IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 24, 2021, accepted August 26, 2021, date of publication September 1, 2021, date of current version September 9, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3109630

CNN-Based Hybrid Optimization for Anomaly Detection of Rudder System

WEILI WANG, RUIFENG YANG, CHENXIA GUO, AND HAO QIN
School of Instrument and Electronics, North University of China, Taiyuan 030051, China
Automatic Test Equipment and System Engineering Research Center of Shanxi Province, North University of China, Taiyuan 030051, China

Corresponding author: Ruifeng Yang (yangruifeng@nuc.edu.cn)

This work was supported in part by Shanxi Provincial Key Research and Development Project of China under Grant 201903D121060.

ABSTRACT In this study, an automatic test platform suitable for steering gears was established, which can test four sets of rudder systems separately. In addition, we propose an anomaly detection method based on deep learning technology to complete the automated multi-fault classification of the steering gear test. This paper combines the particle swarm optimization algorithm and the grey wolf optimization algorithm to optimize the convolutional neural networks (HPSOGWO-CNN). The proposed HPSOGWO-CNN model is constructed in two stages to realize the efficient and high-accuracy anomaly detection of the rudder system. In the first stage, through 10-fold cross validation, the optimal number of search agents of the HPSOGWO algorithm is obtained, and the performance is compared with GWO and PSO algorithms respectively. The results demonstrate that HPSOGWO algorithm is an excellent technique for automatic selection of hyper-parameters. In the second stage, the designed HPSOGWO algorithm is used to fine-tune the hyper-parameters of CNN, and a highly matched model for anomaly detection of rudder system test parameters was finally obtained. The experimental results show that the accuracy of this method is 99.846%, the precision is 99.748%, the recall is 99.498%, the F-score is 99.618%, and Kappa reaches 0.99565. CNN-based hybrid optimization for anomaly detection of rudder system, is advanced in comparison to KNN, SVM, BP, CNN, PSO-CNN, GWO-CNN, MGWO-CNN, WdGWO-CNN, RW-GWO-CNN models, in terms of accuracy, precision, recall, F-score, and kappa, respectively. Moreover, it is not affected by the imbalance samples, and can achieve accurate classification for small training samples.

INDEX TERMS Anomaly detection, data analysis, HPSOGWO-CNN, rudder system.

I. INTRODUCTION

The rudder system is a kind of servo mechanism, which is widely used in the control systems of airplanes, ships, missiles, etc. It receives the control signals from the flight control system and drives the deflection of the rudder surface, so as to control of flight attitude and trajectory. In order to ensure the precise control of the rudder system, it is necessary to analyze the state parameters, static parameters and dynamic parameters of the rudder system during the production process. Initial testing of the rudder system was done manually. With the rapid development of automated test systems, the parameter test process has become efficient and accurate [1]. However, a large number of rudder system parameter test results are still analyzed by manual processing method at present, which leads to the difficulty in ensuring the accuracy of anomaly detection and time-consuming [2]. In the analysis of the rudder system, we judge the performance based on the test results of a set of parameters. If the parameter value is abnormal, it is the work we need to do to locate the specific problem directly, quickly and automatically. In recent years, machine learning technology has also been applied in anomaly detection [3]–[7].

There are few applications of machine learning in the evaluation of steering systems, and only 4 documents have studied it. Table 1 summarizes it.

In Reference [8], the authors used Support Vector Machine (SVM) to diagnose the fault of steering gear and realized intelligent data analysis. And they focused on developing a new decision-points distribution and weight-assignment-oversampling method. The classification accuracy is 91%, the recall is 96.67%, the F-score is 94.2%, and the TNR is 74%. In Reference [9], the intelligent algorithm was applied to the fault detection and location of rudder

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

system and the imbalance data were effectively processed. The highlight of this article is the use of adaptive sampling algorithm considering informative instances (ASCIN) to process the dataset. This technology prevents the loss of important information in the dataset. The classification accuracy is 97.3%, and the TNR is 88.3%. The authors applied Shuffled Frog Leaping Algorithm-based Random Forest (SFLA-RF) algorithm to rudder fault diagnosis in Reference [2]. The advantages of this method for classification are short time and high accuracy. The classification accuracy is 99.787%, and the kappa is 0.99486. Reference [10] broke through the technical bottleneck of low parameter testing efficiency, the classified kappa is 0.9954, and the RMSE is 0.0395.

None of the four papers analyzed and studied the classification performance of the various states of steering gear. In actual engineering, we not only need to ensure that the overall classification accuracy is high, but also that the different states of the steering gear can be accurately distinguished. This article introduces deep learning technology for the first time, and makes a systematic analysis in this regard, breaking the gap of deep learning technology in the classification of test data of the rudder system.

TABLE 1. Summary of related work on rudder system fault diagnosis.

Technique	Contributions	Diagnosis Effect	(Ref.)
PPSO-SVM	Intelligent data analysis	91% accuracy 96.67% recall 94.2% F-score 74% TNR	[8]
WOA-SVM, ASCIN	Handling of unbalanced data.	97.3% accuracy 88.3% TNR	[9]
SFLA-RF	Multi-fault classification.	99.787% accuracy 0.99486 kappa	[2]
SFLA-MWDT	State prediction	0.9954 kappa 0.0395 RMSE	[10]

We also analyzed anomaly detection and fault diagnosis in other areas related to the steering gear experiment, as shown in Table 2. In Reference [11] different fault modes of rolling bearings can be reliably identified. In Reference [12], the author used Random Forest (RF) as a classifier to classify the complex fault of gearbox. Reference [13] showed that SVM has outstanding generalization performance and can obtain high classification accuracy when applied in machine condition monitoring and diagnosis. In Reference [14], CNN was used for feature extraction of time-frequency graph, so as to carry out sensor fault classification. In Reference [15], Deep Neural Networks (DNNs) was designed for fault classification, which can overcome the shortcomings of artificial neural network. Reference [16] can achieve high-precision fault diagnosis when the workload changes. In Reference [17], Independent Component Analysis (ICA) was used for feature extraction of induction motor data, and multi-classification fault identification based on SVM was completed. Reference [18] applied the learning ability of Deep Belief Network (DBN) to fault diagnosis of rolling bearing. Reference [19] and [20] combined the powerful capabilities of neural network in feature extraction and deep auto-encoders (DAE) in classification.

TABLE 2. Summary of DL-based methods for fault diagnosis.

Technique	Applications	(Ref.)
PSO, WSVM	Fault diagnosis of rolling element bearings.	[11]
RF	Complex fault location of gearbox.	[12]
SVM	Machine fault diagnosis.	[13]
CNN	Aeroengine sensor fault diagnosis.	[14]
DNNs	Intelligent diagnosis of rotating machinery.	[15]
TICNN	Bearing fault diagnosis.	[16]
ICA, SVMs	Intelligent faults diagnosis of induction motors.	[17]
DBN	Bearing fault diagnosis.	[18]
CNN, DAE	Fault diagnosis of distillation process.	[19]
RNN, DAE	Bearing fault diagnosis.	[20]

To sum up, it can be seen from the above literatures that machine learning has been widely used in various fields of fault diagnosis. Among them, deep learning has a significant effect in mechanical fault diagnosis.

In recent years, Convolutional Neural Network (CNN) and optimization algorithms have been widely used in various fields. The Particle Swarm Optimization (PSO) and the Grey Wolf Optimization (GWO) algorithm have significant effects in optimization. Reference [21] used only pixels and disease labels as the input of CNNs to automatically classify skin lesions, with a level of competence almost equal to dermatologists. In Reference [22], the designed neural network was used to complete binary stress detection and three types of emotion classification. In Reference [23], the contextual deep CNN predicted the corresponding label of each pixel vector to complete hyper-spectral image classification. Reference [24] showed that the classification performance of DNNs can be significantly improved by PSO algorithm, and compared the impact of the swarm size on PSO and neural network architecture on classification performance. In Reference [25], in order to obtain better classification performance, the author adopted PSO algorithm to optimize the parameters in SVM. In Reference [26], the hyper-parameters in the CNN model were optimized by PSO algorithm, which achieved the purpose of improving network performance and made the selection of hyper-parameters automatic. Reference [27] pointed out that some hyper-parameters in CNN had a significant impact, whereas other hyper-parameters were not very important. In Reference [28], ImGWO was used for feature selection, and ImCNN was used for network anomaly classification. In Reference [29], an enhanced GWO algorithm was used to optimize the hyper-parameters of Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM) networks. In Reference [30], the optimal size of system components was obtained by Hybrid PSO and GWO (HPSOGWO) algorithm. Reference [31] showed the superiority of HPSOGWO algorithm in optimizing the path. In Reference [32], the performance of the hybrid optimization algorithm was verified in a practical problem. In Reference [33], through the comparative experiments of various optimization algorithms, the superior

TABLE 3. Investigation of experimental methods used in this article.

