UNIVERSITY
OF TURKU

# Research on Calligraphy Evaluation Technology Based on Deep Learning

Master's thesis (Technology)

Department of Information Technology

2023

Zhi Zhang

Supervisors:

Dr. Paavo Nevalainen

Prof. Tomi Westerlund

**University of Turku**
**Department of Computing, Faculty of Technology**
**Master of Science in Technology Thesis**

**RESEARCHER:** ZHI ZHANG
**Title:** Research on Calligraphy Evaluation Technology Based on Deep Learning
**Programme:** Master's Degree Programme in Information and Communication Technology
**Number of pages:** 60 pages, 0 appendix pages
**Date:** June 2021
The originality of this thesis has been checked in accordance with the University of Turku quality assurance system using the Turnitin Originality Check service.

---

Today, when computer-assisted instruction (CAI) is booming, related research in the field of calligraphy education still hasn't much progress. This main research for the calligraphy beginners to evaluate their works anytime and anywhere. Author uses the literature research and interview to understand the common writing problems of beginners. Then conducts discussion on these problems, design of solutions, research on algorithms, and experimental verification. Based on the ResNet-50 model, through WeChat applet implements for beginners. The main research contents are as follows:
(1) In order to achieve good results in calligraphy judgment, this article uses the ResNet-50 model to judge calligraphy. First, adjust the area of the handwritten calligraphy image as the input of the network to a small block suitable for the network. While training the network, adjust the learning rate, the number of image layers and the number of training samples to achieve the optimal. The research results show that ResNet has certain practicality and reference value in the field of calligraphy judgment. Regarding the possible over-fitting problem, this article proposes to improve the accuracy of the judgment by collecting more data and optimizing the data washing process.
(2) Combining the rise of WeChat applets, in view of the current WeChat applet learning platform development process and the problem of fewer functional modules, this paper uses cloud development functions to develop a calligraphy learning platform based on WeChat applets. While simplifying the development process, it ensures that the functional modules of the platform meet the needs of teachers and beginners, it has certain practicality and commercial value. After the development of the calligraphy learning applet is completed, it will be submitted for official.

**Keywords**： Calligraphy Evaluation, Convolutional Neural Network, Deep Residual Network, Program Development

# Table of Contents

# 1 INTRODUCTION

As the inheritance of China's famous traditional culture, the Chinese character is also one of the oldest characters in the world. Throughout history, its inheritance and development have exceeded 3600 years [1]. For thousands of years, the Chinese have never stopped using Chinese characters and changed to another name. After continuous development and evolution, Chinese character has become the most widely used character in China. Historically, Chinese character has also influenced neighboring countries such as Korea, Japan, and Vietnam, as well as some Asian regions. Many countries even use Chinese characters as their official character [2].

Calligraphy is a form of Expression of the Chinese character. It is written in accordance with the calligraphic style, structural features, and rules of writing while beautifying its structure and stroke. As the precipitation of Chinese culture and the crystallization of the wisdom of the people, calligraphy has now become a work of art with aesthetic feeling, emitting the eternal charm. However, with the development of barbaric aggression and economic globalization of the modern powers, all kinds of foreign cultures have caused great impact on the local culture, and foreign cultures have gradually occupied the field that should belong to the traditional culture [3]. In recent years, with the development and popularization of mobile phones and Internet terminals, people pay less attention to traditional writing skills. Chinese calligraphy is gradually being neglected.

China's Ministry of Education, in order to promote teenagers of traditional calligraphy study, also in perfecting related guidelines, issued in 2013 "the calligraphy education of primary and middle schools guidelines"[4] for adolescent students of calligraphy education made clear requirements, and ask the local education department for instructions to promote the calligraphy education, constantly asked education industry to reinforce the cultivation of student's ability of calligraphy. Although the relevant departments have made efforts, there are still various problems in the promotion of relevant policies in the education reform [5]. To sum up, as following:

(a) Under the background of the rapid development of science and technology, the public pays little attention to the calligraphy culture;
(b) There are defects in the education system, and there is no specific curriculum for calligraphy.

(c) There is also a serious shortage of educational resources, with only a handful of professional calligraphy teachers.

With the advent of mobile technology and the popularization of smart devices, the way students receive information is quite different from the past. Mobile smart devices have unique advantages in terms of convenience and portability. In every stage of the education system, there are a variety of computer-aided teaching systems and promote information management in the teaching field. The intelligent teaching process has made a certain breakthrough, like "Mooc," "Netease Cloud Classroom," "Ape Guidance Online Education," and so on [6]. The combination of "offline teaching" and "online teaching" system can give students more autonomy, arouse their enthusiasm as much as possible, and help them to master the basic principles during class and consolidate them through practice after class [7]. This teaching system, combined with offline self-practice, has the advantages that traditional classroom teaching does not have. For example:

(a) Giving more emphasis to the key points of teaching content.
(b) Improving students' learning efficiency and reduce teachers' working pressure in the teaching process.
(c) Teachers can invest more time in new research and not just teaching work.
(d) The teaching process of mobile intelligent devices highlights the flexibility and autonomy of learning, and students' learning will not be limited.

Meanwhile, with the concept of lifelong learning, studying through mobile devices as a new way of learning is gradually accepted by young men, and mobile Internet technology also provides feasible technical support for the realization of lifelong learning. Complete technical support, a huge user base, and the change of learning concept make mobile learning enter our daily life.

The auxiliary teaching mode based on mobile intelligent devices has been developed rapidly, but for those fields with too strong subjective consciousness, such as calligraphy judgment. First of all, calligraphy education still stays in the state of face-to-face teaching. Basically, every student needs the guidance of a calligraphy teacher to write calligraphy well. However, due to the limitation of time and space, this kind of education is inefficient and fails to arouse the enthusiasm of students. Even in the field of commercial technology, there is no one that can help beginners learn calligraphy auxiliary teaching system. Therefore, using Image processing technology and combining with calligraphy evaluation standards by Lu Weizhong, Tian Yingzhang, and Kong Dewen[8],[9],[10],[11]. Developing an online calligraphy evaluation system based on AI technology, study the feasibility of deep learning technology, and expect that this system can enable calligraphy beginners to evaluate and improve their calligraphy works anytime and anywhere.

The first chapter briefly introduces the current situation of calligraphy teaching and learning, combined with the advantages of mobile device learning, to explain the reasons for the study of the calligraphy evaluation system. The remaining chapters are assigned as follows:

The second chapter introduces the related technical background and related theoretical research. Firstly, it introduces the Research of "Computer Vision" in the calligraphy field. Secondly, it introduces the common image classification methods, the overview of deep learning and "Convolutional Neural Network," and the calligraphy classification method based on CNN. Finally, the background of the WeChat Mini program and the "mobile learning concept" is introduced.

The third chapter introduces the knowledge of CNN and the R-CNN network and the data sources used in this article. Then use the data to train an R-CNN model and process the original calligraphy images, drawing a conclusion through the analysis of the experimental results. It is not suitable for the analysis of calligraphy image, but the effect of image classification has given people a lot of inspiration.

The fourth chapter introduces another network called ResNet and further analyzes the depth of the ResNet. It also summarizes the residual module of the ResNet and applies the ResNet to the brush calligraphy judgment classification to get the result. Using data sets for training and continuously optimizing the network. At the end of this chapter, the accuracy rate in the evaluation of calligraphy "Heng" by the ResNet network can be guaranteed to be above 70%, indicating that the deep learning technology is feasible in the application of auxiliary calligraphy practice.

The fifth chapter introduces the development of WeChat Mini Programs with auxiliary functions of calligraphy learning. First, it mainly introduces the significance of developing WeChat Mini Programs from the perspective of technology, function, and users. Secondly, it introduces the development tools of WeChat applets and the convenience brought by cloud development functions. Finally, some details of page development are introduced.

The sixth chapter is a summary and outlook of the article, first introduced the main work, then introduced the direction of optimization handwritten calligraphy auxiliary systems. In the future, it can be optimized by improving the accuracy of image classification and the user experience of the WeChat applet.

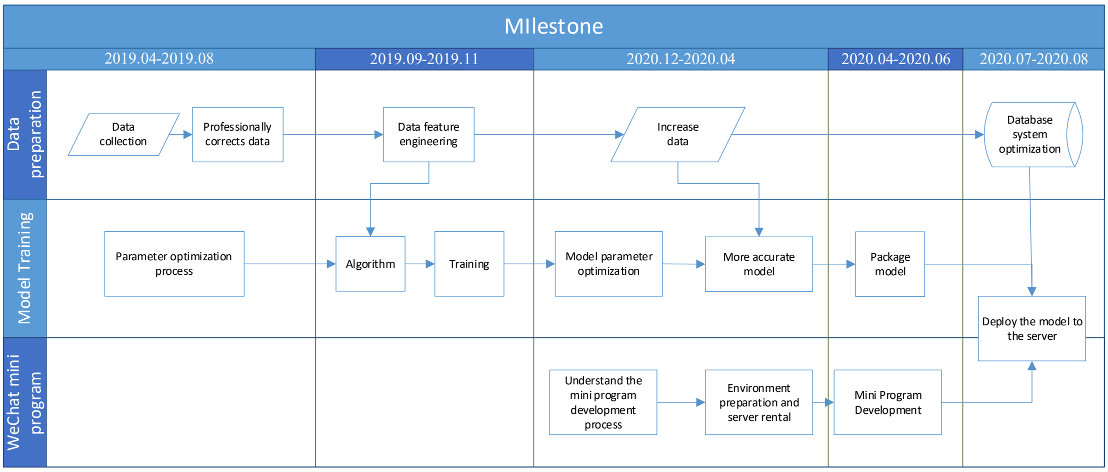



**Fig 1.1** Milestone

Finally, attach the milestones of this article in the development process of handwriting calligraphy assistance system, as shown in the figure 1.1.

# 2 BACKGROUND & RELATED WORKS

## 2.1 Computer vision in the field of Chinese calligraphy

As early as 1966, IBM employees Casey and Nag invented a template matching method to recognize Chinese characters [12], [13]. Since then, Japanese researchers have come up with a variety of methods and built a device that can recognize 4,000 Chinese characters [14]. China began to carry out recognition research in the 1970s, and after many years of exploration, many problems of Chinese character recognition have been solved [14]. In the field of handwritten character recognition, with the deepening of the research. In recent years, the characteristics of the elastic network [15], [16]. Gabor features and gradient features, such as the identification method, can achieve good effect [17]. Gabor double elastic network characteristics. The research of handwriting recognition based on shape similarity distance [18] has also obtained good results.

Wan Hualin, Zhuang Yueting et al. proposed to compare the image similarity by the features of image texture and geometric shape, and to evaluate the calligraphy font by the similarity with the defined standard font [19], [20]. Yu-Kai et al. proposed a skeleton similarity-based calligraphy word retrieval method [21]. This method firstly extracted the skeleton of the candidate word, and found the relative distance on the skeleton of the image by calculating the pixel points in a certain range around the image, and then screened by comparing the skeleton similarity [22]. Jian Liqiong proposed a character recognition method based on Hu moment. After extracting the Hu moment of the image, the k-nearest neighbor method was used to distinguish the image [23]. However, the evaluation content of this system was not specific enough, and guidance for single words was still lacking.

## 2.2 Background

In 2012, the Alex Net model was proposed and won the champion of ImageNet Large-scale visual recognition Competition [24]. As the number of LAYERS of CNN increases, the image classification performance of the network gets better and better. More deep CNN models have been proposed one after another, such as Google Net at the 19-layer VGG layer [25],[26]. The classification accuracy of images has been greatly improved, which is closely related to the great increase of CNN depth. However, an extremely deep convolutional neural network will not only cause the gradient to disappear but also increase the risk of network overfitting, thus affecting the image classification accuracy of the network model. The initial research method is mainly to alleviate the problem of network gradient disappearance by initializing the network model and conducting hierarchical training [27].

At present, deep CNN mostly uses the activation function ReLU to alleviate the problem of network gradient disappearance. Compared with the Sigmoid function, the ReLU function is more effective in alleviating the gradient disappearance [28]. Direct supervision can be used to train the deep CNN model. However, in the process of training the deep CNN model. Once part of the input of activation function ReLU enters the hard saturated region, the corresponding weight cannot be updated quickly, which is called "Gradient Vanishing" [29].

The phenomenon of neuronal death will lead to the convergence of CNN. As a result, many new activation functions have been proposed. For example, the PReLU function introduces additional parameters to improve its performance. ELU (Exponential Linear Unit) function combines the Sigmoid and ReLU functions, and the problem of gradient vanishing of the network model can be greatly alleviated [30]. The PELU (Parametric Exponential Linear Unit) function adds parameters to the ELU function. By training the network model, the parameters of the network model are updated in time to control the performance of the activation function, so that the bias drift and gradient vanishing can be better controlled when training the network.

The depth of the CNN is further increased, but the accuracy of the model is not improved. Instead, the phenomenon of the gradient vanishing, which led to higher errors. He Kaiming and others proposed the ResNet model, which is better than ordinary CNN in terms of convergence and classification performance [31]. The main innovation of the ResNet model is the introduction of a residual structure into the network, which can quickly transfer the results of the previous layer to the next layer of the network, so that after the network model is further deepened, its error will not continue to increase, Which allows the ResNet model to have more layers and more accurate accuracy at the same time.

The image classification method based on the ResNet model has no specific application in handwritten calligraphy classification, and it is currently in an undeveloped field. The goal of handwriting calligraphy classification can be divided into two aspects. One is to extract the texture structure features of calligraphy in calligraphy images, and the other is to measure and learn the extracted text features. When describing calligraphy characteristics, the desired characteristics have stable performance and powerful, distinguishing effects [32]. Before the advent of deep learning-based methods, researchers' most common method was to use artificially designed features to describe the structure of calligraphy.

## 2.3 Common image classification methods

### 2.3.1 Traditional methods

The traditional image classification method to create a new model of image recognition has four steps: Low-level feature learning, feature coding, spatial feature constraints, and classifier design [27],[33],[34]. As following:

(a) **Low-level feature learning and extraction:** In the image, a large number of local features are obtained according to a fixed step size and size for learning. Commonly used local features include SIFT (Scale-Invariant Feature Transform), HOG (Histogram of Oriented Gradient), LBP (Local Binary Pattern), etc. Usually using multiple feature learning to prevent useful missing information.

(b) **Feature coding:** The previous step includes redundancy and noise, which should improve the Expression of feature robustness. The previous step can be coded using the feature transformation algorithm. Commonly used methods include vector quantization coding, sparse coding, and Fisher vector coding.

(c) **Spatial feature constraints:** Taking the MAX value or average value in each dimension feature in the space can get the feature representation without deformation.

(d) **Classification by classifier**: After the third step, image learning can use a vector of uniform dimensions, and then classify the image. Frequently used classifier: SVM (Support Vector Machine).

### 2.3.2 Supervised classification

(1) Minimum distance classification

The basic theory is simple, and the classification accuracy is relatively low, but the advantage is that the calculation time is short, so it is used in the classification task of fast browsing. As shown in Figure 2.1:

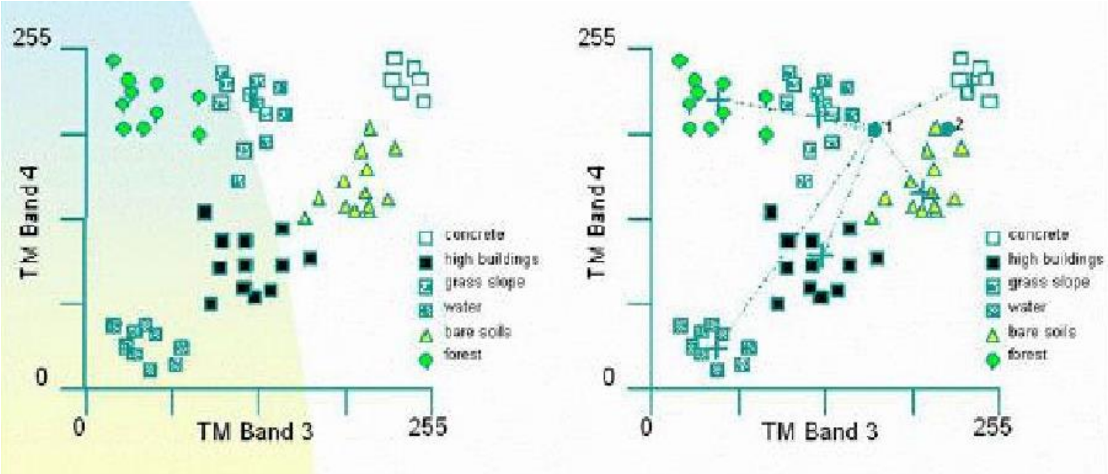



**Fig 2.1** Minimum distance classification

(2) Multi-level cutting classification

It is helpful to directly understand how to cut the feature space and how to associate the pixels waiting to be classified with their types. As shown in Figure 2.2:

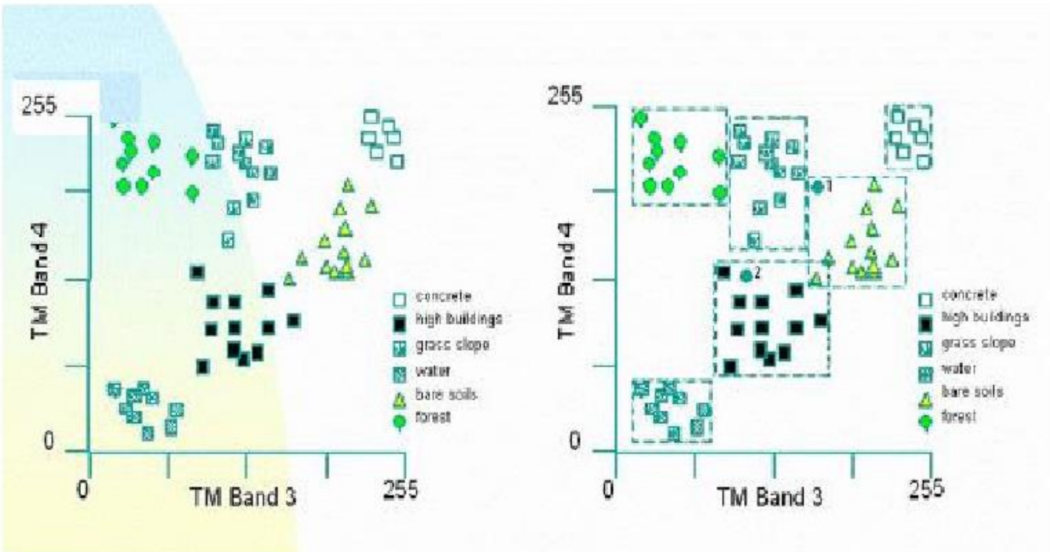



**Fig 2.2** Multi-level cutting classification

(3) Characteristic curve window

The classification of different features depends on the selection of feature parameters and window size, and the selection of different feature parameters and window size needs to be determined according to the distribution status in the space of different feature parameters.

