Prediction of Corrosion Resistance of B10 Copper-Nickel Alloy Based on Optimal Convolutional Neural Network

Yuehua Dong¹, Donghai Hu¹, Zhigang Wang²
¹School of Information Engineering, Jiangxi University of Science and Technology, Ganzhou, Jiangxi, 341000, China
²School of Materials Science and Engineering, Jiangxi University of Science and Technology, Ganzhou, Jiangxi, 341000, China
*Corresponding author’s & first author’s e-mail: 9119930002@jxust.edu.cn

Abstract. Aiming at the defects of insufficient number of corrosion resistance prediction models and simple feature extraction of B10 copper-nickel alloy, a corrosion resistance prediction model of B10 copper-nickel alloy based on optimized convolutional neural network is proposed. The convolutional neural network architecture for grain boundary image characteristics is proposed to conclude a step-by-step convolution operation by analysing the traditional convolution operation process, and it proves theoretically that such a step-by-step convolution operation can reduce the consumption of additional parameters; learning pool proposed single-channel operation can reduce the loss of feature information; the multi-layer feature fusion learning strategy makes the expressive force of deep network extraction features diversified. The results of multiple experiments demonstrate that the improved algorithm introduced can improve the prediction accuracy of the model in image classification comparing with the traditional convolutional neural network model, and the improved convolutional neural network model can better achieve the prediction of corrosion resistance of B10 copper-nickel alloy.

1. Introduction
The current deep learning technology represented by Convolutional Neural Network (CNN) has made great breakthroughs in many ways, especially in the field of computer vision, such as image classification [1], target detection [2], image segmentation [3], etc., and its application has extended into many livelihood industries such as construction [4], forestry [5], medical industry [6], etc.

With the introduction of various network models, increasing optimizing strategies have been discovered as well. Cheng et al. [7] put forward a short link to directly map the bottom feature map to the high-level ResNet structure, which enables some data to skip some training levels that have not been perfected in the forward propagation, thus improving the training effect of the network. Girshick et al. [8] proposed that the R-CNN can convert the conventional features into the features extracted by deep convolutional network when extracting features. They use SVM for classification and outputs the results after non-maximum suppression (NMS). The operation effectively improves the accuracy of the model. But it takes too much time to calculate. Fast R-CNN [9] without SVM classifier is thus put forward with neural network adopted to classify, enabling the classification network and the feature extraction network to be trained at the same time, which not only improves the accuracy of the model but also reduces the time consumption. However, fast R-CNN has its drawbacks in its needs of using the Selective Search extraction box first, which often takes up most of the time consumption.
Therefore, RPN (Region Proposal Network) is applied to replace Selective Search in the improved Fast R-CNN \cite{10}, which improves the detection speed and optimizes the detection result.

Judging from the current trend of research, there are hundreds of thousand possibilities in the application prospect of convolutional neural network, but still very limited research on applying convolutional neural network to grain boundary images of B10 copper-nickel alloy, because: 1) Grain boundary images with performance labels are difficult to obtain. Although grain boundary images can be obtained by electron backscatter diffraction technology and orientation imaging micrograph, the corrosion resistance label of each grain boundary image can only be obtained by the corresponding sample passing physical corrosion test and resistance test, so to accumulate and precipitate to obtain sufficient training data sets needs a long time. 2) The grain boundary images are the boundary lines (grain boundaries) that intersect with each other. The connectivity between grain boundaries and the intersection angle (intergranular angle) exert an important influence on the corrosion resistance of metals, which not only limits the input of the model, but also further leads to the scarcity of training sets. On the other hand, since the intergranular failure in the microstructure of alloy materials will lead to the reduction of its service life, reliability and use value, and the distribution of grain boundaries plays a key role in the intergranular failure of metals and the penetration of corrosion paths, accurate control of the microstructure of materials can be used to improve its properties. It is urgent to establish a direct relationship between grain boundary structure and corrosion resistance so as to predict and control the corrosion resistance of B10 copper-nickel alloy in a more accurate way and understand the relationship between grain boundary structure and intergranular corrosion resistance.

