Hindawi
Mathematical Problems in Engineering
Volume 2020, Article ID 5357146, 12 pages
https://doi.org/10.1155/2020/5357146

Research Article

Aeroengine Control System Sensor Fault Diagnosis Based on CWT and CNN

Linfeng Gou,[1] Huihui Li,[1] Hua Zheng,[1] Huacong Li,[1] and Xiaoning Pei[2]

[1]*School of Power and Energy, Northwestern Polytechnical University, Xi'an 710072, Shaanxi, China*
[2]*School of Automation, Northwestern Polytechnical University, Xi'an 710072, Shaanxi, China*

Correspondence should be addressed to Huihui Li; lihuihui@mail.nwpu.edu.cn

Received 27 September 2019; Revised 6 December 2019; Accepted 26 December 2019; Published 22 January 2020

Academic Editor: Xiaoliang Jin

The aeroengine control system is a piece of complex thermal machinery which works under high-speed, high-load, and high-temperature environmental conditions over lengthy periods of time; it must be designed for the utmost reliability and safety to function effectively. The consequences of sensor faults are often extremely serious. The inherent complexity of the engine structure creates difficulty in establishing accurate mathematical models for the model-based sensor fault diagnosis. This paper proposes an intelligent fault diagnosis method for aeroengine sensors combining a deep learning algorithm with time-frequency analysis wherein the signal recognition problem is transformed into an image recognition problem. The continuous wavelet transform (CWT) is first applied to seven common health condition signals in an engine control system sensor in order to generate scalograms that capture the characteristics of the signal. A convolutional neural network (CNN) model trained with preprocessed and labeled datasets is then used to extract the features of a time-frequency graph based on which faults can be identified and isolated. This method does not require modeling and design thresholds, so it has strong robustness and accuracy rate of over 97%. The trained model effectively reveals faults in sensor signals and allows for accurate identification of fault types.

1. Introduction

The aeroengine control system is a piece of complex thermal machinery which works within high-speed, high-load, and high-temperature conditions over long periods of time. It must have the utmost reliability and safety. It works on the basis of obtaining accurate parameter signals from sensors [1]; however, the inherently large quantity of sensor measurement data, wide distribution, and unique installation location make it prone to faults particularly within the high-temperature, high-pressure, and strong vibration conditions. Generally speaking, the proportion of sensor faults in total system faults exceeds 75% [2]. Timely and effective sensor fault diagnosis is of crucial significance in terms of the safety and reliability of the aeroengine.

A common approach to sensor fault diagnosis is the redundancy analysis technique, which may be model-based, data-driven, or a hybrid of the two [3]. The earliest method is based on an analytical model [4] wherein fault is diagnosed without prior knowledge or experience via the observer, particle filter, Kalman filter, or other tools. It is necessary to build a highly accurate model of the diagnosed object to effectively operate this method. The reliability of models of complex nonlinear systems such as aeroengines decreases as modeling uncertainty and nonlinear complexity increase; thus, the reliability of the diagnostic system decreases.

Data-driven diagnostics that avoid model-building problems have been developed as a response to the above problem. Offline or online learning of test data is followed by feature extraction, classification or regression, and other technologies for fault detection without full detail regarding the working principle of physical objects or the analytic model. Among these techniques, the neural network [5] is most widely used for fault diagnosis due to its self-learning capability and ability to fit arbitrary continuous nonlinear functions. For example, Botros [6] used the Laplace transform-wavelet transform to extract vibration fault signal features from an aeroengine followed by an artificial neural network to classify the faults. The shallow network is prone to overfitting problems [2]. Moreover, the input characteristics (e.g., mean value, variance and energy, and root mean square) of the shallow neural network are selected manually, so it is prone to other issues such as data underutilization, sensitivity to interference, and information loss. The quality of features is the key to the performance of the diagnostic system [7, 8].

Intelligent diagnosis methods based on deep learning [9], which can be used to automatically extract the intrinsic characteristics of data, have attracted a great deal of research attention. Deep neural networks are not prone to overfitting or gradient attenuation problems. The convolutional neural network [10] (CNN) is one of the commonly used deep networks and is now the dominant method applied in intelligent diagnosis systems [11, 12]. Verstraete [13] extracted vibration signal representations in the time and frequency domains as an input to a CNN and then classified various fault characteristics of a rotating machine accordingly at up to 96% accuracy. Alaskar [14] studied a damaged signal with noise using CNN techniques and then generated images from MFPT data and Case Western dataset using short-time Fourier transform. This trained CNN model detected patterns in the signal with 98% and 99% accuracy, respectively. Chen [15] used a deep convolutional neural network (DCNN) method for the life cycle prediction of engine units with special focus on the late stages of the fault. The original features, data preprocessing operation, and time window sample-making process can be selected to achieve predictions very close to the actual RUL values. Liu et al. [16] proposed an effective and reliable method based on the convolution neural network (CNN) and discrete wavelet transformation (DWT) to identify the fault conditions of planetary gearboxes. The experimental results demonstrated the effectiveness and feasibility of the proposed method.

In addition, for the aeroengine control system, highly accurate sensor data are crucial for effective control of the system. The real-time monitoring of sensor health status is particularly important. Online fault diagnosis may serve to monitor sensor data in real time so as to detect sensor faults and inform timely maintenance decisions. Online fault diagnosis methods have strong adaptive ability and indeed can enhance engine safety. The aeroengine is the most widely varied thermal machine. Its working conditions are highly complex and changeable, so online fault diagnosis for its sensors is relatively difficult. Neural networks require an abundance of data—and calculation processes—for training, and the online fault diagnosis method requires the system to be in the correct working state for a period of time to function properly. Although the real-time performance of offline fault diagnosis is poor, it is easy to realize and provides relatively comprehensive analyses based on a large quantity of historical data. Most data-driven fault diagnosis methods adopt offline training and online operation [17].

In the recent years, development of modern aeronautical technology leads to a complex engine control system, where high reliability, quality, and safety are required in very harsh environment. The study of efficient, reliable, and fast sensor fault diagnosis method is of great significance for the efficient and reliable operation of aeroengine and the reduction of maintenance cost. The above studies evaluated the effectiveness of fault diagnosis based on CNN. Li and Qu [2] applied CNN to fault diagnosis of aeroengine sensors. However, they only focused on the time-domain data in the feature learning of CNN. In the paper, we proposed a novel fault diagnosis method combining continuous wavelet transform (CWT) and CNN, since CWT has the characteristics of fast response speed and containing a large amount of information. The experimental dataset consists of a large amount of simulation fault data and the actual aeroengine fault history data. The CNN uses a series of time-frequency diagrams of the experimental dataset as the input to explore the inner characteristics for malfunctional sensors. The efficacy of the proposed method was verified for aeroengine sensor fault diagnosis in the study.

The paper is organized as follows: Section 2 elucidates the proposed method CNN + CWT for fault diagnosis. Section 3 introduces the dataset acquisition experiment to collect the fault signal. In Section 4, comparative experiments are conducted to verify the effectiveness of the developed method. Finally, the paper's conclusions are drawn in Section 5.