(4) Maximum likelihood ratio classification

Frequently used. Perform calculations on each pixel, and divide the attribution probability of different types into the type with the highest attribution probability. As shown in Figure 2.3:

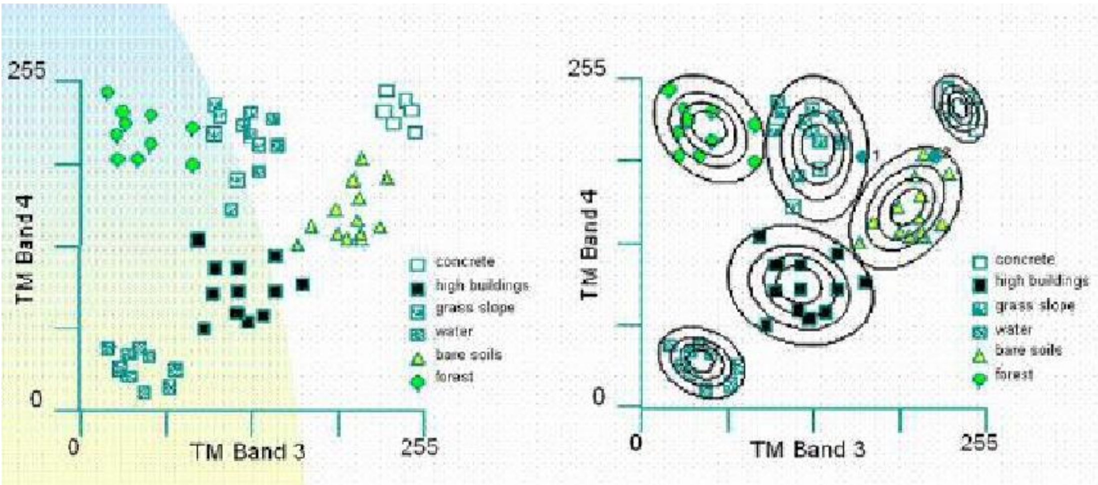



**Fig 2.3** Maximum likelihood ratio classification

### 2.3.3 Unsupervised classification

(1) Hierarchical Clustering

Use "distance" in the spatial distribution to evaluate the similarity of samples and classify or merge their distribution into different groups. The geographical significance of each group can be determined according to the ground truth or by comparing the known category data.

(2) Dynamic clustering

In the initial situation, images were classified roughly and then based on certain principles, in different types, from the new combination of data to the end of reasonable classification.

**The difference between supervised and unsupervised classification:**

The basic difference is whether the training set can be used to obtain the prior type. The former is to select feature parameters according to the samples provided by the training set and establish a discriminant function to classify [35]. The latter does not need to know too much a priori category knowledge, because ground truth spectrum statistics can be used for the particularity, so use it to classify. So the latter is simpler than the former and has a certain accuracy classification.

### 2.3.4 Image classification algorithm

(1) Minimum distance algorithm

In the simple and rude algorithm, we use the data published by Stanford University [36]. There are a total of 60,000 $32 \times 32 \times 3$ image data, and there are ten categories; Each image belongs to one of these ten categories; we can use 5000 as the training set, Use 10 million as the test set. Use the nearest neighbor to calculate the most similar picture in the category. As shown in Figure 2.4:

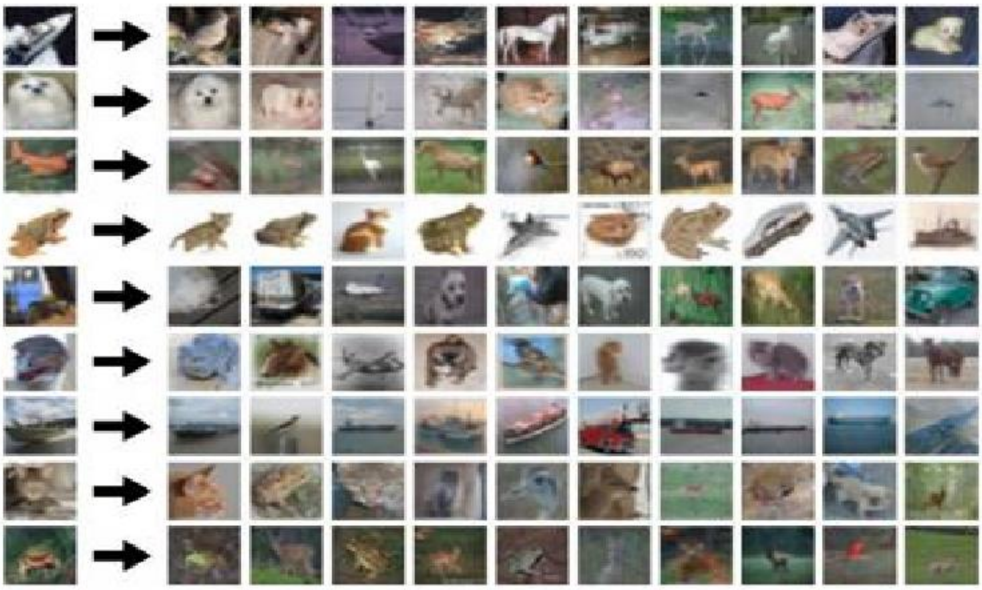


**Fig 2.4** Minimum distance algorithm classification results

The result obtained by the nearest neighbor algorithm is as shown in the figure above, and it can be found that there are still many wrong classifications. The principle of the nearest neighbor algorithm is to remember only the data of each category when training, and then calculate the distance between the image in the test data set and each image in the training data set when predicting, and then look at the image category in the training data set closest to him. This category is the category we want to predict.

Generally, we can expand this $32 \times 32 \times 3$, and then calculate the distance between the two vectors, the L1 distance is:

$$d_1(I_1, I_2) = \sum_p \left| I_1^p - I_2^p \right| \tag{2.1}$$

This is the L1 distance calculation in a color channel, which subtracts the corresponding positions and then adds up the numbers in each square.

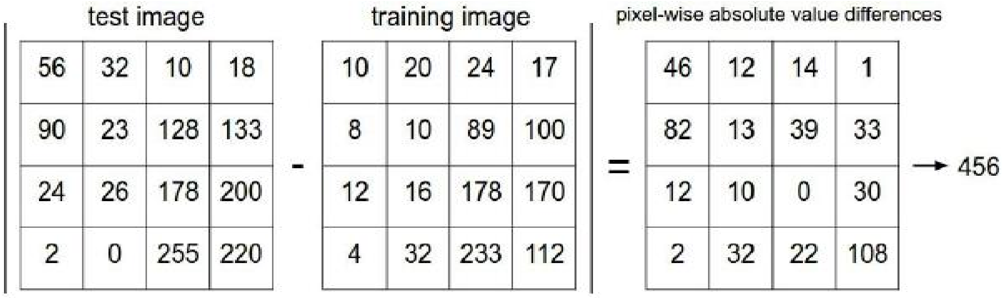



**Fig 2.5** L1 distance calculation diagram

(2) Nearest Neighbor algorithm

This is a relatively simple method: first, "remember" all the images (training set) annotated by the system, and when encountering unlabeled images (images in the test data set), it will be combined with the "remember" image Compare "similarity," find relatively similar and labeled images, let the type of this image become the type of image to be classified, this is the concept of the name "Nearest Neighbor"[37].

The method of judging the "similar" of images: For example, directly find the sum of the subtraction of pixel values at the same position (the distance between two images L1), or use the distance L2 between two images, in fact, find the same one first Make the difference of the position pixel value and then square. Finally, sum up all the square values and extract all squares. The correct description is: mark the image as $I_1$ and $I_2$, $I_1^k$ represents the pixel value of the image $I_1$ at position k, then the L1 and L2 distances of the images $I_1$ and $I_2$ can be expressed respectively for:

$$d_1(I_1, I_2) = \sum_p \left| I_1^k - I_2^k \right|$$

$$d_2(I_1, I_2) = \sqrt{\sum_p \left( I_1^k - I_2^k \right)^2} \tag{2.2}$$

"Nearest Neighbor" is easy to implement, but there are many shortcomings: the speed of the test is relatively slow. When determining an image, each must be compared with the training data set. The calculation amount for each test is very large, especially when it is encountered. In the picture test, a larger space is still needed to store the training data set, but the result is that the accuracy rate is relatively low.

(3) Linear Classification

Use a single "Fully Connected Layer," input all the pixel values of the image, get the results through the calculation of the fully connected layer, and output various types of "scores." The final selected image category has the highest score.

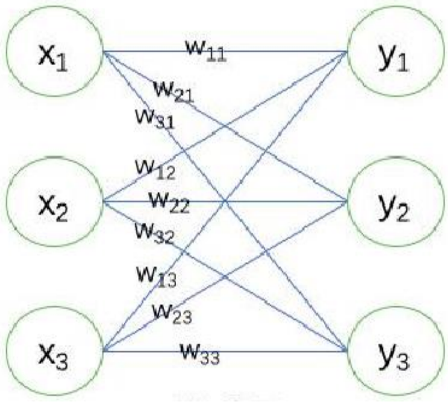



**Fig 2.6** Fully Connected Layer

As shown in Figure 2.6, $X_1$, $X_2$, $X_3$ are all input pixel values. In a natural image, there may be many inputs, such as the CIFAR-10 data set [38]. The image size is $3\times32\times32$ (3 is the RGB Channels), then there are $3\times32\times32$ inputs; $W_{11}$, $W_{21}$, $W_{31}$ represent the weights, $y_1$, $y_2$, $y_3$ are the scores of each category of output, such as $y_i$ represents the score of the cat category. The output calculation method is:

$$y_1 = w_{11} * x_1 + w_{12} * x_2 + w_{13} * x_3 \tag{2.3}$$

$$y_2 = w_{21} * x_1 + w_{22} * x_2 + w_{23} * x_3 \tag{2.4}$$

$$y_3 = w_{31} * x_1 + w_{32} * x_2 + w_{33} * x_3 \tag{2.5}$$

Vectoring it and let:

$$X = [x_1 \;\; x_2 \;\; x_3]^T$$
$$Y = [y_1 \;\; y_2 \;\; y_3]^T \tag{2.6}$$

$$W = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \tag{2.7}$$

Then, the result of Y:

$$Y = W \bullet X \tag{2.8}$$

In the final implementation, a bias will be added to the score of each category. It is similar to the original "proportional function," which always passes through the origin. Now add an offset to become an ordinary "primary function," which no longer necessarily passes through the origin. This can make the Expression more general, more expressive ability. That is:

$$y_1 = w_{11} * x_1 + w_{12} * x_2 + w_{13} * x_3 + b_1 \tag{2.9}$$

$$y_2 = w_{21} * x_1 + w_{22} * x_2 + w_{23} * x_3 + b_2 \tag{2.10}$$

$$y_3 = w_{31} * x_1 + w_{32} * x_2 + w_{33} * x_3 + b_3 \tag{2.11}$$

Similarly, it can also be vectorized. For simplicity, the offset can be combined in the input X, so that:

$$X = [x_1 \;\; x_2 \;\; x_3 \;\; 1]^T$$
$$Y = [y_1 \;\; y_2 \;\; y_3 \;\; 1]^T \tag{2.12}$$

$$W = \begin{bmatrix} w_{11} & w_{12} & w_{13} & b_1 \\ w_{21} & w_{22} & w_{23} & b_2 \\ w_{31} & w_{32} & w_{33} & b_3 \end{bmatrix} \tag{2.13}$$

It can be found that the result $Y = W \bullet X$ is consistent with the above.

## 2.4 Overview of deep learning

Artificial intelligence is a new type of science and technology that simulates theories that challenge the human brain and studies how to apply this technology to application systems. In recent years, AI has used computer programs to challenge the development of the human brain with significant breakthroughs, but related technologies in medicine are still in the development stage. So far, deep learning is relatively perfect. Deep learning was proposed by Hinton et al. in 2006 [39]. It can itself be the development of neural networks, in the new environment of machine learning, and an unsupervised one. The purpose of deep learning is to establish and stimulate the neural network of the human brain to understand learning. It can simulate the mechanism of the human brain to interpret images, sounds, and texts. Strictly speaking, the meaning of deep learning is very broad, but in current research, when it comes to it, everyone will think of the large deep CNN proposed by SERMANET et al. [40].

The essence of deep learning: create a machine learning model with multiple hidden layers and a large number of training samples to learn. It can learn many useful features, and ultimately improve the accuracy of testing or classification. It is now used in other fields such as computer vision, speech recognition, and NLP (natural language processing).

## 2.5 Overview of Convolutional Neural Networks

### 2.5.1 Convolutional neural network structure

A typical CNN for image processing is generated by a series of convolutional networks, including a series of data reduction in the pooling layer. Similar to the low-level vision processing method in the human brain, convolutional network diagnosis obtains image features [41], such as checking organs or liver and gallstones, followed by high-level features, such as obtaining partial and full-level features. CNN generally uses one or more category labels as output.

CNN has multiple hidden layers. The neurons in each layer can receive the data and information from the previous layer and perform exercises. It should be noted that the neurons in each layer can be independent of each other. The network architecture is shown in Figure 2-7:

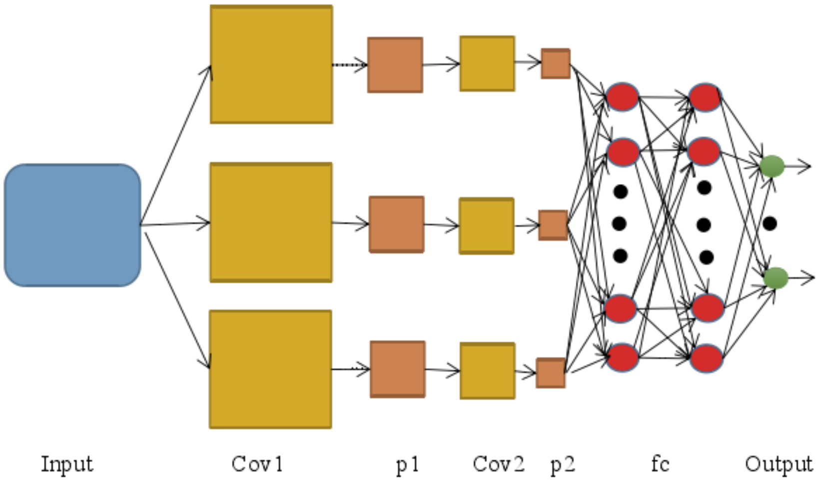



**Fig 2.7** Structure of CNN

As shown in Figure 2.8, CNN consists of these parts: input layer (Input), convolutional layer (Cov1, Cov2), pooling layer (p1, p2), fully connected layer (FC), output layer (Output).

Input layer: Input the image directly into the network, obtain the features after training, and preprocess the original image data. The processing process is two steps:

The input layer directly inputs the image into the network, obtains the characteristics after training, and preprocesses the original image data. The processing process is two steps [41]:

(1) **Grayscale:** When processing color images, we should process RGB channels in sequence, but it takes time. Therefore, it is necessary to increase the processing speed and perform gray-scale processing on the color image to reduce the required data value.

(2) **Image Enhancement:** to enhance the useful information in the image, in order to optimize the visual experience of the image, in the application of determining the image, increase the global or local features of the image, so that the originally blurred image becomes clear or enhances the interesting features, zoom The difference between the features of different objects in the image prevents uninteresting features, allows them to optimize the image quality, enrich the amount of information, enhance the image interpretation and recognition capabilities, and meet some special analysis.

(3) **Convolutional layer:** Convolution operation is actually input. If convolution is treated as a black box, the output can also be input.

The following example describes its operation process:

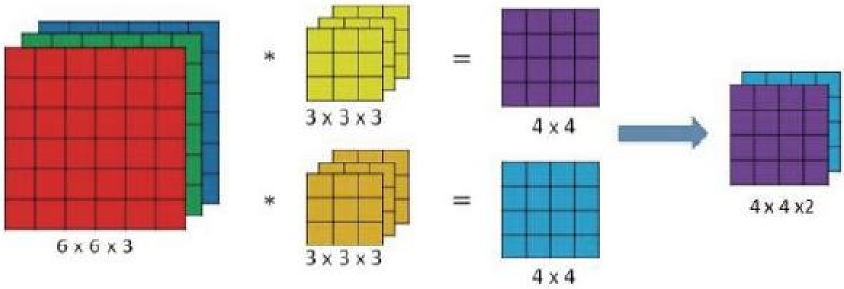



**Fig 2.8** CNN calculation process

In Figure 2.8, the image input size is 6x6x6, the number of channels is 3, there are two convolution kernels, the size is 3×3, three channels, and the convolution operation is performed. The window sliding method is still used, from left to right, from top to bottom, the corresponding positions of the three channels are multiplied and summed, and the output is a 4×4×2 Feature map.

**Pooling layer:** The operation of special processing data on CNN. After pooling, the size of the image features can be reduced, which can solve the problem of a large amount of calculation if the result of the previous layer is input. 2×2 pooling usually reduces the feature map size by 50%. Later, a convolution kernel with a size of 3×3, a step size of 2, and pad=0 are often used instead of the pooling layer, and similar results will also appear. Figure 2.9 shows the commonly used pooling method.