The research in this paper is aimed at the relationship between the microscopic grain boundary structure of B10 copper-nickel alloy and the corrosion resistance of the alloy. The grain boundary features of the alloy are no longer manually selected, but autonomously learned by neural network. In order to better predict the corrosion resistance of B10 copper-nickel alloy, this paper first proposes a network architecture suitable for grain boundary image processing. After preprocessing grain boundary images by using the sensitivity of grain boundary connectivity model \cite{11}, a new step-by-step convolutional algorithm is provided to reduce the consumption of additional parameters. Then, the balance between the parameter quantity and the training effect is ensured through the maximum pooling in the shallow layer and the learning single channel pooling operation strategy in the deep layer. Finally, the fusion learning of multi-layer features make the features participating in the training more distinguishing, thus obtaining more accurate classification and prediction results.

2. Relevant Basic Theories

2.1. Pre-training Process of Convolutional Neural Network

Based on traditional neural network, CNN, a type of feedforward neural network, is improved on the function and form of its structure layer. The high-level semantic features of an image can finally be obtained by extracting image features. The deeper the network layer is, the more abstract the extracted feature representation is and the more obvious the main discrimination features are, better representing the image theme. Thus the identification ability in the image classification and prediction task becomes stronger. The basic structure of CNN consists of five layers, namely input, convolution, pooling, full connection, and output. Convolution layers and pooling layers are usually constructed in multiple and intersecting structures. Each neuron that outputs the characteristic surface in the convolution layer is locally connected with its input, and the corresponding input and connection weights are weighted and summed and then added with the bias to obtain the input value of the neuron. CNN is named for its similarity to convolution. The network is divided into two processes in training: forward propagation and back propagation.

1) Forward propagation process. In this stage, the output value is calculated as follows:

\[ y^{(i)} = f \left( b^i + \sum_{j \in m} W^i_j * x_j^{(i-1)} \right) \] (1)
In the formula, $y^{(i)}$ is the output of the i-th convolution layer, $f(x)$ represents the selected activation function, $b^i$ is the offset, $W^i_j$ is the convolution kernel weight corresponding to the j-th input channel of the i-layer, $*$ is convolution operation, $x^{(i-1)}$ is the input vector, and $m$ is the input feature channel set. Sigmoid, ReLU, etc. are commonly used for the selection of activation functions.

2) Back propagation process. The final result of the forward propagation process is to output the label prediction result for each sample, and according to the result and the set prediction target value, the objective function of the network is defined as formula 2:

$$E(W) = \min \sum_{i=1}^{N} L(z_i) + \lambda \|W\|^2$$

In the formula, $N$ represents the number of samples, $L(x)$ is the selected loss function, $z_i$ is the output of the last layer in the forward propagation process, and $\lambda$ represents the proportion of the network weight $W$ in this iteration. $L(x)$ needs to be determined according to different application environments, commonly including exponential loss function, Hinge loss function, cross entropy loss function, etc. This paper selects the cross entropy loss function based on Softmax classifier. Softmax converts the last output $z$ of the network into probability form by exponential, and its calculation formula is as follows:

$$p_i(z) = \frac{e^{z_i}}{\sum_{j=1}^{m} e^{z_j}}, i = 1,2,\ldots,m$$

Where $e^{z_i}$ is the network output index of category $i$, $\sum_{j=1}^{m} e^{z_i}$ is the sum of the indexes of network output of all categories. At the same time, the probability of the category $z_i$ is predicted according to $p_i(z)$. On this basis, the loss function is defined as:

$$L(z_i) = -\log p_i(z)$$

The gradient descent algorithm is used to derive the convolution kernel parameters and bias of each layer to obtain its updated value until the loss function reaches the minimum.

2.2. Introduction to Pooling Layer and Feature Fusion

The pooling layer is followed by the convolution layer, and each neuron of the pooling layer performs pooling operations on the local acceptance domain. It aims to make the changes in scale and position invariant and to aggregate responses within and across feature maps.

Pooling layer can be improved in three directions: manual pooling, random pooling and learning pooling. Common pooling, such as maximum pooling and mean pooling, belong to manual pooling, while random pooling is to normalize the value of the feature map in a pooling window, and randomly sample and select the normalized value as its probability value, and the larger the element value, the greater the probability of being selected. Learning-based pooling aims to adapt the pooling to the data set to ensure that training errors are minimized in the training phase. In this paper, the improvement direction to the pooling layer is learning-based pooling.