Technique	Objective	Contributions	(Ref.)
CNNs	Diagnosis of skin cancer.	Artificial intelligence in skin cancer classification.	[21]
CNN	Stress detection and emotion classification.	Demonstrate the potential of deep neural networks.	[22]
CNN, FCN	hyper-spectral image classification.	Build a wider and deeper contextual deep CNN.	[23]
PSO, DNNs	Get high quality hyper-parameters.	Solve the difficulty of designing DNNs hyper-parameters.	[24]
PSO, CNN, SVM	Classification of disease.	Combination of feature extraction and classification techniques.	[25]
PSO, CNN	Classification of Lung nodule.	Improve the network performance.	[26]
GWO, CNN	Predict the hygrothermal time series.	Capturing the complex patterns of the hygrothermal response.	[27]
GWO, CNN	Network log big data anomaly.	Realize real-time processing of stream data.	[28]
GWO, CNN-LSTM	Time series prediction and classification.	Improve rate of convergence.	[29]
HPSOGWO	Obtain optimal size of system components.	Comparison of different optimization methods.	[30]
HPSOGWO	Minimizes the path distance.	Propose a multi-objective path planning algorithm.	[31]
HPSOGWO	Search for the best solution.	Verify the performance of the algorithm.	[32]
HPSOGWO	Solution of single-area unit commitment problem.	Comparison of different optimization methods.	[33]

aspects of HPSOGWO algorithm were obtained. Table 3 summarizes the relevant algorithms.

From the survey in Table 3, CNN has a significant performance on classification. Refined from the cited literature, the PSO algorithm can effectively fine-tune the hyper-parameters of the CNN network, and the rate of convergence is fast, but the accuracy needs to be improved. The GWO algorithm is more excellent in terms of optimization accuracy. In recent years, the traditional machine learning algorithm has been applied to rudder system fault diagnosis, which can achieve automatic classification of abnormal data, but the classification performance for small samples needs to be improved. Inspired by the above research, we try to use the CNN optimized by the HPSOGWO algorithm to extract and classify the test parameters of the rudder system in order to achieve outstanding accuracy of fault location. Thus, the rudder system parameter analysis can be automated.

The main contributions of this paper are summarized as follows:

1) An automatic test platform for steering gears has been built to realize parameter testing of up to four sets of steering gears at the same time.

2) A convolutional neural network model combining particle swarm optimization algorithm and grey wolf optimization algorithm (HPSOGWO-CNN) is proposed for anomaly detection of rudder system. This model can obtain superior classification performance in the rudder system.

3) In this paper, the HPSOGWO algorithm is used to solve the problem of difficult selection of CNN hyper-parameters.

4) This article solves the problem of inaccurate classification caused by imbalanced steering gear samples.

The rest of this article is structured as follows. The Section 2 introduces the system model. In the Section 3, the optimal CNN model for rudder system testing is constructed. In the Section 4, the performance of the model is verified by a series of comparative experiments, which is followed by a summary of the Section 5.

II. PROPOSED SYSTEM MODEL

In this study, we propose a new model named HPSOGWO-CNN, which uses the HPSOGWO algorithm to optimize the hyper-parameters in the 1D CNN to obtain the best 1D CNN structure. Among them, the optimization algorithm is a hybrid variant that combines PSO and GWO variants together.

A. GREY WOLF OPTIMIZER (GWO)

The Grey Wolf Optimizer (GWO) was originally proposed by Mirjalili *et al.* [34]. Grey wolves have a very strict social hierarchy. The leaders of the pack are called Alphas (α). To be specific, Alphas manage the team and are answer for making decisions. The second largest scale of the grey Wolf is Beta (β). Beta wolves obey the Alphas and help the Alphas make decisions. The lowest level of grey wolves is Omega (ω), which is a group of the pack who are completely obedient to other wolves. The other wolves are called Delta (δ). They obey Alpha and Beta, but dominate Omega.

The GWO optimization process mainly includes the social hierarchy, encircling prey, hunting, attacking prey and search for prey.

1) SOCIAL HIERARCHY

In order to design the GWO, we established a mathematical model of the social hierarchy of wolves. Among them, the optimal solution is Alpha. Similarly, the second and third optimal solutions are called Beta and Delta. The remaining candidate solutions are named Omega. The algorithm optimization process is dominated by Alpha, Beta and Delta.

2) ENCIRCLING PREY

The mathematical model of encircling behavior of each search agent in the wolf pack is as follows. The position of the grey wolf is updated by (1) and (2).

$$\vec{D} = \left| \vec{C} \cdot \vec{X_p}(t) - \vec{X}(t) \right| \tag{1}$$

$$\vec{X}(t+1) = \vec{X_p}(t) - \vec{A} \cdot \vec{D} \tag{2}$$

where t indicates the current iteration, $\vec{A}$ and $\vec{C}$ are coefficient vectors, $\vec{X_p}$ is the position vector of the prey, and $\vec{X}$ indicates the position vector of a grey wolf.

The vectors $\vec{A}$ and $\vec{C}$ are given by the following mathematical formula:

$$\vec{A} = 2\vec{a} \cdot \vec{r_1} - \vec{a} \tag{3}$$
$$\vec{C} = 2\vec{r_2} \tag{4}$$

where components of $\vec{a}$ are linearly decreased from 2 to 0 over the course of iterations and $\vec{r_1}$, $\vec{r_2}$ are random vectors in [0,1].

3) HUNTING

We set Alpha, Beta and Delta as the three optimal solutions. Then update the position of Omega and other wolves according to the location information of Alpha, Beta and Delta, as shown in Fig. 1.

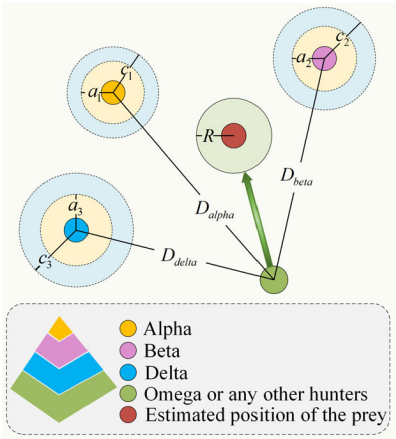


FIGURE 1. Steps to update the location of the grey wolves.

The mathematical model of this behavior can be expressed as follows:

$$\vec{D_\alpha} = \left|\vec{C_1} \cdot \vec{X_\alpha} - \vec{X}\right| \tag{5}$$
$$\vec{D_\beta} = \left|\vec{C_1} \cdot \vec{X_\beta} - \vec{X}\right| \tag{6}$$
$$\vec{D_\delta} = \left|\vec{C_1} \cdot \vec{X_\delta} - \vec{X}\right| \tag{7}$$
$$\vec{X_1} = \vec{X_\alpha} - \vec{A}_1 \cdot \vec{D_\alpha} \tag{8}$$
$$\vec{X_2} = \vec{X_\beta} - \vec{A}_2 \cdot \vec{D_\beta} \tag{9}$$
$$\vec{X_3} = \vec{X_\delta} - \vec{A}_3 \cdot \vec{D_\delta} \tag{10}$$
$$\vec{X}(t+1) = \left(\vec{X_1} + \vec{X_2} + \vec{X_3}\right)/3 \tag{11}$$

4) ATTACKING PREY AND SEARCH FOR PREY

According to (3), the decrease of $\vec{a}$ value will cause the fluctuation of $\vec{A}$ value. When $\vec{A}$ is in the [-1,1] interval, the next position of the agent can be at any position between the wolf and its prey. On the one hand, $|A| < 1$ forced the wolf to attack its prey. On the other hand, $|A| > 1$ forces the wolf to separate from its prey, hoping to find more suitable prey.

B. PARTICLE SWARM OPTIMIZATION (PSO)

Particle swarm optimization was first proposed by Kennedy and Eberhart [35]. The algorithm is composed of particles, each of which has only two attributes: velocity and position. Update the velocity and position of each particle through (12) and (13), and finally obtain the optimal solution.

$$v_i^{k+1} = v_i^k + c_1 r_1 \left(P_{best}^k - x_i^k\right) + c_2 r_2 \left(g_{best} - x_i^k\right) \tag{12}$$
$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{13}$$

where i refers to the particle in the swarm. k is the number of iterations. r_1 and r_2 values represent random numbers in the range [0,1]. The coefficients $c_1$and c_2 represent the optimization parameters. P_{best} represents the best position of the individual, g_{best} represents the best position of the population.

C. HYBRID PSO-GWO (HPSOGWO)

Although PSO technology is subject to some limitations, it has some advantages, such as simplicity, durability, and easy implementation. The disadvantage is that it is easy to fall into a local minimum [36]. And, GWO algorithm has the characteristics of strong convergence performance, few parameters, and easy implementation. It avoids local trapping and maintains the balance between exploration and exploitation. But there is a lack of communication between individual positions and group positions. In this way, both these extraordinary features of PSO and GWO are incorporated into the algorithm of HPSOGWO [37].