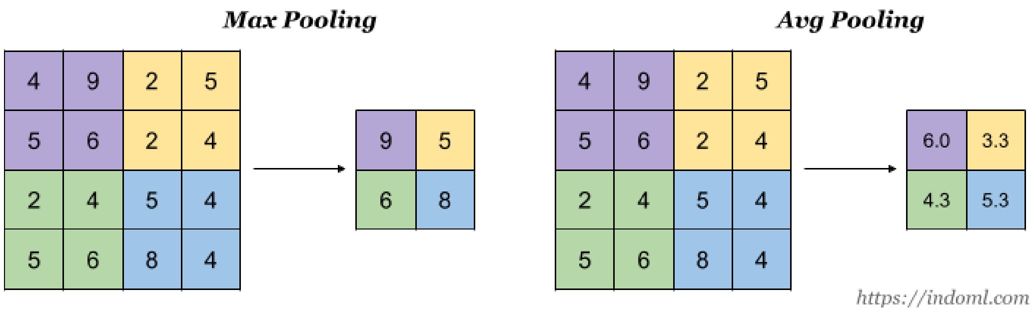



**Fig 2.9** Average pooling and Max pooling

**Activation function:** Linear operations are operations like convolution operations and pooling operations. However, a large amount of data in nature has a non-linear relationship when it is classified, so non-linear elements need to be added so that the network can solve the non-linear relationship. Common ReLU activation functions are as follows:

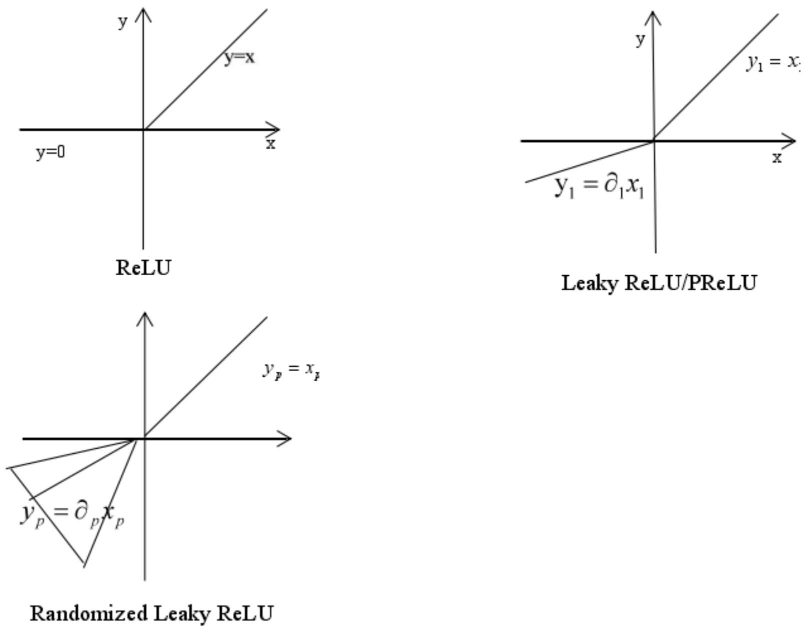



**Fig 2.10** Activation function

**Full connected layer (FC layer):** This layer is the layer that consumes the most parameters. If the input of this layer is 4×4×100, the output is 512, so 4×4×100×512 parameters are required; compared with the usual convolutional layer, if the volume of product core is 4×4, the output is 512, and the required parameters are 4×4×512. Usually, the network architecture includes two FC layers, and the output of the second FC layer corresponds to the number of output categories.

**Output layer:** Determine the output content according to specific tasks.

Today, although CNN is a powerful tool in machine learning and a highly parallelizable algorithm, it also has shortcomings.

(1) CNN requires a large number of labeled training samples, but it is difficult to satisfy because there is no organization in the calligraphy field to collect calligraphy pictures, and the classification methods of different genres are inconsistent.
(2) CNN must have a large amount of calculation and memory because the depth of CNN needs to be trained. Otherwise, the training process will take longer.
(3) As the depth of CNN increases, it becomes a complex training process. The complexity is over-fitting and convergence. This generally requires repeated adjustments to the network architecture or parameters to ensure that all layers are learning at a considerable speed.

### 2.5.2 LeNet

The first successful use of CNN was proposed by Yann Le Cun in 1990 [42]. The most famous among them is the LeNet structure, which can get zip codes, numbers, etc. Let's introduce LeNet-5 ("-5": 5 layers displayed) used in major deep learning models today. The difference between LeNet-5 is that its structure is conv1->pool->conv2->pool2->fc layer. Invariably, after the convolution layer immediately pooled layer mode.

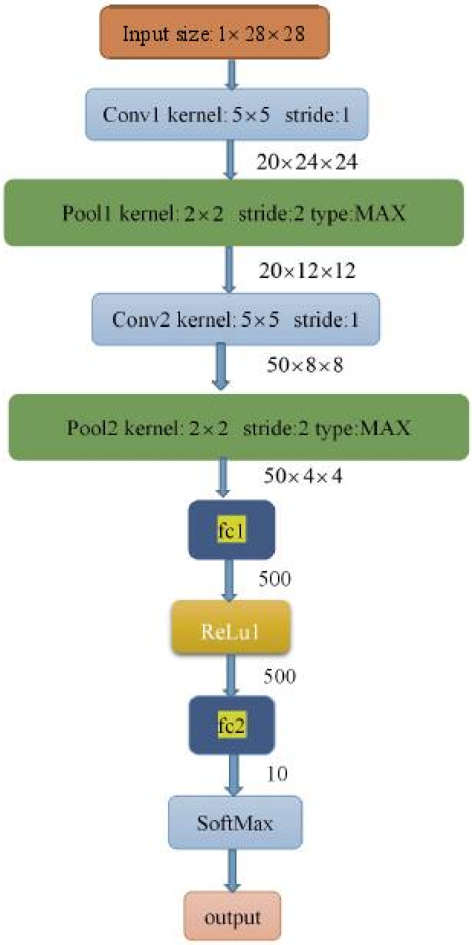



**Fig 2.11** LeNet-5 network architecture

### 2.5.3 GoogleNet

Szeged et al. participated in the study of Google's convolutional network [43]. Its prominent role is to increase the Inception block to replace the manual selection of the convolution type, and then stack the Inception module (deepen the depth) to form the Inception network. Remove the FC layer (remove most of the parameters), apply global average pooling, and reduce the amount of many parameters.

There are 22 layers in the network structure, and tables are used instead of structural diagrams. As shown in Table 2.1, it starts with three layers of ordinary convolution. Then it is divided into three groups of sub-networks, the first sub-network has 2 Inception blocks, the second group has 5 Inception blocks, and the third group has 2 Inception blocks. Next, connect to the global average pooling layer and the FC layer. It can be seen that although the FC layer is replaced with global mean pooling, the table shows that there is still an FC layer at the end, which is beneficial to fine-tune the network to different data sets.

**Table 2.1** LeNet-5 network architecture

| type | patch size/ stride | output size | depth | #1×1 | #3×3 reduce | #3×3 | #5×5 reduce | #5×5 | pool proj | params | ops |
|---|---|---|---|---|---|---|---|---|---|---|---|
| convolution | 7×7/2 | 112×112×64 | 1 | | | | | | | 2.7K | 34M |
| max pool | 3×3/2 | 56×56×64 | 0 | | | | | | | | |
| convolution | 3×3/1 | 56×56×192 | 2 | | 64 | 192 | | | | 112K | 360M |
| max pool | 3×3/2 | 28×28×192 | 0 | | | | | | | | |
| inception (3a) | | 28×28×256 | 2 | 64 | 96 | 128 | 16 | 32 | 32 | 159K | 128M |
| inception (3b) | | 28×28×480 | 2 | 128 | 128 | 192 | 32 | 96 | 64 | 380K | 304M |
| max pool | 3×3/2 | 14×14×480 | 0 | | | | | | | | |
| inception (4a) | | 14×14×512 | 2 | 192 | 96 | 208 | 16 | 48 | 64 | 364K | 73M |
| inception (4b) | | 14×14×512 | 2 | 160 | 112 | 224 | 24 | 64 | 64 | 437K | 88M |
| inception (4c) | | 14×14×512 | 2 | 128 | 128 | 256 | 24 | 64 | 64 | 463K | 100M |
| inception (4d) | | 14×14×528 | 2 | 112 | 144 | 288 | 32 | 64 | 64 | 580K | 119M |
| inception (4e) | | 14×14×832 | 2 | 256 | 160 | 320 | 32 | 128 | 128 | 840K | 170M |
| max pool | 3×3/2 | 7×7×832 | 0 | | | | | | | | |
| inception (5a) | | 7×7×832 | 2 | 256 | 160 | 320 | 32 | 128 | 128 | 1072K | 54M |
| inception (5b) | | 7×7×1024 | 2 | 384 | 192 | 384 | 48 | 128 | 128 | 1388K | 71M |
| avg pool | 7×7/1 | 1×1×1024 | 0 | | | | | | | | |
| dropout (40%) | | 1×1×1024 | 0 | | | | | | | | |
| linear | | 1×1×1000 | 1 | | | | | | | 1000K | 1M |
| softmax | | 1×1×1000 | 0 | | | | | | | | |

## 2.6 Mobile learning

### 2.6.1 Mobile learning research

In the early days of mobile learning, researchers did not have a unified definition of the concept of mobile learning. Although researchers define mobile learning from different perspectives, they all affirm the important role of mobile devices in mobile learning. The realization of mobile learning requires the help of mobile devices that can connect to the Internet. Teachers and learners can get rid of time and space. Obtain teaching resources anytime, anywhere. At the same time, European and American countries have launched research projects on mobile learning to carry out research on mobile learning from the macro and micro levels [7],[45].

With the advancement of a series of research projects, the theory of mobile learning has been further enriched, and researchers have a further understanding of mobile learning. Simply transforming traditional teaching resources into mobile learning resources cannot give full play to the advantages of mobile learning. Therefore, researchers gradually shifted their focus **from "technology" to "learning"** and began to study learning models suitable for mobile learning, combining mobile learning with physical classrooms, and enriching mobile learning theories. Through teaching practice, researchers generally believe that mobile learning has its natural advantages in terms of convenience and flexibility [46]. In addition to focusing on technical equipment and learning modes, researchers have also carried out related research on the mobile learning environment, focusing on human-computer interaction and improving the learning experience of learners, including mobile learning resource screens, mobile terminals, and subtitles. Learners have a significant impact on mobile learning activities.

Based on the current research on mobile learning, the author believes that mobile learning is a learning method that can carry out learning activities at any time and place without being restricted by time and space with the help of mobile devices in a mobile environment. Mobile learning theory provides design reference for the development of a mobile learning platform based on the WeChat mini program in this article. In the development process, the development work should be combined with the flexibility, interactivity, and diversity of mobile learning to make the functions of the mobile learning platform as complete as possible, meet the learning needs of learners, and enhance the learning experience of learners.

### 2.6.2 Development of WeChat Mini Program

In 2016, Zhang Xiaolong, the "father" of WeChat, announced the birth of the WeChat mini program at the opening ceremony. On January 9, 2017, the lightweight application "WeChat Mini Program" was officially launched [47]. The WeChat mini program embodies the characteristics of "at your fingertips and at any time." When using a mini-program, the user does not need to download or install it, just use the scan or search function provided by WeChat to find the corresponding mini program. With the continuous development of WeChat Mini Programs, it has integrated applications in many fields such as society, entertainment, government affairs, and public transportation. Applications that users do not use frequently and stay for a short time are particularly suitable for development with WeChat mini program. Users can use a certain function without downloading the application and without subsequent management.


**Fig 2.12** Concept of WeChat Mini Program

On January 9, 2020, Tencent held a WeChat open class and released a three-year performance table of the mini program in the open class. After three years of exploration and development, by A2019, the daily active users of Mini Programs exceeded 300 million, and the cumulative annual transaction volume exceeded 800 billion yuan. Compared with the previous year, the number of visits per capita and the number of uses per capita of Mini Programs both increased, increasing by 45% and 98% respectively [48]. At present, the application of WeChat Mini Programs in the education field includes well-known MOOC learning platforms and learning open forums.

### 2.6.3 Blended learning theory

After the development of the mobile learning platform is completed, it can be used as an auxiliary means of classroom teaching. Learners can use their spare time to preview or consolidate their knowledge on the mobile learning platform, discuss difficulties and doubts encountered in learning with teachers and other learners on the platform, and improve learning efficiency. In addition, teachers can also use the mobile learning platform to carry out practical activities in a variety of teaching forms, such as flipped classrooms.

## 2.7 Chapter Summary

This chapter first introduces the commonly used image classification algorithm and status. Then it introduces the concept of Deep Learning and related content of CNN. It also briefly introduces the calligraphy evaluation and classification method based on CNN. Finally, it briefly introduces the popular "mobile learning concept" and the history of WeChat mini-programs.

# 3 CALLIGRAPHY CLASSIFICATION BASED ON R-CNN

Traditional object detection and classification are based on image recognition. But then an algorithm that applies deep learning to object detection appeared. Ross Girshick et al. proposed R-CNN (Region with CNN features) in a conference paper in CVPR in 2014 [49], which combined the region proposal and CNN convolution features. Up for target detection. This chapter first describes the network structure of R-CNN and introduces our **databases**: One part of the calligraphy picture data comes from the Calligraphy Association of Fudan University, and the other part comes from the Calligraphy Association of Shanghai University of Finance and Economics. The main two italics calligraphy structures are "Heng" and "Pie." There are more than 20,000 original data, divided into six categories.

## 3.1 CNN & R-CNN

CNN architecture has a convolutional layer, FC layer, etc., and after using the softmax multi-class classifier and cross-entropy loss function operation, the feature dimension of the image has dropped to a very low state, and the fully connected layer is used to integrate all the local information after feature synthesis, And then get the global information, and output a feature vector.

R-CNN is the first algorithm to apply deep learning to target detection smoothly. It is usually possible to use the exhaustive method to select the area frame where all objects may exist in the image, obtain the features from it, and use the image discrimination method to classify. After obtaining all the effective areas of the classification, use the non-maximum value to suppress the resulting output.

According to the traditional idea of target detection, R-CNN still uses the extraction frame to obtain features, image classification, and non-maximum suppression to implement target detection [25]. Only when obtaining the features, the traditional features (such as SIFT, HOG, etc.) are replaced by the features obtained by the deep convolutional network. The overall framework of R-CNN is shown in Figure 3.1:

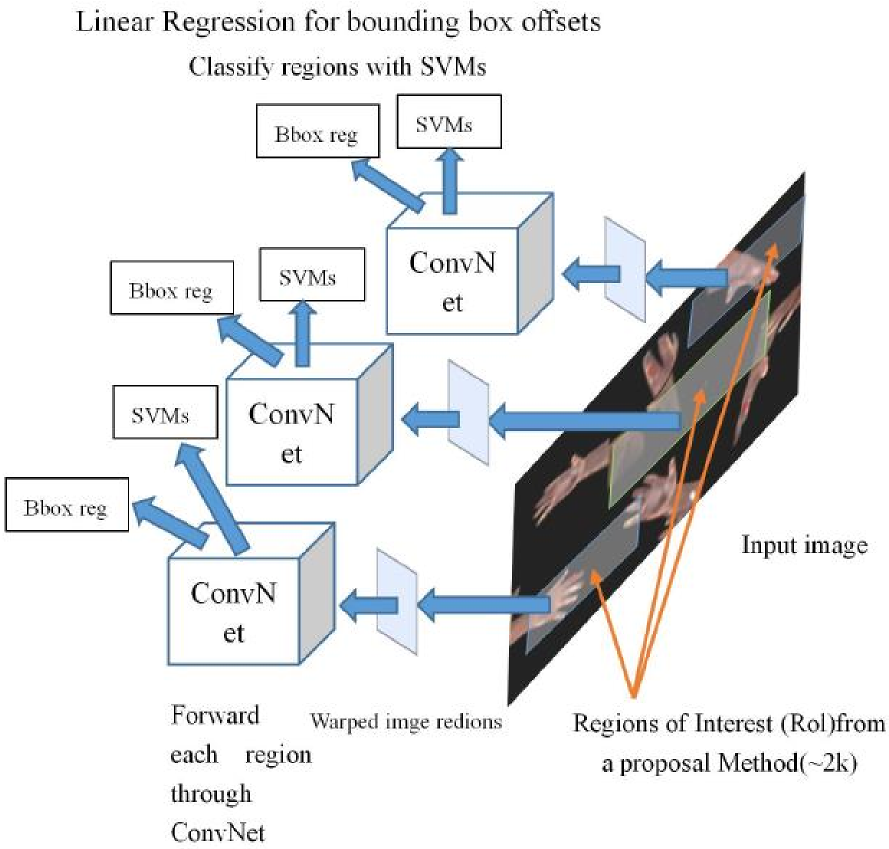



**Fig 3.1** Framework of R-CNN

Take a picture, R-CNN uses the selective search method to estimate that the number of candidate regions is 2000. After that, the candidate frame is resized to the same size and added to the CNN model to finally obtain the feature vector. Next, it will operate in the multi-class SVM classifier, and the candidate area is predicted, including the probability value of each type of object. Each class trains a single SVM, and the probability of belonging to this class in the feature vector can be derived. In order to improve the positioning accuracy, R-CNN finally trained a bounding box regression model to correct the precise position of the frame.

## 3.2 Network implementation

### 3.2.1 Training process

(1) Obtain the **region proposal** in the input picture, and use the CNN network to obtain the features for each candidate area (the size of the proposal is reduced to the same 227×227 to match the CNN network). Prepare the region proposal. To obtain 2000

region proposals from all training images, the method used is selective search.

(2) **Positive samples and negative samples.** Assume that one of the region proposals is compared with all the ground truth in the current image. When the IoU with the largest repetition rate of the two is greater than or equal to 0.5, the ground truth is the positive sample of its region proposal, and if the IoU is less than or equal to 0.5, it is a negative sample.

