IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 19, 2020, accepted July 16, 2020, date of publication July 24, 2020, date of current version August 5, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3011728

Intelligent Recognition of Ferrographic Images Combining Optimal CNN With Transfer Learning Introducing Virtual Images

HONGWEI FAN[1,2], SHUOQI GAO[1], XUHUI ZHANG[1,2], (Member, IEEE), XIANGANG CAO[1,2], HONGWEI MA[1,2], AND QI LIU[1]

[1]School of Mechanical Engineering, Xi'an University of Science and Technology, Xi'an 710054, China
[2]Shaanxi Key Laboratory of Mine Electromechanical Equipment Intelligent Monitoring, Xi'an University of Science and Technology, Xi'an 710054, China

Corresponding author: Hongwei Fan (hw_fan@xust.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 51875451, Grant 51834006, and Grant 51974228; and in part by the project of the Shaanxi Key Laboratory of Mine Electromechanical Equipment Intelligent Monitoring of China under Grant SKL-MEEIM201910.

ABSTRACT Ferrography analysis(FA) is an important approach to detect the wear state of mechanical equipment. Ferrographic image recognition based on deep learning needs a large number of image samples. However, the ferrographic images of mechanical equipment are difficult to obtain enough high-quality samples in a short time due to the complexity and low efficiency of the ferrogram making. Therefore, the recognition method for small sample ferrographic images based on the convolutional neural network(CNN) and transfer learning(TL) is proposed. Based on the similarity of samples, the virtual ferrographic image set is designed as the source data of the pretraining model, the tested CNN model is constructed by using the TL. Based on the AlexNet frame, this paper studies the influence of the CNN internal factors including network structure, convolution parameters, activation function, optimization mode, learning rate and the external factors on the classification effect of test samples, and the L2 regularizer is added to solve the overfitting. According to the classification result of test samples, an optimal parameter combination is obtained to establish an intelligent recognition model of ferrographic images based on CNN and TL with the recognition accuracy of 93.75%. Moreover, the t-SNE is used to realize the wear particle recognition process visualization, which proves the effectiveness of the proposed algorithm. This work provides an effective way for the ferrographic image recognition of wear particles under small samples.

INDEX TERMS Ferrographic image, convolutional neural network, transfer learning, wear condition recognition.

I. INTRODUCTION

Wear is one of the main causes of mechanical failure. The particles produced by wear contain a lot of wear information, such as the location, type and degree of wear, etc.. Wear particles mostly exist in the lubricating oil from mechanical equipment. By using the oil analysis technology, the wear status monitoring and identification of equipment can be carried out, and then the potential problems can be found in time, and the equipment can be effectively maintained. Oil analysis technology includes ferrography analysis(FA), spectrum analysis and particle size analysis, etc., which can reveal the evolution trend of wear state and the relationship between the wear state and mechanism [1]–[4]. Among these methods, the FA is most widely used.

FA is mainly based on the analysis of wear particle images. According to the shape, size and texture of wear particles, the wear type and even wear location of mechanical equipment can be determined [5]. The wear condition diagnosis of equipment are carried out by extracting the features such as color [6], surface texture [7], boundary dimension [8] and the relationship between quantitative correlation features and wear morphology [9]. The traditional wear particle recognition process includes image preprocessing, segmentation [10], [11], feature extraction and pattern classification [12]. This process is complex and mainly depends on manual

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

recognition, which has a poor universality. The high cost, low efficiency and poor accuracy of manual identification can be solved by establishing an automatic classification system [13], and the identification process is objective [14], [15]. As a kind of deep learning model the CNN has a good ability of feature extraction and generalization and can realize the automatic classification of wear particles [16], and has been applied in the image recognition of wear particles. The wear particle images collected from the oil sample are often blurred due to the oil pollution and poor light conditions. Wu *et al.* [17] proposed a kind of method to improve image quality by pixel level restoration using larger kernel image, compared to the traditional method its calculation efficiency is higher and the more features are extracted. Szatmari *et al.* [18] applied the CNN model to the separation of metal wear particles and bubbles, they established an online fault monitoring system, and the preliminary experiments show that the method has strong robustness and anti-noise ability. Wang [19] built a CNN model to identify seven kinds of ferrographic images based on deposition chain ferrograph and block debris image, aiming at the problems of poor universality of traditional recognition methods. Wu [20] built a Wear-Net wear particle image classification model and wear-SSD target detection model based on Reference [19] to classify the single type debris and detect the composite debris. Although both have achieved good results, there is also a lack of sufficient data in the model design. Peng *et al.* [21] proposed a classification model of wear particles considering overlapping particles, they used the Inception-v3 model to automatically extract the characteristics of wear particles and designed a new network with three classifiers to determine whether there are fatigue, oxide and spherical particles in the ferrographic images. In addition, they also proposed a FECNN model with one-dimensional convolution operation in CNN to identify wear particles [22]. Wang *et al.* [23] proposed an integrated model of BP neural network and CNN for wear debris classification, the test results show that the identification rates are all over 80%. An *et al.* [24] used a CNN model and TL to identify the fatigue and serious sliding particles, and the accuracy is 89.35%. Peng *et al.* [25] combined TL and support vector machine(SVM) to identify four types of wear debris including cutting, sphere, fatigue and severe sliding particles, and they proposed a method to identify the type of wear particles by extracting the characteristics of wear particles in dynamic video [26].

The above researches are beneficial to the intelligent recognition of ferrographic images based on the machine learning, but there is still no optimal solution for small sample images. In order to solve the low efficiency problem of ferrographic image production, this paper proposes a new intelligent recognition approach based on the CNN and TL by introducing virtual image set to deal with a small sample of ferrographic images. The structure of this paper is as follows: Section 1 is the basis of CNN and TL used, Section 2 is the acquisition method of virtual and measured image set, Section 3 is to build the recognition model and select the best parameters, Section 4 is the results and discussion, and the final section is the conclusion.

II. BASIS OF METHOD

A. CONVOLUTIONAL NEURAL NETWORK

CNN is an important branch of deep learning, which is a feedforward neural network [27], [28] according to the biological receptive field mechanism. The typical structure of CNN is generally composed of the input layer, convolution layer(Conv), pooling layer(Pool), fully-connected layer(FC) and output layer, as shown in Figure 1.

As shown in Figure.1, Conv is used to extract the features of input images. The input images are extracted by the inner product operation with the convolution kernel composed of weight matrix to generate the featured images. The formula of operation is:

$$X^{(k)} = f(\sum_{i=0}^{n}\sum_{j=0}^{n} W_{i,j}^{(k)} I_{l+i,m+j}^{(k-1)} + b^{(k)}) \tag{1}$$

where $X^{(k)}$ is the featured map of the output, $I_{l+i,m+j}^{(k-1)}$ is the featured map from the upper layer, $W_{i,j}^{(k)}$ is the weight matrix, $b^{(k)}$ is the bias, f is the activation function. l, m are two dimensions of the featured graph of the previous layer, and k is the current layer. When $k = 1, I$ indicates the input images.

Pool is to reduce the dimension of the featured map obtained by the convolution, which makes the feature robust to the position change of the input images. Meanwhile, Pool can reduce the amount of model parameters. Pool is usually the max or average pooling, i.e., taking the maximum or average value of the eigenvalue in the pooling area.

FC is to map the high-dimensional features from the Conv and Pool to the sample marker space to realize the classification. The high-dimensional feature is the input of the fully-connected layer, and each feature participates in the calculation as a neuron. The calculation formula is as follows:

$$z_i^{(k)} = X_i^{(k-1)} W^{(k)} + b^{(k)} \tag{2}$$

where $z_i^{(k)}$ is the featured map of the fully-connected layer, $X_i^{(k-1)}$ is the featured map processed by dimension reduction in the $(k-1)$-th layer.

For the output feature mapping, the Softmax function is used to calculate the likelihood probability of each category, and the category corresponding to the maximum likelihood probability is taken as the result of data classification and output by the output layer. The Softmax function is as follows:

$$P(y_i) = \frac{e^{z_i^{(k)}}}{\sum_{u=1}^{K} e^{z_u^{(k)}}} \tag{3}$$

where y_i represents the predicted sample category, $P(y_i)$ is the probability predicted as y_i category, and K is the number of sample categories. Especially, $\sum_{i=1}^{K} P(y_i) = 1$.

Then, the classification result is:

$$P(\hat{y}) = \max(P(y_i)), \quad 0 < i < K \tag{4}$$

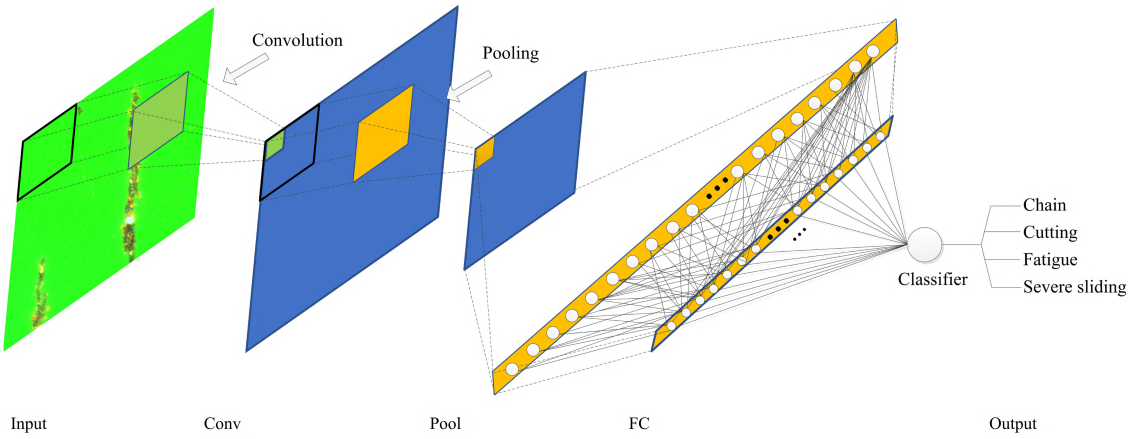


FIGURE 1. Structure of a typical CNN model.

where $P(\hat{y})$ represents the probability that the prediction is $\hat{y}$, y_i represents the real. When $\hat{y} = y_i$, the prediction result is correct; otherwise, wrong.

Assuming M is the total number of samples, and N is the number of correct predictions, then the prediction accuracy is:

$$p = \frac{N}{M} \tag{5}$$

Generally, the cross entropy function is used as the loss function to calculate the error. The error expression between the predicted value and the real value is:

$$L_{loss} = -\frac{1}{M}\sum_{i=1}^{K} y_i \log\left(\hat{y}\right) + (1 - y_i)\log\left(1 - \hat{y}\right) \tag{6}$$

The cross entropy is a part of KL divergence, which is used to measure the difference between two different probability distributions. The smaller the cross entropy is, the smaller the KL divergence is, and the closer the two distributions are, i.e, the more accurate the predicted result is.

B. TRANSFER LEARNING

TL is a new kind of machine learning method to deal with small samples, which relaxes the hypothesis that the training and test set are independent and equally distributed and the number of training samples is sufficient [29], aiming to transfer the existing knowledge to solve the learning problem of only a few number of labeled samples in the target area [30].

Figure 2 shows the schematic diagram of TL method, the large-scale image set is used to train a neural network model, and then part of the network structure and its parameters in the pretraining model is transferred to the classification model. In this paper, the parameters of the convolution layer trained by the pretraining model is applied to the classification model by TL scheme. The target image set is used to retrain the model. In the training process, all the convolution layers are frozen, and only the fully-connected layer participates in the network updating. Finally, the classification results are obtained.