Feature fusion is a new technology. Considering that one of the obstacles of machine learning algorithms is that it has to rely on processing data to work, and the algorithm can only make predictions based on the data which are composed of relevant variables (i.e. Features), in case the calculated features cannot clearly display the predicted signals, then bias will not take the model to the next layer. The advantages of feature fusion are: 1) the correlation between data points can infer important features; 2) data can be synthesized between different data sets; 3) recognizing the relationship between different individuals can help to obtain new features. Feature fusion can enrich the diversity of features, thus promoting in-depth-wise learning.

2.3. Brief Introduction of Prediction Principle of Corrosion Resistance

The grain boundary images and Nyquist impedance test results of four representative B10 copper-nickel alloys with different corrosion resistance is shown in Fig. 1, where (a) to (d) are alloy grain boundary distribution images with different nickel contents. And Fig.2 are corrosion resistance quantitative curves obtained from the above physical tests. In Table 1, S1, S2, S3 and S4 are four groups of copper-nickel alloy with different $Y$ contents. As shown in the curves, representative
samples with corrosion resistance of S3, S4, S2, and S1 from high to low can be seen corresponding to four labels of strong, rather strong, general, and weak respectively. On the basis of the existing research [11], the corrosion resistance of materials highly correlates with the grain boundary structure of materials. Thereby, a training model can be established to independently learn grain boundary characteristics and achieve the highest possible classification accuracy with a large number of grain boundary images and corresponding labels. For a given grain boundary image with unknown properties, after the training of the model is completed, the output of the model will give probability values belonging to four performance strengths, and the corrosion resistance strength will be determined in accordance with the category of the highest probability values. For instance, if the four output probability values of the current image model are 0.1, 0.8, 0.1, 0 and respectively, it is determined to be in the S4 sample classification with strong corrosion resistance.

![Figure 1. Grain boundary distribution characteristics of B10 alloy with different reduction ratios](image)

**Table 1. Alloy composition table with different Y contents**

| Sample | Ni  | Fe  | Mn  | S   | C   | Y      |
|--------|-----|-----|-----|-----|-----|--------|
| S1     | 11.48 | 1.76 | 0.99 | <0.001 | 0.0037 |
| S2     | 11.43 | 1.63 | 0.96 | 0.001 | 0.0084 | 0.0039 |
| S3     | 11.44 | 1.61 | 0.96 | <0.001 | 0.0084 | 0.0210 |
| S4     | 11.58 | 1.67 | 0.96 | 0.0012 | 0.0084 | 0.0530 |
2.4. Image Preprocessing
In the analysis of grain boundary images, both image resolution and grain boundary selection will influence the design of analysis algorithm as well as the calculation accuracy. Therefore, grain boundary images need to be preprocessed before analysing the information contained in the images. Pre-treatment operation is aimed to remove non-critical characteristics or unnecessary grain boundaries such as small angle grain boundaries and special grain boundaries, while retaining random grain boundaries with poor intergranular corrosion resistance. For the five grain boundary images in Fig. 1 (taking the S2 as examples), the four grain boundary images are first converted into gray scale images, and the gray scale histogram is calculated and the binarization segmentation threshold is selected accordingly. In this paper, the binary segmentation threshold is set to 50, which is more beneficial to the study of black random boundaries.

3. Optimized Convolutional Neural Network
In this paper, the prediction of corrosion resistance of five different types of B10 copper-nickel alloy is mainly divided into three parts: image pre-processing, image feature extraction and processing, and classification of full connection layers. Compared with traditional image classification methods, Convolutional neural network has the ability of feature extraction and autonomous learning. In addition, the number of neurons required by the full connection layer is reduced by sharing weights, the network structure is simplified, and the computation amount is reduced. CNN can apply the learned features to other prediction tasks through migration learning, thus enhancing universality and prediction accuracy. Therefore, this paper uses convolutional neural network to extract the grain boundary distribution characteristics of different types of B10 copper-nickel alloy, and improves the performance of the network model by three methods: step convolution, pooling strategy selection with different network depths, and multi-layer feature fusion.