(3) **Pre-training:** The key to this step is that because there are fewer labeled samples in the detection problem, large-scale training is difficult. The learning model is AlexNet, which includes 5 convolutional layers and 2 FC layers. In the Keras framework, 60% of the data set is used to train in advance, and the classifier is trained on this data set.

(4) **Fine-tuning:** This process is shown in Figure 3.2. Because the size of the region proposal obtained in the second step in Figure 3.3 is different, we need to transform the size to a consistent size state, so we need to transform the region proposal to 227 ×227. And directly expand the region proposal of any size and aspect ratio to the specified size. Then the pre-trained network in the third step in Figure 3.3 is used as input, and then training.

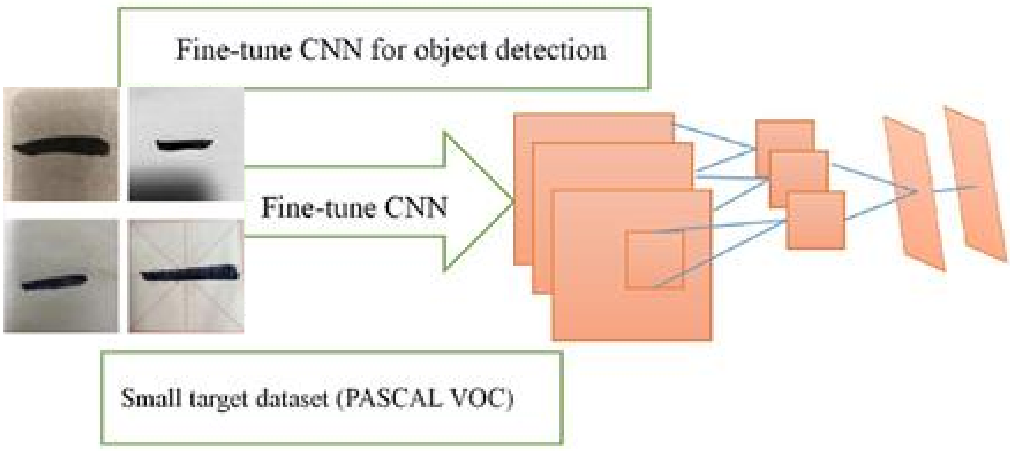



**Fig 3.2** Fine-tuning process

(5) Train different SVM classifiers for different types.

(6) **Regression.** The training process uses the feature 6×6×256 dimension of pool5 and the ground truth of the area frame to train each type of regression separately.

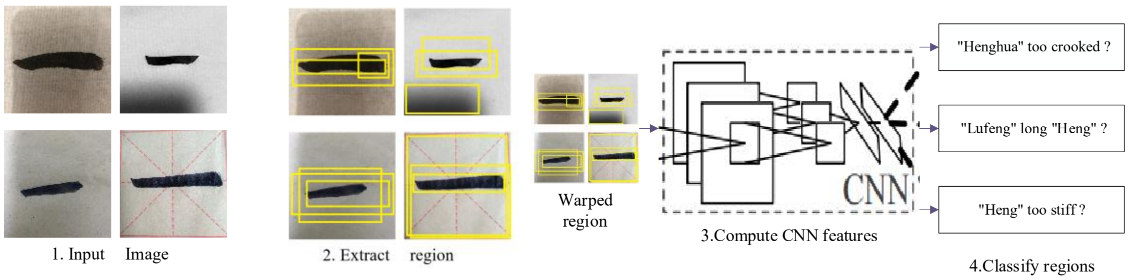



**Fig 3.3** Training process

### 3.2.2 Testing process

(1) Image input, from which 2000 region proposals are obtained, and the selective search method is used for them.

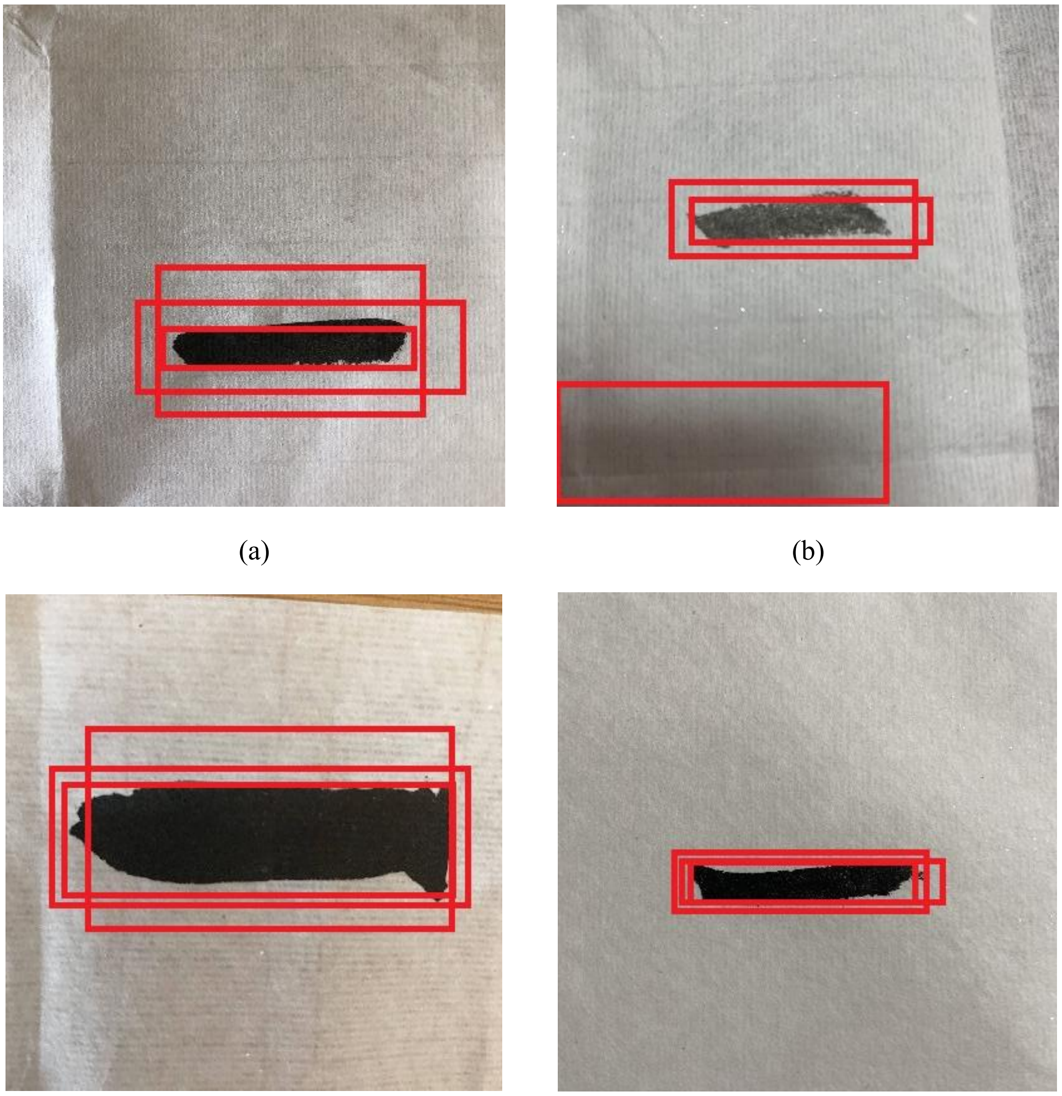
(2) Transform all the region proposals so that all of them are transformed into a uniform size, and what can be input is the trained CNN model.

(3) In each classification result, use the type trained by the SVM classifier to score the obtained features, and obtain the region proposal in each column of the score matrix, and it needs to be deleted. The deletion operation uses the Non-maximum suppression (NMS) method. The final result of this method is to leave some of the highest scores and delete some of them with a high overlap area.

(4) The region proposal obtained by the regression uses the features of the pool5 layer. After a series of training and testing processes, R-CNN finally completed the classification of the pictures.

### 3.2.3 Related parameters

In the R-CNN test process in the previous section, after step 4 is completed, the score of a certain object category is obtained, and a situation will be encountered at this time. As an example, as shown in Figure 3.4, the range of calligraphy text may be surrounded by multiple region proposals. In this state, non-maximum suppression operations must be used to reduce overlapping areas and remove regions with lower scores.



(a) (b) (c) (d)

**Fig 3.4** Calligraphy image scoring

**Non-Maximum Suppression (NMS):** Search for the score value in the result, interpret the high value in the region proposal, and then the deleted region proposal meets the condition that it exceeds the maximum value with the highest score IOU. After deleting, continue to check the score value, still repeat the steps of the previous step, and continue the deletion operation, until the end, the condition that is satisfied is that there are no remaining region proposals [50]. Do this for all the columns so that you can sort after obtaining a part of the region proposal in each category at the end. What needs to be done is to sort and let the score decide. Before using the score, you need to know the size of each bounding box, and select the bounding box with the largest score for calculation, calculate its IoU, and remove the bounding box with IoU greater than the specified value. Until the candidate area is 0, stop the repeated steps. Next, continue to repeat the above steps for the next category, and the candidate box deletion condition is that the score is less than a certain value. The stopping condition is that no candidate frame of this type exists before the next classification can be continued.

The design parameters of the R-CNN regression box are shown in Figure 3.5.

- The **yellow** window P represents Region Proposal.
- The **green** window G represents Ground Truth.
- The **red** window $\hat{G}$ represents the prediction of Region Proposal after regression.

The current goal is to find the P to $\hat{G}$ Linear transformation (When the IoU of Region Proposal and Ground Truth is greater than 0.6, it can be regarded as a linear transformation), making $\hat{G}$ and G closer.

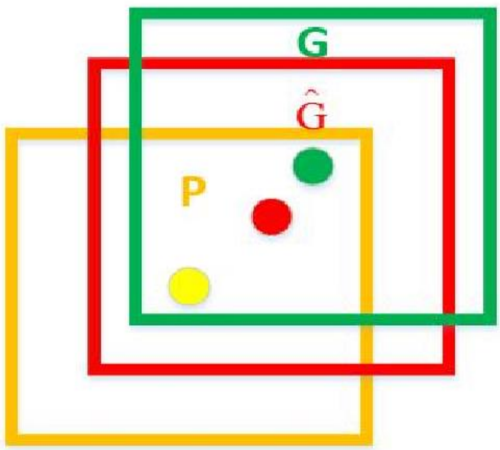



**Fig 3.5** Non-maximum suppression process

First, define the mathematical Expression of the P window as $P^i = (P^i{}_x, P^i{}_y, P^i{}_w, P^i{}_h)$, where $(P^i{}_x, P^i{}_y)$ represents the center point coordinates of the $i$ -th window, $(P^i{}_w, P^i{}_h)$ are the width and height of the $i$ -th window, respectively. The mathematical Expression of the window G as $G^i = (G^i{}_x, G^i{}_y, G^i{}_w, G^i{}_h)$ . The mathematical Expression of $\hat{G}^i$ the window is $\hat{G}^i = (\hat{G}^i{}_x, \hat{G}^i{}_y, \hat{G}^i{}_w, \hat{G}^i{}_h)$ , the superscript $i$ is omitted below.

Four transformation functions are defined here [ $d_x(P), d_y(P), d_w(P), d_h(P)$ ]. $d_x(P), d_y(P)$ Change x and y through translation, and $d_w(P), d_h(P)$ change w and h through zooming, which is expressed according to the following four formulas.

$$\hat{G}_x = P_w \cdot d_x(P) + P_x \tag{3.1}$$

$$\hat{G}_y = P_w \cdot d_y(P) + P_y \tag{3.2}$$

$$\hat{G}_w = P_w \cdot e^{d_w(P)} \tag{3.3}$$

$$\hat{G}_h = P_h \cdot e^{d_h(P)} \tag{3.4}$$

## 3.3 Network Architecture

- **AlexNet**

The AlexNet architecture appeared in the 2012 Imagenet recognition competition [51]. The network architecture was proposed by Alext and others, making CNN the core algorithm for image processing. Although LeNet-5 that appeared in the last century, is a classic, it can only be applied in certain fields because of the limitations of complex real-world scenarios. However, due to the rapid development of features such as SVM, LeNet-5 does not actually have many applications. According to the proposal of ReLU and dropout, and the big data that broke out in the GPU Internet era, CNN has made significant progress, making deep learning the forefront of artificial intelligence is the AlexNet architecture. Figure 3.6 shows the network architecture of AlexNet:

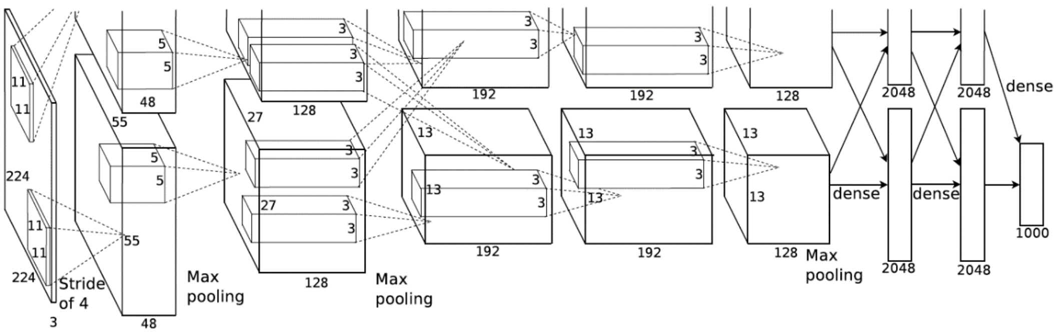



**Fig 3.6** AlexNet simple network architecture

AlexNet has a total of 8 layers of architecture, the first 5 are convolutional layers, the first two convolutional layers and the fifth convolutional layer have pooling layers, and the rest do not. The final three layers are the FC layer, with an estimated 650,000 neurons and about 60 million training parameters.

- **VGG**

The text classification framework proposed by Oxford Visual Geometry Group in 2014 is VGG [26]. Compared with the original architecture, this architecture is more about widening and deepening the network architecture. The focus is on five groups of convolutions, and the Max Pooling space between each group is reduced. The number of consecutive convolutions used multiple times in the same group is 3×3, the number of convolution kernels is increased from relatively shallow 64 to 512 relatively deep, and the number of convolution kernels in the same group is the same. After convolution, add two FC layers, and then the classification layer. Table 3.1 is the network structure of VGG Net.

**Table 3.1** Architecture of VGG Net

| ConvNet Configuration | | | | | |
|---|---|---|---|---|---|
| A | A-LRN | B | C | D | E |
| 11 weight layers | 11 weight layers | 13 weight layers | 16 weight layers | 16 weight layers | 19 weight layers |
| input (224 × 224 RGB image) | | | | | |
| conv3-64 | conv3-64<br>**LRN** | conv3-64<br>**conv3-64** | conv3-64<br>conv3-64 | conv3-64<br>conv3-64 | conv3-64<br>conv3-64 |
| maxpool | | | | | |
| conv3-128 | conv3-128 | conv3-128<br>**conv3-128** | conv3-128<br>conv3-128 | conv3-128<br>conv3-128 | conv3-128<br>conv3-128 |
| maxpool | | | | | |
| conv3-256<br>conv3-256 | conv3-256<br>conv3-256 | conv3-256<br>conv3-256 | conv3-256<br>conv3-256<br>**conv1-256** | conv3-256<br>conv3-256<br>**conv3-256** | conv3-256<br>conv3-256<br>conv3-256<br>**conv3-256** |
| maxpool | | | | | |
| conv3-512<br>conv3-512 | conv3-512<br>conv3-512 | conv3-512<br>conv3-512 | conv3-512<br>conv3-512<br>**conv1-512** | conv3-512<br>conv3-512<br>**conv3-512** | conv3-512<br>conv3-512<br>conv3-512<br>**conv3-512** |
| maxpool | | | | | |
| conv3-512<br>conv3-512 | conv3-512<br>conv3-512 | conv3-512<br>conv3-512 | conv3-512<br>conv3-512<br>**conv1-512** | conv3-512<br>conv3-512<br>**conv3-512** | conv3-512<br>conv3-512<br>conv3-512<br>**conv3-512** |
| maxpool | | | | | |
| FC-4096 | | | | | |
| FC-4096 | | | | | |
| FC-1000 | | | | | |
| soft-max | | | | | |

The structure of the VGG model is relatively simple. The key is to show that increasing the network depth can improve performance. Its most eye-catching part is that it has a very unified architecture and can only perform 3×3 convolution and 2×2 pooling from start to finish. The disadvantage is that the amount of calculation is huge because it needs to use a lot of memory and parameters (140M). There are 11, 13, 16, 19 layers of models. Among them, column D in Table 3.1 is commonly used VGG16, and column E is VGG19.

The biggest feature of the VGG Net architecture is the use of a **small 3×3 convolution kernel** instead of a 5×5 or 7×7 convolution kernel. The reasons are as follows:

(1) The minimum size of 3×3 is to capture eight neighborhood information of pixels.
(2) The finite step size of two 3×3 stacked convolutional layers is 5×5. The step size of the three 3×3 stacked convolutional layers is 7×7, so a stack of small convolutional layers can be used to replace the large convolutional layer, and the step size does not change.
(3) Compared with a large convolutional layer, multiple 3×3 convolutional layers have many Non-linear relationships, making the function more judgmental.
(4) There are fewer parameters for multiple 3×3 convolutional layers.