The modified governing equations of the hybrid algorithm are as follows:

$$\vec{D_\alpha} = \left|\vec{c_1} \cdot \vec{X_\alpha} - \omega * \vec{X}\right| \tag{14}$$
$$\vec{D_\beta} = \left|\vec{c_2} \cdot \vec{X_\beta} - \omega * \vec{X}\right| \tag{15}$$
$$\vec{D_\delta} = \left|\vec{c_3} \cdot \vec{X_\delta} - \omega * \vec{X}\right| \tag{16}$$

The velocity and position update equations obtained by combining PSO and GWO algorithm are as follows:

$$v_i^{k+1} = \omega * \left(v_i^k + c_1 r_1 \left(x_1 - x_i^k\right) + c_2 r_2 \left(x_2 - x_i^k\right) + c_3 r_3 \left(x_3 - x_i^k\right)\right) \tag{17}$$
$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{18}$$

In (17), ω represents the inertia weight parameter.

D. FRAMEWORK CONSTRUCTION OF 1D CNN MODEL

In this study, a new 1D CNN model is proposed for the anomaly detection of rudder systems. Specifically, the 1D CNN model designed in this paper includes three convolution layers, two pooling layers, one flatten layer, two dropout layers and two dense layers. The structure of 1D CNN is described as follows:

1) INPUT LAYER

The input layer is used to accept preprocessed rudder system test data.

2) CONVOLUTION LAYER

This layer has multiple filters and performs most of the computation. The convolution is passing through this layer. Generally speaking, convolution generates a new spectrum representation by sliding the kernel with a certain ''stride'' across the entire spectrum. Then the output feature map is passed to the activation function to achieve nonlinear changes in the network layer. The activation function used in the convolutional layer designed in this paper is ReLU. Obtain the number of filters n_i and filter size S_F in the convolutional layers through the HPSOGWO optimization algorithm.

3) POOLING LAYER

It is usually used for feature dimensionality reduction to achieve the purpose of compressing data and reducing the number of parameters in the training process, thereby reducing overfitting and improving the fault tolerance of the model. There are two main types of pooling layers: maximum pooling and average pooling. This article uses maximum pooling. Also, HPSOGWO algorithm is used to determine the pool size S_P.

4) FLATTEN LAYER

The flatten layer is used to ''flatten'' the output of the pooling layer, that is, to convert it into a vector. It is usually used for the transition from the convolution layer or pooling layer to the fully connected layer.

5) DROPOUT LAYER

The dropout layer achieves the effect of using random deactivation of hidden units to prevent overfitting. The introduction of this randomness forces the network to become redundant, so that the network does not match the training samples well, thereby increasing the generalization ability of the network. In the training process, we randomly sample according to a certain probability to change the network structure, which is equivalent to training different networks. For the two dropout layers in this experiment, select two dropout ratio values of 0.25 and 0.5, respectively.

6) DENSE LAYER

This layer contains a large number of neurons, which are used to connect neurons in this layer with those in other layers. When the activation function of the dense layer is set to Softmax, the layer can be regarded as a classification layer [38]. The model designed in this paper has two dense layers. The number of units C in the first dense layer is determined by the optimization algorithm. And the second dense layer is used for classification. This experiment needs to divide the test results of the rudder system into 11 categories, so the number of units in this layer is 11.

III. EXPERIMENT VALIDATION

A. RUDDER SYSTEM TESTING EQUIPMENT

This paper establishes an automatic test platform suitable for rudder system testing, which can automatically complete rudder system data collection and processing, performance index testing and other functions. The test equipment is mainly composed of a main control industrial computer, power supply, capture card, signal conditioning circuit, driver, pneumatic steering gear and multi-function data acquisition card. It can test four sets of rudder systems separately, which greatly improves the test efficiency. The system automatically analyzes the rudder system parameters returned to the industrial control computer to meet the requirements of anomaly detection.

It is well known that the rudder must be evaluated before use to ensure that all parameters are in normal condition. If the parameter is abnormal, it needs to be analyzed and adjusted immediately, which is directly related to the performance of the steering gear. Therefore, how to improve the accuracy of the anomaly detection of rudder system is very necessary. Fig. 2 shows a brief flow chart of the rudder system test.

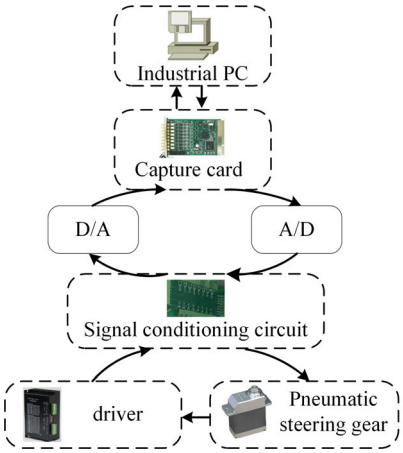


FIGURE 2. Overview of the rudder system test process.

B. DATASET DESCRIPTION

In this study, 19490 historical test data are used. The test items of the pneumatic rudder system included in this dataset are as follows: transient time, overshoot, steady-state errors, hysteresis, band width, etc. These parameters reflect the dynamic and static characteristics of the pneumatic rudder system, and are a comprehensive index for evaluating the performance of the rudder system.

Here, the columns in the dataset are called ''features.'' There are 10 types of features in our work, corresponding to 10 types of errors. The ''labels'' indicate 11 test statuses, consisting of a qualified type and 10 types of faults. Furthermore, the test data of each rudder system consist of 10 characteristic values and a label. The overview of the dataset is shown in Table 4.

TABLE 4. Dataset overview.

Parameters	Detailed
Transient Time	The shortest time from the initial steady state to the new equilibrium state of the elastic load rudder system when the step control signal is input.
Overshoot	The percentage of the ratio of the maximum output minus the steady-state value to the steady-state value of the elastic load rudder system when the step control signal is input.
Steady-State Errors	The deviation of a system from a steady state to a new steady state of the elastic load rudder system when the step control signal is input.
No-Load Band Width	Frequency band width of rudder system input sinusoidal control signal.
Load Band Width	Frequency band width of elastic load rudder system input sinusoidal control signal.
Hysteresis Characteristics	The inconsistency degree between the positive stroke characteristic and the back stroke characteristic of the elastic load rudder system when the triangular wave control signal is input.
Zero Precision1	Rudder feedback signal compensation of rudder system input positive control signal and then reset to zero.
Zero Precision2	Rudder feedback signal compensation of rudder system input negative control signal and then reset to zero.
Electrical Stroke1	Rudder feedback voltage when the positive DC control signal is input.
Electrical Stroke2	Rudder feedback voltage when the negative DC control signal is input.

C. EXPERIMENTAL 1D CNN ARCHITECTURE

1) EXPERIMENTAL SETUP

In this experiment, we hope to obtain the best network architecture, so the HPSOGWO algorithm is used to fine-tune the hyper-parameters of 1D CNN.

The experimental setup is as follows: The number of samples selected by one training of the neural network is called batch size. The loss function will vibrate and not converge due to the unreasonable number of batch-sizes; increasing the number of batch-sizes within a certain range can not only shorten the training time of neural network, but also improve the accuracy of training. Therefore, we define the hyper-parameter as: $batch - sizes = 512$. The training of 1D CNN model is performed using the adaptive moment estimation (Adam) algorithm [39]. In the optimization, the fitness function selects the accuracy value. Each neural network is trained for 100 epochs. The parameters of HPSOGWO algorithm are set as follows: maximum number of iterations is 10, $c_1 = c_2 = c_3 = 0.5$, $\omega \in [0.5, 1)$. Moreover, the dataset are partitioned by 10-fold cross validation. The specific method is as follows: first, preprocess the data in the dataset D; second, randomly shuffle the order; then divide it into 10 mutually exclusive subsets of similar size. The training set is composed of the union of 9 subsets (T), and the testing set is composed of the remaining subsets (Ψ). Therefore, 10 times of training and testing are carried out, and the results of 10 tests are averaged to get the final result. The above experimental settings are used to analyze the performance of the optimized algorithms of HPSOGWO, PSO and GWO.

The process of using HPSOGWO algorithm to optimize CNN can be summarized as follows.

1) Set the population of grey wolf $Pop = 30$, the maximum number of iterations $Max\_iterations = 10$, the optimization dimension $Dim = 6$, and set the upper and lower boundaries of the hyper-parameters according to Table 5.
2) Initialize the population of grey wolf $X_i (i = 1, 2, \ldots, 6)$, which is the value of six hyper-parameters.
3) Population boundary check. Prevent the initialized hyper-parameters exceeding from the upper and lower boundaries.
4) Evaluate the fitness of the population. The specific process is to substituting the initialized hyper-parameters into CNN and training CNN to obtain the accuracy of multi-fault classification of the rudder system. The accuracy is the fitness value.
5) Sort the fitness.
6) Sort the population position according to the fitness. Specifically, let the fitness correspond to the hyper-parameters one by one.
7) Set the best three grey wolves $X_\alpha, X_\beta, X_\delta$, that is, the three sets of hyper-parameters due to the three highest fitness.
8) $Iterations < 10$, if $Pop < 30$, update parameters a, A, c, ω, calculate the velocity and position of the search agents according to (17) and (18).
9) Increase the number of Pop by 1.
10) Go back to Step 8-9, until $Pop \geq 30$.
11) Go back to Step 3-6. Update the fitness corresponding to X_α to the global optimal solution, and X_α is the global optimal position.
12) Increase the number of *Iterations* by 1.
13) Go back to Step 8-12, until $Iterations \geq 10$. The final highest fitness is the global optimal solution, and the global optimal position is the optimal hyper-parameters for constructing the network.
14) In order to obtain a more reliable and stable CNN model, this paper uses 10-fold cross-validation training, replaces the training set 10 times, repeats the above steps, and finally obtains 10 sets of optimal hyper-parameters.