2. The Fault Diagnosis Method Based on CNN and CWT

This section introduces the process of the proposed method. A diagram of the proposed fault diagnosis scheme, the CNN + CWT, is provided in Figure 1. In the process, the sensor signal $x(k)$ is first normalized to obtain the signal $y(k)$. A continuous wavelet transform is then applied to the dataset with labels to obtain time-frequency characteristics. The time-frequency graphs of certain types of faults are very similar, so the simple wavelet transform cannot accurately identify them. It is necessary to find a more effective classification method. Deep learning technology has proven effective for data analysis, and the CNN deep learning model is an intelligent method for deep data mining. The CNN can not only distinguish the general outline of information but also distinguish subtle differences invisible to the human eye. The CNN has strong multivariable processing ability, so a time-frequency graph can be used as the input to train it. Actual fault data from an operable engine were input into the trained CNN model in this study as a verification set. Six types of faults were classified, respectively.

2.1. Sensor Signal Preprocessing. In practice, the units of measurement for different variables are often different. To eliminate the dimensional effect of variables and ensure each variable has the same expressive power, data are often standardized prior to analysis. As shown in equation (1), the normalization of measurement signals can make the mean value of data approach 0 and the variance approach 1, which reduces errors in the signal acquisition process while bringing down the computational complexity and computation time.

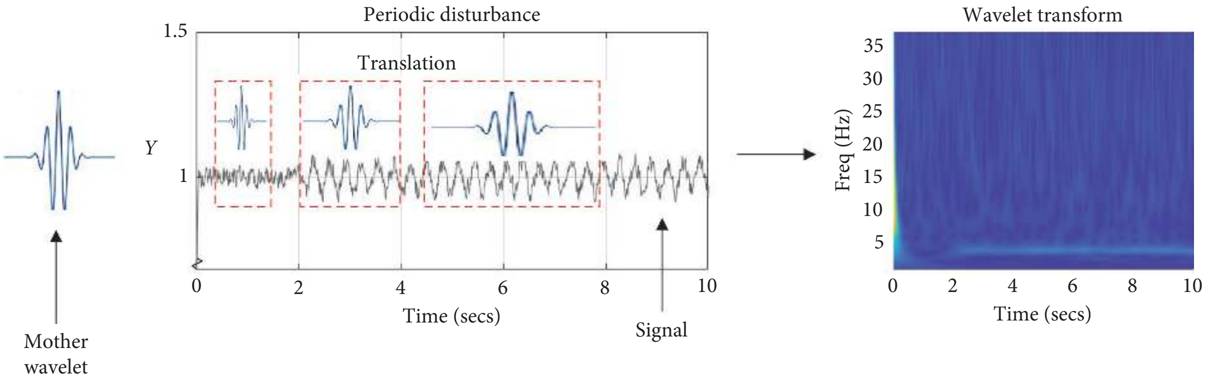


Figure 1: Flowchart of the proposed method.

$$y(k) = \frac{x(k) - \overline{x}(k)}{\sigma(x(k))}, \tag{1}$$

where

$$\overline{x}(k) = \frac{1}{n}\sum_{i=1}^{n} xi(k),$$

$$\sigma(x(k)) = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(xi(k) - \overline{x}(k))^2}. \tag{2}$$

where x(k) is the sensor signal to be processed and y(k) is the processed sensor signal.

2.2. Continuous Wavelet Transform (CWT). In sensor fault diagnosis, the sensor signal containing time information is recorded with data points indexed by time series [18]. The sensor signal is time-varying and nonstationary, so its global characteristics in the time or frequency domains are insufficient for subsequent analysis; it is often simply impossible to capture the internal law of the fault state. It is necessary to determine the changes in the signal spectrum with time through joint time-frequency analysis (JTFA). The one-dimensional (1D) sensor fault signal can be converted by JTFA into a multidimensional matrix suitable for CNN processing [19]. The time-frequency graph calculated by JTFA reflects very comprehensive information. The CWT is a widely applied method under JTFA theory which we apply here to process the input data of our neural network.

The CWT functions use a wavelet function system to represent or approximate a signal, which projects the signal onto a 2D time-scale phase plane. Unlike the short-time Fourier transform (STFT), the CWT has an adjustable time-frequency window and can thus resolve the conflict between time and frequency resolutions. A function called the "mother wavelet" is shifted, and then an inner product was created between the function and the signal which will be processed at different scales. This is expressed as follows:

$$W_f(p,q) = \frac{1}{\sqrt{|p|}}\int_{-\infty}^{+\infty} f(x)\overline{\psi\left(\frac{x-q}{p}\right)}\mathrm{d}x, \tag{3}$$

where p is the scaling factor corresponding to frequency information (i.e., the scale parameter). The function of the scale parameter p is to stretch and compress the basic wavelet. q is the translation factor corresponding to space-time information (i.e., the transformation parameter), which reflects the displacement of the mother wavelet. $\psi(x)$ is the mother wavelet. $\overline{\psi(x)}$ is the complex conjugate of $\psi(x)$. $W_f(p,q)$ is the signal after wavelet transformation, and $f(x)$ is the mathematical expression of the signal to be processed.

The CWT is a linear transformation, so it has superposition, time-shift invariance, telescopic covariance, and other properties which allow us to use them to scale the basic wavelet. A large scale corresponds to a large time window, and a small scale corresponds to a small window. Thus, the signal can be decomposed at different resolutions on different time and frequency scales. The basic wavelet scaling and translation constitute the continuous wavelet sequence, as shown in equation (4). The wavelet cluster is the basis of the function space and acts as the observation window of the processed signal in the wavelet transform. Signal amplitudes vary at different times, at different frequencies, and under different health conditions. To this effect, prominent and comprehensive feature information can be obtained via CWT.

$$\psi_{p,q}(x) = \frac{1}{\sqrt{|p|}}\psi\left(\frac{x-q}{p}\right). \tag{4}$$

Under different scale factors, continuous wavelet transform produces coefficients of different parts of the output signal of equation (4). These coefficients contain the complete time-frequency information of the sensor signal and, combined with time series and scale series, can be used to obtain scalograms. Next, the 2D time-frequency images are resized so that the CNN can process them by edge clipping and down-sampling. Figure 2 shows the conversion process with the CWT time-frequency representation of an aeroengine sensor as an example.

2.3. Convolutional Neural Network (CNN). Deep learning is a complex machine learning algorithm that can reveal the internal rules and presentation levels of sample data. It is a very flexible and powerful tool. The deep learning model used in this paper, the CNN [20], is a supervised learning