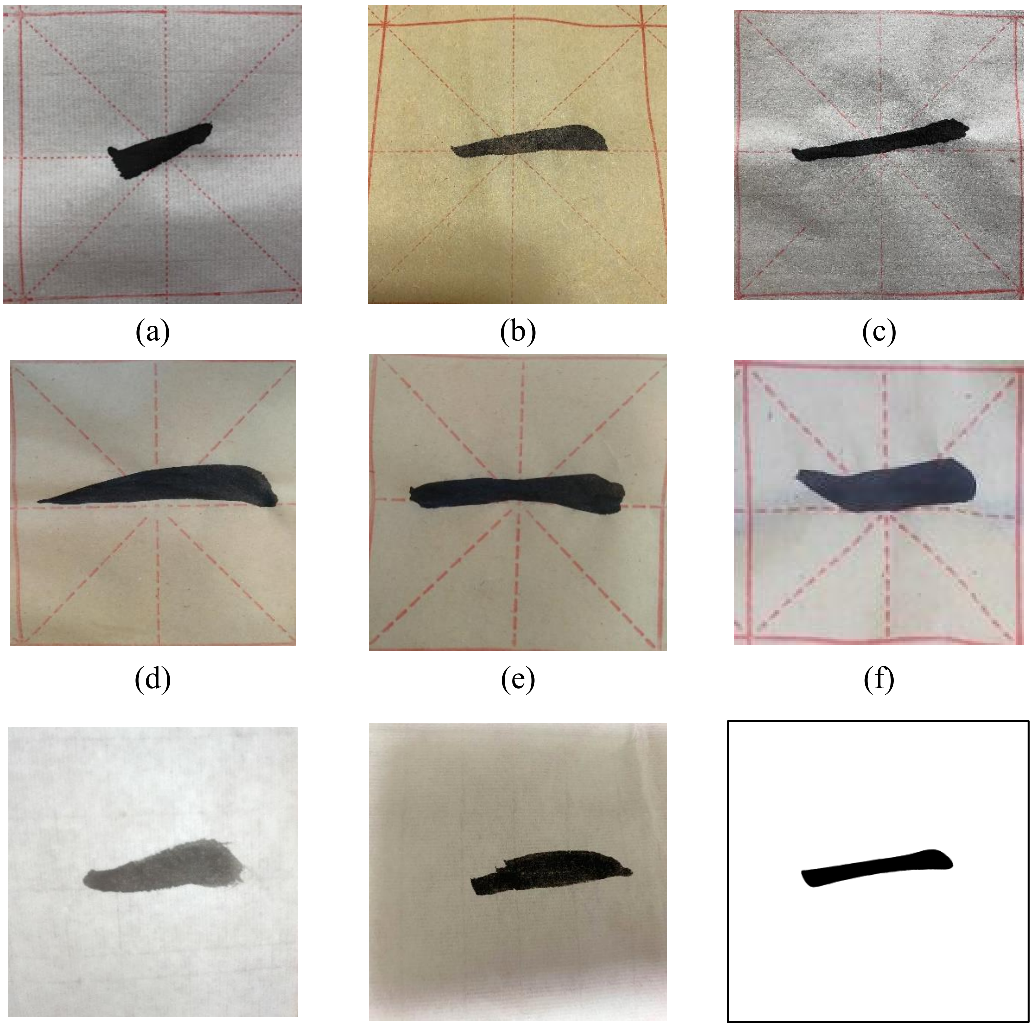
## 3.4 Calligraphy data analysis

### 3.4.1 Private Dataset

Looking at the entire calligraphy industry, no company or institution has collected calligraphic text image data. The main purpose of this article is to study the feasibility of automatic calligraphy correction. Unlike in the medical field, there is sufficient data available for training and testing. So this article uses data collected by the author himself, one part of the calligraphy picture data comes from the Calligraphy Association of Fudan University, and the other part comes from the Calligraphy Association of Shanghai University of Finance and Economics. The main two italics calligraphy structures are "Heng" and "Pie." **Part of the data has been uploaded to Github, and the link is:**

https://github.com/Jay-Z-zhang/Calligraphy/tree/master/data

The data used in the training and testing process of this article contains 30,000 handwritten calligraphy images, and there are eight types of tagged high-resolution images. Using our private data set, a convolutional neural network model was trained to divide more than 20,000 high-resolution images into six categories.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

**Fig 3.7** Examples of data set (a) "Henghua" too crooked (b) Half split head and tail cross (d) Standard structure (d) "Lufeng" long "Heng" (e) "Heng" too stiff (f) "Xingbi partial peak" (g) "Henghua" too short (h) Thickness varies unevenly (i) Standard structure

Figure 3.7 is an example of our data set of vehicle types, which is suitable for all types of identification tasks of stroke "Heng."The training data set has six models, including standard structure, "Xingbi partial peak," "HuiFeng not in place," "Qifeng bloated," "horizontal drawing too long," and "horizontal drawing too short." The number of pictures of the corresponding types of models is 7822, 5558, 4922, 2883, 1476, 1392, among which the picture sizes are 1600×1200 and 1920×1080. The data set has rich image information, including pictures with different font sizes, different colors, different lighting conditions, and shooting angles. Using it to train a convolutional neural network model can well improve the generalization ability of the network.

### 3.4.2 Data Augmentation

Convolutional neural networks contain a large number of parameters [52]. To train a large-scale network model, a large amount of sample data is needed; otherwise, it will cause overfitting due to insufficient data. The data set used in this article is not very large, so it needs to be expanded. The data set can be expanded by acquiring new data, but this method needs to collect data and label the acquired new data, which is cumbersome and costs too much. Data Augmentation techniques are commonly used to expand data sets. There are many data enhancement methods, such as flipping, cropping, rotating, zooming, translation, and adding noise. These four data enhancement methods are shown in Figure 3.8 to Figure 3.11. Among them, Figure 3.8 is angle rotation, Figure 3.9 is horizontal flip, Figure 3.10 is random cropping, and Figure 3.11 is adding Gaussian noise.

The specific operations for data enhancement are as follows:

**Angle rotation:** Rotate the original picture every 15 degrees to get a new picture. In Figure 3.8, the two pictures on the left are original images, and the two pictures on the right are examples of new pictures obtained after angular rotation.

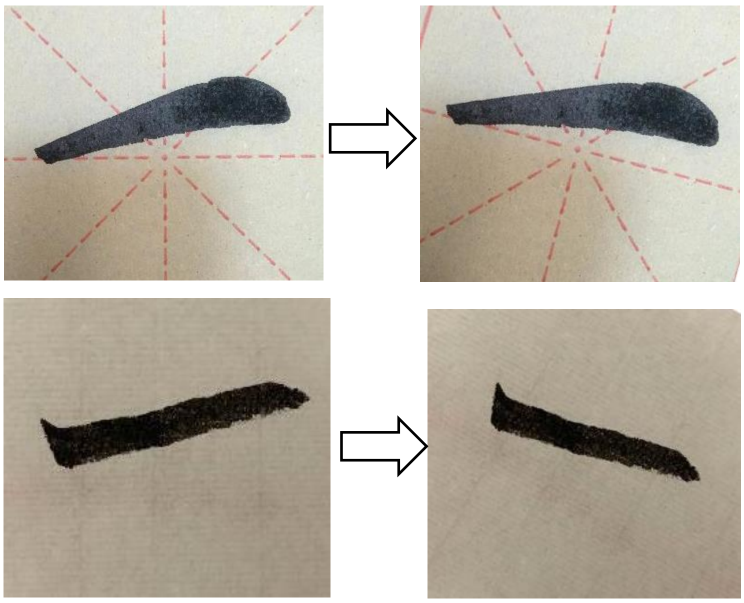


**Fig 3.8** Examples of angle rotation

**Flip horizontally:** Flip the original picture horizontally to get a new picture. In Figure 3.9, the two pictures on the left are original images, and the two pictures on the right are examples of new pictures obtained after horizontal flipping.

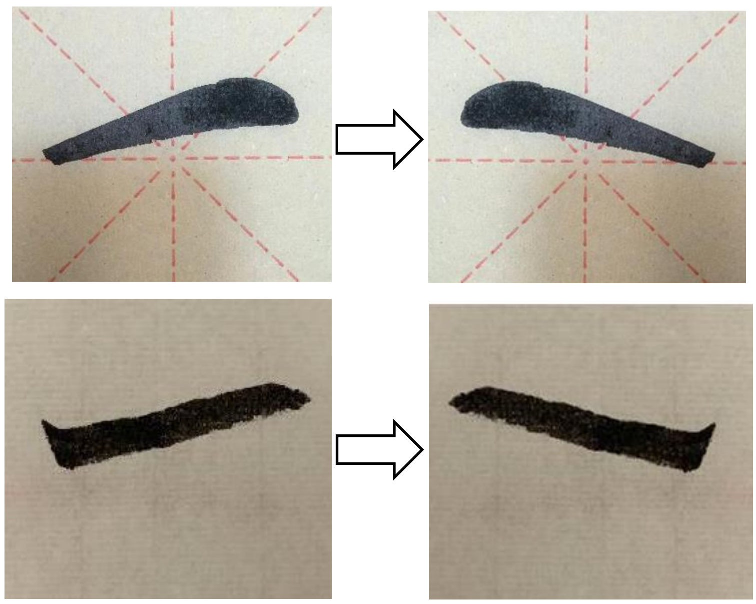


**Fig 3.9** Examples of angle rotation

**Random cropping:** Randomly crop the original picture 80% to get a new picture. In Figure 3.10, the two pictures on the left are original images, and the two pictures on the right are examples of new pictures obtained after random cropping.

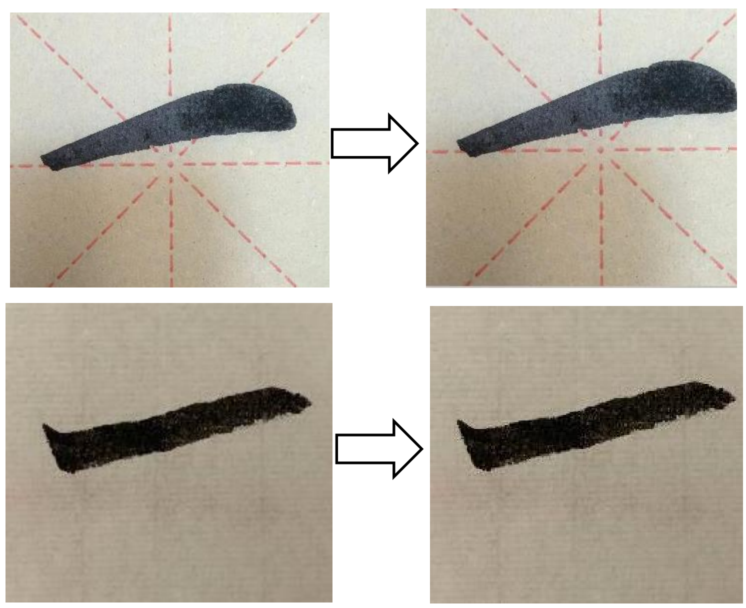


**Fig 3.10** Examples of random cropping

**Increase Gaussian noise:** Add Gaussian noise to the original picture to get a new picture. In Figure 3.11, the two pictures on the left are original images, and the two pictures on the right are examples of new pictures obtained by adding Gaussian noise.

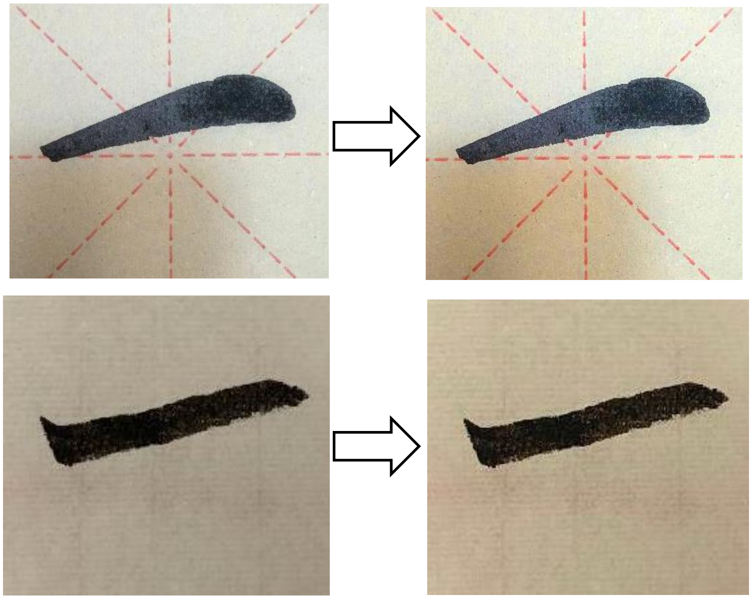


**Fig 3.11** Examples of adding Gaussian noise

Using the above four data set enhancement methods, the training data is generated in batches from the original limited data set through the Python program to expand the data set. After the original data is enhanced, it can reduce the over-fitting of the neural network and improve the recognition effect of the CNN in this experiment on the calligraphy image of the brush.

### 3.4.3 Data divide

Before the data Augmentation, all data is reduced to 64×64 size images. Since the width and height of the original images are different, the vehicle targets in the reduced data set will be deformed. Use this data set to convolutional neural network Model training will make the model have a certain resistance to deformation. In addition, the data set needs to be divided. This article divides the data set of each type of vehicle into a training set, a validation set, and a test set at a ratio of **3:1:1.**

After the data is divided, the training set data needs to be expanded. This article expands the data by flipping and cutting operations. First, the training data set is expanded to 2 times by horizontal mirroring, and then the expanded sample data is randomly cropped to 54×54, and then adjusted to 64×64. Each image is randomly cropped four times, so the training data set will be expanded to 10 times. The distribution of the number of pictures of various types of vehicle models in the data set is shown in Table 3.2.

**Table 3.2** Data distribution

| Type | Original | Expansion | Training | Testing |
|---|---|---|---|---|
| Standard | 7822 | 39110 | 4693 | 1564 |
| "Xingbi partial peak" | 5558 | 27790 | 3335 | 1111 |
| "HuiFeng not in place." | 4922 | 24610 | 2953 | 984 |
| "Qifeng bloated" | 2883 | 14415 | 1730 | 577 |
| "Horizontal draw too long." | 1476 | 7380 | 887 | 295 |
| "Horizontal draw too short." | 1392 | 6960 | 835 | 278 |

### 3.4.4 Normalization

After the calligraphy data is prepared, the data cannot be directly inputted into the convolutional neural network for training, and the data needs to be standardized; that is, the input data is restricted to a certain range while eliminating the differences between samples.

$$x' = \frac{x - \overline{x}}{\sigma} \tag{3.5}$$

The specific operation is to subtract the mean value of the image elements, and then divide by the standard deviation. The network models of all experiments in this article are built with the **Keras** deep learning framework. Keras is simple and easy to use. The documentation is complete, and the content of the documentation is well organized. Good scalability, support Tensorflow, theano, mxnet, cntk. And the custom layer is also easy to write. The user population is very wide, and it is easy to find colleagues who use Keras to communicate.

## 3.5 Result analysis

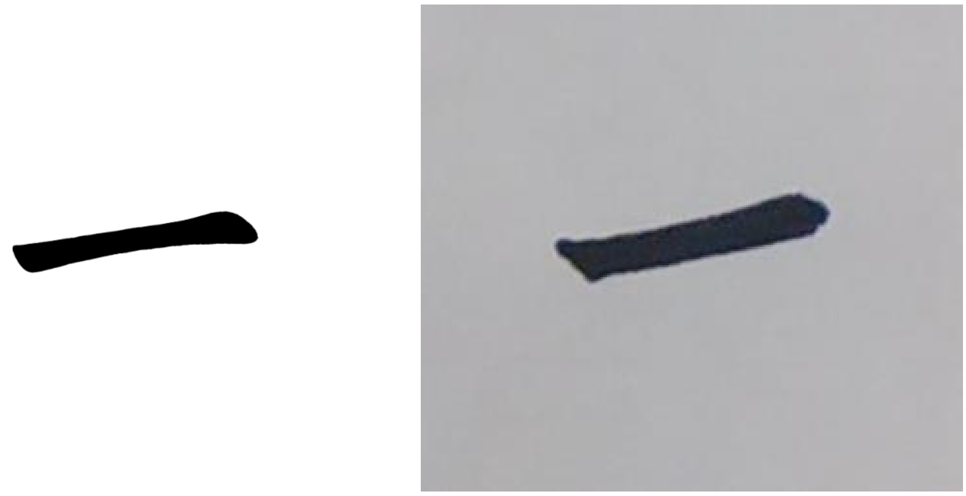
### 3.5.1 Comparative analysis results

Let the structure used be applied to the test data, and the final result is shown in Table 3.3. Analysis of the error rate and the Top2 error rate (the probability of not including the accurate class in the first or first two classes predicted), the result is not evenly distributed, so it has reference value.

**Table 3.3** CNN test error results

| NO. | Model | TOP-1 | TOP-2 |
|---|---|---|---|
| 1 | AlexNet | 0.38623 | 0.13262 |
| 2 | | 0.34627 | 0.14639 |
| 3 | VGG-16 | 0.33124 | 0.13132 |
| 4 | | 0.33539 | 0.10280 |
| 5 | R-CNN | 0.27361 | 0.0831 |
| 6 | | 0.22415 | 0.0966 |

This article begins to use an undisclosed data set, input writing brush calligraphy images to train the R-CNN network architecture, and then use the trained network classification test. The classification result is shown in Figure 3.12. The picture on the left is the standard "Heng" structure image, and the right is the image with defects in calligraphy, which is called "Henghua too long" in the professional field.



(a) (b)

**Fig 3.12** Testing result (a) Standard (b) “Henghua” too long

### 3.5.2 Summary of results

To sum up, the **advantages of R-CNN**, use the selective search method to improve the screening speed of the candidate domain (compared to the traditional method). The problem of insufficient labeled data in the target detection training process is solved, and more accurate positioning is obtained.

The **shortcomings of R-CNN** are redundant calculations. Because of the need to zoom operation of the candidate area, the target will be deformed. It is not suitable for the analysis of calligraphic images. But its effectiveness in image classification has given people a lot of inspiration.

## 3.6 Chapter Summary

This chapter first introduces the related knowledge of the R-CNN network and the data sources used. Then use the trained R-CNN model to process the original calligraphy image, and draw a conclusion through the analysis of the experimental results. It is not suitable for the analysis of calligraphy image, but the effect of image classification has given people a lot of inspiration.

