3390/app10031062.

[15] T. Berghout, L. Mouss, O. Kadri, et al., "Aircraft engines Remaining Useful Life prediction with an adaptive denoising online sequential Extreme Learning Machine," Engineering Applications of Artificial Intelligence, pp. 96, 2020, doi: 10.1016/j.engappai.2020.103936 .