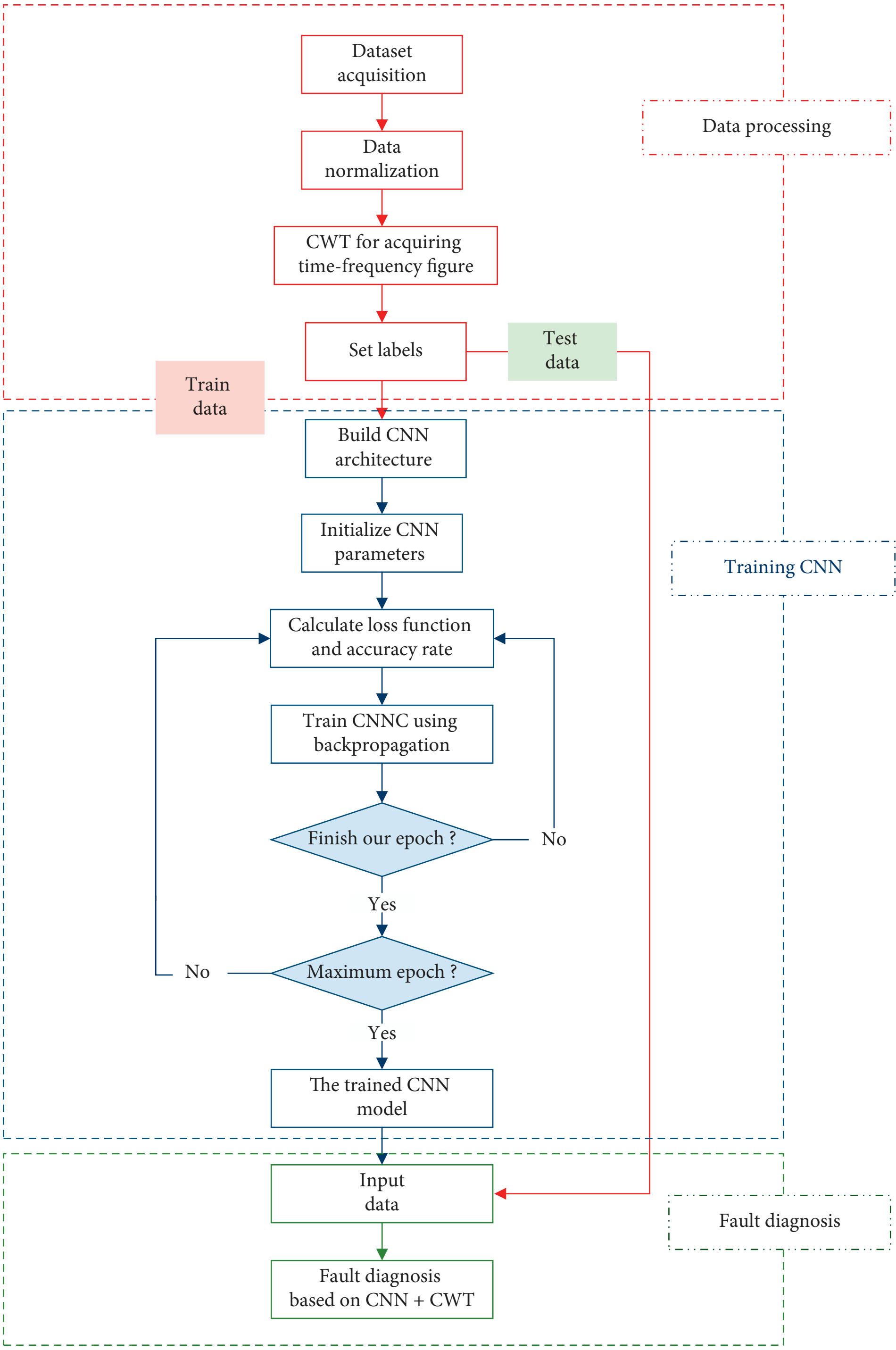


Figure 2: Schematic diagram of CWT.

method [10] under which a convolution layer is utilized to extract features. The CNN has advantages over other networks due to its adoption of weight replication, receptive fields, and subsampling [8]; it has low complexity, strong antinoise ability, and other advantageous properties. The typical structural network of a convolutional nerve consists of an input layer, convolution layer, pooling layer, fully connected layer, and output layer [21, 22].

Convolution is a special filtering method for significant characteristic extraction. Under normal conditions, each convolutional layer contains multiple convolutional kernels. In the convolution layer, the feature maps of the previous layer are convolved with learnable kernels and pass through activation functions for nonlinear transformation. Finally, an output feature map is produced that conforms to convolution theory. Each output map integrates convolution with multiple input maps. The convolution process is expressed as follows:

$$x_k^n = \phi\left(\sum_{i\in Mk} x_i^{n-1} * \beta_{ik}^n + b_k^n\right), \tag{5}$$

where x_k^n is the j^{th} feature graph of the i^{th} convolution layer, $\phi(\,)$ denotes the activation function, M_k is the input graph set, β is the convolution kernel, and b_k^n represents the corresponding bias.

The pooling layer compresses the input feature graph to reduce its size and simplifies the network computation process. Feature compression serves to extract the main features while decreasing the dimensional complexity of the output and its sensitivity to environmental changes while retaining as much valid information as possible [23]. This can be expressed as

$$x_k^n = \phi\left(w_k^n \mathrm{down}\left(x_k^{n-1}\right) + b_k^n\right), \tag{6}$$

where down( )is the subsampling function, w_k^n denotes the weight matrix, and the meaning of other parameters is the same as in the above equation.

The fully connected layer is similar to a traditional neural network in that it is connected to all the neurons in the previous layer. Many classification models, such as the Softmax regression model as-applied in this study, can be used as a fully connected layer. Assume there is a dataset for training the model and the samples of the set have corresponding tags in the tag sets $\{y^i\}_{i=1}^N$. $P(y^i = k \mid x^i)$ is the probability of the sample matching the correct label, which can be calculated by the Softmax regression model. The output result is as follows:

$$R = \begin{bmatrix} P\left(y^i = 1 \mid x^i; \theta_1^T x^i\right) \\ P\left(y^i = 2 \mid x^i; \theta_2^T x^i\right) \\ \vdots \\ P\left(y^i = K \mid x^i; \theta_k^T x^i\right) \end{bmatrix} = \frac{1}{\sum_{j=1}^k e^{\theta_j^T x^i}} \begin{bmatrix} e^{\theta_1^T x^i} \\ e^{\theta_2^T x^i} \\ \vdots \\ e^{\theta_k^T x^i} \end{bmatrix}, \tag{7}$$

where x is the input eigenvector of the i^{th} sample and k represents the amount of labels. $\theta = [\theta_1, \theta_2, \ldots, \theta_k]^T$, and R represent the parameters and output of the classification model, respectively.

The deep network adopted in this study is based on the framework of LeNet-5 with some modifications, namely, a dropout layer is added behind the fully connected layer to prevent overfitting [23]. The key idea is to randomly drop units (along with their connections) from the neural network during training. Dropout forces a neural unit to work with other randomly selected neural units. This prevents units from coadapting too much and enhances generalization. During training, samples from an exponential number of different "thinned" networks were dropped out. At test time, it is easy to approximate the effect of averaging the predictions of all these thinned networks by simply using a single unthinned network that has smaller weights [24]. This significantly reduces overfitting and simulates a neural network with a large number of different network structures and in turn make the nodes in the network more robust.

The trainable parameters of the CNN model were first initialized, and then optimized by the back-propagation (BP) algorithm and the adaptive moment estimation (Adam) algorithm [25] in the PyTorch deep learning framework. The optimization process serves to calculate the error between the real value and the predicted value. The trainable parameters (weights and bias) can then be updated quickly and fine-tuned until the training error is minimal.

We measure the training error by a function called cross-entropy loss function. Generally speaking, when using neural networks to solve classification problems, the cross-entropy cost is considered slightly better than the mean square error loss (MSE). The MSE causes the loss of useful information and may result in gradient dispersion. The cross-entropy loss function $F(\theta)$ is expressed as follows:

$$F(y_c, y_l) = -\frac{1}{n}\sum_{\text{class}} \left[y_c \ln y_l + (1 - y_c)\ln(1 - y_l)\right], \tag{8}$$

where n indicates the dimension of training data. Class is the tag category into which signals need to be classified. yc represents the output of the sensor fault signal of the neural network, and yl is the mark result of the sensor fault signal.

To summarize, we first standardized and time-frequency analyzed the sensor signals of the aeroengine control system for seven health conditions in order to generate scalograms. The input for CNN is the tagged scalograms, and the output is the health conditions of the sensor. The output results can be used to identify and isolate sensor faults.