3.1. Network Architecture
Figure 3 shows the network layer structure composed of 6 convolutional layers and 3 fully connected layers. The activation function of each convolutional layer is ReLU. The specific structural parameters are as follows:

- **Input:** 799 × 611 × 1 Binary grain boundary image
- Convolution layer 1: 11 × 11 × 128 convolution kernels, step=4, no padding (p = 0); maximum pooling: 3 × 3 kernel, step=3; number of parameters: 61kByte.
Convolution layer 2: $5 \times 5 \times 256$ convolution kernels, step=3 and 4 sides each filled with a layer of 0 ($p = 1$); maximum pooling: $3 \times 3$ convolution kernel, step=1; number of parameters: 3.1Mbyte.

Convolution layer 3: $3 \times 3 \times 512$ convolution kernels, step=1 and no zero padding; no maximum pooling layer; number of parameters: 4.5MByte.

Convolution layer 4: $3 \times 3 \times 512$ convolution kernels, step=1, no zero padding; no maximum pooling layer; number of parameters: 9.0MByte.

Convolution layer 5: $3 \times 3 \times 256$ convolution kernels, step=1 and no zero padding; no maximum pooling layer; number of parameters: 10kByte.

Convolutional layer 6: $1 \times 1 \times 256$ convolution kernels, step=1 and no padding; single-channel pooling layer (SC-pooling): $3 \times 3$ kernel convolution, step=2 and 4 sides filled with 2 layers of 0 ($p = 2$); number of parameters: 2.5MByte.

The fully connected layers FC1, FC2, and FC3 are 3840, 480, and 5 dimensions, respectively. Since the input of the first layer is connected by the output of the previous layer into a vector as input. So the number of parameters in the first layer is 0, and the number of parameters in the second and third layers are 7.0MByte and 9.4kByte, respectively.

3.2. Step-by-Step Convolution
For a characteristic output channel, the traditional convolution can be divided into two steps: in the first step, each filter scans its corresponding characteristic input channel and outputs $M$ intermediate response channels (the filter is different for each characteristic output channel); in the second step, the $M$ intermediate response channels are added together to generate a characteristic output channel (as shown in Fig. 4).

Figure 3. Deep Neural Network Architecture

Figure 4. Conventional convolution
Obviously, the traditional convolution operation uses all the input feature channels to generate each output channel, which will consume a large number of additional parameters. In order to overcome this shortcoming, this paper proposes a convolution operation as follows:

In the first step, each filter is applied to only one intermediate channel instead of all intermediate channels. These filters are shared in each region of their corresponding characteristic input channels and are still updated by the back-propagation algorithm in the training phase. The input-output mapping through this filter can be expressed as:

\[
Y_{m,n}^{k} = \sum_{m,n \in \Omega} W_{m,n}^{k} X_{m,n}^{k}
\]

Among them, \( W^{k} \) is the filter in the No. \( k \) characteristic input channel, \( X_{m,n}^{k} \) is the eigenvalue located in the \((m, n)\) position in the No. \( k \) characteristic input channel. \( \Omega \) represents the area of the current processing feature, \( Y_{m,n}^{k} \) is the eigenvalue at the \((m, n)\) position in No. \( k \) output eigenchannel.

The difference between the two operations is that in the new convolution operation, each filter only depends on the corresponding characteristic input channel and does not use other input channels to generate the intermediate channel. Assuming that the size of each filter is \( w \times h \) and there are \( p \) characteristic input channels, then the single channel filter can be represented by \( w \times h \times 1 \times p \), where 1 means that each single channel filter depends on a single characteristic input channel, and because it is a single channel, the number of intermediate channel is also \( p \). Simple calculation shows that the filter in this step needs \( S_1 = w \times h \times p \) parameters, which is obviously much smaller than the parameter \( C_1 = w \times h \times p \times q \) consumed by the traditional convolution pool.