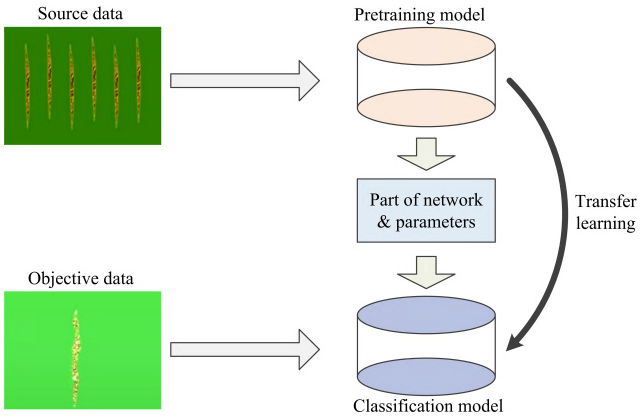


FIGURE 2. Schematic diagram of the TL used.

III. SAMPLE PREPARATION

A. MEASURED SAMPLE PREPARATION

1) EXPERIMENTAL SETUP

In this paper, for the lubricating oil of a gearbox as shown in Figure 3, an analytical ferrograph as shown in Figure 4 was used to make the ferrogram. In Figure 4, the ferrogram making system deposits the wear particles on the glass substrate through a magnetic field generator, and then obtains the digital images through an imaging system composed of a microscope and an upper computer. The parameters of analytical ferrograph are shown in Table 1.

2) FERROGRAM PREPARATION

As shown in Figure 5, the process of preparing the wear particle ferrogram by an analytical ferrograph shown in Figure 4 includes five key steps: sampling, dilution, water bath oscillation, ferrogram making and drying.

As shown in Figure 5, take 6ml oil into a test tube, add 2ml tetrachloroethylene as the diluent, and oscillate in a warm

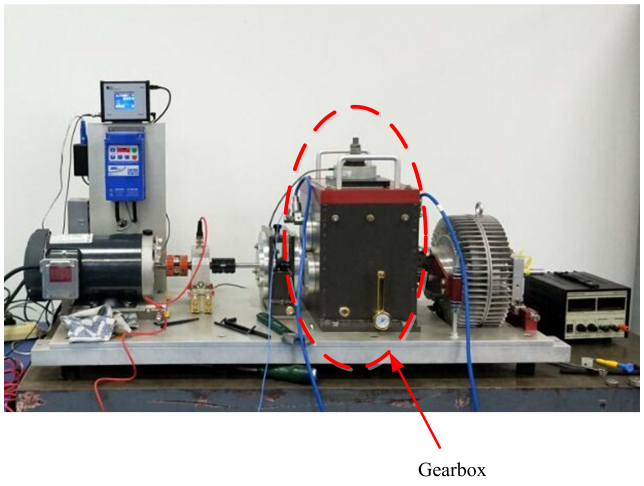


FIGURE 3. Gearbox platform used for oil collection.

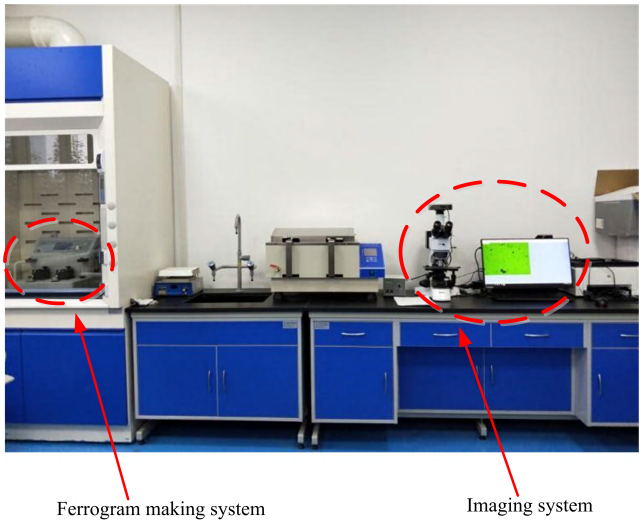


FIGURE 4. Ferrographic image preparation platform.

water bath for 30min at the frequency of 68 times/min at 65° to make the oil fully diluted to ensure that the stacked particles are separated during the process and the particles are evenly distributed. The ferrogram making principle of analytical ferrograph is shown in Figure 5 (d), a micro quantitative pump is used to press out the lubrication oil in the tube and make it drop on the ferrogram, the wear particles in the oil are adsorbed on the ferrogram under a strong magnetic field, and then the residual oil on the ferrogram is dried by a heating device, finally an effective ferrogram is made after the temperature dropping to the room temperature.

3) DIGITAL IMAGE ACQUISITION

Put the prepared ferrogram on the stage of a microscope, and observe the distribution of the wear particles. Then capture the debris image by an image acquisition software on the upper computer. The wear particle types include four categories and six species [31]. Different types of wear particles

TABLE 1. Parameters of ferrographic image preparation platform.

Project	Parameter
Oil sample / solvent consumption	6ml, 2ml
Water bath oscillation temperature / frequency	65℃, 68 times/min
Water bath heating temperature / time	330℃~140℃, 90s
Microscope min / max magnification	10× objective lense, 50× objective lense
Microscope background light intensity / image capture ISO	Secondary, automatic
Image capture / save / process resolution	2592×1944, 2592×944, 220×160

TABLE 2. Types and image characteristics of wear particles.

Type of wear particles	Image feature
Chain debris	Wear Particle distributed in chain along the magnetic line on the ferrograph
Cutting debris	Chip-like, spiral and arc-shaped
Fatigue debris	Thin block, smooth surface with pitting, irregular outline
Severe sliding debris	Smooth surface with obvious parallel scratches or cracks, and the straight edge

produced by different wear mechanisms have different image characteristics. Table 2 shows the main types of wear particles and their image characteristics.

The ferrographic images obtained by an imaging system shown in Figure 4 include the chain, cutting, fatigue and severe sliding debris, as shown in Figure 6.

When we capture the ferrographic images, the problem of particle accumulation is inevitable due to the influence of light conditions and oil state. It is necessary to select the effective samples from the original samples. Finally, 640 valid samples were used for the model training, 160 samples per category. According to training set : test set=8:2, these samples were randomly divided.

B. DESIGN OF VIRTUAL FERROGRAPHIC IMAGE

According to the debris image obtained by the above experiment, the corresponding virtual ferrographic images were designed, as shown in Figure 7.

According to the sample similarity principle, the virtual ferrographic images were designed, which consists of 50 images for each category, i.e., 200 images for 4 categories. Using the random rotation, brightness and contrast

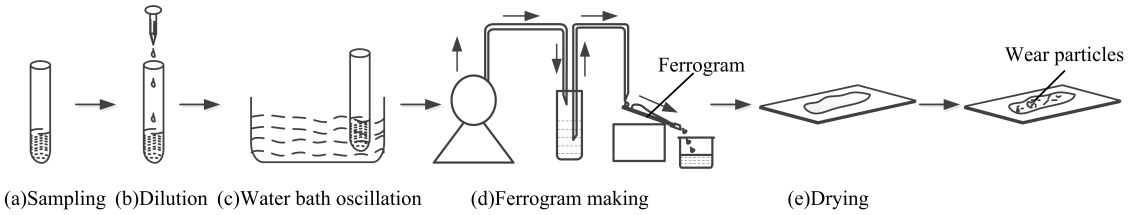


FIGURE 5. Preparation process of ferrogram.

adjustment, and other data enhancement methods, the virtual image samples are expanded to 6500 images for each category, i.e., a total of 26000 images. According to training set : verification set : test set=6:2:2, these virtual images were randomly divided to train the model.

IV. MODEL RESEARCH

Based on the CNN, a pretraining model and a classification model for TL are designed. The pretraining model uses the virtual ferrographic image set, and the classification model uses the measured ferrographic image set. Based on the AlexNet frame, this paper studies the model from the aspects of network structure, parameters, activation function, optimizer and overfitting solution, and analyzes the classification effect of the model with different parameters. The pretraining model is to train the convolution layer parameters for the image feature extraction, so the convolution layer is the research focus, including the convolution layer structure, parameters, activation function and others. The classification model is to classify the images by the high-dimensional features extracted from the convolution layer, so the generalization ability, convergence speed and accuracy are the main research points, including the fully-connected layer structure, optimization method, learning rate and regular term processing. Take the classification accuracy rate and model convergence rate as the evaluation indexes, When the iteration stops, the optimal combination of network parameters will be determined, and the intelligent recognition model for small sample ferrographic images will be obtained. The initial parameter settings of the model are shown in Table 3.

A. NETWORK STRUCTURE

The network structure is the first problem to be considered in a CNN model. The recognition effect of wear particles and the generalization ability of the model are closely related to the complexity of the network structure. In this paper, we used the AlexNet as a basic network frame, and adjusted the location of the pooling layer properly without changing the numbers of the convolution layer and the pooling layer, so as to select the network structure more suitable for the target image classification.

1) POOLING LAYER

Figure 8 is the schematic diagram of AlexNet-based network structure, which is composed of five convolution layers and three fully-connected layers, including three pooling layers.

TABLE 3. Initial parameter settings of the model.

Parameter	Value
Kernel(length×width×number×stride)	3×3×4×1
Pooling(mode, length×width×stride)	Max pooling, 2×2×2
Weight initialization(initialization mode, mean, variance)	Truncated normal distribution, mean=0, variance=0.01
Bias	0
Activation function	ReLU
Loss function	Cross entropy
Classification function	Softmax
Optimization method	Stochastic gradient descent
Learning rate	0.001
Batch size(virtual data set)	Train set=30, validation set=10, test set=10
Batch size(measured data set)	Train set=40, test set=10
BN layer	None
Regularization	None
Dropout	None

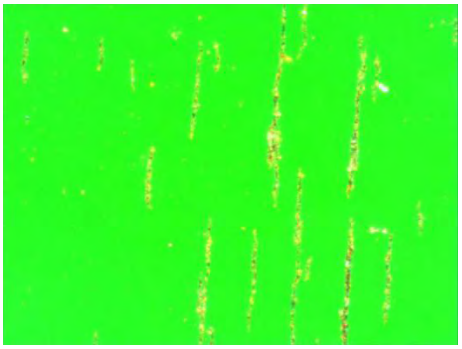
TABLE 4. Quantitative results of the model when the pooling layer is located at different positions.

Pooling layer	Pool3	Pool4	Pool5
Accuracy	0.8519	0.8442	0.7615
Error	0.4165	0.5320	0.8243

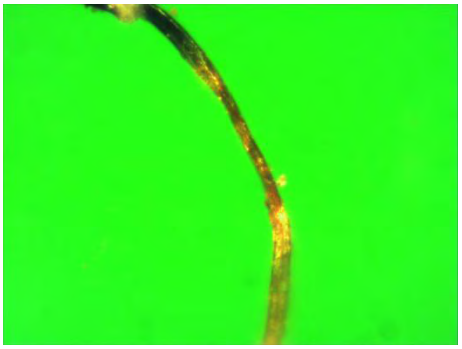
By properly adjusting the position of the third pooling layer, the recognition accuracy and error of the test set are studied, and the optimal structure can be selected.

In Figure 8, the third pooling layer is Pool5, and its position is adjusted successively after Conv3 and Conv4, i.e., Pool3 and Pool4. The performance of the model under the Pool3, Pool4 and Pool5 is studied respectively. The classification accuracy and error of the test set are shown in Figure 9 and Table 4.

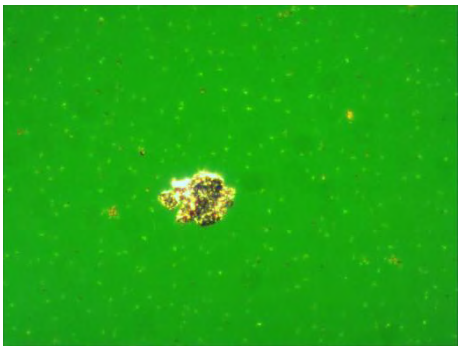
It can be seen from Figure 9 (a) and Table 4 that the test set accuracy of the model under the Pool5 is lower than those under the Pool4 and Pool3, and the result of Pool3 is slightly higher than that of Pool4. Accordingly, in Figure 9 (b), the error value of the model is the smallest when the training process is stopped under the Pool3. Therefore, the Pool3 is the best choice, i.e., the third pooling layer should be placed after the convolution layer 3.