2.4. Performance Evaluation. After training the model, we used accuracy rate to evaluate its quality and performance. We calculated the accuracy of training dataset after each iteration while training the deep network and then conducted a reliability test and calculated the accuracy of the test dataset. When the predicted value and the real value are the same, the accuracy rate calculation result is 1; otherwise, it is 0. The average value of numerous calculation results is the final accuracy rate value.

$$\mathrm{AC}(y_c, y_l) = \frac{\mathrm{sum}\,(y_c == y_l)}{N}, \tag{9}$$

where $\mathrm{AC}(y_c, y_l)$ represents accuracy rate, the summing function is the fault classification result of the neural network output, and calculates the same number between yc and yl; N is the dimension of yc.

3. Dataset Acquisition Experiment

3.1. Fault Signal Simulation. CNN model training requires a large amount of data. Too small a dataset produces a trained model that is imprecise and lacks generalization ability. The real fault data of a certain engine are insufficient to effectively train the model, so we analyzed the fault modes of sensors used in aeroengine control systems, simulated various sensor faults, and obtained the output data of sensors through a series of simulation experiments.

The sensor fault changes the output of the sensor and renders the control system which is unable to accurately provide measured information, resulting in a large error in the feedback value. This further degrades the performance of the control system even to the point of failure. According to their cause, sensor faults can be classified into different categories. Certain sensor faults occur suddenly, where the sensor output value changes suddenly due to structural damage or other reasons such as open fault or short-circuit fault. Some sensor faults occur slowly during engine operation and have smaller amplitude, e.g., bias fault, spike fault, drift fault, or periodic disturbance fault. Table 1 summarizes the causes for several faults and provides a simulation scheme thereof [26].

After determining the fault simulation method, we determined the sensor simulation model. In most cases, first-order or second-order inertial elements [27] are used to represent sensors in the simulation model. The latter was selected in this study. Its transfer function is as follows:

$$G(s) = \frac{w_{\mathrm{n}}^2}{s^2 + 2\xi w_n s + w_n^2} \cdot e^{-\tau s}, \tag{10}$$

where $\xi = 1.25, w_n = 9, \tau = 0.12$.

Considering the sensors of the aeroengine control system that tend to work alongside disturbances such as ambient noise, we generated a Gaussian noise $G(0, \sigma)$ with a mean of zero and variance of σ for subsequent analysis. The signal was then artificially mixed with the output of the sensor simulation model to make the output result closer to real-world signal values. We simulated the output signals of the sensor under seven different health conditions in MATLAB as shown in Figure 3.

The parameter records of the engine used in this experiment were obtained under different working conditions (e.g., cruise state and maximum state) and flight conditions (e.g., different heights, Ma). Hence, the component-level model of an engine was used as the simulation verification model. The engine mathematical model was run at the design point working condition on the ground and the off-design point working condition on the upper air, respectively. The input of the aeroengine model is step signals. The data obtained from the output signal of the model through the sensor model were then used as experimental data. The diagnosis performance can be evaluated effectively by random fault simulation. The fault types, fault time, and fault amplitudes of each simulation in our case are random. We randomly changed the occurrence time and size of the fault, as well as the slope, periodic signal frequency, and other parameters to secure a sufficiently large amount of data. When the simulation was complete, we sampled the sensor signal every 0.01 seconds to build the dataset. Finally, 2,500 sets of sensor fault data were input to the MATLAB simulation platform.

3.2. Dataset Acquisition. A large amount of data is required for aeroengine control system fault diagnosis, as discussed above. Here, a portion of the sensor signal set was collected from the records of a certain type of aircraft engine running to fault under various working surroundings and fault types. The other portion of data was derived from the MATLAB simulation. In this case, nine key characteristic parameters of aeroengines that could be measured by sensors are considered: power-lever angle sensor PLA, inlet air temperature T_1, gas generator rotational speed N_{L}, power turbine rotational speed N_{H}, compressor inlet temperature T_{25}, compressor outlet pressure P_3, high-pressure power turbine outlet temperature$T_{4.5}$, low-pressure turbine outlet temperature T_5, and low-pressure turbine outlet pressure P_5. Seven possible sensor health conditions are considered for each sensor. We obtained 500 data samples for each health condition, resulting in a total of 3,500 datasets.

The CNN is a supervised learning method that requires marked input and output data. In other words, in a supervised learning process, the categories of each training sample and test dataset should be known. We labeled each type of (preprocessed) data for each sensor health condition as shown in Table 2.

Next, we performed time-frequency analysis of the preprocessed and labeled data. We then obtained a dataset composed of time-frequency graphs (Figure 4). Finally, the dataset was arbitrarily divided into two parts: a training set and test set (about 70% and 30% of the total data, respectively). We processed the time-frequency graph to treat the images, so they suited the CNN (224×224). The processed dataset was then saved as the CNN input dataset.

4. Experiments and Discussion

4.1. Experimental Setup. In this section, an experiment was conducted to show the effectiveness of our proposed method based on the dataset above. The experiment settings are shown in Figure 5.

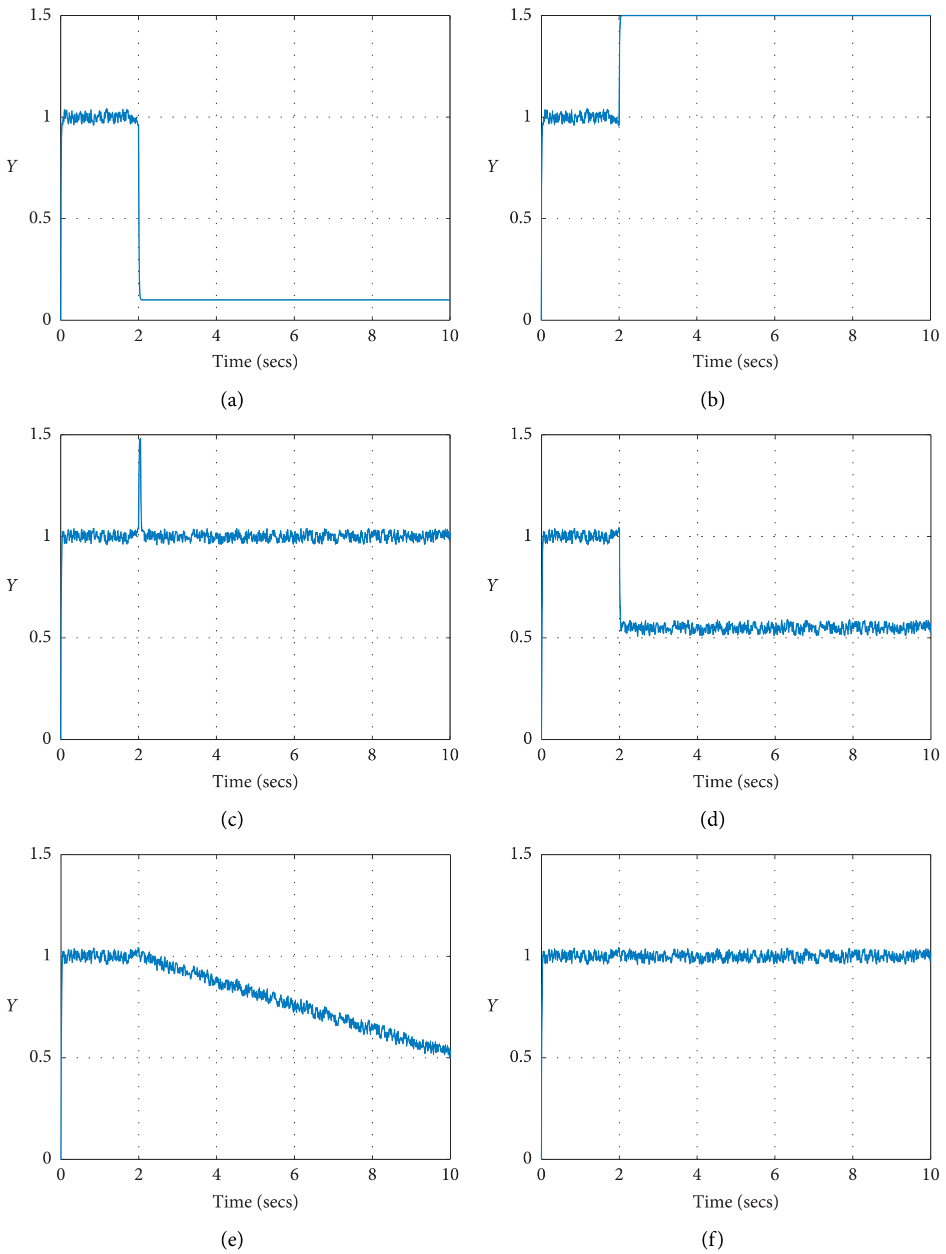
4.2. Experimental Parameter Selection. In the wavelet transform of this experiment, we used the Morlet wavelet. It is a basic wavelet which is often used for continuous wavelet transform. After several experiments, we found that a scale factor setting of 1–100 worked best for our experimental conditions. For each given scale p, the wavelet coefficients were computed for $q = 1$ to $q = \text{length}$ (i.e., the length of the signal).