In the second step, \( p \) intermediate channels are taken as inputs and combined with filters represented by \( 1 \times 1 \times p \times q \) to generate \( q \) characteristic output channels. It can also be calculated that the filter in this step needs \( S_2 = p \times q \) parameters, while the parameter consumed in the traditional convolution step is \( C_2 = 0 \). Combined with the previous step, the new convolution operation consumes a total of parameters \( S = S_1 + S_2 = w \times h \times p + p \times q \), while the traditional convolution consumes parameters \( C = C_1 + C_2 = w \times h \times p \times q \). Comparing the two, it can be seen that when the output channel is large, \( S \) is much smaller than \( C \). Therefore, in the network architecture of this paper, a new convolution operation is selected to replace the traditional convolution operation at the higher level of the output channel.

A schematic diagram of the above new convolution operation is shown in Fig. 5

See Table 2 for the comparison of detailed parameters between the new and old convolution.

| Accuracy | Traditional convolution | Stepwise convolution |
|----------|-------------------------|---------------------|
| filter   | \( w \times h \times p \times q \) | \( w \times h \times 1 \times p \), \( 1 \times 1 \times p \times q \) |
| Parameter amount | \( w \times h \times p \times q \) | 2.25M+256k |
3.3. Pooling Strategy Selection for Different Network Layers

Previous studies have shown that the purpose of pooling operation is to compress the input feature map. On the one hand, it makes the feature map smaller and simplifies the computational complexity of the network. On the one hand, feature compression is carried out to extract the main features. This means that the pooling operation is inevitably accompanied by the loss of information, especially in the deep layer of the network structure. The pooling objects are often extracted high-level feature images, and the loss of information has a greater impact on the classification accuracy of the model.

In the network architecture of this paper, the traditional maximum pooling is selected for the shallow pooling operation. For 799 × 611 input images, this is conducive to quickly reducing the computation. Since the grain boundary image to be trained contains only two kind of pixel values, and the distribution of boundaries in the image is relatively sparse, selecting maximum pooling in the shallow layer of the network also ensures that if there are boundary pixels in the processing area, they will be selected and retained. In the deep layer of the network structure, inspired by the above-mentioned single-channel filter, the pooling operation is replaced by a learning single-channel filter, and the parameters of the filter are updated with the back propagation algorithm in training. This learning-based pooling can ensure that training errors are minimized in the training phase, and at the same time, the loss of feature information can be reduced in the deep layer, thus obtaining better performance than the maximum pooling.

3.4. Multi-layer Feature Fusion Learning

The features extracted from the shallow layer of convolutional neural network contain a lot of image details, while the features extracted from the deep layer contain more abstract features. The fusion of the details and abstract features extracted from different network layers is more conducive to the diversification of expressive force. Through deconvolutional visualization of the feature maps obtained by each layer of the network in this paper, it can be seen that convolution layer 2 mainly extracts low-level features with edge attributes, convolution layer 3 extracts texture features with irregular changes in local attributes, and convolution layer 6 learns complete and key features with discrimination. According to the own characteristics of grain boundary images in this paper, the features extracted from convolution layers 2, 3 and 6 with certain distinguishing significance are fused and relearned. The specific fusion architecture is shown in Fig. 6.

In Fig. 6, there is a multi-layer feature fusion module in the dotted line box. The fusion steps can be summarized as 4 steps:

1) Three characteristic graphs of 64 channels are randomly selected from each of the three convolution layers (i.e. 64 groups, 3 graphs per group) and normalized to 10 × 6 size by Resize function.
2) Pixel value in each feature map are connected into vector, and the similarity between the three vectors transformed from the same set of feature maps is calculated by using the cosine similarity measurement formula, and the calculated three results are added together.
3) 16 groups of feature vectors with the smallest similarity are selected, and three vectors in each group are aggregated into one feature vector.
4) 16 feature vectors are all connected into FC1 and then fed into the classifier. The cosine similarity measurement formula is as follows:

\[
sim = \cos(\theta) = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} (A_i)^2} \times \sqrt{\sum_{i=1}^{n} (B_i)^2}}
\]