(a) Chain debris



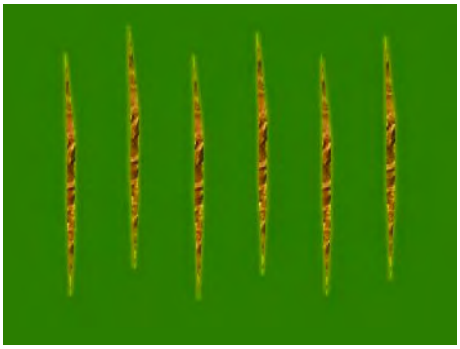
(b) Cutting debris



(c) Fatigue debris

(d) Severe sliding debris

FIGURE 6. **Wear particle image captured in the experiment.**



(a) Chain debris

(b) Cutting debris

(c) Fatigue debris

(d) Severe sliding debris

FIGURE 7. **Virtual ferrographic images designed.**

2) FULLY-CONNECTED LAYER

The TL of the pretraining model is only used for the feature extraction, i.e., the convolution layer, so the fully-connected layer needs to be redesigned and trained in the classification model. Increasing the number of fully-connected layers will deepen the network structure, make the model more complex and generate more random factors. In this paper, on the

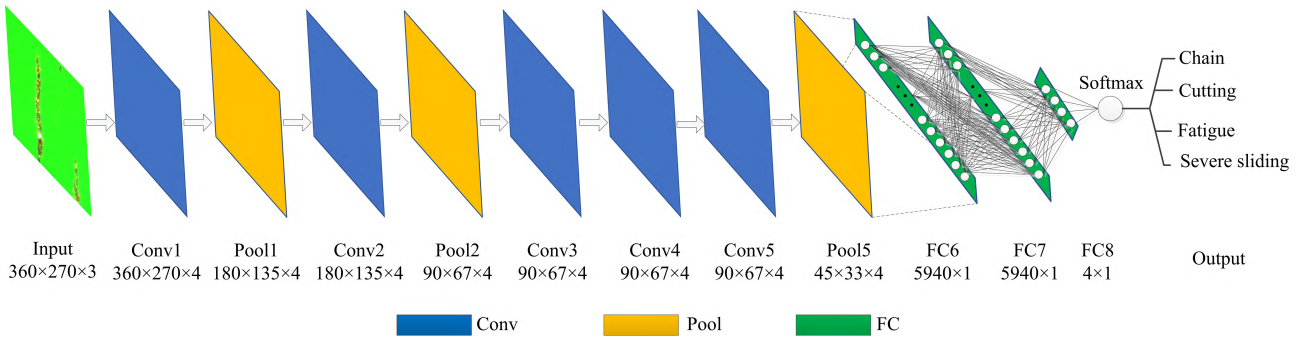


FIGURE 8. Schematic diagram of AlexNet-based CNN structure.

basis of AlexNet frame, the number of fully-connected layers is decreased layer by layer, and the effect on the model classification is studied. It can be seen from Figure 8 that AlexNet frame contains three fully-connected layers. Therefore, the performance of models with 1, 2 and 3 fully-connected layers is studied here, and the results are shown in Figure 10 and Table 5.

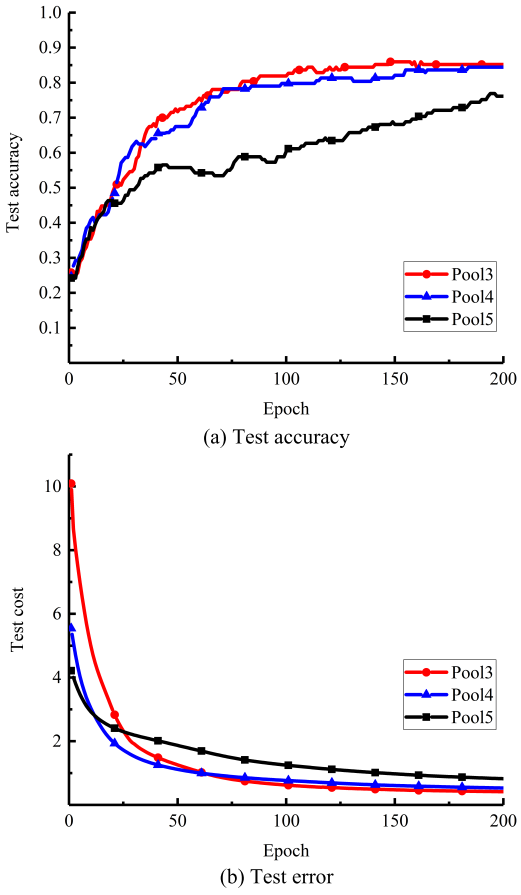


FIGURE 9. Performance curves of the model when the pooling layer is located at different positions.

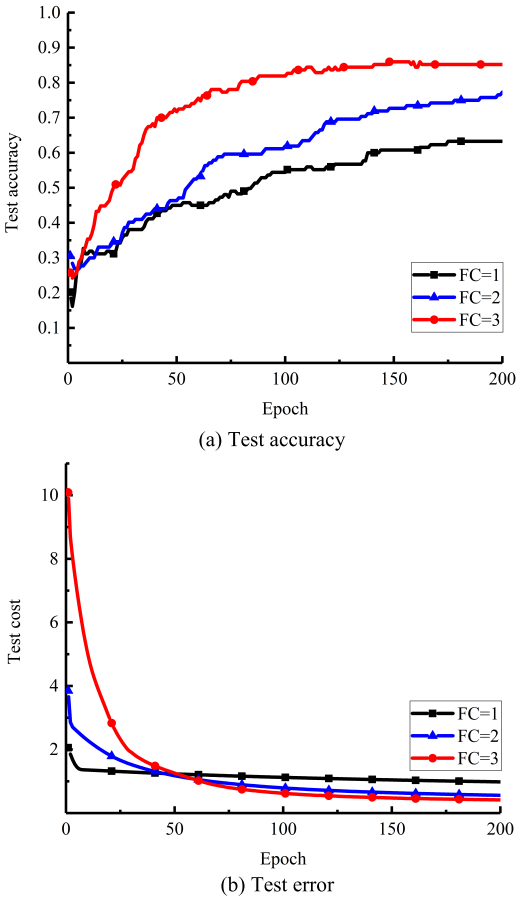


FIGURE 10. Effect of the number of fully-connected layers on the model.

TABLE 5. Quantitative results of the number of fully-connected layers on the model.

FC number	1	2	3
Accuracy	0.6327	0.7731	0.8519
Error	0.9839	0.5603	0.4165

The results in Figure 10 and Table 5 show that with the decrease of the number of fully-connected layers, the recognition accuracy of the model decreases, and the model tends to be underfitting. Meanwhile, the error increases with the

TABLE 6. Quantitative results of the model under different parameter initialization methods.

Initialization methods	Truncated normal distribution	Normal distribution	Uniform distribution
Accuracy	0.8519	0.5442	0.5769
Error	0.4165	1.9832	3.0104

decrease of the total number of connecting layers. In this paper, the network structure of three fully-connected layers shows a better classification effect. Therefore, three fully-connected layers are selected to build the classification model.

B. INITIALIZATION OF WEIGHT PARAMETER

This paper studies three initialization methods of weight parameters, including the normal distribution, the truncated normal distribution and the uniform distribution. The formulae are:

$$f(x) = \frac{1}{\sqrt{2\pi}\delta} e^{-\frac{1}{2}(\frac{x-\mu}{\delta})^2} \tag{7}$$

$$f(x;\mu,\delta,a,b) = \frac{\frac{1}{\delta}\phi(\frac{x-\mu}{\delta})}{\Phi(\frac{b-\mu}{\delta}) - \Phi(\frac{a-\mu}{\delta})} \tag{8}$$

$$f(x) = \begin{cases} \dfrac{1}{d-c}, & c < x < d \\ 0, & \text{others} \end{cases} \tag{9}$$

Equations (7) and (8) respectively represent the probability density, μ is the mean value and δ is the standard deviation of the normal distribution and truncated normal distribution, a and b are the upper and lower limits of variable value range in the truncated normal distribution. The difference between the truncated snormal distribution and the normal distribution is that the truncated normal distribution limits the value range of variables, $a = \mu - 2\delta$, $b = \mu + 2\delta$. Equation (9) represents the probability density of the uniform distribution, c and d are upper and lower limits respectively. Different weight initialization methods produce the parameters that obey different distribution, and the larger or smaller weight is not conducive to classification. If the weight is too large, the gradient explodes easily, and if the weight is too small, the gradient will disappear. This paper studies the classification of the model based on the above three initialization methods of weight parameters, and the results are shown in Figure 11 and Table 6.

It can be seen from Figure 11 that when the parameters are initialized with the normal distribution and uniform distribution, the performance of the model is relatively close. Compared with normal distribution data, the truncated normal distribution data are more centralized, more data are distributed near the mean value, and the model training speed is faster. In Table 6, when the truncated normal distribution is used for the initialization, the recognition accuracy of model is higher, the convergence speed is faster, and the error is smaller. Therefore, this paper chooses the truncated normal distribution to initialize the weight parameters.

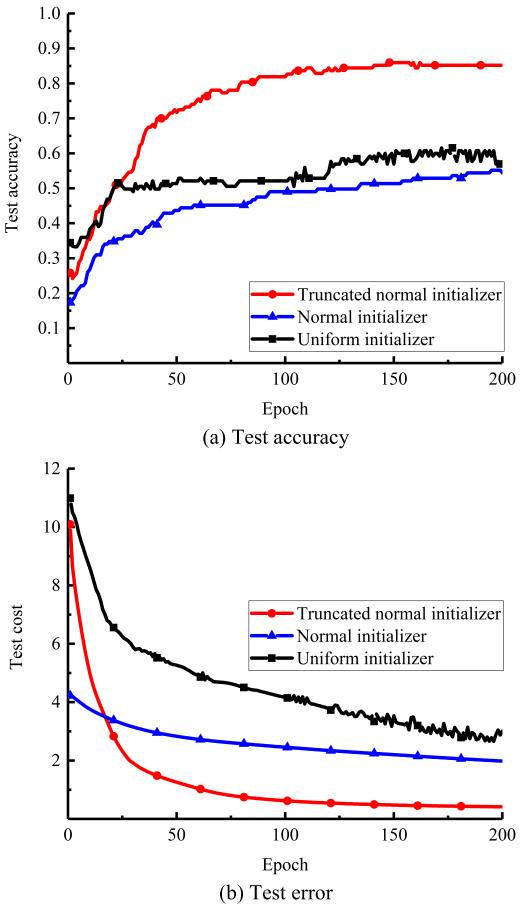


FIGURE 11. Performance curves of the model under different parameter initialization methods.

C. HYPERPARAMETER IN CONVOLUTION LAYER

The convolution layer extracts the image features by the convolution operation between the convolution kernel and image. Choosing the reasonable parameters can enrich the image features extracted in the convolution process and make the model classification better. The convolution operation parameters include the convolution kernel size, number and convolution stride.