2) IMPACT OF THE SEARCH AGENT SIZES ON HPSOGWO

The influence of the number of search agents in the HPSOGWO algorithm on the experimental results is analyzed by the experiment. The following search agent sizes are investigated: $SA = \{10, 20, 30, 40\}$, and Table 5 summarizes all hyper-parameters that need to be fine-tuned to obtain the best classification results.

Use the testing set to evaluate the classification performance of the 1D CNN optimized by the HPSOGWO algorithm. As mentioned earlier, we perform 10-fold cross validation. Specifically, 10 repeated independent experiments are carried out for different search agent sizes. In Fig. 3, we show the classification accuracy of the HPSOGWO algorithm optimized for all search agent sizes. It can be proved

TABLE 5. The search range of the hyper-parameters.

Hyper-Parameter	Range
Number of filters n_1	$(2^3;2^6)$
Number of filters n_2	$(2^5;2^7)$
Number of filters n_3	$(2^6;2^8)$
Number of units C	$(2^6;2^9)$
Filter Size S_F	(2;4)
Pool Size S_P	(2;4)

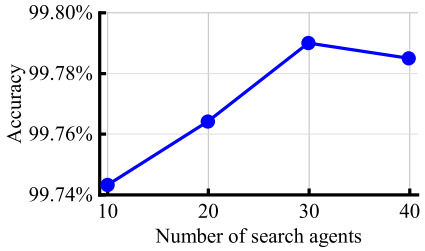


FIGURE 3. In the HPSOGWO algorithm, the classification accuracy of different number of search agents on Ψ.

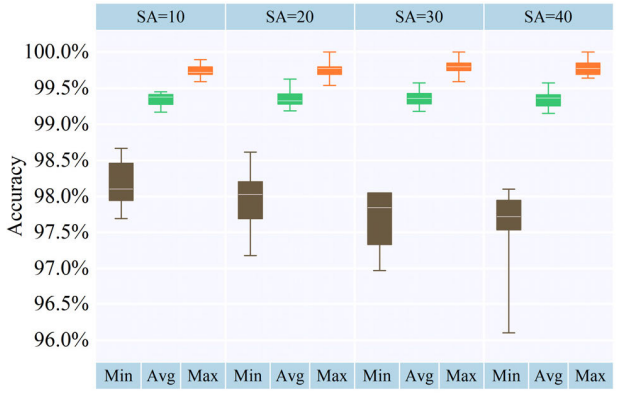


FIGURE 4. Minimum, average and maximum accuracy obtained for Ψ using the optimized 1D CNN. The white line in the box-plot is the median line.

TABLE 6. Comparison of averages value.

Number of search agents	Average value of Min	Average value of Avg	Average value of Max
SA=10	98.158%	99.351%	99.743%
SA=20	97.932%	99.350%	99.764%
SA=30	**97.650%**	**99.357%**	**99.790%**
SA=40	97.558%	99.348%	99.784%

from the analysis graph that when 30 search agents are used, the classification accuracy is the highest.

In order to verify the consistency of the hyper-parameter quality of HPSOGWO algorithm obtained in 10 independent experiments, the box plot shown in Fig. 4 is drawn. Table 6 lists the average values of various results. Table 7 lists the accuracy results obtained from 10 independent experiments conducted with different search agents. In Fig. 4 and Table 6, we show that the performance of the accuracy in

TABLE 7. The results of accuracy over 10 independent runs.

Run	SA=10	SA=20	SA=30	SA=40
1	99.692%	99.795%	99.795%	99.743%
2	99.590%	99.538%	99.590%	99.641%
3	99.795%	99.795%	99.846%	99.846%
4	99.846%	99.743%	99.795%	99.795%
5	99.692%	99.743%	99.692%	99.692%
6	99.795%	99.846%	99.846%	99.897%
7	99.897%	100.000%	100.000%	100.000%
8	99.692%	99.795%	99.743%	99.743%
9	99.743%	99.692%	99.795%	99.795%
10	99.692%	99.692%	99.795%	99.692%
Avg.	**99.743%**	**99.764%**	**99.790%**	**99.784%**

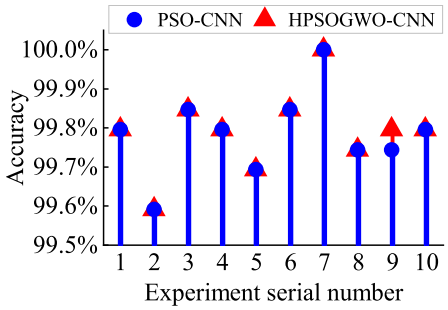
testing set when using HPSOGWO algorithm to optimize 1D CNN network under four different search agent sizes. Interestingly, increasing the number of search agents may lead to the deterioration of the lowest accuracy in ten optimization experiments. In other words, increasing the size of search agents may not be able to effectively improve the initial low-quality positions. In addition, it can be seen from Table 7 that when the number of search agents is 30, the average accuracy of 10 experiments can reach 99.790%. Compared with the number of search agents of 10, 20 and 40, the accuracy is increased by 0.047%, 0.026% and 0.006%, respectively.

3) ANALYSIS OF THREE OPTIMIZATION ALGORITHMS

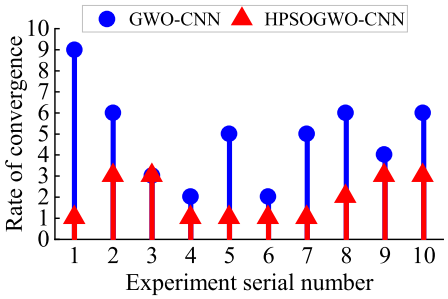
In order to verify the superiority of the HPSOGWO algorithm in the optimization of 1D CNN hyper-parameters over the PSO algorithm and the GWO algorithm, we conducted the following two experiments. The population size of the three different algorithms is 30. As shown in Fig. 5 (a), compared with the PSO-CNN algorithm, the HPSOGWO-CNN algorithm has higher classification accuracy in Ψ when optimizing 1D CNN in 10 independent repeated experiments.

Another excellent feature of our proposed HPSOGWO-CNN model is that it converges faster than the GWO-CNN model. This is because, compared with the GWO algorithm, the HPSOGWO algorithm adopts the velocity and position update formula of the PSO algorithm, which improves the exploration ability of the GWO algorithm. In Fig. 5 (b), we show the number of iterations required in each of the 10 independent repeated experiments when the fitness function value reaches the optimal value for the first time. For example: in the experiment with serial number 1, when the HPSOGWO algorithm is used to optimize 1D CNN, the fitness function value reaches the best value after the first iteration is completed; however, under the same conditions of T and Ψ, using the GWO algorithm, the fitness function value can reach the best value at the 9th iteration.

Overall, it can be seen from this that the HPSOGWO algorithm is an effective technique for automatic selection technology of hyper-parameters in neural networks, and its performance is superior to that of the other two algorithms alone.

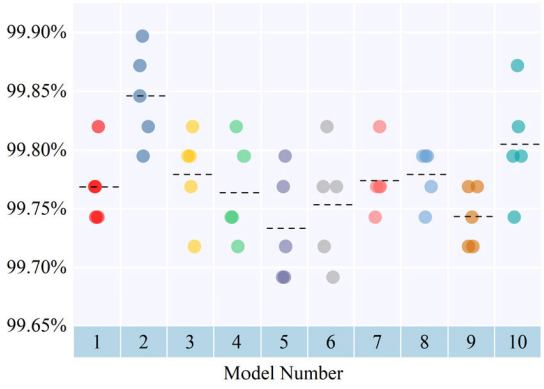


(a) Comparison of accuracy between PSO-CNN and HPSOGWO-CNN

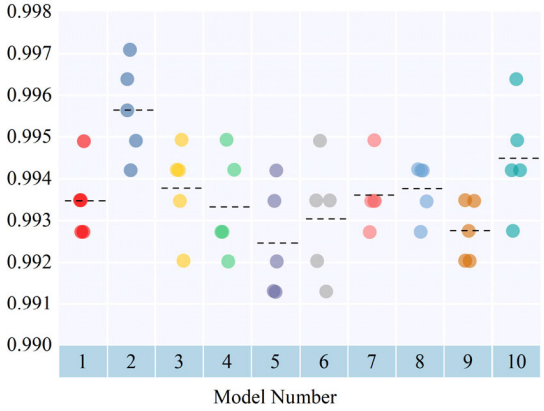


(b) Comparison of rate of convergence between GWO-CNN and HPSOGWO-CNN

FIGURE 5. Performance comparison results of optimization algorithms. (a). In 10 independent repeated experiments, the accuracy of PSO-CNN and HPSOGWO-CNN model on Ψ. (b). In 10 independent repeated experiments, the rate of convergence of PSO-CNN and HPSOGWO-CNN model on Ψ.