Due to the complex structure of the deep learning model, it is difficult to select appropriate parameters to establish an effective model. In order for the scheme we designed to have good performance, we determined the appropriate key parameters based on the experience of our predecessors [2, 28] and the classical convolutional neural network model LeNet-5. These parameters include the size and number of the filter, the ratio of the dropout layer, the learning rate, and the size of input data (Table 3).

The architecture of the CNN is shown in Figure 6. The CNN structure designed in this study includes one input layer, two convolution layers, two pooling layers, and three fully connected layers. A dropout layer was added behind the

Table 1: Common sensor types, fault causes, and simulation methods.

Fault	Reason	Method of simulation
Short circuit	Pollution caused by the bridge road corrosion line short connection	Signal is close to 0.1
Open circuit	Signal line is broken, chip pin is not connected	Signal approaches maximum
Spike	Random disturbance in power supply and ground wire, surge	Add a pulse signal to the original signal
Bias	Bias current or bias voltage	Add a small constant or random signal to the original signal
Drift	Temperature drift	Signal is offset by a certain rate
Periodic disturbance	50 Hz interference from the power supply	Signal of a certain frequency superimposed on the original signal



(a)

(b)

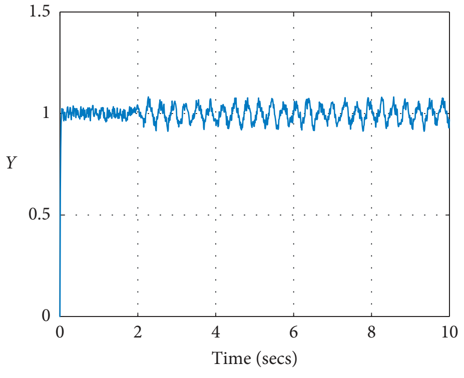
(c)

(d)

(e)

(f)

Figure 3: Continued.

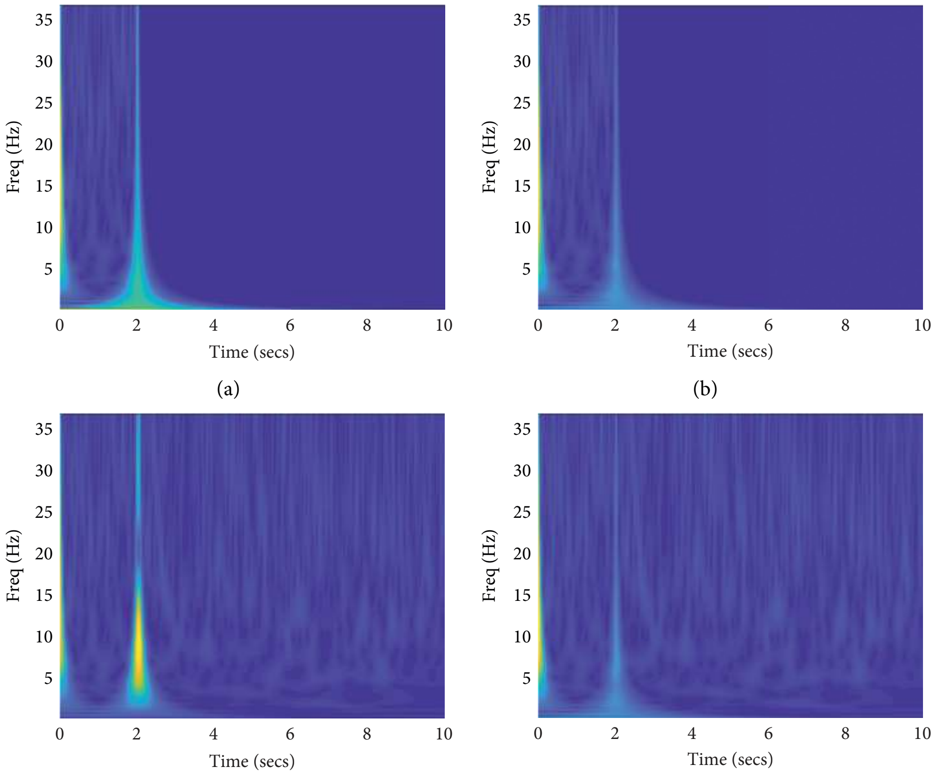


(g)

Figure 3: Sensor signals of 7 health conditions. (a) Short circuit. (b) Open circuit. (c) Spike. (d) Bias. (e) Drift. (f) Normal. (g) Periodic disturbance.

Table 2: Fault labels.

Fault	Short circuit	Open circuit	Spike	Bias	Drift	Normal	Periodic disturbance
Label	0	1	2	3	4	5	6



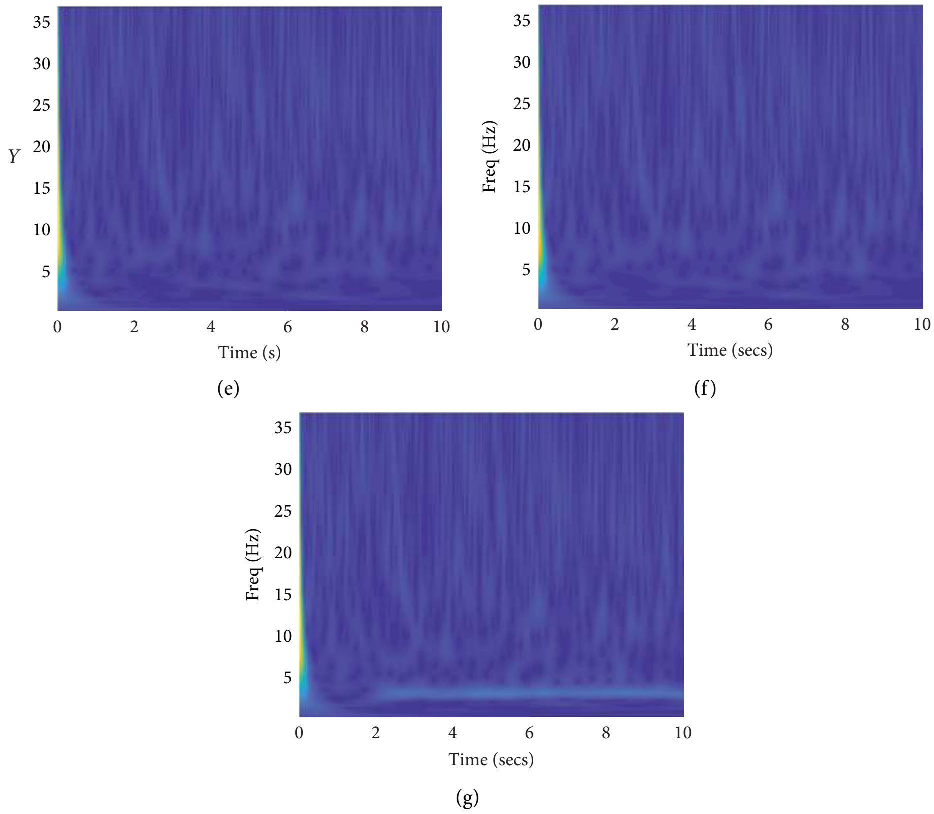
(a)