1) CONVOLUTION KERNEL SIZE

The convolution kernel is composed of multi-dimension data, and its function is similar to that of filter. The size of convolution kernel is related to the range of receptive field. The larger the convolution kernel is, the larger the range of receptive field is, the more parameters are, and the slower the network operation speed is. When 1×1 kernel is selected, the receptive field is a number. When the multiple 1×1 convolution kernels are used, each input can be regarded as a neuron. The process of convolution operation is the linear operation between multiple input neurons and multiple weights. At this

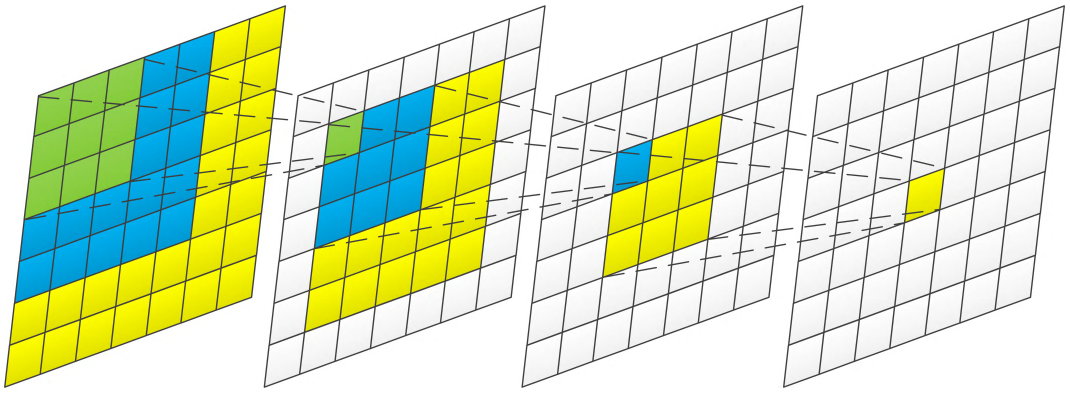


FIGURE 12. Schematic diagram of equivalent receptive field of convolution kernel with $s_{conv} = 1$.

TABLE 7. Quantitative results of the model under different convolution kernel sizes.

Sizes of kernel	3×3	5×5	7×7
Accuracy	0.8519	0.4712	0.6135
Error	0.4165	9.4759	3.7204

time, the convolution layer can be regarded as the fully-connected layer. Therefore, 1 × 1 convolution kernel is not used.

The schematic diagram of the receptive field corresponding to convolution kernels of 3 × 3, 5 × 5 and 7 × 7 is shown in Figure 12. When the convolution stride $s_{conv} = 1$, it can be seen from Figure 12 that the receptive field of two 3 × 3 convolution kernels is the same as that of one 5 × 5 convolution kernel, and the receptive field of three 3 × 3 convolution kernels is the same as that of one 7 × 7 convolution kernel.

In this paper, we study the classification effect of the model when the convolution kernel size is 3 × 3, 5 × 5 and 7 × 7 respectively. The results are shown in Figure 13 and Table 7.

It can be seen from Figure 13 that under the same number of convolution kernels, when the size of convolution kernels is 3 × 3, compared with the large-scale convolution kernels, the parameters are less, the calculation speed is faster, the performance on the test set is better than the other two large convolution kernels. And in Table 7, the model accuracy is higher, and the convergence speed is faster. Therefore, the convolution kernel of 3 × 3 is chosen in this paper.

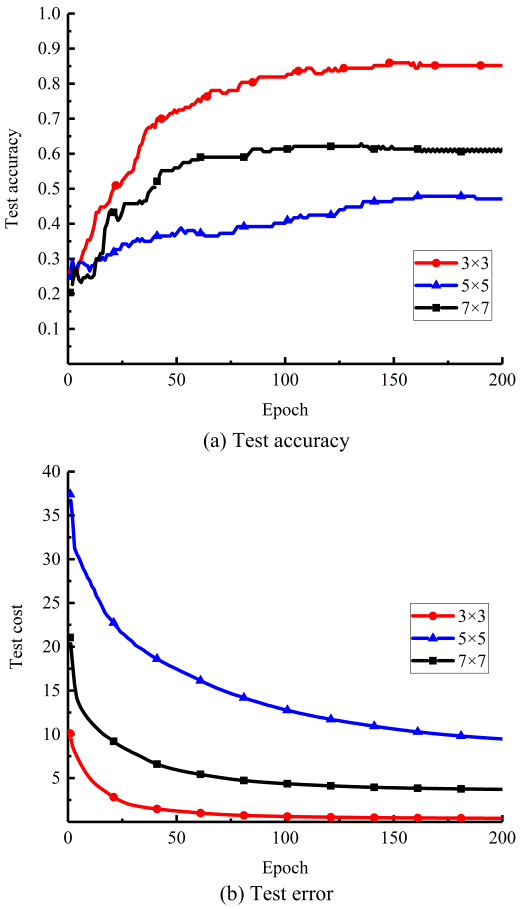


FIGURE 13. Performance curves of the model under different convolution kernel sizes.

2) NUMBER OF KERNELS

Only one convolution kernel is needed for a single channel image convolution. However, for the RGB three channel color image, each channel needs a convolution kernel to do the convolution operation with the image. Using the multiple convolution kernels can be regarded as describing the image from multiple perspectives, reflecting more complete information of the image and more extracted feature information. The more convolution kernels are, the better the effect is. Too many convolution kernels may interfere with the extracted image information and make the model overfitting. This paper studies the performance of the model under the 4, 8, 16, 32 and 64 convolution kernels, and the results are shown in Figure 14 and Table 8.

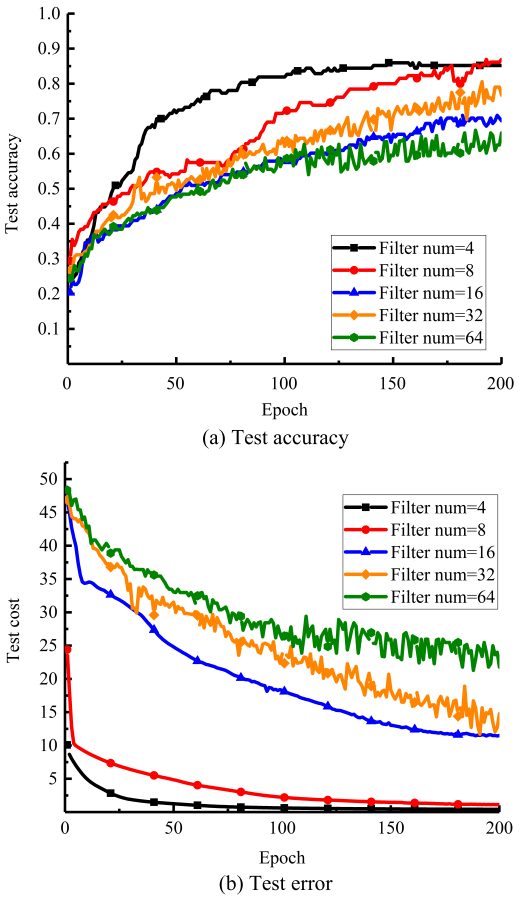


FIGURE 14. Performance curves of the model under different convolution kernels.

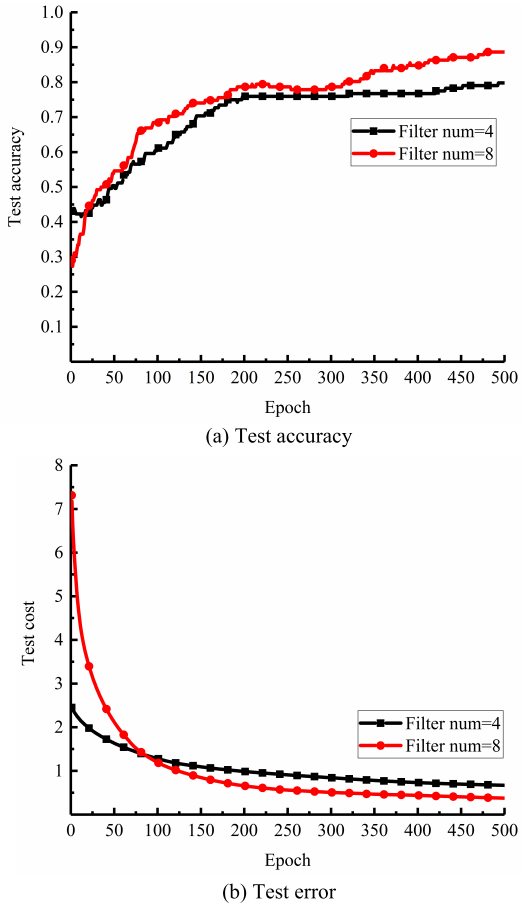


FIGURE 15. Performance curves of the model after increasing the number of iterations.

TABLE 8. Quantitative results of the model under different convolution kernels.

Number of kernel	4	8	16	32	64
Accuracy	0.8519	0.8692	0.6942	0.7673	0.6596
Error	0.4165	1.1307	11.4983	14.7860	21.7846

TABLE 9. Quantitative results of the model after increasing the number of iterations.

Number of kernel	4	8
Accuracy	0.7981	0.8865
Error	0.6710	0.3771

It can be seen from Figure 14 and Table 8 that the number of convolution kernels has no regular effect on the model. When the number of convolution kernels is greater than 8, the error increases gradually, the convergence speed of the model slows down, and the volatility of the data is also increasing.

As the accuracy of the model is close when the number of convolution kernels is 4 and 8, and the accuracy is still rising when the number of convolution kernels is 8, increase the number of iterations to obtain the better training results. The test results of the model after the number of iterations increased are shown in Figure 15, and the quantitative results are shown in Table 9.

It can be seen from Figure 15 and Table 9 that with the increase of iteration times, the model can obtain the higher accuracy and lower error under the 8 convolution kernels. Therefore, 8 convolution kernels are selected to construct the model.

3) CONVOLUTION STRIDE

The convolution stride represents the amount of data between two convolution operations in the receptive field. When the stride is less than or equal to the kernel size, the receptive fields of the two convolution operations will be closely connected or overlapped. The convolution layer can extract the feature information of the image completely, and the model classification is good. When the stride is greater than the convolution kernel size, the receptive fields of the convolution operation will lose part of the image information, and the extracted features are insufficient, and the model classification result is poor. This paper studies the performance of the

TABLE 10. Quantitative results of the model under different convolution steps when the convolution kernel size is 3 × 3.

s_{conv}	1	2	3	4
Accuracy	0.8692	0.7327	0.4981	0.5115
Error	1.1307	0.6852	1.0399	1.0718

model when the strides are 1, 2, 3 and 4 respectively, and the results are shown in Figure 16 and Table 10.

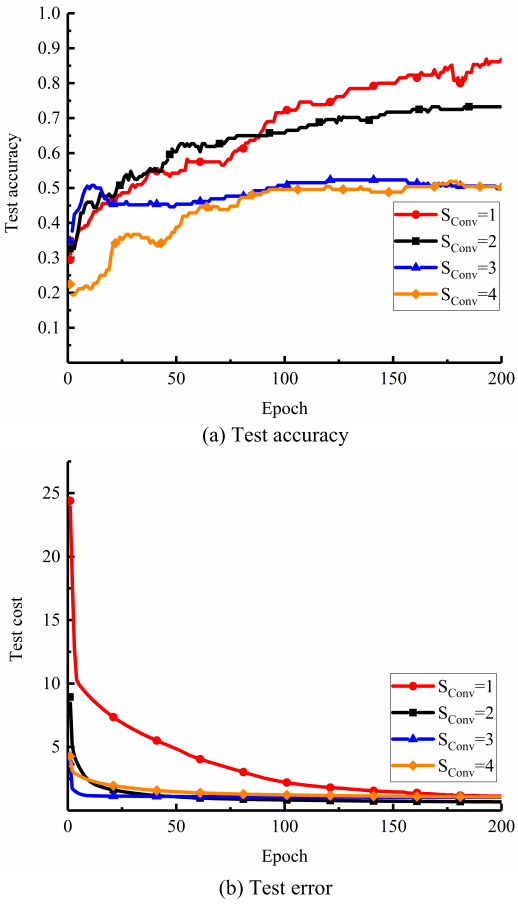


FIGURE 16. Performance curves of the model under different convolution steps when the convolution kernel size is 3 × 3.