Where \(A, B\) are eigenvectors.
4. Experimental Results and Analysis

The experiment aims to prove that: 1) the introduced improved algorithm can improve the effect of CNN-based method in image classification; 2) applying the network architecture and algorithm to grain boundary images can achieve good prediction of grain boundary corrosion resistance. In order to verify the effectiveness of the method proposed in this paper, the research adopts TensorFlow 1.5 to implement the optimized in-depth-wise learning framework shown in Fig. 2, with the pre-trained network model, the training machine environment of Ubuntu16.10 system. The experiment is conducted in an environment equipped with the Intel Core i5-7300H CPU of 3.1GHz, 8G memory and GTX1050Ti graphics card. The verification of the four experiments in sections 3.1 to 3.4 adopts the common data set CIFAR10 and CIFAR100, and the verification of the experiments in section 3.5 adopts the B10 copper-nickel alloy grain boundary data set provided by the School of Materials Science and Engineering, including 500 grain boundary images of 5 kinds representing different corrosion resistance.

4.1. Comparison of Network Architecture with Stepwise Convolution and Traditional Convolution

In this experiment, under the condition of maintaining the strategy of feature fusion and pooling, the performance difference between distributed convolution and traditional convolution is tested, and CIFAR10 and CIFAR100 are compared with the error rate and the number of network parameters as evaluation indexes. The error rate is defined as:

\[
\text{Error rate} = \frac{\text{Images Misclassified}}{\text{Images classified}} \times 100\%
\]

The comparison results are shown in Table 2. For CIFAR10 data sets, the step-by-step convolution in this paper reduces the error rate from 10.97% to 9.72% in traditional convolution, thus improving the performance by 1.25%, and for CIFAR100 data sets, the error rate is reduced from 30.12% to 29.25%. In addition, the parameter is reduced from 4.5 M to 2.506 M. The results show that the step-by-step convolution is beneficial to the improvement of classification accuracy and the reduction of parameters. The reasons are: on the one hand, the traditional convolution filter is a $W \times H \times Q$ three-dimensional tensor, while the proposed step-by-step convolution single-channel filter is a $W \times H$ two-dimensional tensor. Therefore, the number of parameters of the proposed method is one order less than that of the traditional convolution filter. On the other hand, under a certain amount of training data, the parameter reduction ability of this method can deal with the over-fitting problem more effectively, so this method has higher classification accuracy than traditional convolution.

| Table 3. Comparison of the Results of Two Schemes. |
|-----------------------------------------------|
| Error rate | Traditional convolution | Stepwise convolution |
| CIFAR10     | 10.97                 | 9.72               |
| CIFAR100    | 30.12                 | 29.25              |
| Parameter amount | 4.5M           | 2.25M+256k         |

Figure 6. Multi-layer Feature Fusion Learning Framework
4.2. **Comparison with Network Architecture with the Deep Maximum Pooling**

Under the condition of maintaining feature fusion and step-by-step convolution, this experiment tests the performance differences of different pooling strategies, and compares CIFAR10 with error rate and training error convergence speed as evaluation indexes. To this end, 5,000 iterations (each iteration traverses 20,000 samples) are defined as 1 Epoch.

The error convergence is shown in Fig. 7, from the 2nd to 14th Epoch, the training error of manual maximum pooling is smaller than that of single-channel learning pooling, which is because the feature of maximum pooling loss in the initial stage of training does not have sufficient influence on the accuracy of recognition. However, the error convergence amplitude of single channel pooling is larger than that of manual maximum pooling. The training error of single-channel pooling is already smaller than manual maximum pooling, starting with the 15th Epoch. Because the learning single-channel pooling filter will also update the parameters in the filter with back propagation. It can be better adapted to the extraction of main distinguishing features, and can reduce the loss of deep features. And it is more conducive to the convergence of training error to minimum.