(a) Comparison of accuracy.



(b) Comparison of Kappa.

FIGURE 6. After full training (epoch = 200), the accuracy and kappa of the 10 performing hyper-parameter combinations are shown. For each combination, the dots represent the experimental results, whereas the dotted line represents the average of all five replicates.

4) PERFORMANCE EVALUATION

Through the above experiments using 10-fold cross validation, ten different optimal solutions were finally obtained in ten experiments. Therefore, the 10 best solutions of the HPSOGWO-CNN model are shown in Table 8.

TABLE 8. The ten best performing solutions of the HPSOGWO-CNN model.

Model Number	n_1	n_2	n_3	C	S_F	S_P
1	34	87	126	392	2	2
2	**60**	**32**	**64**	**351**	**2**	**2**
3	32	109	85	190	2	2
4	57	32	64	105	2	2
5	32	32	112	408	2	2
6	37	128	64	512	2	2
7	60	32	64	324	2	2
8	45	80	139	377	2	2
9	64	128	189	484	2	2
10	64	128	64	512	2	2

In this experiment, we divide the training set and the testing set at a ratio of 8:2. Furthermore, due to the differences in initialization weights and deviations, repeated training of the network with the same hyper-parameters will result in different training results each time, in other words, different classification performance obtained [27]. Therefore, in order to obtain a more stable network structure, we performed five repetitive training on these ten best combinations, as shown in Figs. 6. Through the analysis of accuracy and kappa coefficient, it is concluded that the average performance of model 2 is the best.

The hyper-parameters of model 2 are used to construct the 1D CNN designed in this experiment. As shown in Table 9, the 1D CNN consists of 3 1D convolution layers, 2 1D pooling layers and 1 flatten layer in the convolution part. The first layer, the second layer and the fourth layer of the 1D CNN structure are convolution layers, which contain 60, 32 and 64 1D filters respectively. The size of each convolution layer filter is 2, the stride size is 1, and the output uses the ReLU activation function. The third layer and the fifth layer are the max pooling layer, and the pool size is 2. The sixth layer is the flatten layer. In general, data are input in the form of 10×1 vectors and output in 192×1 vectors through these layers. Next, as shown in Fig. 7, all the output data are connected into one vector, which is fed into a dense layer with 351 units. And dropout ratio value is selected to be as 0.25. The ReLU activation function is used for each unit. The last dense layer is the output layer, and dropout ratio value is selected to be as 0.5. Using Softmax as the activation function, 11 classification results are obtained. These 11 types of results divide the data into qualified types and 10 types of faults, thus realizing the anomaly detection of the rudder system.

TABLE 9. The architecture of 1D CNN.

Layer Number	Layer Type	Hyperparameters	
Layer 1	Convolution Layer	Filter Size	2
		Number of filters	60
		Activation Layer	ReLU
		Stride	1
		Padding	same
Layer 2	Convolution Layer	Filter Size	2
		Number of filters	32
		Activation Layer	ReLU
		Stride	1
		Padding	same
Layer 3	Pooling Layer	Pool Size	2
		Pooling type	Max-Pooling
		padding	same
Layer 4	Convolution Layer	Filter Size	2
		Number of filters	64
		Activation Layer	ReLU
		stride	1
		Padding	same
Layer 5	Pooling Layer	Pool Size	2
		Pooling type	Max-Pooling
		padding	same
Layer 6	Flatten Layer		
Layer 7	Dropout Layer	Rate	0.25
Layer 8	Dense Layer	Activation Layer	ReLU
		units	351
Layer 9	Dropout Layer	Rate	0.5
Layer 10	Dense Layer	Activation Layer	Softmax
		units	11

IV. ANALYSIS AND DISCUSSION

A. EVALUATION METRICS

The following parameters are used to evaluate the performance of the proposed model: accuracy, precision, recall, F-score and kappa.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{19}$$

$$\Pr ecision = \frac{TP}{TP + FP} \tag{20}$$

$$\mathrm{Re}\, call = \frac{TP}{TP + FN} \tag{21}$$

$$F - score = 2 \times \frac{\Pr ecision \times \mathrm{Re}\, call}{\Pr ecision + \mathrm{Re}\, call} \tag{22}$$

$$kappa = \frac{N \sum_{i=1}^{r} x_{ii} - \sum_{i=1}^{r} (x_{i+} \times x_{+i})}{N^2 - \sum_{i=1}^{r} (x_{i+} \times x_{+i})} \tag{23}$$

where TP, TN, FP and FN represent true positive, true negative, false positive and false negative, respectively. TP refers to the normal category that is classified as normal, and FN refers to the normal category that is classified as abnormal. On the contrary, an abnormal category classified as a normal category is called FP, and an abnormal category classified as an abnormal category is called TN.

B. COMPARISON OF PERFORMANCE INDEXES OF DIFFERENT MODELS

In order to thoroughly evaluate the proposed model, we compared the proposed HPSOGWO-CNN model with traditional machine learning algorithms, other neural network algorithms, unoptimized CNN, and CNN models optimized by different optimization algorithms. The experimental results of classification performance of each model are the average of 5 repeated experiments.

There are various ways to improve GWO algorithm [40], [41]. In order to analyze the advantages of HPSOGWO algorithm in optimizing CNN, this paper compares it with MGWO [42], RW-GWO [43] and WdGWO [44]. Among them, MGWO and RW-GWO both improved the parameter update mechanism, and WdGWO improved the grey wolf individual position update mechanism.

The model settings for the comparison experiments are as follows: The BP model uses a structure that removes the convolution part of the HPSOGWO-CNN model. Specifically, we can analyze the influence of the convolution part of HPSOGWO-CNN on the classification performance by extracting the BP model formed after the flatten layer of the HPSOGWO-CNN model. The values of the hyper-parameters of the CNN model are the middle value of the optimization range of each hyper-parameter. In addition, the PSO-CNN, GWO-CNN, MGWO-CNN, RW-GWO-CNN, and WdGWO-CNN models use the same training set and testing set as the HPSOGWO-CNN model in the experiment. Specifically, the hyper-parameters of the models constructed by different optimization algorithms are shown in Table 10.

TABLE 10. Hyperparameters of different models.

	n_1	n_2	n_3	C	S_F	S_P
CNN	28	48	96	224	3	3
PSO-CNN	47	71	116	478	3	2
GWO-CNN	15	113	132	116	2	2
MGWO-CNN	48	42	91	113	2	3
WdGWO-CNN	16	60	114	200	2	2
RW-GWO-CNN	48	53	161	160	2	2
HPSOGWO-CNN	60	32	64	351	2	2

In order to analyze the complexity of the model, the space complexity of the proposed model is analyzed. Space complexity is mainly affected by the number of parameters in the model. The larger the space complexity, the larger the amount of data needed for training the model.

TABLE 11. Total parameters of different models.

	Total parameters
CNN	63819
PSO-CNN	207185
GWO-CNN	80851
MGWO-CNN	50007
WdGWO-CNN	86633
RW-GWO-CNN	101723
HPSOGWO-CNN	**79827**

Table 11 shows that the total parameters of the MGWO-CNN model are the lowest, and that of the PSO-CNN model

TABLE 12. Comparison of classification performance of different models.

	Accuracy	Precision	Recall	F-score	Kappa	Time(s)
KNN	96.280%	96.016%	86.481%	89.791%	0.88952	0.902
SVM	97.948%	97.857%	92.129%	94.412%	0.94047	0.647
BP	99.266%	98.600%	97.876%	98.190%	0.97923	22.137
CNN	99.564%	99.135%	98.529%	98.809%	0.98767	61.256
PSO-GNN	99.631%	99.214%	99.037%	99.139%	0.98957	74.608
GWO-CNN	99.728%	99.439%	99.300%	99.362%	0.99231	78.074
MGWO-CNN	99.764%	99.576%	99.224%	99.391%	0.99332	59.468
WdGWO-CNN	99.774%	99.585%	99.267%	99.418%	0.99361	61.963
RW-GWO-CNN	99.779%	99.571%	99.293%	99.426%	0.99376	69.176
HPSOGWO-CNN	**99.846%**	**99.748%**	**99.498%**	**99.618%**	**0.99565**	**56.288**

are the highest. Among them, the total parameters of the proposed model are 79827, which is relatively low.

The classification performance index shown in Table 12 is the average value obtained from 5 repeated experiments. Compared with KNN and SVM, the accuracy of our proposed model is improved by 3.566% and 1.898%, respectively. Recall, F-score and Kappa all perform poorly in the two traditional machine learning algorithms. Compared with the BP model, the accuracy is increased by 0.58%, which indicates that the feature extraction of data through the convolution layer can effectively improve the classification accuracy. The accuracy of the CNN model without optimization is 0.282% lower than that of the CNN model optimized by the HPSOGWO algorithm. In other words, the optimization algorithm is indeed effective in improving the classification performance.