As shown in Figure 16, with the increase of convolution step, the performance of the model on the test set becomes worse and worse. When $s_{conv} = 1$, the overlapped part of the receptive field is the most, the feature extracted by kernels is more abundant, and the performance of the model in the test set is the best. In Table 10, when $s_{conv} = 1$, the accuracy is larger than others. Therefore, the stride is taken as $s_{conv} = 1$.

D. HYPERPARAMETER IN POOLING LAYER

Pooling is also called subsampling. A filter similar to the convolution kernel is used to take the maximum or average value of the specified region of the characteristic graph output by the convolution operation. The output data of pooling is far less than the input data, so the pooling can reduce the dimension of data.

TABLE 11. Quantitative results of the model under different pooling methods.

Pooling method	Max pooling	Average pooling
Accuracy	0.7865	0.6442
Error	0.6582	0.9129

1) POOLING METHOD

There are two ways of pooling: max pooling and average pooling. Max pooling is to take the maximum eigenvalue of the specified area in the feature map as the output value; average pooling is to take the average eigenvalue. For the two pooling methods, the results are shown in Figure 17 and Table 11.

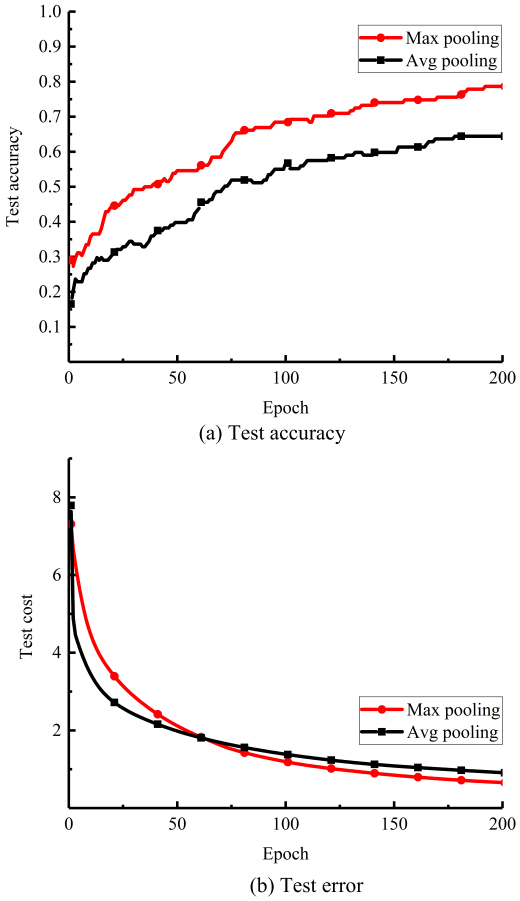


FIGURE 17. Performance curves of the model under different pooling methods.

It can be seen from Figure 17 that under two pooling methods, the max pooling can obtain higher accuracy, faster convergence speed and lower error, so the maximum pooling is selected.

2) POOLING AREA SIZE

The pooling size is similar to the kernel except that it performs different operations. The pooling area specifies the size of the

TABLE 12. Quantitative results of the model under different pooling area sizes.

C	2	3	4
Accuracy	0.8981	0.5615	0.6981
Error	1.0931	1.1215	1.4268

area where the pooling operation is performed, and then run the pooling function within the area. When the side length of the pooling area $C = 1$, it is equivalent to copy the feature map, which has no effect on the training process, so the minimum value is 2. The size of the pooling area affects the size of the output characteristic map of the pooling layer. When the pooling area is too large, the amount of data in the output characteristic map is insufficient, resulting in the poor classification results. The results are shown in Figure 18 and Table 12.

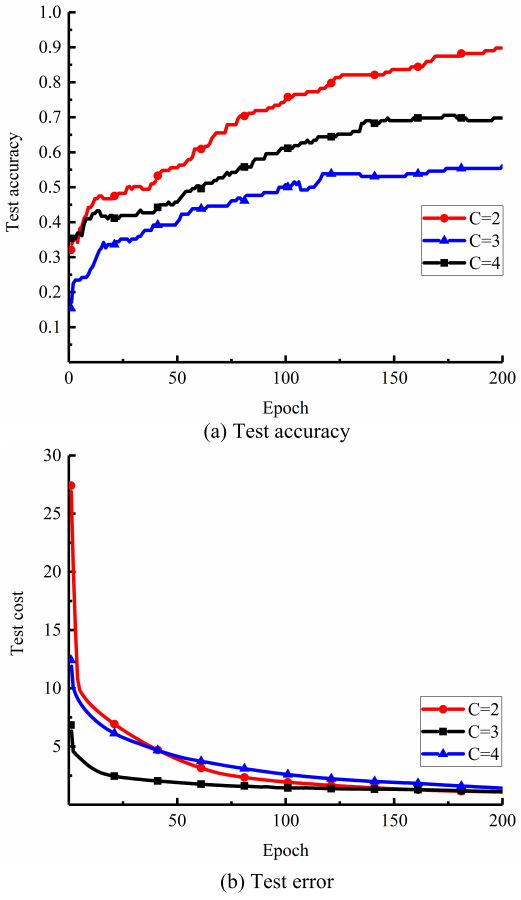


FIGURE 18. Performance curves of the model under different pooling area sizes.

As shown in Figure 18 and Table 12, when the side length of pooling area $C = 2$, the model has the highest accuracy, the smallest error and the best performance on the test set. Therefore, the pooling area size should be 2×2.

TABLE 13. Quantitative results of the model under different pooling steps.

s_{pool}	2	3
Accuracy	0.8981	0.6154
Error	1.0931	1.1298

3) POOLING STRIDE

The pooling stride s_{pool} is similar to the convolution step size s_{conv}, which refers to the interval of each movement of the area when the pooling operation is performed. When $s_{pool} > C$, part of the eigenvalues will be lost, resulting in the poor model classification effect due to the insufficient feature data. When $s_{pool} = 2$ and $s_{pool} = 3$ respectively, the performance of the model is studied, and the results are shown in Figure 19 and Table 13.

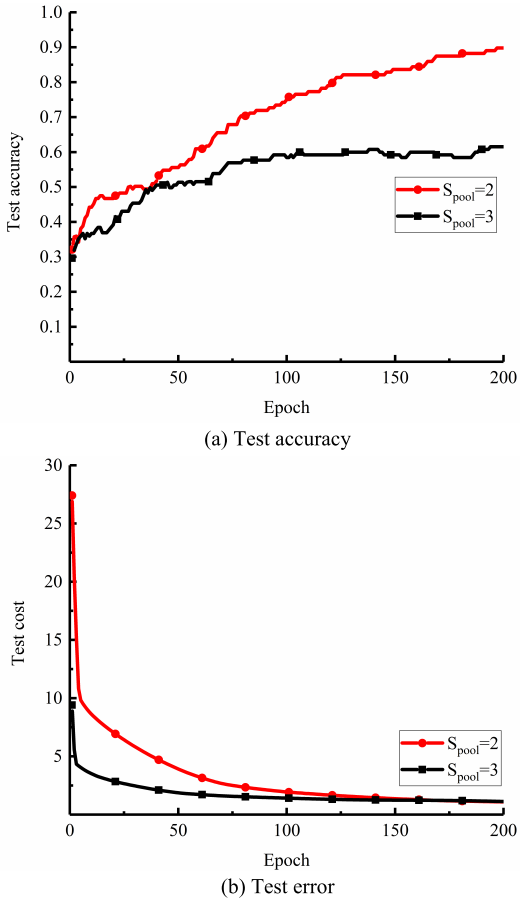


FIGURE 19. Performance curves of the model under different pooling steps.

The results show that in the case of $C = 2\times2$, when $s_{pool} > 2$, the feature extraction of the featured map from the pooling layer to the convolution layer is not complete, and there is no effective feature classification, the recognition accuracy of the model is reduced, and the performance is poor. Therefore, in this paper the pooling step size $s_{pool} = 2$.

E. ACTIVATION FUNCTION

For the output value of the linear operation in the convolution or fully-connected layer, its characteristic is often not linearly separable, so it is necessary to introduce the nonlinear factor to meet the classification requirement. The activation function can realize the non-linearity of the output result of linear operation. The expressions of sigmoid, tanh and ReLU functions are as follows:

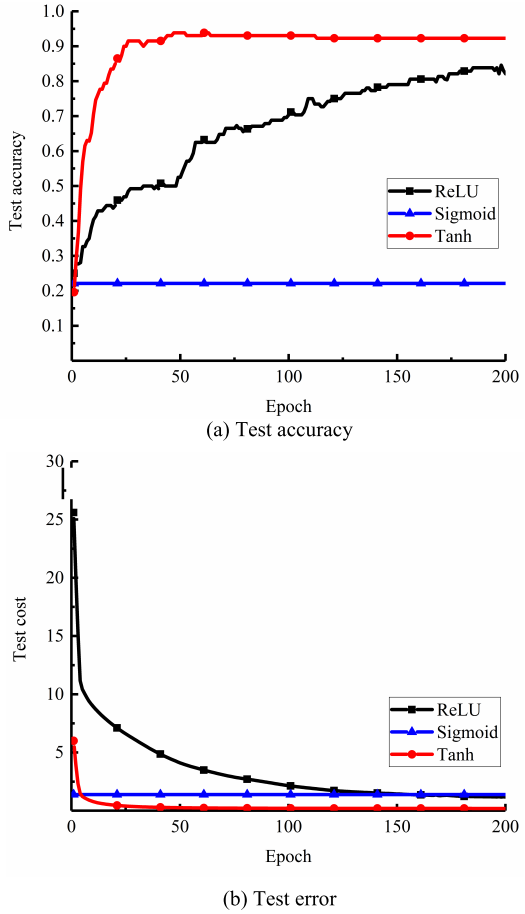
$$\text{sigmoid}(z) = \frac{1}{1+e^{-z}} \tag{10}$$

$$\tanh(z) = \frac{e^{z}-e^{-z}}{e^{z}+e^{-z}} \tag{11}$$

$$\text{ReLU}(z) = \max(z, 0) \tag{12}$$

where z is the output value of the linear operation.

With the action of three different activation functions, the performance of the model is shown in Figure 20 and the quantitative result is shown in Table 14.



(a) Test accuracy

(b) Test error

FIGURE 20. Performance curves of the model under different activation functions.

As shown in Figure 20, the sigmoid function has an input value distributed in the saturated area of the function, so that the derivative of the function is 0 when the error back propagation occurs, the gradient of the loss function disappears, and the parameters are not fully trained. After the migration, the parameters do not have the ability to extract the image features, so the accuracy is kept at about 20%, and the model still keeps the result of random classification. Compared with the accuracy and error of the model in the test set, the tanh function converges faster than ReLU. And we can get the highest accuracy for the tanh function in Table 14. Therefore, the tanh is selected as the activation function.

TABLE 14. Quantitative results of the model under different activation functions.

Activation function	ReLU	Sigmoid	Tanh
Accuracy	0.8212	0.2212	0.9231
Error	1.1790	1.3882	0.1904

F. OPTIMIZATION METHOD

The classification accuracy of the model is an important index to evaluate the model effect. The higher the accuracy is, the smaller the error between the predicted and the actual category is. The model optimization is to minimize the error and achieve the highest classification accuracy. The error between the predicted and actual value is reflected by the loss function. Therefore, solving the minimum value of the loss function is a key step in the model optimization. The gradient direction of the loss function is the fastest increasing direction of the function, so the minimum value of the function can be obtained along the gradient decreasing direction. The optimization methods based on the gradient descent scheme include Mini-batch gradient descent (Mini-BGD), gradient descent with momentum (Momentum) and Nesterov accelerated gradient (NAG).