![Figure 7. Comparison of Results of Two Schemes](image)

The comparison results of error rates are shown in Table 4. For the CIFAR10 datasets, the deep single channel pooling strategy in this paper reduces the error rate from 13.56% of the maximum pooling to 9.72%, which improves performance by 3.84%. It should be noted that learning-based deep layer single-channel pooling is better than step-by-step convolution in improving classification accuracy. The reason here is that step-by-step convolution focuses on reducing the number of parameters consumed in training and reducing the amount of calculation. Although simplifying parameters can promote the solution of over-fitting problem, it is still not as good as the accuracy improvement intensity brought by the preservation of deep discrimination features.

| Table 4. Comparison of Error Rates of Two Schemes. |
|-----------------------------------------------|
| Error rate | Maximum pooling | Single channel pooling |
| CIFAR10     | 13.56            | 9.72                 |

4.3. **Comparison with Network Architecture without Feature Fusion**

Under the condition of maintaining pooling strategy and step-by-step convolution, this experiment tests the performance difference caused by feature fusion or not, and also compares CIFAR10 and CIFAR100 with error rate as evaluation index.

The comparison results of error rates are shown in Table 5. For these two data sets, the performance of feature fusion has been improved by 2.61% and 2.29% respectively. From the perspective of improving accuracy, it can be seen that among the three improvement methods, single-channel pooling is the best, feature fusion and step-by-step convolution are the lowest in turn, step-by-
step convolution is the best in terms of simplified computation, feature fusion and single-channel pooling are the lowest in turn.

Table 5. Comparison of Error rates of Two schemes

| Error rate   | No feature fusion | feature fusion |
|--------------|-------------------|----------------|
| CIFAR10      | 12.33             | 9.72           |
| CIFAR100     | 31.54             | 29.25          |

4.4. Comparison with other methods
In this experiment, the proposed Optimal CNN and Enhanced ALEXNET [13], Supplement CNN [14], R-CNN [15] and Integrated CNN [16] are compared with CIFAR10 and CIFAR100 data sets. Detailed comparative data are shown in Table 6, where it can be seen that the accuracy of this method is slightly superior under medium training data. However, under larger data sets, the best accuracy cannot be achieved. Considering that the main application scenario of the model in this paper is the grain boundary image of the alloy, it is very difficult to collect enough training images, and it is also impossible to use some conventional methods such as folding and cropping to process grain boundary images to increase training sets. Therefore, abandoning some more complicated improvement schemes to allocate computing resources in a targeted way is beneficial to the practical engineering application effect.

Table 6. Comparison of Error Rates of Five Models

| Error rate       | CIFAR10 | CIFAR100 |
|------------------|---------|----------|
| Strengthen Alexnet [13] | 9.80    | 28.15    |
| Supplement CNN [14]  | 12.67   | 34.16    |
| R-CNN [15]        | 10.26   | 31.75    |
| Integrate CNN [16] | 11.63   | 30.05    |
| Algorithm of this paper | 9.72    | 29.25    |

4.5. Classification prediction results of grain boundary images
In this experiment, the optimal network model established in this paper is applied to the classification of grain boundary images. 50 images are taken for each type of grain boundary image as the test set and the remaining 2250 images as the training sets. The experiment is also compared with the above four improved network models. The comparison results of classification accuracy can be seen in Table 7.

Table 7. Comparison of Accuracy of Five Models

| Accuracy       | Grain boundary image |
|----------------|----------------------|
| Strengthen Alexnet [13] | 79.75   |
| Supplement CNN [14]  | 77.14    |
| R-CNN [15]        | 77.92    |
| Integrate CNN [16] | 81.33    |
| Algorithm of this paper | 81.76    |

Owing to the limitation of the number of training sets, the classification accuracy of the five network models has decreased by different ranges, but the optimal network model proposed in this paper still enjoys the highest classification accuracy because there is not much feature information that can be dug in grain boundary images. However, high-level features cannot be well preserved by the
improved maximum mean pooling in reference [13], while other improved schemes in this paper, such as the hash coding introduced in the full connection layer, are more prone to dealing with large-scale image classification processing tasks, so the classification accuracy of CIFAR100 in Section 3.4 experiments is higher than that of this model, but it still cannot meet the requirements in the current classification task environment. In reference [14], the strategy of inverting the feature map extracted from convolution layer and acting on the activation function together with the original feature map is not applicable to the application environment of this paper, because the image to be processed is itself a binary boundary image, and this optimization operation has no obvious advantages, so its performance is the worst among the five models. What’s worth mentioning is that the integrated convolutional neural network in reference [16] has reduced the accuracy difference from 1.91% to 0.23% from the experiment of CIFAR10, the public data set in section 3.4, to this experiment. It is still lower than the method in this paper, but may be effective to classify and predict grain boundary images by using image complexity and integrated network to enhance training effect.