In the experiment, six optimization algorithms of PSO, GWO, MGWO, WdGWO, RW-GWO and HPSOGWO are compared to optimize the classification performance of CNN. The results show that the PSO-CNN model performs the worst among the 5 performance index evaluations of accuracy, precision, recall, F-score and kappa, which are all lower than the CNN model optimized by GWO and improved GWO. Among them, the accuracy of the HPSOGWO-CNN model is 99.846%, the precision is 99.748%, the recall is 99.498%, and the F-score is 99.618%. Kappa is 0.99565, which is the closest to 1 compared with the other 9 models. Among the 4 improved GWO algorithms, the evaluation indicators of HPSOGWO-CNN stand out.

In addition, we report the program execution time of different models. Specifically, the running time required by GWO-CNN is 78.074s, and the running time of the 4 improved GWO algorithms has been shortened. Among them, the running time of HPSOGWO-CNN is the shortest, reaching 56.288s. On the premise of ensuring high-quality, the CNN model structure optimized by the HPSOGWO algorithm can reduce its execution time.

The non-parametric statistical hypothesis tests of the proposed model and other models are shown in Table 13. The progressive significance p_1 is obtained using the Mann-Whitney test. It is an approximate normal calculation probability and is suitable for data with a large sample size. The precision significance p_2 uses Kruskal test, which is the probability obtained by the exact test, and is suitable for data with a small sample size. The p- value reflects whether the difference between the two models is statistically significant. p- value <0.05 indicates that there is a significant difference between the two models. Compared with the proposed model, the non-parametric test values of the two methods are both less than 0.05, indicating that the comparative experiment in this paper is meaningful [45].

TABLE 13. Non-parametric statistical hypothesis tests of proposed method and other methods.

Compare Techniques	p_1 - value	p_2 - value
	Mann-Whitney Test	Kruskal Test
KNN	0.008	0.005
SVM	0.008	0.005
BP	0.008	0.009
CNN	0.008	0.008
PSO-CNN	0.008	0.009
GWO-CNN	0.008	0.009
MGWO-CNN	0.032	0.019
WdGWO-CNN	0.008	0.005
RW-GWO-CNN	0.032	0.019

In summary, the proposed HPSOGWO-CNN model has the best level and excellent performance in all aspects of these 10 models. It is an outstanding model for the anomaly detection of the rudder system.

C. COMPARISON OF PERFORMANCE INDEXES OF DIFFERENT CATEGORIES

In our work, the HPSOGWO-CNN model performed well in the overall evaluation performance indicators. Then, in order to further analyze the classification of each category, we plotted the confusion matrix of the HPSOGWO-CNN model in Table 14, as well as the accuracy, precision and F-score of each category of each model in Table 15. The confusion matrices of the above experiments all selected the best one from these five experiments. Class Q in Table 15 indicates that the rudder system is qualified, whereas class FA to FJ mean the single fault (fault A to fault J).

It is worth noting that the KNN and SVM machine learning algorithms are greatly affected by the imbalance of data, and they are less effective in the classification of FE, FF, FG and FJ. Especially the KNN model, in the FF classification,

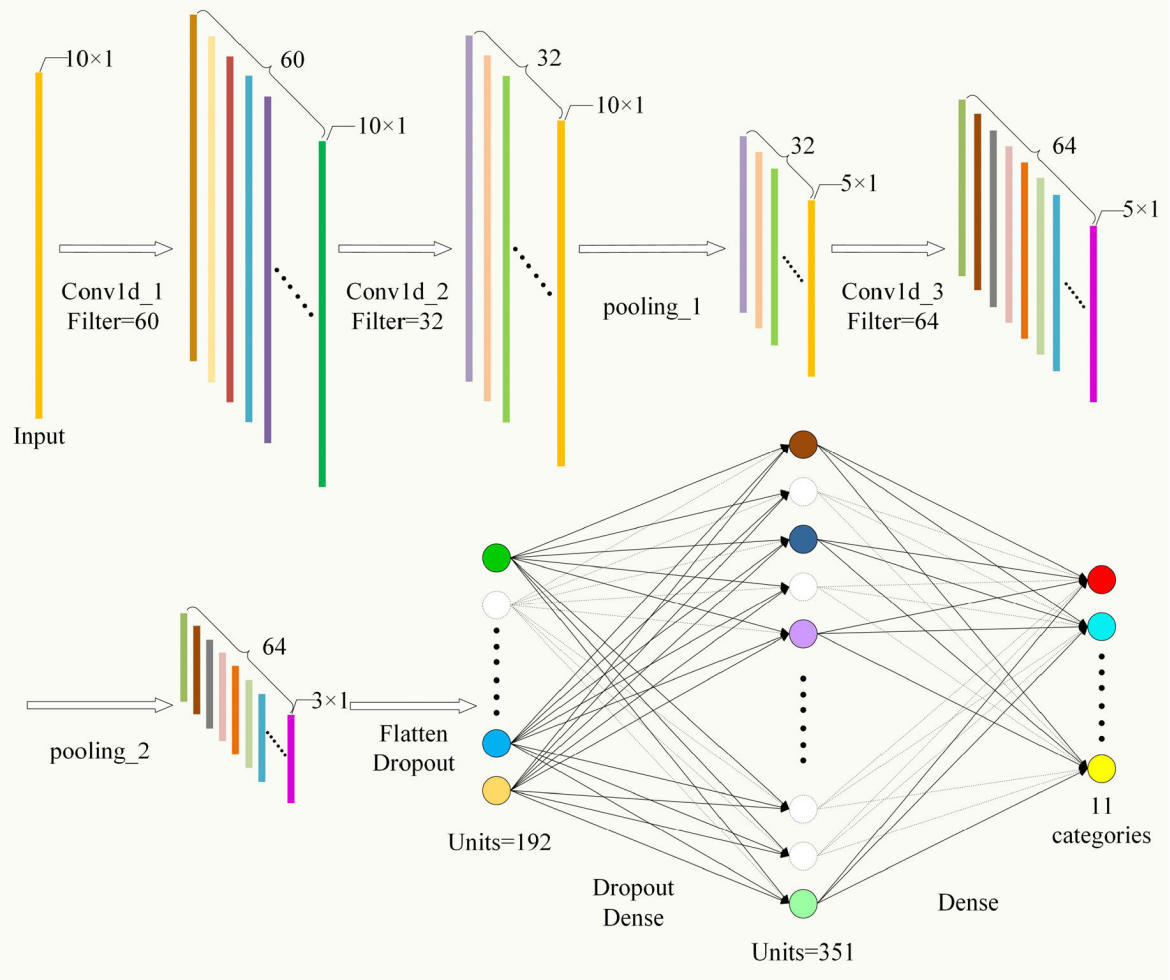


FIGURE 7. Structure diagram of the proposed 1D CNN.

TABLE 14. A confusion matrix of the HPSOGWO-CNN model.

		Prediction results										
		Q	FA	FB	FC	FD	FE	FF	FG	FH	FI	FJ
Actual results	Q	3119			**1**							
	FA		154									
	FB			45								
	FC				50							
	FD					155						
	FE						55					
	FF							71				
	FG	**2**							54			
	FH									43		
	FI										35	
	FJ	**1**										113

the accuracy is less than 50%, cannot achieve the classification performance.

When the neural network model is used for classification, it is less affected by data imbalance. Basic neural network models, such as BP and CNN, improve the classification performance of each type of fault to more than 90%.

Next, compare the classification performance of the model obtained by using PSO and GWO to optimize the CNN hyper-parameters. In terms of accuracy, FA, FB, FC, FH, FI, FJ, PSO-CNN and GWO-CNN have reached 100%. Among FD, FE, FF, and FG, the classification accuracy of GWO-CNN is better than that of PSO-CNN. In terms of precision and F-score, in addition to the classification performance of FA and FE, the GWO-CNN model is equal to or better than the PSO-CNN model in the classification performance of the other nine abnormal categories. It can

TABLE 15. Accuracy, precision and F-score of each class of each model.