1) MiNi-BGD

The essence of Mini-batch sample training method is to use the stochastic gradient descent (SGD) method to update parameters in each small batch, and calculate the gradient of the loss function in each batch respectively. The parameter updating method is as follows:

$$\begin{cases} w_{new} = w_{old} - \eta \dfrac{\partial L(w_{i:i+q})}{\partial w_{i:i+q}} \\ b_{new} = b_{old} - \eta \dfrac{\partial L(b_{i:i+q})}{\partial b_{i:i+q}} \end{cases} \tag{13}$$

where q is the number of samples per batch. At each time, only a small part of data in the training set is used to calculate the gradient and update the parameters. Therefore, the direction of gradient descent is not always strictly in the direction of the minimum value.

2) MOMENTUM

The Momentum method is an improvement of Mini-BGD, which uses the exponential weighted average to obtain the mean value of different parameter gradients in the current

Mini-batch, and then updates the parameters with the mean value of gradients.

If $\frac{\partial L(w_{i:i+q})}{\partial w_{i:i+q}} = \nabla W$, $\frac{\partial L(b_{i:i+q})}{\partial b_{i:i+q}} = \nabla b$, the parameter is updated as follows:

$$\begin{cases} w_{new} = w_{old} - \eta V_{\nabla W_r} \\ b_{new} = b_{old} - \eta V_{\nabla b_r} \end{cases} \tag{14}$$

where $V_{\nabla W_r}$ and $V_{\nabla b_r}$ represent the gradient mean value of the sum of w and b, and the calculation method is:

$$\begin{cases} V_{\nabla W_r} = \alpha V_{\nabla W_{r-1}} + (1-\alpha)\nabla W_r \\ V_{\nabla b_r} = \alpha V_{\nabla b_{r-1}} + (1-\alpha)\nabla b_r \end{cases} \tag{15}$$

where $V_{\nabla W_{r-1}}$ and $V_{\nabla b_{r-1}}$ represent the mean value of the gradients of w and b in $r-1$ batch, ∇W_r and ∇b_r represent the gradients of w and b in t batch, and α is the hyperparameter controlling the weighted average of the index.

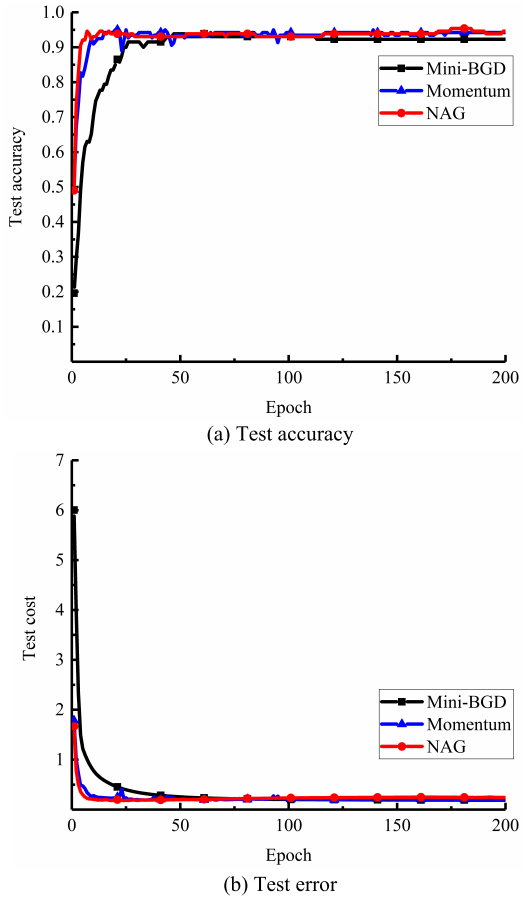


FIGURE 21. Performance curves of the model under different optimization methods.

3) NAG

NAG is an improvement of Momentum method. The gradient of the loss function in the next batch data is calculated in advance, and then it is combined with the gradient descent direction of the loss function in the previous batch, so as to modify the gradient descent direction and obtain the true gradient direction of loss function in the current batch, so as to make the gradient descent faster.

Assuming that g_t represents the cumulative value of the gradient of the t-th iteration, there is:

$$g_t = \gamma g_{t-1} + \eta \nabla L(w - \gamma g_{t-1}) \tag{16}$$

In Equation (16), g_{t-1} indicates the gradient accumulation value in the $t-1$ batch, γ is the gradient attenuation rate, $w - \gamma g_{t-1}$ means to update the weight parameter according to the gradient in the first batch to obtain the weight parameter in the next batch, $\nabla L(w - \gamma g_{t-1})$ represents the gradient of the loss function with respect to the weight in the next batch, g_t represents the actual gradient of the loss function with respect to the weight in the t batch. Therefore, the updating mode of

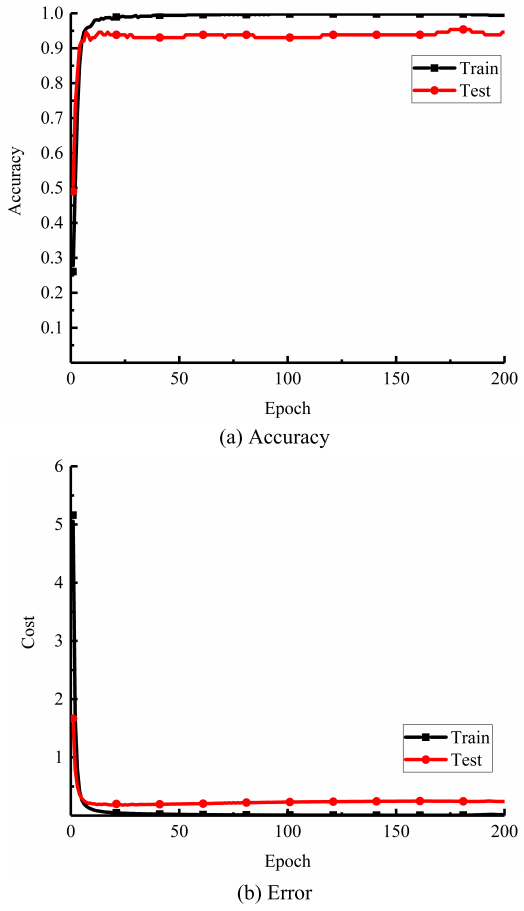


FIGURE 22. Training and test performance curves of the model.

TABLE 15. Quantitative results of the model under different optimization methods.

Optimization method	Mini-BGD	Momentum	NAG
Accuracy	0.9231	0.9423	0.9462
Error	0.1904	0.2102	0.2426

parameters under the NAG method is:

$$\begin{cases} w_{new} = w_{old} - g_t(w) \\ b_{new} = b_{old} - g_t(b) \end{cases} \tag{17}$$

Three different optimization methods were adopted, and the performance of the model is shown in Figure 21 and Table 15.

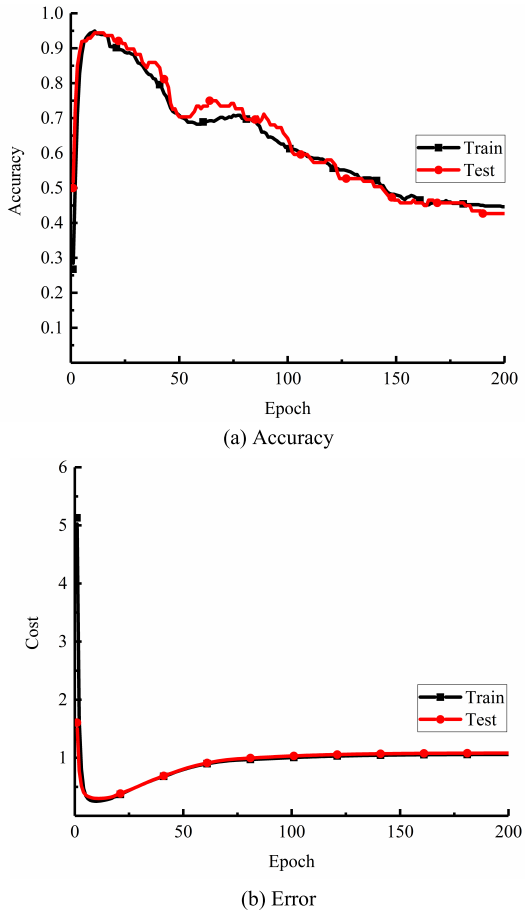


FIGURE 23. Performance curves of the model after adding regularization.

It can be seen from Table 15 that the above three optimization methods have little influence on the accuracy of the model, the difference is that the gradient descent speed is different. The convergence rates of Momentum and NAG are faster than that of Mini-BGD in Figure 21. Compared with Momentum, NAG is faster and the fluctuation is less, i.e., the model is more stable. Therefore, NAG is used as the optimization method to update the parameters.

G. OVERFITTING

Overfitting is a common phenomenon in the process of model training. The reason is that the model can achieve good results in the training set, but the performance in the test set is very poor. In this case, some noise features in the training set are over fitted, resulting in the insufficient generalization ability. There are two indexes to judge overfitting: deviation and variance. Deviation refers to the error rate of training set, and variance refers to the difference between the error rate of test set and training set. Figure 22 is the training and test result of the model. And Table 16 shows the quantitative classification result of the model.

TABLE 16. Quantitative results of the model.

Accuracy		Deviation	Variance
Train set	Test set		
0.9942	0.9462	0.0057	0.0481

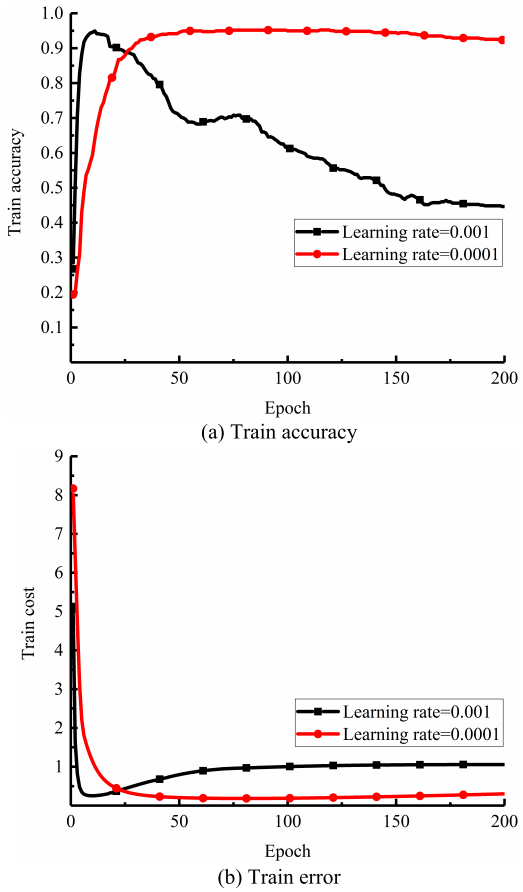


FIGURE 24. Performance curves of the model under different learning rates.

It can be seen from the Table 16 that the deviation of the model is 0.57% and the variance is 4.81%. The performance of the model in the test set is not as good as that in the training set, and there is overfitting. By adding L2 regularizer [32] to regularize the objective function, the overfitting can be prevented.

Regularization is to control the parameters by adding the restriction rules to the objective function, so as to avoid too many parameters in the training process. The objective function of adding the regular term can be expressed as follows:

$$L^* = L_{loss} + \Phi(w) \tag{18}$$

where L^* represents the objective function, L_{loss} represents the loss function of cross entropy, and $\Phi(w)$ represents the regular term.

As for L2 regularizer, the formula is:

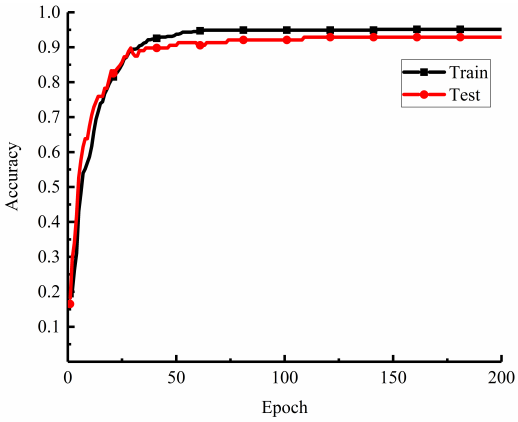
$$\Phi_{l_2}(w) = \frac{\lambda}{2} \|w\|_2^2 \tag{19}$$

where, λ represents the regular coefficient.