(b)

(c)

(d)

Figure 4: Continued.



(e)

(f)

(g)

Figure 4: Scalograms of 7 health conditions. (a) Short circuit. (b) Open circuit. (c) Spike. (d) Bias. (e) Drift. (f) Normal. (g) Periodic disturbance.

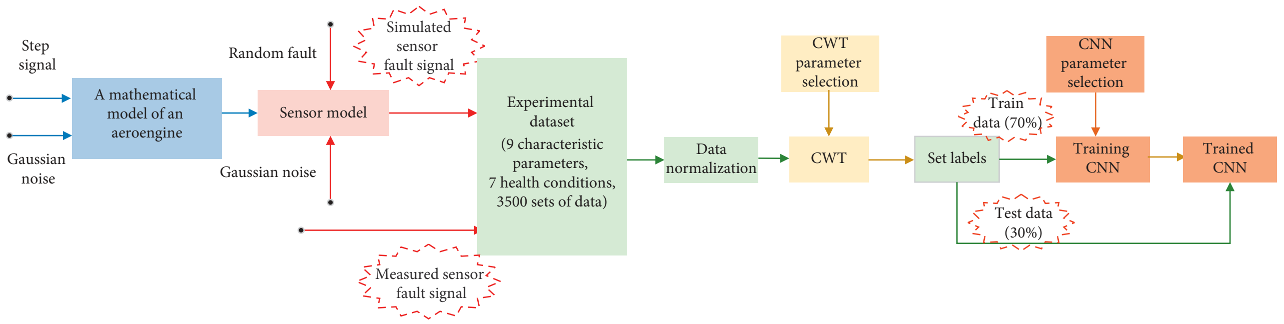


Figure 5: Flowchart of the experiment.

Table 3: Network parameters.

Name	Parameter	The size of output feature
Input layer	Input data	$3 \times 224 \times 224$
Convolutional layer ($C1$)	6 filters, size $5 * 5$, stride 1	$6 * 220 * 220$
Pooling layer ($P1$)	Filter size $2 * 2$, stride 1	$6 * 110 * 110$
Convolutional layer ($C2$)	16 filters, size $5 * 5$, stride 1	$16 * 106 * 106$
Pooling layer ($P2$)	Filter size $2 * 2$, stride 1	$16 * 53 * 53$
Fully connected layer ($F1$)	120 nodes, dropout = 0.5	$44944 * 120$
Fully connected layer ($F2$)	84 nodes, dropout = 0.5	$120 * 84$
Fully connected layer ($F3$)	7 nodes	$84 * 7$
Output layer	Output data	$7 * 1$
Learning rate	0.00004	None
Training dataset	2450	None
Test dataset	1050	None

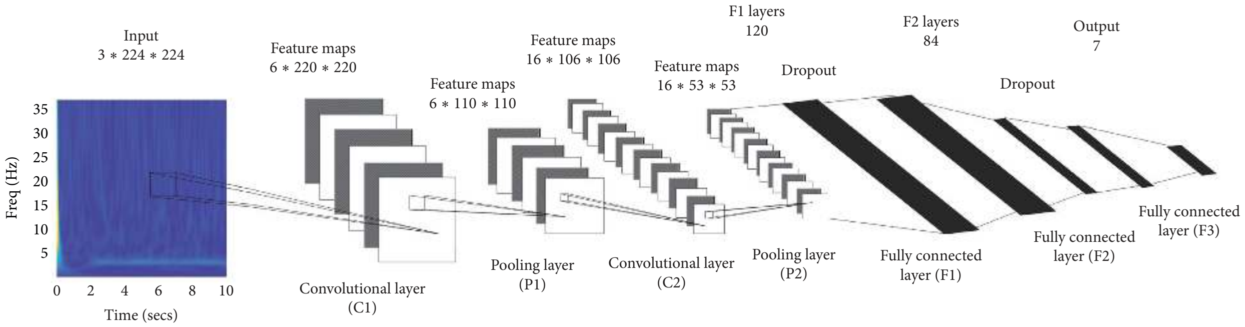


Figure 6: CNN architecture.

fully connected layer to prevent overfitting. The input data of the network are time-frequency images $3 \times 224 \times 224$ in size and the output is 7×1. There are 6 filters in the first convolutional layer ($C1$). The size of the output feature graph is $6 \times 220 \times 220$. An output feature graph $6 \times 110 \times 110$ in size falls after the first pooling layer ($P1$). The second convolution layer ($C2$) has 16 filters. The size of the output feature graph is $16 \times 106 \times 106$. The output feature diagram after the second maximum pooling layer ($P2$) is $16 \times 53 \times 53$ in size. The fault signals of sensors are classified by three fully connected layers with dimensions of 44944×120, 120×84, and 84×7, respectively.

5. Results

To validate the proposed CNN + CWT method, we used a radial basis function network (RBF) for fault detection and isolation of the same dataset. The RBF is a feedforward neural network which maps from the nonlinear separable pattern space to the linear separable state space. It is often used in sensor fault diagnosis, which makes it comparable to the proposed method.

We selected the loss curve and accuracy rate curve to describe the quality of the trained deep network. The abscissa of these two graphs is the training iteration; the ordinate is training loss, equation (8) and accuracy rate, equation (9), respectively. The loss curve (Figure 7) of the CNN was drawn by comparing the actual label and classification results of sensor fault signals. Figure 6 shows the loss curve of 35,000 iterations.

The accuracy rate of the CNN (Figure 8) was calculated by each iteration, and the bias was updated after training the weight. Figure 8 shows loss accuracy variations with the number of iterations when different fault types were input to the deep network. The input of the RBF neural network is the time-frequency signal of the aeroengine control system sensor. We set the target error as 1×10^{-8} and stopped the network training process when the average error function met the target.

The RBF training process is shown in Figure 9. The trained RBF network is very large with 1,354 neurons. For further illustration, the final test results of the trained CNN and RBF models are shown in Table 4.