# 4 CALLIGRAPHY CLASSIFICATION BASED ON ResNet

## 4.1 Calligraphy image classification based on ResNet

### 4.1.1 Network architecture

The texture characteristics of brush calligraphy are quite complicated, and the characteristics of manual marking need to be completed by experienced calligraphy teachers, and everyone needs to use the same set of judgment standards. The most noteworthy point is that the features of manual marking cannot accurately display the text structure area, so the generalization ability is poor. With the popularity of deep learning, it is used in computer vision [52].

ResNet (Residual Network) won the gold medal in ImageNet image classification, image object localization, and image object detection competitions in 2015 and is well-known. Due to the increase in the depth of network training and the decrease in accuracy, ResNet reduces the difficulty of training deep networks through the learning of residual modules. On the basis of the original idea (BN, small convolution kernel, full convolution network), add Residual Block. Each Residual Block includes two paths, one of which is a direct connection path of the input feature, and the remaining one performs 2 to 3 convolution operations on the feature to obtain the residual of this feature, and finally adds the features on the two paths. The Residual Block architecture is shown in Figure 4.1:

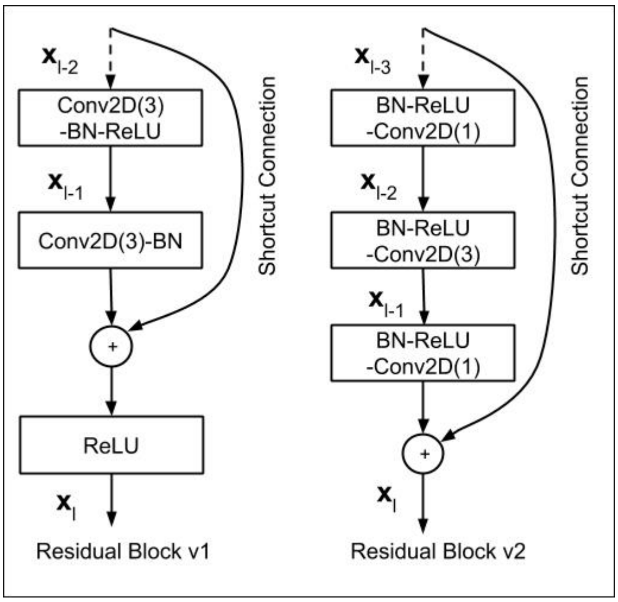



**Fig 4.1** Residual Block structure

ResNet training has the concept of rapid convergence, and successfully trained hundreds or even thousands of CNN. Aiming at the actual problem of brush calligraphy judgment, this paper constructs a recognition model based on the ResNet-50 network, whether the brush calligraphy characters are standard. The architecture of this model is shown in Figure 4.2. The numbers on the convolutional layer and residual block in the figure indicate the output quantity of the feature map.

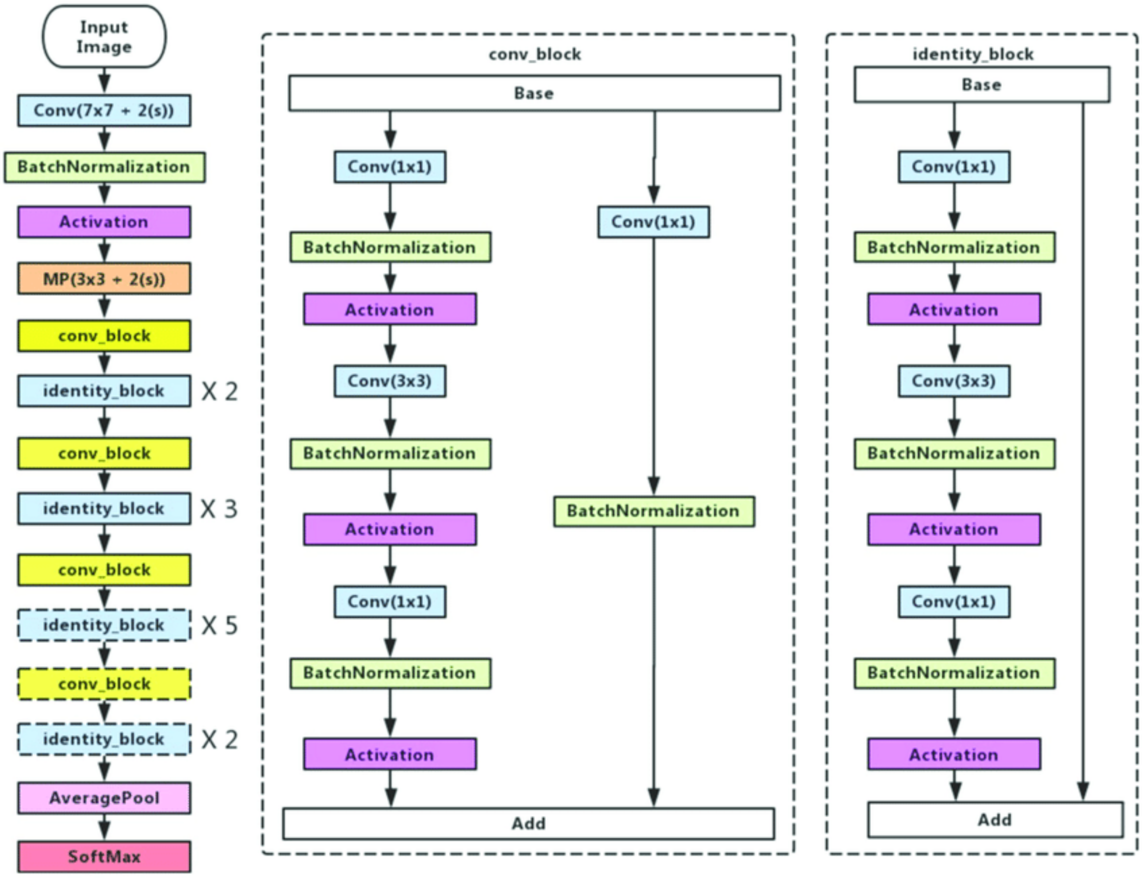



**Fig 4.2** Architecture of ResNet-50

The specific structure of the network is introduced: the calligraphy image is firstly enhanced by data (an increase of noise, mirroring, rotation), and then reaches the input layer of the network. The three pictures reach the first convolutional layer from the input layer, and the size of the convolution kernel is 7×7. Sixty-four feature maps; after the 3×3 maximum pooling layer, 64 feature maps are obtained. The entire network is divided into four residual modules after the first convolutional layer. After the operation of the four residual modules, the output 2048 feature maps; finally, through the average pooling layer and the FC layer, through softmax, the classification result is obtained. The calligraphy judge, the overall structure of the process, is shown in Figure 4.3.

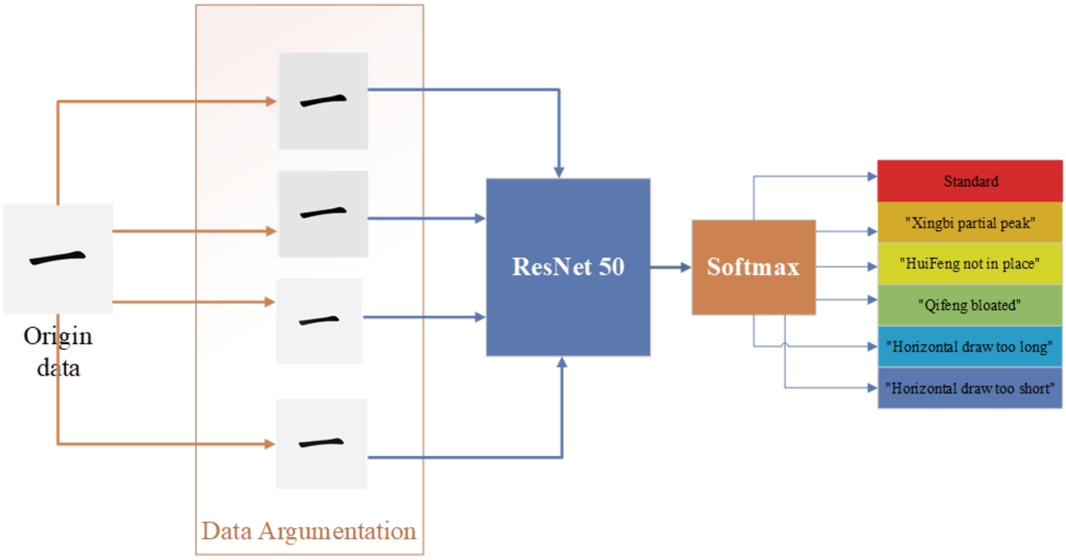



**Fig 4.3** Calligraphy standards judge the overall structure of the network

Because the network used in the data set in this article is ResNet50, ResNet50 is briefly introduced. The main point of ResNet50 is that because of their high usage rate, and special instruction is needed. Its specific structure is given, as shown in Table 4.1:

**Table 4.1** ResNet-50

| layer name | output size | 18-layer | 34-layer | 50-layer | 101-layer | 152-layer |
|---|---|---|---|---|---|---|
| conv1 | 112×112 | 7×7, 64, stride 2 | | | | |
| | | 3×3 max pool, stride 2 | | | | |
| conv2_x | 56×56 | $\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$ | $\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$ | $\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$ | $\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$ | $\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$ |
| conv3_x | 28×28 | $\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$ | $\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$ | $\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$ | $\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$ | $\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$ |
| conv4_x | 14×14 | $\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$ | $\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$ | $\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$ | $\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$ | $\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$ |
| conv5_x | 7×7 | $\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$ | $\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$ | $\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$ | $\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$ | $\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$ |
| | 1×1 | average pool, 1000-d fc, softmax | | | | |
| FLOPs | | $1.8\times10^9$ | $3.6\times10^9$ | $3.8\times10^9$ | $7.6\times10^9$ | $11.3\times10^9$ |

Table 4.1 is the architecture of ResNet with different depths, which explains five kinds of depths of ResNet. Regardless of the depth of the ResNet network, there are five stages, namely: conv1, conv2_x, conv3_x, conv4_x, conv5_x. For example: in the 101-layer, "-layer" represents how many layers of the network, the number of input convolutions should be 7×7×64 at the beginning, but then it needs to go through a complicated process, which requires 33 building blocks, the formula It is 3+4+23+3=33. What needs to be known is that the number of channels of the block is 3. Therefore, the

first calculation result of the network is 99 layers, and the formula is 33x3=99. However, the point worth noting is FC layers. It needs to add one more, so the formula is 1+99+1=101, and the output is 101 layers.

### 4.1.2 Pre-activation and projection

After the data passes through the first single convolutional layer, before entering the first Residual Unit, use an activation function to process it, and then go down. And after the data passes through the last Residual Unit and enters the average pooling layer, additional activation functions are used for processing.

For deep networks of 50 layers and above that use Bottleneck Design, pre-activation must be done first, and last need projection.

### 4.1.3 Residual Block

The two structures of the Residual Block are shown in Figure 4.1. The "curved arc" on the right side of the structure is the "shortcut connection." When it comes to identity mapping, it also explains the core of ResNet. Shortcut here means a cross-layer connection. A path from x to y is set up in the Highway Network. The ResNet shortcut has no weight, and the network is stable and easy to learn. Prove that as the network depth increases, the performance will gradually get better.

### 4.1.3 Loss Function

In the classification, extensive use of cross-entropy loss function is, the distance between two probability distributions, specification two probability distributions p, q, q let p describe the cross-entropy is:

$$H(p,q) = -\sum p(x) \cdot \log q(x) \tag{4.1}$$

Note: Cross entropy describes the distance between two probability distributions, but the output neural network may not be a probability distribution. The probability distribution characterizes the probability of occurrence of different events. When the total number of things is limited, the probability distribution function $p(X = x)$ is satisfied:

$$\forall x, p(X=x) \in [0,1] \quad (\sum_i p(X=x_i)=1) \tag{4.2}$$

It can be described as that the probability of anything happening is [0, 1], and the sum of the probability of all things happening is 1. Assuming that "a single sample belongs to a certain category" in the classification is regarded as a single probability event, the accurate result of the training data is called a single probability distribution. Since the probability of "a sample is classified as the error type" is 0, the probability of "a sample is classified as the correct type" is 1.

## 4.2 Training and testing result

### 4.2.1 Software and hardware environment

In the experiments in this article, a desktop computer is used as the experimental environment, the CPU is AMD Ryzen 5 1500x, the main memory is 16G, the GPU is NVidia GeForce GTX1060 Ti, and the GPU memory is 6G. Keras is the deep learning framework used, and the programming language uses Python. In order to facilitate the download and management of Python software packages, the software framework Keras and the programming language Python use Anaconda for installation and configuration. Anaconda can be used to switch between multiple versions of Python without conflict randomly.

### 4.2.2 Experimental results

First, classify and label calligraphy images (calligraphy expert operation). The data set used for skin images includes 24053 images. Then 14432 images are used as training set samples and 4810 as test set samples. Before being sent to ResNet, the image is preprocessed and compressed into 64×64 pixels. Then the image is processed with data enhancement. Define the test accuracy rate as:

$$ACC = \frac{\text{Top-1 predicts the correct number}}{\text{Total number of test images}} \tag{4.3}$$

Set the epoch as 10. The following Table 4.2 is a good three-round training loss, and test accuracy based on the ResNet model experiment:

**Table 4.2** Training Results

| No. | Epoch | Train-loss | Accuracy | Training time(min) |
|---|---|---|---|---|
| 1 | 20 | 0.35347 | 0.7968 | 39.5 |
| 2 | 10 | 0.36141 | 0.7654 | 17.2 |
| 3 | 10 | 0.35094 | 0.7349 | 19.7 |
| 4 | 5 | 0.34622 | 0.7245 | 7.2 |
| 5 | 5 | 0.28415 | 0.7141 | 48.5 |

Through the experimental data, it is found that because the used data set itself has good imaging quality, clear outlines, and obvious features, coupled with image preprocessing and data enhancement, the accuracy rate is above 70%. This result is good for calligraphy practice, and it has certain reference value. Change the learning rate; we can get the following figure:

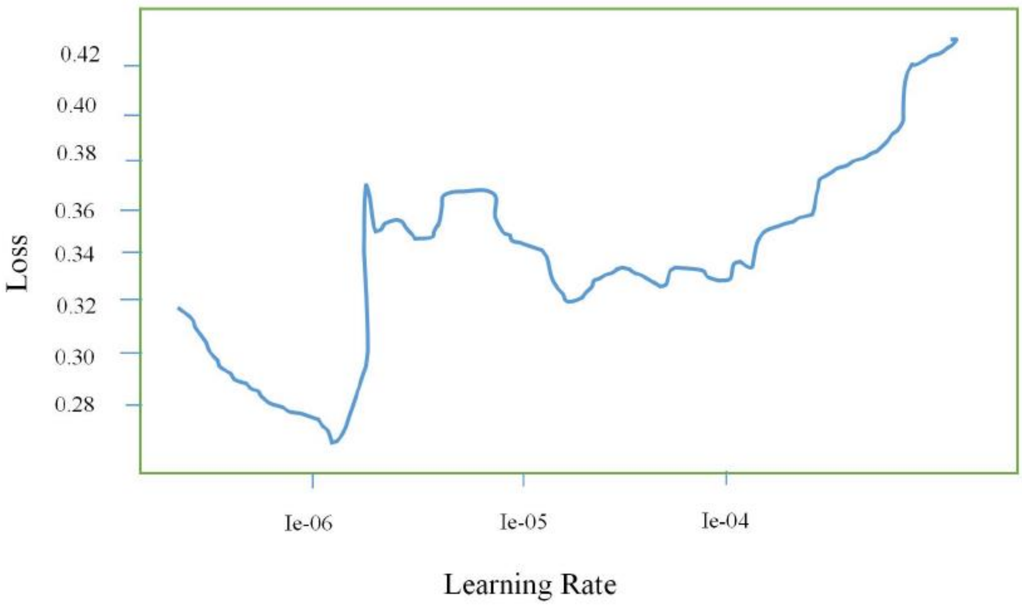



**Fig 4.4** Loss function under different learning rate

In the experiment, this article tries to experiment with different learning rates, and the obtained loss value is combined into the curve shown in Figure 4.4. It is not uncommon to draw a conclusion: Appropriately increasing the learning rate can reduce the loss value, but if you continue to increase the learning rate, the loss value will increase rapidly.

## 4.3 Chapter Summary

This chapter first introduces the ResNet network and further analyzes the depth of the ResNet network. It also summarizes the residual module of the ResNet network and applies the ResNet network to the brush calligraphy judgment classification to get the result. Use data sets for training and continuously optimize the network. In the end, the accuracy rate in the evaluation of calligraphy "Heng" by the ResNet network can be guaranteed to be above 70%, indicating that the deep learning technology is feasible in the application of auxiliary calligraphy practice.

# 5 DEVELOPMENT OF CALLIGRAPHY PRACTICE PLATFORM BASED ON WECHAT MINI PROGRAM

## 5.1 Requirements analysis

### 5.1.1 Functional Requirements

The WeChat Mini program developed in this article is a calligraphy learning platform. The user is a calligraphy learner and needs to satisfy the learner's function of obtaining calligraphy judgment results.

Therefore, the mobile learning platform needs to be guided by mobile learning theory to realize the characteristics of flexibility, interactivity, and diversity of mobile learning. At the same time, contextual cognition and learning theory emphasize the role of the environment in learning, so mobile learning platforms need to provide learners with a good learning environment with learning, interaction, and evaluation functions. In view of the above situation, the platform intends to set up three modules:

- Image upload and correction module
- Interactive ranking module
- Image display module

### 5.1.2 Equipment requirements

To develop a calligraphy learning platform based on WeChat applets, PC devices, and mobile smart devices are required. The main function of the PC device is to write program code, while the main function of the mobile smart device is to use the "machine test" to check the application effect of the applet on the mobile smart device.

- **PC equipment**

The development of WeChat Mini Programs requires fewer PC devices. The operating system of the PC device can be Windows or IOS, but the Windows operating system requires a Windows 7 version and above. PC equipment mainly uses the "WeChat Developer Tool" (IDE) officially provided by WeChat to write program code, upload, review, and release programs.