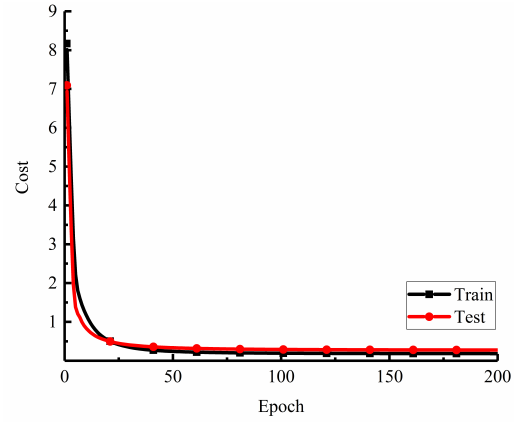
With the L2 regularizer, the performance of the model is shown in Figure 23.

It can be seen from Figure 23 that variance decreased from 4.81% to 1.97%. The L2 regularizer can effectively alleviate the overfitting. But the accuracy firstly increases and then decreases, and the error firstly decreases and then rebounds. The difference between the maximum value of accuracy and the value at the end of training is 50.29%. According to the results, the learning rate is too large, and when the gradient drops it crosses the minimum value of the objective function resulting in the error rebounding.

When the learning rate decreases from 0.001 to 0.0001, the performance of the model improves significantly. Figure 24 shows the performance of the model on the training set when the learning rate is set as 0.001 and 0.0001.

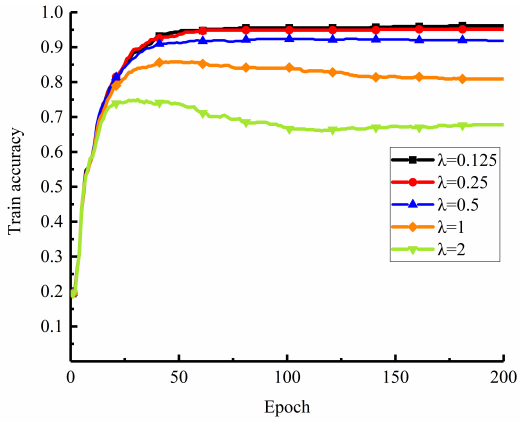


(a) Accuracy

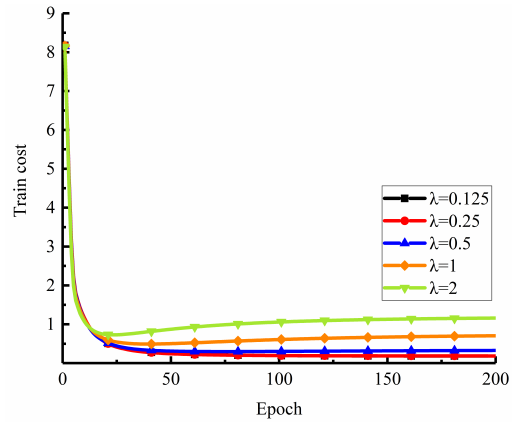


(b) Error

FIGURE 25. Performance curves of the model under exponential decay learning rate.



(a) Train accuracy



(b) Train error

FIGURE 26. Performance curves of the model under different regular term coefficients.

TABLE 17. Quantitative results of the improved model.

Accuracy		Deviation	Variance
Train set	Test set		
0.9514	0.9289	0.0486	0.0226

TABLE 18. Quantitative results of the model under different regular term coefficients.

λ	Accuracy		Deviation	Variance
	Train	Test		
0.125	0.9615	0.9135	0.0385	0.0481
0.25	0.9514	0.9289	0.0486	0.0226
0.5	0.9183	0.9212	0.0817	-0.0029
1	0.8091	0.8615	0.1909	-0.0524
2	0.6779	0.7115	0.3221	-0.0337

It can be seen from Figure 24 that when the learning rate decreases, although the performance of the model has picked up, the overall change trend has not changed, and the accuracy rate still shows the trend of increasing firstly and

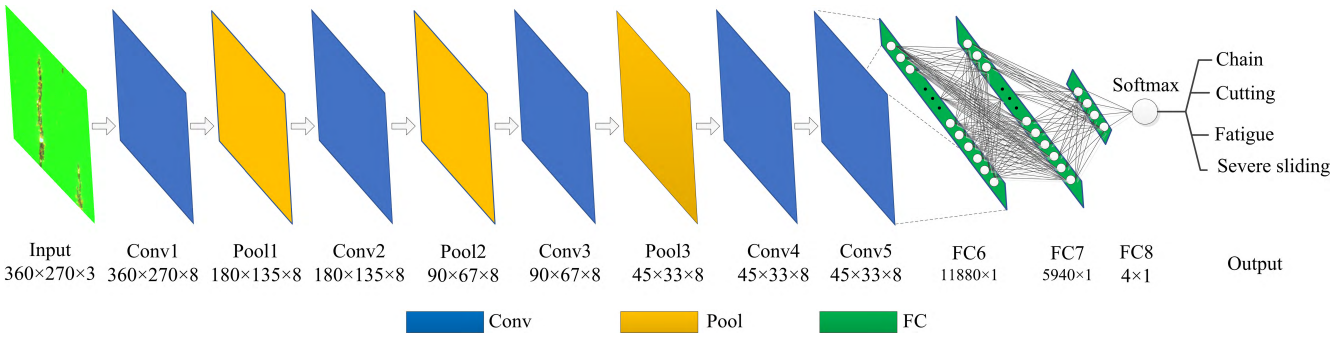


FIGURE 27. Final structure of the optimal model.

TABLE 19. Final parameters of the optimal model.

Parameter	Value	
	Pretraining model	Classification model
Kernel(length×width×number×stride)	3×3×8×1	
Pooling(mode, length×width×stride)	Max pooling, 2×2×2	
Weight initialization(initialization mode, mean, variance)	Truncated normal distribution, mean=0, variance=0.01	
Bias	0	
Activation function	Tanh	
Loss function	Cross entropy	
Classification function	Softmax	
Optimization method	NAG	
Learning rate	0.001	Exponential decay mode (initial learning rate =0.0001)
Batch size	Train set=30, validation set=10, test set=10	Train set=40, test set=10
Regularization	None	L2 regularization, λ =0.25

then decreasing. The difference between the maximum value of accuracy and the value at the end of training is still 2.88%. Therefore, a decay method of learning rate is considered to make the model converge quickly by using a large initial learning rate. With the gradual decline of learning rate in the model training, the gradient will not cross the minimum value in the process of decline, and keep the trend of continuous decline.

H. LEARNING RATE

Learning rate plays an important role in finding the minimum value of loss function. When the learning rate is too large, the step of gradient descent is too large, and the loss function may cross the minimum value point, resulting in the situation that the error firstly drops then rises, and finally oscillates in a certain range; when the learning rate is too small, the parameter updating speed is too slow. Therefore, in this paper the learning rate is set in the way of exponential decay. The learning rate decays at a certain rate during the training process, so that the model converges rapidly without crossing the minimum value. The training speed and accuracy of the model are taken into account. The expression is as follows:

$$\eta_{decay} = \eta * \alpha^{\frac{s}{s_{decay}}} \tag{20}$$

where η_{decay} represents the learning rate after attenuation, η represents the initial learning rate, α represents the decay rate of learning rate, s represents all iterations, and s_{decay} represents the decay stride of learning rate.

It can be seen from Figure 24 that when the learning rate is 0.0001, the performance of the model is better than 0.001, so 0.0001 is taken as the initial learning rate.

The model is trained in the form of exponential decay, and the results are shown in Figure 25.

It can be seen from Figure 25 that after the learning rate is set in the exponential decay mode, the performance of the model is stable.

I. COEFFICIENT OF REGULARIZER

After the learning rate is set in the exponential decay mode, the model classification results are shown in Table 17. Under the L2 regularization, the variance of the model is reduced from 4.81% to 2.26%. The test results are closer to the training results, which effectively suppresses overfitting.

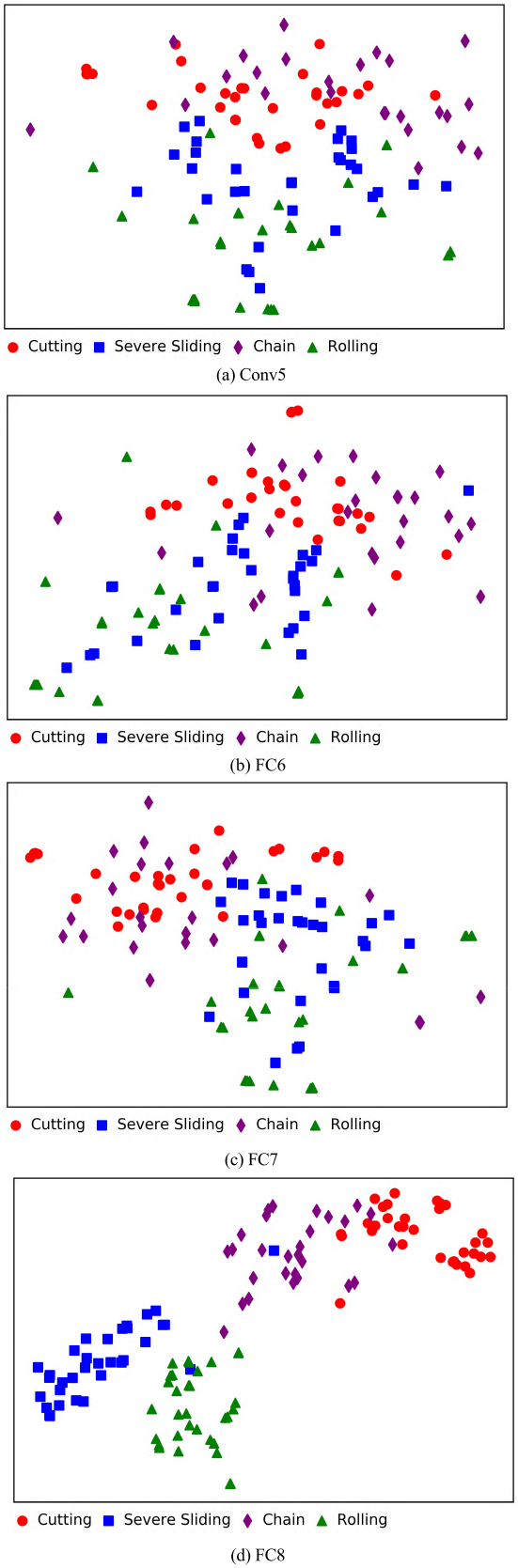


FIGURE 28. Visualization process of fully-connected layer.

In Equation (21), as a coefficient of regular term, λ indicates the degree of parameter restriction. In this paper, five regularization coefficients are selected, and the results are shown in Figure 26 and Table 18.

It can be seen from Figure 26 and Table 18 that as the coefficient of the regular term increases gradually, the restriction on the parameters increases gradually, which makes the model deviation show an upward trend and the model tends to be underfitting. Considering the deviation and variance, choose $\lambda = 0.25$, at this time the results of test set and training set are the closest, and the deviation is in an acceptable range.

V. RESULTS AND DISCUSSION

A. OPTIMAL MODEL

Based on the above model research, an optimal CNN model for the intelligent recognition of small sample ferrographic images is established. Figure 27 is the structural diagram of the optimal model. The network structure of the pretraining model is completely consistent with the classification model. The parameter settings are shown in Table 19.