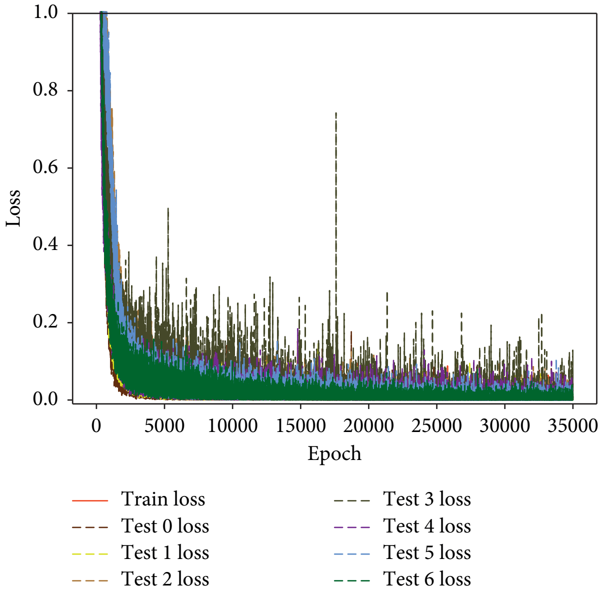


Figure 7: Loss curve.

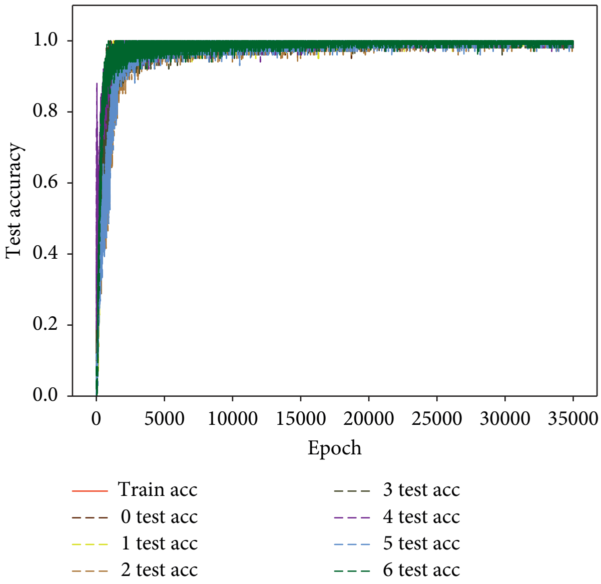


Figure 8: Accuracy rate curve.

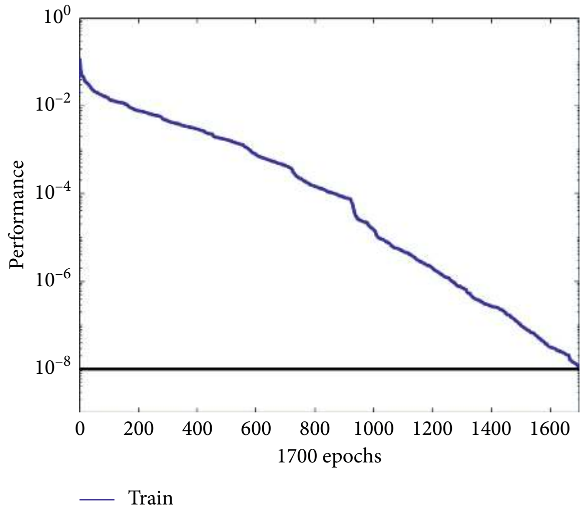


Figure 9: Training mean square error curve of the RBF neural network.

Table 4: Fault diagnosis efficacy of different methods.

Fault detection method		Accuracy rate (%)
CNN + CWT	Training data	98
	Short-circuit fault test set	99.5
	Open-circuit fault test set	98.7
	Spike fault test set	99.09
	Bias fault test set	98.02
	Drift fault test set	97.03
	Normal fault test set	99.09
	Periodic disturbance fault test set	99.6
RBF	Test dataset	91.45

The CNN + CWT method has higher accuracy than the RBF neural network at more than 97% versus only around 91%. The RBF neural network has limited ability to express complex functions, and its input only contains the time-domain information of the signal. By contrast, the proposed method combines time domain and frequency domain information for the purposes of deep learning. Deep learning expresses complex functions with relatively few parameters in extracting essential characteristics from relatively few sample sets, ultimately forming a very effective fault diagnosis scheme. The proposed method is indeed more discriminant than the RBF; the CNN model can effectively identify and isolate aeroengine sensor faults.

6. Conclusion

An intelligent approach combining time-frequency analysis and CNN methodology was established in this study, the CNN + CWT, to transform the signal recognition problem into an image recognition problem for effective aeroengine sensor fault diagnosis. Traditional methods require burdensome modeling and manual feature extraction processes; improper threshold settings lead to false alarms. The proposed method was designed to resolve these and other problems with the traditional sensor fault diagnosis methods. The deep structure of this method allows abundant essential characteristics of datasets to be extracted. Sensor signals representing different health conditions were collected from the records of a certain type of aircraft engine running and simulated in MATLAB. Time-frequency graph features were obtained by using a continuous wavelet transform for the collected and simulated sensor fault signals. We then used historical data features to train the CNN model. Finally, we validated the CNN + CWT by comparison against the RBF neural network method. The accuracy of our trained CNN model reached 97%, exceeding that of the RBF neural network. Experimental results demonstrated its feasibility.

While good experimental results have been obtained by the proposed method, further optimization is still necessary since the current training time is longer than most shallow networks in the literature. In addition, further research is yet needed to recover fault signals through in-depth learning algorithms. It is also necessary to obtain information such as the occurrence time and size of faults to make fully effective fault decisions.

Data Availability

The aeroengine control system sensor fault data used in this study are included within the supplementary information files.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported by National Science and Technology Major Project (2017-V-0011-0062) and the Fundamental Research Funds for the Central Universities (Grant no. 31020190MS702).

Supplementary Materials

The supplementary material file contains the experimental sensor signal set used to produce the dataset upon which we validated the proposed method (Section 2). The datasets were sourced from the records of a certain type of aircraft engine under various fault types and the simulation experiments of MATLAB. The data are given in.mat format including 70% training data and 30% test data, respectively. The dataset can be divided into seven types: normal data and six types of fault data. More details can be found in the data_description.docx in the supplementary material zip file. (*Supplementary Materials*)

References

[1] Q. Yang, S. Li, and Y. Cao, “Multiple model-based detection and estimation scheme for gas turbine sensor and gas path fault simultaneous diagnosis,” *Journal of Mechanical Science and Technology*, vol. 33, no. 4, pp. 1959–1972, 2019.

[2] J. Li and W. Qu, "Aero-engine sensor fault diagnosis based on convolutional neural network," in *Proceedings of the 2018 37th Chinese Control Conference (CCC)*, pp. 6049–6054, IEEE, Wuhan, China, July 2018.

[3] X. Li, Q. Ding, and J.-Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliability Engineering & System Safety*, vol. 172, pp. 1–11, 2018.

[4] P. M. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy," *Automatica*, vol. 26, no. 3, pp. 459–474, 1990.

[5] J.-D. Wu and J.-C. Chen, "Continuous wavelet transform technique for fault signal diagnosis of internal combustion engines," *NDT & E International*, vol. 39, no. 4, pp. 304–311, 2006.

[6] K. K. Botros, G. Kibrya, and A. Glover, "A demonstration of artificial neural-networks-based data mining for gas-turbine-driven compressor stations," *Journal of Engineering for Gas Turbines and Power*, vol. 124, no. 2, pp. 284–297, 2002.

[7] D. Lee, V. Siu, R. Cruz et al., "Convolutional neural net and bearing fault analysis," in *Proceedings of the International Conference on Data Mining (DMIN). The Steering Committee of the World Congress in Computer Science*, p. 194, Computer Engineering and Applied Computing (WorldComp), Warangal, India, December 2018.

[8] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[9] S. Lawrence, C. L. Giles, A. C. Ah Chung Tsoi, and A. D. Back, "Face recognition: a convolutional neural-network approach," *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 98–113, 1997.