5. Conclusion
Aiming to verify that convolutional neural network can be applied to the classification and prediction of grain boundary images, the architecture of convolutional neural network is first proposed in this paper. Three improved strategies for convolutional neural network are put forward. And then the model with the other four latest improved models is compared through experiments to test the effectiveness of the model for image classification. And eventually the model to the classification task of grain boundary images is applied. The results of the experiment verify the effectiveness of the model, which proves to be of a good application prospect due to the limited number of prediction models that can be established between grain boundary distribution structure and metal corrosion resistance. The follow-up is to further improve the accuracy of the model thus reaching a higher application level.

Acknowledgments
I would like to thank my tutor Dong for her careful guidance and my classmates for their help. I would like to thank the school for providing the platform and help. I also want to thank my parents and friends for their encouragement.

References
[1] Anthimopoulos M, Christodoulidis S, Ebner L, et al. Lung Pattern Classification for Interstitial Lung Diseases Using a Deep Convolutional Neural Network.[J]. IEEE Transactions on Medical Imaging, 2016, 35(5):1207- 1216.
[2] Cai Z, Fan Q, Feris R S, et al. A unified multi-scale deep convolutional neural network for fast object detection[C] //European conference on computer vision. Springer, Cham, 2016: 354-370.
[3] Du C, Gao S. Image segmentation- based multi-focus image fusion through multiscale convolutional neural network[J]. IEEE access, 2017, 5: 15750-15761.
[4] Wang Xiaosong, Peng Yifan, Lu Le, et al. Tienet: text-image embedding network for common thorax disease classification and reporting in chest x-rays [C]//Proc of IEEE Conference on Computer Vision and Pattern Recognition.2018:9049-9058.
[5] Li Yaping, Gao WeiXin. Research on X-ray welding image defect detection based on convolution neural network[J]. Journal of Physics: Conference Series,2019,1237(3).
[6] Roth H R, Lu L, Seff A, et al. A new 2.5 D representation for lymph node detection using random sets of deep convolutional neural network observations[C]/International conference on medical image computing and computer-assisted intervention. Springer, Ch- am, 2014: 520-527.
[7] Cheng M M, Mitra N J, Huang X, et al. Global contrast based salient region detection[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(3): 569-582.
[8] Girshick R, Donahue J, Darrell T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.

[9] Girshick R. Fast r-cnn[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1440-1448.

[10] Ren S, He K, Girshick R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[C] //Advances in neural information processing systems. 2015: 91-99.

[11] Zhang Y, Feng X, Song C, et al. Quantification of grain boundary connectivity for predicting inter-granular corrosion resistance in BFe10-1-1 copper–nickel alloy[J]. MRS Communications, 2019, 9(1): 251-257.

[12] Yang Baohua, Qi Lin, Wang Mengxuan, Hussain Saddam, Wang Huabin, Wang Bing, Ning Jingming. Cross-Category Tea Polyphenols Evaluation Model Based on Feature Fusion of Electronic Nose and Hyperspectral Imagery.[J]. Sensors (Basel, Switzerland),2019,20(1).

[13] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016:770-778.

[14] ZamirA R, Sax A, Shen W, et al. Taskonomy: disentangling task transfer learning [C/ Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2018:3712-3722.

[15] Mills Kyle, Ryczko Kevin, Luchak Iryna, Domurad Adam, Beeler Chris, Tamblyn Isaac. Extensive deep neural networks for transferring small scale learning to large scale systems.[J]. Chemical science,2019,10(15).

[16] Mingyuan Xin, Yong Wang. Research on image classification model based on deep convolution neural network[J]. EURASIP Journal on Image and Video Processing,2019,2019(1).

[17] Sensor Research; Researchers from Hohai University Report on Findings in Sensor Research (Hyperspectral Remote Sensing Image Classification Based on Maximum Overlap Pooling Convolutional Neural Network)[J]. Computers, Networks & Communications,2019.