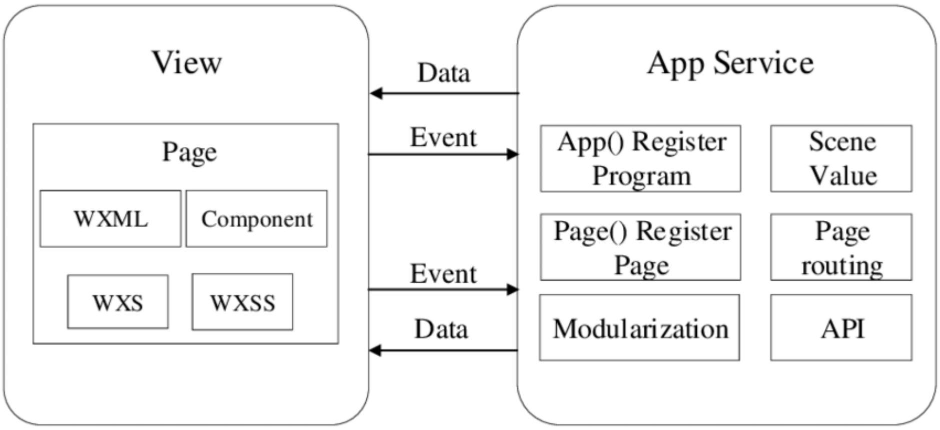


**Fig 5.1** Structure of WeChat

In addition, in the process of developing WeChat applets, the official cloud development functions provided by WeChat will be used. PC devices must also use visualization tools in the IDE to configure cloud information and manage resources.

- **Mobile smart devices**

Native apps are generally large in size, especially some video and game applications, which usually have higher requirements for hardware devices such as device memory and processors. If the hardware device cannot meet the requirements of the native App, the program will freeze or become unresponsive during the usage, which will affect the user's experience. The mini-program is attached to the WeChat App. As long as the device can install the WeChat software, the user can experience the applet. At the same time, WeChat requires the size of the mini-program should be less than 1M. Therefore, compared to the native app, it has lower requirements for device hardware.

Mobile smart devices need to install WeChat software, which is mainly used for real machine testing. After the developer writes the program code on the PC, he needs to use the WeChat software in the mobile smart device to scan the QR code provided by the WeChat developer to preview the small program. After the code is scanned successfully, the developer can proceed on the mobile smart device Preview, monitor the performance of the applet, and test the actual use of the applet.

## 5.2 Design Principles

The official WeChat document provides guidelines for the design of mini-programs, including requirements such as friendly and polite, clear and clear, convenient and elegant, and unified and stable to guide developers to develop standardized WeChat mini-programs [55]. The main three principles, as shown in figure 5.2:

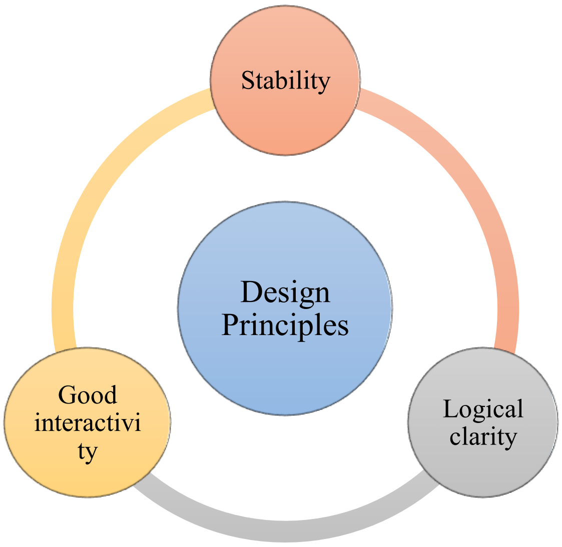



**Fig 5.2** Design principles

**Stability** is an essential attribute of an information system. After the calligraphy learning platform goes online, if it fails to run smoothly, there are phenomena such as freezing or unresponsiveness, it will inevitably affect the learner's experience and cause learners to give up using the calligraphy learning platform. When testing the program, it is necessary to involve the various functions of the program, to detect the influence of special values on the program, to avoid the logic error of the program, and to affect the use of learners.

**Good interactivity:** The operating equipment of the calligraphy learning platform is mainly concentrated on the mobile terminal, and the display interface of the mobile terminal is usually small. Therefore, when designing a mobile learning platform, we need to pay attention to the size of various function buttons, and we should fully consider whether they are suitable for finger clicks. When designing buttons, they should be clean and clear, so that learners can clearly understand the function of each button, and improve the ease of use of the mobile learning platform.

**Logical clarity:** Pages with the same functions should have a uniform style structure, including interactive buttons, font colors, and font sizes. The chaotic page structure design will increase the learner's use burden and affect the learner's operating experience. Therefore, when designing the page, avoid redundant elements on the page, causing misoperation.

## 5.3 Overall System design

### 5.3.1 MINA framework

MINA is a framework for developing WeChat mini-programs, similar to Vue[47], and its operating mechanism is "response-binding" mode. The MINA framework is very simple, including a view layer and a logic layer. When the program is running, it only needs to modify the data in the logic layer, and the view layer can be updated according to the results of the logic layer feedback.

The MINA framework diagram is shown in Figure 5.3. The interfaces, modes, and mechanisms required for the operation of mini Programs are all included in the framework.

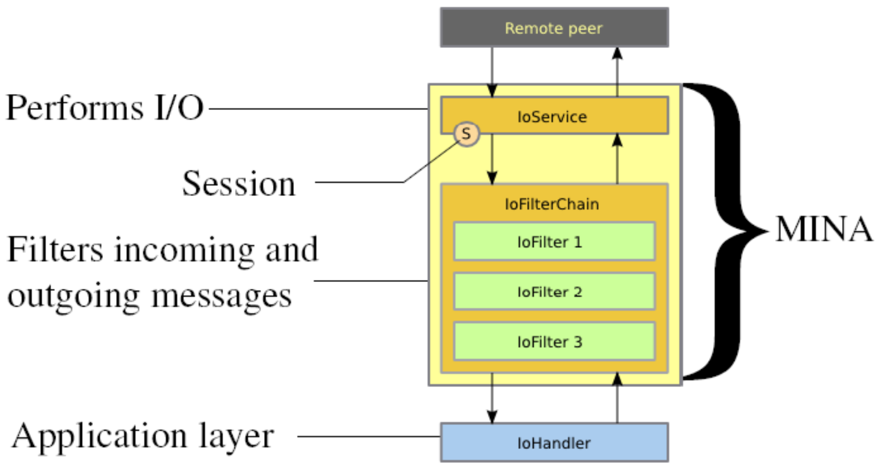



**Fig 5.3** MINA frame structure diagram

- **View layer**

The view layer is mainly responsible for providing the user operation interface and rendering the data information provided by the logic layer to the front-end page. A complete WeChat mini program page contains WXML, WXSS, JS, and JSON files, among which WXML files and js files are required.

The functions of each file are shown in Table 5.1. WeChat introduced a markup language (WXML, Wei Xin Markup language) different from HTML to describe the page structure of WeChat applets. At the same time, WeChat launched WXSS (WeiXin Style Sheet) to build the style of pages.

**Table 5.1** Functions of each file on the Mini Program

| File format | Function |
|---|---|
| **. wxml** | Describe the structure of the mini program |
| **. wxss** | Describe the style of the Mini Program page |
| **.js** | Describe the logical structure and various data of the Mini Program |
| **. json** | Describe various configuration information of the Mini Program |

- **Logic layer**

The logic layer is mainly responsible for processing the logical interaction part of data and programs. It is developed in JavaScript language, supports ES5 and ES6, and is used to realize the interactive function of learners and WeChat mini-programs. All kinds of events and data are processed by the logic layer. When the logic layer receives a request from the view layer to modify data, the logic layer processes the request sent by the view layer according to the pre-specified rules and methods. After the logic layer is processed, the processing result is fed back to the view layer. Finally, the view layer re-renders the page according to the feedback result from the logic layer.

### 5.3.2 Cloud development function

The calligraphy learning platform needs to record various data information generated during the learning process of learners. WeChat official stipulates that the size of a mini-program cannot exceed 1M, so it is impossible for developers to place various resources required by the platform inside the mini-program. Therefore, it is also necessary to use cloud development functions to manage various resources of WeChat mini-programs.

The cloud development function provides three major functions: cloud function, cloud storage, and cloud database. The concept of the back-end is weakened, and the work of developers building and maintaining servers is eliminated. Developers can use the cloud to complete their core business. If you use the traditional way of building a server to provide support for the backend of the applet, developers also need to build a server, register domain names, operate and maintain, etc., while the cloud development function eliminates these complicated procedures for developers. In the cloud development console, developers have the highest usage rights and can manage all data in the database, storage, and cloud functions.

- **Cloud function**

Cloud function refers to a set of program codes running in the cloud environment. Developers can use cloud functions to implement complex services without building a server. When the cloud function processes business, it creates cloud function instances in the cloud. The platform is responsible for a series of tasks such as the creation, destruction, and management of cloud function instances.

- **Cloud storage**

The cloud storage function provides developers with the function of storing resources, which can be uploaded to the cloud through the front-end page of the applet and the cloud development console. After the developer uploads the resources to the cloud, the cloud will assign a cloud file ID to each resource. The applet component supports the cloud file ID. According to the cloud file, the applet front-end page will obtain the specified cloud resource.

- **Cloud database**

The cloud database does not require developers to build their own database, is a JSON format database, and data is stored in the cloud database in the form of a collection. The collective data in the cloud database can be rewritten through the cloud function, cloud development console, and the front-end page of the applet. The cloud database provides developers with the functions of adding, deleting, modifying, and checking as well as authority management functions. The functions of insert, delete, modify, and query provides developers with functions to process data. Permission management allows developers to set user permissions to access the collection.

There are three main purposes for using cloud development functions to develop WeChat mini-program, as following table 5.2:

**Table 5.2** Purpose of using cloud development functions

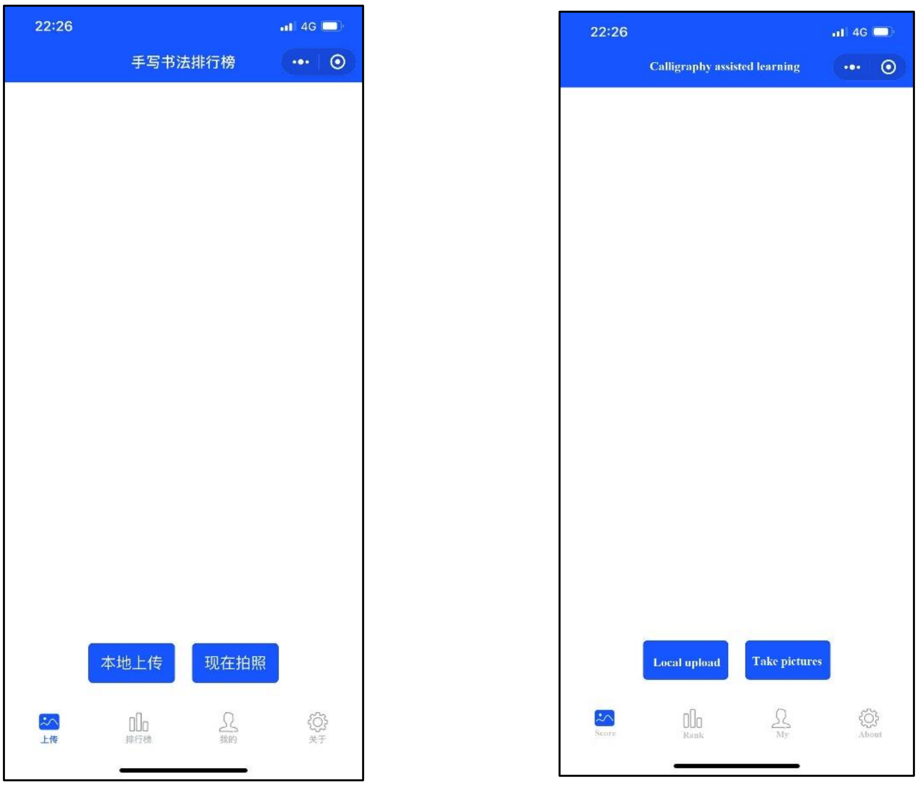
| Cloud | Function |
|---|---|
| **storage** | Storing teaching resources are stored in the cloud, and the teaching resources are displayed on the applet through the API provided by cloud development; |
| **database** | Storing all kinds of data generated during the learning process of learners, such as interactive comment information; |
| **function** | Developing the core business of the WeChat Mini Program. Compared with traditional servers, cloud development functions weaken the concept of back-end, without the need to build servers. |

### 5.3.3 System page design

In order to simplify the content of the applet, the development of this article did not use the traditional homepage, login interface, and business interface. In order to facilitate users to use quickly, the small program designed in this article omits the registration and login process and only has a calligraphy evaluation page, a ranking display page, and an extension page (instructions, calligraphy knowledge).

- **Calligraphy evaluation page**

It is also the homepage of the program. Learners can take pictures and upload calligraphy images to the server on this page, or upload them locally, and then the server will display the returned results on this page. The page structure of the program is shown in Figure 5.4.




(a) (b)

**Fig 5.4** Calligraphy evaluation page (a) Evaluation page—Chinese (b) Evaluation page—English

- **Leaderboard page**

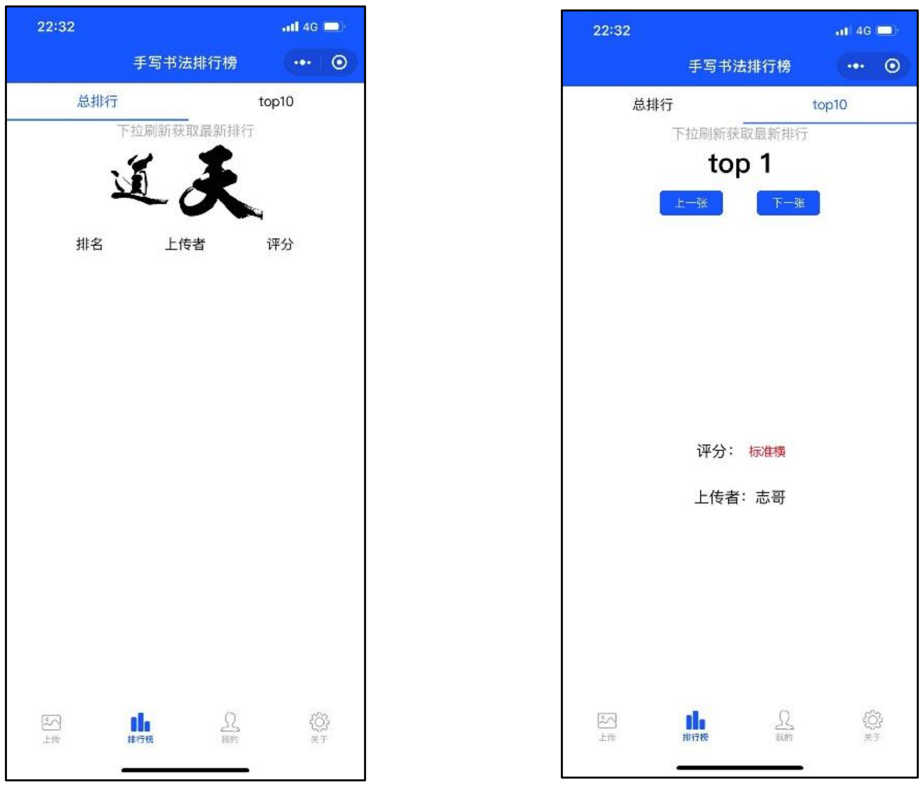
It is also the interactive page of this article. This page provides a display platform for calligraphy learners, sorting according to the probability of judgment from high to low, displaying the ID of the uploader, and the score is equal to the probability value * 100. In addition to the overall ranking accident, there is a special display that was also made for the top 10.

When calligraphy learners upload their own calligraphy pictures, it is equivalent to adding a new record to the database. The recorded content includes the learner's user information, calligraphy images, upload time, and evaluation results. The relevant fields of the database are shown in Table 5.3.

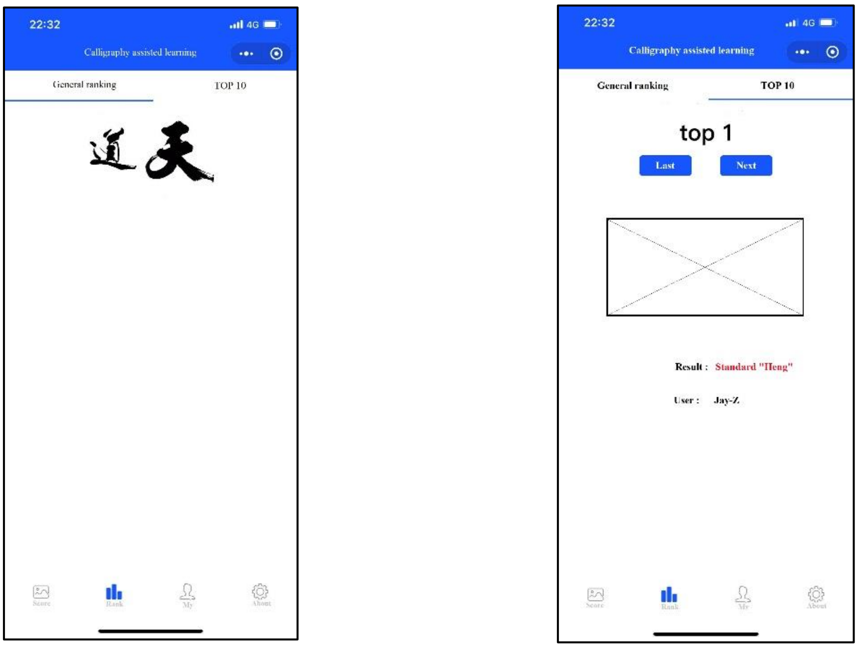
**Table 5.3** Learner upload record datasheet

| Field | key | datatype | description |
|---|---|---|---|
| **_id** | Yes | String | Serial number |
| **_openid** | - | String | Uploader id |
| **imageUrl** | - | String | URL link of uploaded image |
| **time** | - | Date time | Upload time |
| **judgeRes** | - | Float | The result of image judgment |
| **userInfo** | - | String | Uploader's WeChat information, including WeChat user's nickname, city, and gender |

The page structure of the program is shown in Figure 5.5.