[10] R. Girshick, J. Donahue, T. Darrell et al., "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587, Boston, MA, USA, June 2014.

[11] J. Tompson, R. Goroshin, A. Jain et al., "Efficient object localization using convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 648–656, Boston, MA, USA, June 2015.

[12] J. Wang, J. Zhuang, L. Duan et al., "A multi-scale convolution neural network for featureless fault diagnosis," in *Proceedings of the 2016 International Symposium on Flexible Automation (ISFA)*, pp. 65–70, IEEE, Cleveland, OH, USA, August 2016.

[13] D. Verstraete, A. Ferrada, E. L. Droguett, V. Meruane, and M. Modarres, "Deep learning enabled fault diagnosis using time-frequency image analysis of rolling element bearings," *Shock and Vibration*, vol. 2017, Article ID 5067651, 17 pages, 2017.

[14] H. Alaskar, "Deep learning-based model architecture for time-frequency images analysis[J]," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 12, 2018.

[15] R. Chen, X. Huang, L. Xu, X. Zhang, and Y. Zhang, "Intelligent fault diagnosis method of planetary gearboxes based on convolution neural network and discrete wavelet transform," *Computers in Industry*, vol. 106, pp. 48–59, 2019.

[16] C. L. Liu, W. H. Hsaio, and Y. C. Tu, "Time series classification with multivariate convolutional neural network," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 6, pp. 4788–4797, 2018.

[17] S. X. Ding, *Data-driven Design of Fault Diagnosis and Fault-Tolerant Control systems*, Springer, London, UK, 2014.

[18] H. Sun, Z. He, Y. Zi et al., "Multiwavelet transform and its applications in mechanical fault diagnosis–a review," *Mechanical Systems and Signal Processing*, vol. 43, no. 1-2, pp. 1–24, 2014.

[19] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[20] C. Lu, Z. Wang, and B. Zhou, "Intelligent fault diagnosis of rolling bearing using hierarchical convolutional network based health state classification," *Advanced Engineering Informatics*, vol. 32, pp. 139–151, 2017.

[21] D. Gao, Y. Zhu, X. Wang et al., "A fault diagnosis method of rolling bearing based on complex morlet CWT and CNN," in *Proceedings of the 2018 Prognostics and System Health Management Conference (PHM-Chongqing)*, pp. 1101–1105, IEEE, Chongqing, China, October 2018.

[22] Z. Cui, F. Xue, X. Cai, Y. Cao, G.-g. Wang, and J. Chen, "Detection of malicious code variants based on deep learning," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 7, pp. 3187–3196, 2018.

[23] X. Jin, C. Xu, J. Feng et al., "Deep learning with S-shaped rectified linear activation units," *Computer Science*, vol. 3, pp. 1–8, 2015.

[24] N. Srivastava, G. Hinton, A. Krizhevsky et al., "Dropout: a simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

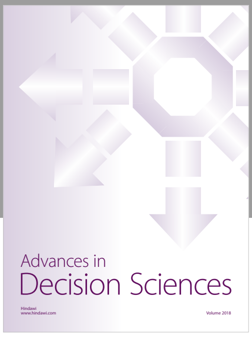
[25] S. Duan, H. Zheng, and J. Liu, "A novel classification method for flutter signals based on the CNN and STFT," *International Journal of Aerospace Engineering*, vol. 2019, Article ID 9375437, 2019.

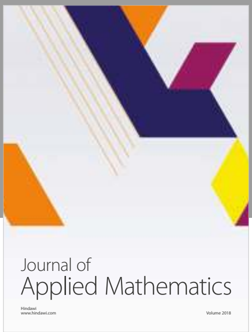
[26] D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," 2014, http://arxiv.org/abs/1412.6980.

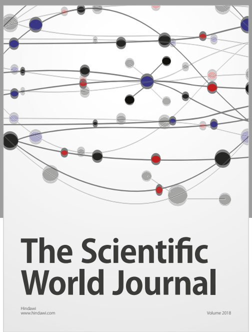
[27] Y. Niu, *Sensor Fault Diagnosis and Signal Recovery Method Based on Artificial Neural Network*, Harbin Institute of Technology, Harbin, China, 1997.

[28] L. Jing, M. Zhao, P. Li, and X. Xu, "A convolutional neural network based feature learning and fault diagnosis method for the condition monitoring of gearbox," *Measurement*, vol. 111, pp. 1–10, 2017.

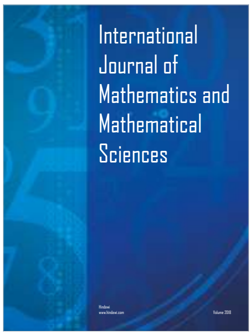
Advances in
Operations Research


Advances in
Decision Sciences

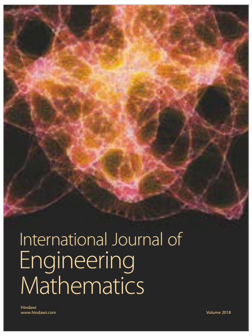

Journal of
Applied Mathematics


The Scientific
World Journal

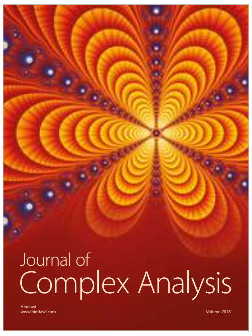
Journal of
Probability and Statistics

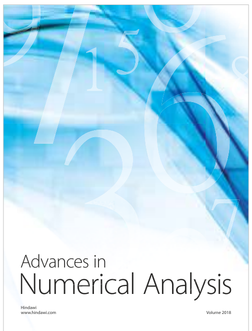

International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Optimization

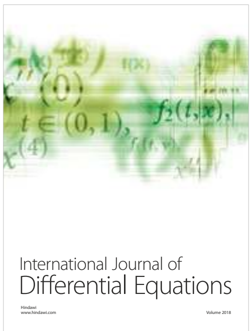

International Journal of
Engineering
Mathematics

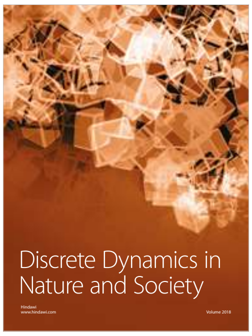
International Journal of
Analysis

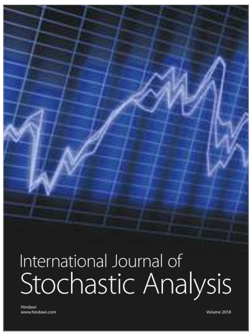

Journal of
Complex Analysis


Advances in
Numerical Analysis

Mathematical Problems
in Engineering

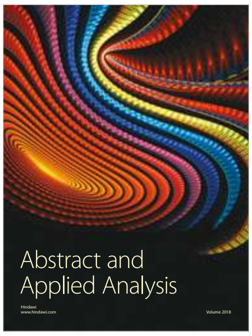

International Journal of
Differential Equations

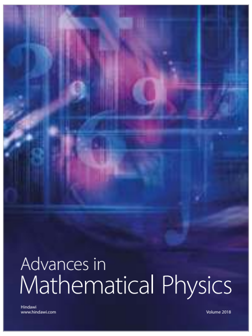

Discrete Dynamics in
Nature and Society


International Journal of
Stochastic Analysis

Journal of
Mathematics

Journal of
Function Spaces


Abstract and
Applied Analysis


Advances in
Mathematical Physics