Category name		Q	FA	FB	FC	FD	FE	FF	FG	FH	FI	FJ
Number of training set samples		12549	604	151	169	615	242	269	240	170	182	401
Number of testing set samples		3120	154	45	50	155	55	71	56	43	35	114
accuracy	KNN	99.10%	96.10%	100.00%	100.00%	96.77%	72.73%	42.25%	76.79%	100.00%	100.00%	67.54%
	SVM	99.49%	100.00%	100.00%	100.00%	100.00%	83.64%	64.79%	83.93%	100.00%	100.00%	81.58%
	BP	99.58%	99.35%	100.00%	100.00%	98.71%	98.18%	94.37%	92.86%	100.00%	100.00%	98.25%
	CNN	99.74%	100.00%	100.00%	100.00%	99.35%	100.00%	100.00%	91.07%	97.67%	100.00%	100.00%
	PSO-CNN	99.90%	100.00%	100.00%	100.00%	99.35%	96.36%	97.18%	92.86%	100.00%	100.00%	100.00%
	GWO-CNN	99.84%	100.00%	100.00%	100.00%	100.00%	98.18%	100.00%	96.43%	100.00%	100.00%	100.00%
	MGWO-CNN	99.87%	100.00%	100.00%	100.00%	100.00%	100.00%	98.59%	94.64%	100.00%	100.00%	100.00%
	WdGWO-CNN	99.90%	100.00%	100.00%	100.00%	100.00%	100.00%	98.59%	96.43%	100.00%	100.00%	100.00%
	RW-GWO-CNN	99.87%	100.00%	100.00%	100.00%	99.35%	100.00%	98.59%	96.43%	100.00%	100.00%	**100.00%**
	HPSOGWO-CNN	**99.97%**	**100.00%**	**100.00%**	**100.00%**	**100.00%**	**100.00%**	**100.00%**	**96.43%**	**100.00%**	**100.00%**	99.12%
precision	KNN	96.41%	96.10%	97.83%	90.91%	95.54%	95.24%	96.77%	100.00%	100.00%	94.59%	92.77%
	SVM	98.01%	97.47%	95.74%	94.34%	96.27%	100.00%	100.00%	100.00%	100.00%	94.59%	100.00%
	BP	99.55%	98.71%	100.00%	94.34%	97.45%	94.74%	98.53%	100.00%	100.00%	100.00%	100.00%
	CNN	99.81%	98.72%	100.00%	98.04%	99.35%	98.21%	97.26%	100.00%	100.00%	100.00%	98.28%
	PSO-CNN	99.71%	100.00%	100.00%	98.04%	99.35%	100.00%	98.57%	100.00%	100.00%	100.00%	100.00%
	GWO-CNN	99.90%	99.35%	100.00%	98.04%	99.36%	98.18%	98.61%	100.00%	100.00%	100.00%	100.00%
	MGWO-CNN	99.87%	100.00%	100.00%	98.04%	98.10%	100.00%	100.00%	100.00%	100.00%	100.00%	100.00%
	WdGWO-CNN	99.90%	100.00%	100.00%	98.04%	99.36%	100.00%	100.00%	100.00%	100.00%	100.00%	99.13%
	RW-GWO-CNN	99.87%	100.00%	100.00%	98.04%	100.00%	96.49%	98.59%	100.00%	100.00%	100.00%	100.00%
	HPSOGWO-CNN	**99.90%**	**100.00%**	**100.00%**	**98.04%**	**100.00%**	**100.00%**	**100.00%**	**100.00%**	**100.00%**	**100.00%**	**100.00%**
F-score	KNN	97.74%	96.10%	98.90%	95.24%	96.15%	82.47%	58.82%	86.87%	100.00%	97.22%	78.17%
	SVM	98.74%	98.72%	97.83%	97.09%	98.10%	91.09%	78.63%	91.26%	100.00%	97.22%	89.86%
	BP	99.57%	99.03%	100.00%	97.09%	98.08%	96.43%	96.40%	96.30%	100.00%	100.00%	99.12%
	CNN	99.78%	99.35%	100.00%	99.01%	99.35%	99.10%	98.61%	95.33%	98.82%	100.00%	99.13%
	PSO-CNN	99.81%	100.00%	100.00%	99.01%	99.35%	98.15%	97.87%	96.30%	100.00%	100.00%	100.00%
	GWO-CNN	99.87%	99.68%	100.00%	99.01%	99.68%	98.18%	99.30%	98.18%	100.00%	100.00%	100.00%
	MGWO-CNN	99.87%	100.00%	100.00%	99.01%	99.04%	100.00%	99.29%	97.25%	100.00%	100.00%	100.00%
	WdGWO-CNN	99.90%	100.00%	100.00%	99.01%	99.68%	100.00%	99.29%	98.18%	100.00%	100.00%	99.56%
	RW-GWO-CNN	99.87%	100.00%	100.00%	99.01%	99.68%	98.21%	98.59%	98.18%	100.00%	100.00%	**100.00%**
	HPSOGWO-CNN	**99.94%**	**100.00%**	**100.00%**	**99.01%**	**100.00%**	**100.00%**	**100.00%**	**98.18%**	**100.00%**	**100.00%**	99.56%

be analyzed that the GWO algorithm is better than the PSO algorithm in optimizing CNN.

Furthermore, in the comparative experiments of four improvements of GWO algorithms, the proposed HPSOGWO-CNN model has the best overall classification performance. The specific performance is as follows: Among the 11 types of steering gear failures, 8 types (FA, FB, FC, FD, FE, FF, FH, FI) have an accuracy of 100%, 9 types (FA, FB, FD, FE, FF, FG, FH, FI, FJ) have an accuracy of 100%, and 7 types (FA, FB, FD, FE, FF, FH, FI) of F-score have reached 100%.

Among them, in the analysis of accuracy, precision and F-score of the HPSOGWO-CNN model, FA, FB, FD, FE, FF, FH, FI categories can all reach 100%. The sample size of the 7 types of test sets is less than 200, indicating that for small samples, the model can still achieve the correct classification.

V. CONCLUSION

Aiming at the anomaly detection of the rudder system, an automatic test platform suitable for the rudder system is established, and a new model named HPSOGWO-CNN is proposed for anomaly diagnosis. As we all know, the performance of neural networks directly depends on their hyper-parameters, and the artificially designed hyper-parameters cannot achieve the best network structure, so this experiment uses the HPSOGWO algorithm to complete the construction of CNN. The designed CNN is used for feature extraction and classification of experimental data for rudder testing. The results of Section 3 show that in the experiment of optimizing the hyper-parameters of the CNN, compared with the PSO and GWO algorithms, the HPSOGWO algorithm has obvious advantages in accuracy and the rate of convergence. In other words, it is a fast and efficient algorithm for hyper-parameters automatic selection of neural network. The results in Section 4 show that our proposed model can achieve 99.846% accuracy, 99.748% precision, 99.498% recall, 99.618% F-score, 0.99565 Kappa in the multi-fault classification of the rudder system, and the classification performance is hardly affected by sample imbalance.

REFERENCES

[1] Z. Peng, B. Li, and Q. Liu, "Design and application of an automatic testing system for servo-valve," *Int. J. Adv. Comput. Technol.*, vol. 4, no. 23, pp. 747–754, 2012.

[2] B. Chang, R. Yang, C. Guo, S. Ge, and L. Li, "A new application of optimized random forest algorithms in intelligent fault location of rudders," *IEEE Access*, vol. 7, pp. 94276–94283, 2019.

[3] Q.-N. Xu, H. Zhou, F. Yu, X.-Q. Wei, and H.-Y. Yang, "Effective model based fault detection scheme for rudder servo system," *J. Central South Univ.*, vol. 21, no. 11, pp. 4172–4183, Nov. 2014.

[4] Q. N. Xu, K. M. Lee, H. Zhou, and H. Y. Yang, "Model-based fault detection and isolation scheme for a rudder servo system," *IEEE Trans. Ind. Electron.*, vol. 62, no. 4, pp. 2384–2396, Apr. 2015.

[5] J. Zhou, P. Li, Y. Zhou, B. Wang, J. Zang, and L. Meng, "Toward new-generation intelligent manufacturing," *Engineering*, vol. 4, no. 1, pp. 11–20, 2018.

[6] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," *IEEE Trans. Med. Imag.*, vol. 35, no. 5, pp. 1285–1298, May 2016.

[7] J. Liu, R. Hao, T. Zhang, and X. Wang, "Vibration fault diagnosis based on stochastic configuration neural networks," *Neurocomputing*, vol. 434, pp. 98–125, Apr. 2021.

[8] L. Li, R. Yang, C. Guo, S. Ge, and B. Chang, "The data learning and anomaly detection based on the rudder system testing facility," *Measurement*, vol. 152, Feb. 2020, Art. no. 107324.

[9] L. Li, R. Yang, C. Guo, S. Ge, and B. Chang, "A novel application of intelligent algorithms in fault detection of rudder system," *IEEE Access*, vol. 7, pp. 170658–170667, 2019.

[10] B. L. Chang, R. Yang, C. X. Guo, S. C. Ge, and L. M. Li, "Performance evaluation and prediction of rudders based on machine learning technology," *Proc. Inst. Mech. Eng. G-J. Aerosp. Eng.*, vol. 233, no. 15, pp. 5746–5757, Dec. 2019.

[11] Z. Liu, H. Cao, X. Chen, Z. He, and Z. Shen, "Multi-fault classification based on wavelet SVM with PSO algorithm to analyze vibration signals from rolling element bearings," *Neurocomputing*, vol. 99, pp. 399–410, Jan. 2013.

[12] T. Chen, Y. Cui, and C. Wu, "Application of lifting wavelet and random forest in compound fault diagnosis of gearbox," *Proc. SPIE*, vol. 5, Mar. 2018, Art. no. 107102A.

[13] A. Widodo and B.-S. Yang, "Support vector machine in machine condition monitoring and fault diagnosis," *Mech. Syst. Signal Process.*, vol. 21, no. 6, pp. 2560–2574, 2007.

[14] L. Gou, H. Li, H. Zheng, H. Li, and X. Pei, "Aeroengine control system sensor fault diagnosis based on CWT and CNN," *Math. Problems Eng.*, vol. 2020, pp. 1–12, Jan. 2020.