The parameters in Table 19 achieved a better effect than others in the above research process. Taking the virtual images as the input of the pretraining model, the pretraining model is built by using the parameters in Table 19. The target image features were extracted by the convolution layer parameters obtained by the pretraining model. The feature vector of the images was used as the input of the classification model to train the fully-connected layers to get the optimal classification model. The performance of the classification model was tested by test set. A total of 160 samples were tested for four wear particle types, of which 150 samples were correctly classified and 10 samples were incorrectly classified, the classification accuracy is 93.75%.

B. VISUALIZATION

The t-SNE algorithm was used to verify the classification performance of the model on the test set, which can map the data of a high-dimensional space to a low-dimensional space while preserving the local characteristics of the data set. By observing the separability of data in a low-dimensional space, it can verify whether the data are separable in a high-dimensional space. The mapping of the high-dimensional data for the fully-connected layer in the 2D space is shown in Figure 28.

Figure 28 (a), (b), (c) and (d) respectively show the mapping of high-dimensional output data by Conv5 and FC6, FC7 and FC8 in 2D space by t-SNE algorithm. It can be seen from Figure 28 that after the convolution and pooling of the original images, the final output featured data via Conv5 is disordered. After two layers of fully-connected layer processing, the featured data began to cluster according to a certain rule. After the third fully-connected layer, the categories of featured data are very clear. The results show that the reduced

data can be divided in a low dimension space, and the high dimension data can also be divided. The classification result of Figure 28 (d) is consistent with the classification accuracy of 93.75%, which proves that the intelligent classification model based on the CNN and TL of ferrographic images established proposed in this paper is enough effective.

VI. CONCLUSION

Wear condition recognition is an important topic in the fault diagnosis of mechanical equipment, and the analytical ferrograph is widely used for the wear particle recognition. In order to deal with the low efficiency and accuracy problem of ferrographic image recognition, a new intelligent approach combining CNN with TL introducing the virtual images is proposed, which is very suitable for the intelligent recognition of small sample ferrographic images.

(1) In order to meet the needs of the classification model, four types of wear particle sample were obtained by an analytical ferrograph. Based on the sample similarity, the virtual ferrographic images corresponding to the different types of wear particle were designed. In addition, the number of ferrographic image samples needed by the training model was obtained by the data enhancement, and the data set was reasonably divided. Introducing the virtual image samples is one of the innovation points in this paper.

(2) The training and classification models were built independently. Based on the AlexNet frame, the influence of each parameter on the classification effect of the CNN model was studied, and the optimal parameter combination was obtained. Using an exponential decay to set the learning rate, the problem that the model is difficult to converge was effectively restrained. By adding a regular term, the overfitting was restrained and the generalization ability of the model was improved.

(3) The intelligent recognition of wear particle ferrographic images under small sample data was realized by using the TL. The visualization of fully-connected layer was realized by using the t-SNE, and the classification accuracy is 93.75%, which proves the deep learning model for the mechanical wear condition recognition proposed in this paper to be effective.

REFERENCES

[1] Y. Peng, T. Wu, S. Wang, and Z. Peng, "Wear state identification using dynamic features of wear debris for on-line purpose," *Wear*, vols. 376–377, pp. 1885–1891, Apr. 2017.

[2] B. Fan, B. Li, S. Feng, J. Mao, and Y.-B. Xie, "Modeling and experimental investigations on the relationship between wear debris concentration and wear rate in lubrication systems," *Tribol. Int.*, vol. 109, pp. 114–123, May 2017.

[3] W. Cao, G. Dong, Y.-B. Xie, and Z. Peng, "Prediction of wear trend of engines via on-line wear debris monitoring," *Tribol. Int.*, vol. 120, pp. 510–519, Apr. 2018.

[4] S. Feng, B. Fan, J. Mao, and Y. Xie, "Prediction on wear of a spur gearbox by on-line wear debris concentration monitoring," *Wear*, vols. 336–337, pp. 1–8, Aug. 2015.

[5] X. P. Yan, Y. B. Xie, and H. L. Xiao, "Development of oil monitoring techniques and its tendency," *China Mech. Eng.*, vol. 8, no. 1, pp. 102–105, Feb. 1997.

[6] Y. Peng, T. Wu, S. Wang, and Z. Peng, "Oxidation wear monitoring based on the color extraction of on-line wear debris," *Wear*, vols. 332–333, pp. 1151–1157, May 2015.

[7] J. Wang, G. Wang, and L. Cheng, "Texture extraction of wear particles based on improved random Hough transform and visual saliency," *Eng. Failure Anal.*, vol. 109, Jan. 2020, Art. no. 104299.

[8] A. Kumar and S. K. Ghosh, "Size distribution analysis of wear debris generated in HEMM engine oil for reliability assessment: A statistical approach," *Measurement*, vol. 131, pp. 412–418, Jan. 2019.

[9] U. Cho and J. A. Tichy, "Quantitative correlation of wear debris morphology: Grouping and classification," *Tribol. Int.*, vol. 33, no. 7, pp. 461–467, Jul. 2000.

[10] J. Wang, L. Zhang, F. Lu, and X. Wang, "The segmentation of wear particles in ferrograph images based on an improved ant colony algorithm," *Wear*, vol. 311, nos. 1–2, pp. 123–129, Mar. 2014.

[11] T. Wu, H. Wu, Y. Du, N. Kwok, and Z. Peng, "Imaged wear debris separation for on-line monitoring using gray level and integrated morphological features," *Wear*, vol. 316, nos. 1–2, pp. 19–29, Aug. 2014.

[12] J. Wang and X. Wang, "A wear particle identification method by combining principal component analysis and grey relational analysis," *Wear*, vol. 304, nos. 1–2, pp. 96–102, Jul. 2013.

[13] G. W. Stachowiak and P. Podsiadlo, "Towards the development of an automated wear particle classification system," *Tribol. Int.*, vol. 39, no. 12, pp. 1615–1623, Dec. 2006.

[14] S. Raadnui, "Wear particle analysis—Utilization of quantitative computer image analysis: A review," *Tribol. Int.*, vol. 38, no. 10, pp. 871–878, Oct. 2005.

[15] T. B. Kirk, D. Panzera, R. V. Anamalay, and Z. L. Xu, "Computer image analysis of wear debris for machine condition monitoring and fault diagnosis," *Wear*, vols. 181–183, pp. 717–722, Mar. 1995.

[16] J. D. Lin, X. Y. Wu, Y. Chai, and H. P. Yin, "Structure optimization of convolutional neural networks: A survey," *Acta Automatica Sinica*, vol. 46, no. 1, pp. 24–37, Jan. 2020.

[17] H. Wu, N. M. Kwok, S. Liu, R. Li, T. Wu, and Z. Peng, "Restoration of defocused ferrograph images using a large kernel convolutional neural network," *Wear*, vols. 426–427, pp. 1740–1747, Apr. 2019.

[18] I. Szátmari, A. Schultz, C. Rekeczky, T. Kozek, T. Roska, and L. O. Chua, "Morphology and autowave metric on CNN applied to bubble-debris classification," *IEEE Trans. Neural Netw.*, vol. 11, no. 6, pp. 1385–1393, Nov. 2000.

[19] L. J. Wang, "Ferrography image classification based on deep learning," M.S. thesis, College Mech. Electronical Eng., NUAA, Nanjing, Jiangsu, 2018.

[20] M. Wu, "Wear particle analysis based on convolutional neural network and image saliency," M.S. thesis, College Mech. Electronical Eng., NUAA, Nanjing, Jiangsu, 2019.

[21] P. Peng and J. Wang, "Wear particle classification considering particle overlapping," *Wear*, vols. 422–423, pp. 119–127, Mar. 2019.

[22] P. Peng and J. Wang, "FECNN: A promising model for wear particle recognition," *Wear*, vols. 432–433, Aug. 2019, Art. no. 202968.

[23] S. Wang, T. H. Wu, T. Shao, and Z. X. Peng, "Integrated model of BP neural network and CNN algorithm for automatic wear debris classification," *Wear*, vols. 426–427, pp. 1761–1770, Apr. 2019.

[24] C. An, H. J. Wei, H. Liu, S. Wu, and Y. P. Zhu, "Intelligent identification of ferrographic wear oarticles based on convolution neural network," *Mod. Manuf. Eng.*, no. 7, pp. 111–114, 2019.

[25] Y. Peng, J. Cai, T. Wu, G. Cao, N. Kwok, S. Zhou, and Z. Peng, "A hybrid convolutional neural network for intelligent wear particle classification," *Tribol. Int.*, vol. 138, pp. 166–173, Oct. 2019.

[26] Y. Peng, J. Cai, T. Wu, G. Cao, N. Kwok, S. Zhou, and Z. Peng, "Online wear characterisation of rolling element bearing using wear particle morphological features," *Wear*, vols. 430–431, pp. 369–375, Jul. 2019.

[27] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.

[28] F. B. Zhou, L. H. Zou, X. J. Liu, and F. Y. Meng, "Micro landform classification method of grid DEM based on convolutional neural network," *Geomatics Inf. Sci. Wuhan Univ.*, to be published, doi: 10.13203/j.whugis20190311.

[29] J. C. Huang, Q. Shu, X. C. Zhu, L. Zhou, H. Z. Liu, and J. Lin, "Robot vision recognition and sorting strategy based on transfer learning," *Comput. Eng. Appl.*, vol. 55, no. 8, pp. 232–237, 2019.

[30] F. Z. Zhuang, P. Luo, Q. He, and Z. Shi, "Survey on transfer learning research," *J. Softw.*, vol. 26, no. 1, pp. 26–39, 2015.

[31] Q. M. Yang, *Wear Particle Analysis-Wear Particle Atlas and Ferrography*, 1st ed. Beijing, China: China Railway Publishing House, 2002, pp. 135–137.
[32] Y.-F. Jin, Z.-Y. Yin, W.-H. Zhou, J.-H. Yin, and J.-F. Shao, "A single-objective EPR based model for creep index of soft clays considering L2 regularization," *Eng. Geol.*, vol. 248, pp. 242–255, Jan. 2019.

HONGWEI FAN received the B.Sc. degree from Beihua University, in 2007, the M.Sc. degree from Northwestern Polytechnical University, in 2010, and the Ph.D. degree from Xi'an Jiaotong University, in 2015. He is currently a Lecturer with the School of Mechanical Engineering, Xi'an University of Science and Technology. His main research interests include intelligent monitoring, and diagnosis and control of equipment.

SHUOQI GAO was born in 1993. He received the B.Sc. degree from the Xi'an University of Science and Technology, in 2016, where he is currently pursuing the master's degree with the School of Mechanical Engineering. His main research interest includes ferrographic image analysis of wear particles.

XUHUI ZHANG (Member, IEEE) received the B.Sc. and M.Sc. degrees from the Xi'an University of Science and Technology, in 1996 and 2002, respectively, and the Ph.D. degree from Xi'an Jiaotong University, in 2009. He is currently a Professor with the School of Mechanical Engineering, Xi'an University of Science and Technology. His main research interest includes intelligent detection and control of equipment.

XIANGANG CAO received the B.Sc. and M.Sc. degrees from the Xi'an University of Science and Technology, in 1994 and 1997, respectively, and the Ph.D. degree from Xi'an Jiaotong University, in 2008. He is currently a Professor with the School of Mechanical Engineering, Xi'an University of Science and Technology. His main research interest includes condition monitoring of equipment.

HONGWEI MA received the B.Sc. and M.Sc. degrees from the Xi'an University of Science and Technology, in 1984 and 1993, respectively, and the Ph.D. degree from Xi'an Jiaotong University, in 1998. He is currently a Professor with the School of Mechanical Engineering, Xi'an University of Science and Technology. His main research interest includes intelligent detection and control of equipment.

QI LIU was born in 1994. He received the B.S. degree from Xi'an University, in 2017. He is currently pursuing the master's degree with the School of Mechanical Engineering, Xi'an University of Science and Technology. His main research interest includes ferrograph design and experiment.