[15] F. Jia, Y. G. Lei, J. Lin, X. Zhou, and N. Lu, "Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data," *Mech. Syst. Signal Process.*, vols. 72–73, pp. 303–315, May 2016.

[16] W. Zhang, C. Li, G. Peng, Y. Chen, and Z. Zhang, "A deep convolutional neural network with new training methods for bearing fault diagnosis under noisy environment and different working load," *Mech. Syst. Signal Process.*, vol. 100, pp. 439–453, Feb. 2018.

[17] A. Widodo, B.-S. Yang, and T. Han, "Combination of independent component analysis and support vector machines for intelligent faults diagnosis of induction motors," *Expert Syst. Appl.*, vol. 32, no. 2, pp. 299–312, 2007.

[18] J. Tao, Y. Liu, and D. Yang, "Bearing fault diagnosis based on deep belief network and multisensor information fusion," *Shock Vibrat.*, vol. 2016, pp. 1–9, Aug. 2016.

[19] C. Li, D. Zhao, S. Mu, W. Zhang, N. Shi, and L. Li, "Fault diagnosis for distillation process based on CNN-DAE," *Chin. J. Chem. Eng.*, vol. 27, no. 3, pp. 598–604, 2019.

[20] H. Liu, J. Zhou, Y. Zheng, W. Jiang, and Y. Zhang, "Fault diagnosis of rolling bearings with recurrent neural network-based autoencoders," *ISA Trans.*, vol. 77, pp. 167–178, Jun. 2018.

[21] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115–118, 2017.

[22] R. Li and Z. Liu, "Stress detection using deep neural networks," *BMC Med. Informat. Decis. Making*, vol. 20, no. S11, p. 285, Dec. 2020.

[23] H. Lee and H. Kwon, "Going deeper with contextual CNN for hyperspectral image classification," *IEEE Trans. Image Process.*, vol. 26, no. 10, pp. 4843–4855, Oct. 2017.

[24] P. R. Lorenzo, J. Nalepa, M. Kawulok, L. S. Ramos, and J. R. Pastor, "Particle swarm optimization for hyper-parameter selection in deep neural networks," in *Proc. Genetic Evol. Comput. Conf.*, New York, NY, USA, 2017, pp. 481–488.

[25] B. Navaneeth and M. Suchetha, "PSO optimized 1-D CNN-SVM architecture for real-time detection and classification applications," *Comput. Biol. Med.*, vol. 108, pp. 85–92, May 2019.

[26] G. L. F. da Silva, T. L. A. Valente, A. C. Silva, A. C. de Paiva, and M. Gattass, "Convolutional neural network-based PSO for lung nodule false positive reduction on CT images," *Comput. Methods Programs Biomed.*, vol. 162, pp. 109–118, Aug. 2018.

[27] A. Tijskens, H. Janssen, and S. Roels, "Optimising convolutional neural networks to predict the hygrothermal performance of building components," *Energies*, vol. 12, no. 20, p. 3966, Oct. 2019.

[28] S. Garg, K. Kaur, N. Kumar, G. Kaddoum, A. Y. Zomaya, and R. Ranjan, "A hybrid deep learning-based model for anomaly detection in cloud datacenter networks," *IEEE Trans. Netw. Service Manage.*, vol. 16, no. 3, pp. 924–935, Sep. 2019.

[29] H. Xie, L. Zhang, and C. P. Lim, "Evolving CNN-LSTM models for time series prediction using enhanced grey wolf optimizer," *IEEE Access*, vol. 8, pp. 161519–161541, 2020.

[30] A. M. Abdelshafy, H. Hassan, and J. Jurasz, "Optimal design of a grid-connected desalination plant powered by renewable energy resources using a hybrid PSO–GWO approach," *Energy Convers. Manage.*, vol. 173, pp. 331–347, Oct. 2018.

[31] F. Gul, W. Rahiman, S. S. N. Alhady, A. Ali, I. Mir, and A. Jalil, "Meta-heuristic approach for solving multi-objective path planning for autonomous guided robot using PSO–GWO optimization algorithm with evolutionary programming," *J. Ambient Intell. Humanized Comput.*, vol. 12, no. 7, pp. 7873–7890, Jul. 2021.

[32] N. H. Son and N. V. Hop, "A hybrid meta-heuristics approach for supplier selection and order allocation problem for supplying risks of recyclable raw materials," *Int. J. Ind. Eng. Comput.*, vol. 12, no. 2, pp. 177–190, 2021.

[33] V. K. Kamboj, "A novel hybrid PSO-GWO approach for unit commitment problem," *Neural Comput. Appl.*, vol. 27, no. 6, pp. 1643–1655, Aug. 2016.

[34] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.

[35] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. ICNN Int. Conf. Neural Netw.*, Perth, WA, Australia, Nov. 1995, pp. 1942–1948.

[36] A. K. Mishra, S. R. Das, P. K. Ray, R. K. Mallick, A. Mohanty, and D. K. Mishra, "PSO-GWO optimized fractional order PID based hybrid shunt active power filter for power quality improvements," *IEEE Access*, vol. 8, pp. 74497–74512, 2020.

[37] N. Singh and S. B. Singh, "Hybrid algorithm of particle swarm optimization and grey wolf optimizer for improving convergence performance," *J. Appl. Math.*, vol. 2017, pp. 1–15, Nov. 2017.

[38] J. Zhu, A. S. Sharma, J. Xu, Y. Xu, T. Jiao, Q. Ouyang, H. Li, and Q. Chen, "Rapid on-site identification of pesticide residues in tea by one-dimensional convolutional neural network coupled with surface-enhanced Raman scattering," *Spectrochim. Acta A, Mol. Biomolecular Spectrosc.*, vol. 246, Feb. 2021, Art. no. 118994.

[39] M. S. Rana, A. Nibali, Z. He, and S. Morgan, "Automated repair of fragmented tracks with 1D CNNs," *Image Vis. Comput.*, vol. 101, Sep. 2020, Art. no. 103982.

[40] H. Faris, I. Aljarah, M. A. Al-Betar, and S. Mirjalili, "Grey wolf optimizer: A review of recent variants and applications," *Neural Comput. Appl.*, vol. 30, no. 2, pp. 413–435, 2018.

[41] G. Negi, A. Kumar, S. Pant, and M. Ram, "GWO: A review and applications," *Int. J. Syst. Assurance Eng. Manage.*, vol. 12, no. 1, pp. 1–8, Feb. 2021.

[42] N. Mittal, U. Singh, and B. S. Sohi, "Modified grey wolf optimizer for global engineering optimization," *Appl. Comput. Intell. Soft Comput.*, vol. 2016, pp. 1–16, Mar. 2016, Art. no. 7950348.

[43] S. Gupta and K. Deep, "A novel random walk grey wolf optimizer," *Swarm Evol. Comput.*, vol. 44, pp. 101–112, Feb. 2019.

[44] M. R. S. Malik, E. R. Mohideen, and L. Ali, "Weighted distance Grey wolf optimizer for global optimization problems," in *Proc. IEEE Int. Conf. Comput. Intell. Comput. Res. (ICCIC)*, Dec. 2015, pp. 1–6, doi: 10.1109/ICCIC.2015.7435714.

[45] N. Ayub, M. Irfan, M. Awais, U. Ali, T. Ali, M. Hamdi, A. Alghamdi, and F. Muhammad, "Big data analytics for short and medium-term electricity load forecasting using an AI techniques ensembler," *Energies*, vol. 13, no. 19, p. 5193, Oct. 2020.

WEILI WANG received the B.S. degree in measurement and control technology and instrumentation from the College of Instrumentation and Electronics, North University of China, Taiyuan, China, in 2019, where she is currently pursuing the master's degree in instrumentation science and technology. Her research interests include rudder testing, deep learning, and intelligent algorithms and their applications.

RUIFENG YANG received the B.S. degree in testing technology and the M.S. and Ph.D. degrees in measurement technology and instruments from the North University of China, Taiyuan, China, in 1992, 1999, and 2005, respectively. From 2010 to 2012, he studied at the Postdoctoral Center for Control Engineering, Beihang University. He is currently a Professor with the School of Instrument and Electronics, North University of China, and the Director of the Automatic Test Equipment and System Engineering Research Center of Shanxi Province. He was awarded a second prize for scientific and technological progress in Shanxi province and also presides over more than ten scientific research projects (include government sponsored researches and projects of school-enterprise cooperation), such as the National Fund and the Doctoral Foundation of the Ministry of Education and has authored or coauthored more than 20 academic papers. His research interests include automated testing and control technology, equipment test detection and system integration, and intelligent instrument.

CHENXIA GUO received the B.S., M.S., and Ph.D. degrees from the North University of China, Taiyuan, China. She is currently an Associate Professor with the School of Instrument and Electronics, North University of China. Her research interests include automated testing and control technology, design and integration of complex electromechanical systems, and vision measurement.

HAO QIN received the B.S. degree in measurement and control technology and instrumentation from the College of Instrumentation and Electronics, North University of China, Taiyuan, China, in 2019, where he is currently pursuing the master's degree in instrumentation science and technology. His research interests include rudder systems, deep learning, and health management.