(a) (b)

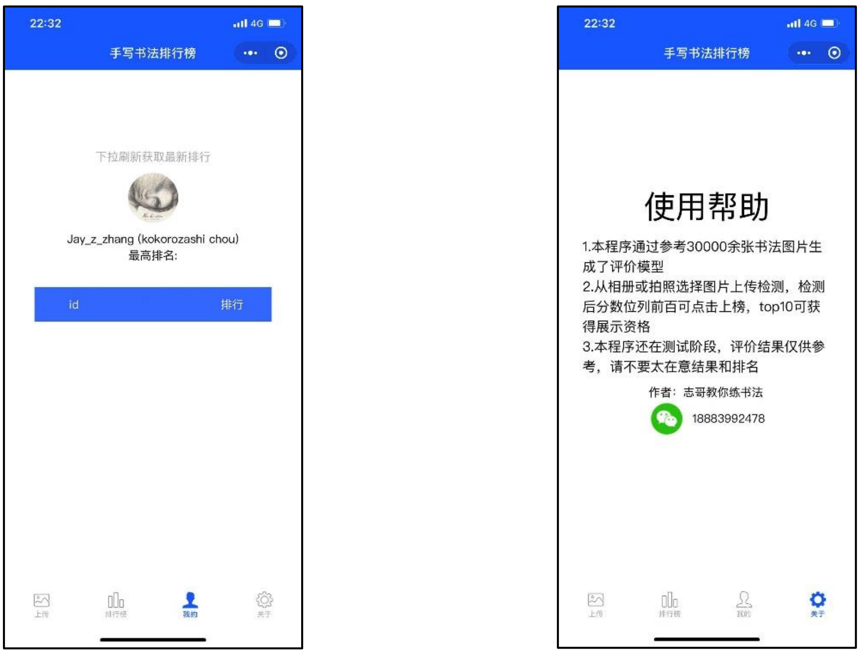



(c) (d)

**Fig 5.5** Leaderboard page (a), (b) Chinese page (c), (d) English page

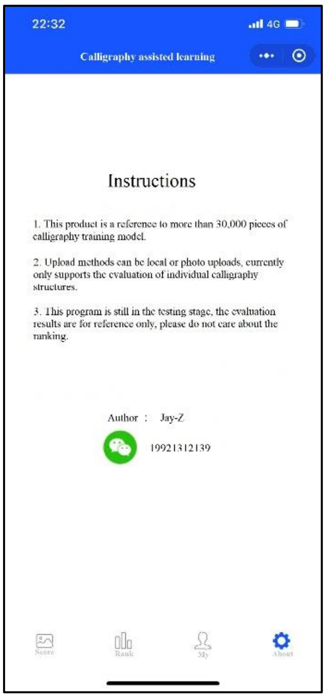
- **Extension page**

Here mainly introduce two extension pages, one page is to display the practitioner's own calligraphy, and the other page is used to give a brief introduction to the mini-program, which can be used as an extension in the future. The page structure of the program is shown in Figure 5.6.




(a) (b)


(c)




(d)

**Fig 5.6** Extension page (a), (b) Chinese page (c), (d) English page

## 5.4 Mini program testing

After the mini-program is developed, it needs to be tested to ensure that the mini-program can normally run after it is online. The WeChat Developer Tool provides an experience scoring tool for mini-programs. With this tool, you can detect whether the mini-programs are defective in design and give corresponding optimization suggestions. The experience score is 100 points.

The use of this tool is also very simple. Open the WeChat developer tool, click the details button in the upper right corner of the interface, and in the local settings, set the debugging base library to version 2.2.0 or above, and the author to version 2.6.1. Next, select the Audits panel in the debugging area and click Run to test. During the test, the developer needs to operate the applet in the preview area. During the operation, the developer needs to use every function button, interaction, and page of the applet as much as possible, so that the experience scoring tool can give a comprehensive evaluation. The score of this applet is shown in Figure 5.7.


**Fig 5.7** Applet experience test score

From the mini-program experience score, it can be seen that the mini-program scores 89 points, 100 points, and 90 points in terms of performance, experience, and best practices, with a total score of 91 points. The mini-program performs well and can be officially launched and opened to learners after passing the review.

## 5.5 Chapter Summary

This chapter mainly introduces the WeChat Mini Programs with auxiliary functions of calligraphy learning. Firstly, it introduces the significance of developing WeChat Mini Programs from the perspective of technology, function, and users. Secondly, it introduces the development tools of WeChat applets and the convenience brought by cloud development functions. Finally, some details of page development are introduced.

# 6 SUMMARY & OUTLOOK

## 6.1 Summary

The development of artificial intelligence-based learning platforms is now the mainstream direction of social development and has disruptive changes to the traditional education system. Therefore, the study of handwritten calligraphy automatic evaluation systems is very important. Now the creation of education platforms is the focus of artificial intelligence research. Using deep learning technology to automatically judge handwritten calligraphy, and classify calligraphy errors, so that calligraphy learners can practice anytime and anywhere and improve their brushwork in time. At the same time, the temporary use of resources for teachers to evaluate and modify calligraphy is also saved. Based on the ResNet network architecture and handwritten calligraphy image feature extraction technology, this paper has obtained good experimental results that have been tested and completed the classification challenge of handwritten calligraphy errors. The main contributions of this paper are as follows:

(1) Read Chinese and English literature in the field of image classification to understand the way of calligraphy appreciation, image classification technology, and the development trend of mobile program development. Through the understanding of the current research status at home and abroad, understanding of the deep learning in Chinese and English versions, further exploration of classification methods in the field of calligraphy education.

(2) Summarize and explain image classification methods, including classic image classification methods, and understand the historical development of image classification methods. Sort out the concepts of CNN and deep learning and other related knowledge, and have a deep understanding of CNN, including network architecture, practical details, activation functions, etc.

(3) Compared with some network architectures for image classification, the author chooses to use the residual ResNet network for image classification and conducts training and testing through the datasets of Fudan University Painting and Calligraphy Association and Shanghai University of Finance and Economics Painting and Calligraphy Association. According to the experimental results, the advantages of R-CNN are:

- It uses the selective search method to speed up the selection of candidate domains (compared with traditional methods);
- Solved the problem of insufficient labeled data in the target detection training process;
- Get more precise positioning. However, the disadvantage of R-CNN is that there are redundant calculations. Because the candidate area needs to be zoomed, it will cause the target to be deformed. Even if this approach makes a major breakthrough in the direction of image classification, it still cannot analyze images in the field of calligraphy.

(4) Due to the inspiration of R-CNN, the ResNet network with the added residual module appeared, which has become a relatively effective method of calligraphic image classification methods. This article uses the ResNet network to train and test in the same data set. The results show that the testing effect of the ResNet network is higher than other networks. In the experiment, the author found that due to the variety and complexity of calligraphy fonts, it may not be conducive to the test samples. According to the recall curve of the confusion matrix, the author believes that if the quality of the training data is optimized, the resulting probability of calligraphy judgment can be further improved.

(5) In the latter part of this article, by combing the development context of calligraphy learning and WeChat applets, combined with the characteristics of learning on the Internet, the current calligraphy learning platform is based on WeChat applets has a complicated development process and fewer functional modules. The use of cloud development functions to develop a calligraphy learning platform based on WeChat applets weakens the concept of the backend, allowing developers to focus on the development of the core business. While simplifying the development process, it ensures that the functional modules of the platform meet the needs of teachers and learners. After the development of the calligraphy learning platform is completed, internal tests will be conducted to optimize the calligraphy learning platform. After the development of the WeChat Mini Program is completed, submit it to WeChat official review. Finally, it is used by learners for empirical research.

## 6.2 Future research directions

This article develops a WeChat mini program for automatic evaluation and correction of handwritten calligraphy based on the ResNet network based on the research of classification methods based on deep learning. Even if the results of the calligraphy review and revision have reference value, there is a fatal problem in the paper: the number of images for training and testing is not enough, so we still need to consider the future direction. The following is the key content:

(1) The biggest problem in this article is the impact of training the network. This influencing factor is the lack of data. Therefore, the first thing to do is to get more data from calligraphy institutions or other channels, optimize training according to the amount of data, and get results.
(2) At present, when training and testing the network, the time we need is too long, and the speed is too slow. Therefore, the direction of future breakthrough progress is to consider what kind of network can be added to make the training and testing process consume less time.
(3) Suppose we can get good experimental conclusions after repeated training and testing. That way, it can be applied to other calligraphy type evaluation and reform fields to create more value.
(4) With the advancement and development of technology, the functions of WeChat Mini Programs will become stronger and stronger, and their applications in the education field will become more extensive. At the same time, with the acceleration of the pace of life, the use of mobile smart devices to carry out learning activities will become the trend of future learning. Future Research on WeChat Mini Programs in the education field can be explored from the following two aspects:
    - Integrate WeChat Mini Programs with physical classrooms, and explore efficient online and offline teaching models, so that WeChat Mini Programs can have more applications in the education field.
    - The evaluation index system for mini-programs can be further studied. At present, there are few relevant types of research on the evaluation of WeChat applets, and there is still a lack of unified and standardized standards for evaluating WeChat applets. Moreover, different types of small programs should be evaluated with different evaluation criteria.

# References


[1] Chen Ding. Dot-guided Writing instrument and its manufacturing method: China. CN200610157297.8 [P].2008-06-11.

[2] Ministry of Education, PRC. Guidelines for Primary and Secondary School Students' Education [S]. Beijing: Beijing Normal University Press, 2013.

[3] Hu Jingxuan. On the Dilemma of Calligraphy Education in Primary and Secondary Schools and its Solutions [J]. Hua Zhang, 2013(2):247-248.

[4] The Guidelines for Calligraphy Education in Primary and Secondary Schools, Ministry of Education on January 18, 2013

[5] Shi D, Gunn S R, Damper R I. Active radical modeling for handwritten Chinese Characters [C]// International Conference on Document Analysis and Recognition, 2001. Proceedings. IEEE, 2001:236-240.

[6] Rogers Y, Price S, Fitzpatrick G, et al. Ambient Wood: Designing New Forms of Digital Augmentation for Learning Outdoors[C]// Third International Conference for Interaction Design &Children. ACM Press, 2004.

[7] Korucu AT, Alkan A. Differences between m-learning (mobile learning) and e-learning, basic terminology and usage of m-learning in education [J]. Procedia Social & Behavioral Sciences, 2011, 15 (none):1925-1930.

[8] Yang Jinbang. Principles and Methods of Calligraphy Appreciation [J]. Popular Literature and Art, 2017(2): 137.

[9] Wang Weilin. Arch thief -- The aesthetic pursuit, style characteristics and influence of Samanweng's Calligraphy [J]. Art of Calligraphy and Painting, 2011(6):63-64.

[10] Liu Yi. Appreciation of Calligraphy [J]. Chinese Painting and Calligraphy, 2011(5):78-81.

[11] Li Feng. An Analysis of the Appreciation Principle of Calligraphy [J]. Science and Education Wenhui (Mid-Ten-day issue), 2009(11):263-264.

[12] Y. Lecun, L. Bottou, Y. Bengio， et al. Gradient-based learning applied to document recognition [J].Proceedings of the IEEE, 1998, 86(11):2278-2324

[13] A. Al Bayati, N. Sulaiman, G.Sadiq. A Modified Conjugate Gradient Formula for Back Propagation Neural Network Algorithm [].Journal of Computer Science, 2009, 5(11):849-856.

[14] Loschky LC, Sethi A, Simons D J,et al. The importance of information

localization in scene gist recognition. [J]. J Exp Psychol Hum Percept Perform, 2007, 33(6):1431-1450.

[15] Liu Chenglin. High Accuracy handwritten Chinese character recognition using quadratic classifiers with discriminative feature extraction[C]//Proceedings of 18th International Conference on Pattern Recognition, 2006:942-945.

[16] K.He, X.Zhang, S.Ren, et al. Deep Residual Learning for Image Recognition [C].IEEE Conference on Computer Vision and Pattern Recognition(CVPR)，Los Alamitors，2016, 770-778

[17] Ge Yong, Huo Qiang, Feng Zhidan. Offline recognition of handwritten Chinese characters using Gabor features, CDHMM modeling and MCE training [C]//Proceeding of IEEE International Conference on Acoustics，Speech and Signal Processing 2002，2002:1053-1056.

[18] Xue J, Ding x, Liu C, et al. Location and interpretation of destination addresses on handwritten Chinese envelopes [J]. Pattern Recognition Letters, 2001, 22(6-7):639-656.

[19] W. S. Lai, J.B. Huang, N. Ahuja, et al. Fast and Accurate Image Super-Resolution with Deep Laplacian Pyramid Networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,2019 41(11):2599-2613

[20] Zheng Yuanpan, Li Guangyang, Li Ye.Summary of Research on Application of Deep Learning in Image Recognition[J].Computer Engineering and Applications,2019,55(12).

[21] Guan Yin. Flower recognition system based on residual network migration learning [J]. Computer Engineering and Applications, 2019,55(1).

[22] Wang Heng et al. research on breast cancer pathological image classification based on ResNet50 network [J].Journal of China Jiliang University,2019,30(1).

[23] Hu Mingkuei．Visual pattern recognition by moment invariants[J]．IEEE Transactions on Information Theory, 1962，8(1):179-187．

[24] N. Dalal, B. Triggs. Histograms of Oriented Gradients for Human Detection [C]. IEEE Conference on Computer Vision and Pattern Recognition, San Diego，2005,886-839

[25] N. Zhang. Computing Optimised Parallel Speeded-Up Robust Features (P-SURF) on Multi-Core Processors[].International Journal of Parallel Programming, 2010,38(2):138-158

[26] Q.C.zhang,L.T.Yang,Z.K.Chen et al.A Drop connect Deep Computation

Model for Highly Heterogeneous Data Feature Learning in Mobile Sensing Networks[J].IEEE Network,2018, 32(4):22-27
[27] J. Y. Choi, K. S.Sung, Y.K. Yang. Multiple Vehicles Detection and Tracking based on Scale-Invariant Feature Transform[C].Intelligent Transportation Systems Conference, Seattle, 2007, 528-533
[28] R.Memisevic, C.Zach, G.E.Hinto,et al.Gated Softmax Classification[C]. Proceedings of the 23rd International Conference on Neural Information Processing Systems，Vancouver,2010,1603-1611
[29] L.Wang，Y.Yang,R.Min,et al.Accelerating deep neural network training with inconsistent stochastic gradient descent[J].Neural Networks,2017,93(000):219-229,11
[30] Pan S J, Yang Q. A Survey on Transfer Learning [J]. IEEE Transactions on Knowledge and Data Engineering, 2010,22(10).
[31] Ma Xiaohu, Pan Zhigeng, Zhang Fuyan. Spline Chinese character library based on stroke description and its application [J]. Chinese Journal of Computers, 1996, 19(3): 8l-88.
[32] B. Pedro，M. A. Ricardo. On the performance of GoogLeNet and AlexNet applied to sketches [C]. Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, 2016, 1124-1128
[33] K. Li, Z. Wu, K.C. Peng, et al. Tell me where to look: guided attention inference network [C]. Salt Lake City, UT, United States: Proceedings of 31th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018:9215-9223
[34] W. Fei, M. Jiang, Q. Chen, et al. Residual Attention Network for Image Classification [C]. Honolulu, HI, United States: Proceedings of 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017:6450-6458
[35] C.Saunders, M.O.Stitson, J. Weston et al. Support vector machine[J].Computer Science, 2002,1(4):1-28
[36] S.Xu, H.Jiang, T.Jin, F.C.Lau, Y.Pan, Automatic generation of Chinese calligraphic writing with style imitation, IEEE Intell.Syst.24 (2)(2009)167–178.
[37] 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.[8]

[38] Chollft F. Xception: Deep Learning with Depthwise Separable Convolutions [C]// Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2017:1800-1807.

[39] Hinton G E, Salakhutdinov R R. Reducing the Dimensionality of Data with Neural Networks [J]. Science, 2006.

[40] HowardA G, Zhu M, Chen B, et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications [EB/OL]. (2019-04-10)[2020-05-01].https://arxiv.org/pdf/1704.04861.pdf.

[41] N.Srivastava, G.Hinton, A. Krizhevsky, et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting [J].Journal of Machine Learning Research, 2014, 15 (1):1929-1958

[42] Fang Yamei, Wang Hongjun, Huang Kuangyu, etc. Image preprocessing method for 3D reconstruction of target under complex background［J］. Progress in Laser and Optoelectronics, 2019, 56(13): 118-126.

[43] Z.Zhang, J.w U, KYU, Chinese calligraphy specific style rendering system, in: 10TH annual joint conference on digital libraries, 2010, brand.99–108.

[44] Zeng Qian.The establishment of a second-hand goods transaction and sharing platform for colleges and universities based on the WeChat applet[J].Modern Business,2019.2:52-53

[45] Qiu Yueye.The vitality of WeChat applets[J].21st Century Business Review,2018.Z1:84-86

[46] Yang Qi, Zhang Liping.The development of WeChat applets from the perspective of Internet ecology[J]. News Forum,2017.2:22-24

[47] Yu Shengquan, Mao Fang. Informal learning—a new field of e-Learning research and practice [J].Education Research, 2005(10):19-24.

[48] Yuan Hong.The mobile reading behavior in the informal learning of young people [J].Library Forum, 2019, 39(11):113-121.

[49] C.Szegedy, W.Liu, Y.Jia, et al. Going Deeper with Convolutions[J].Computer Vision and Pattern Recognition(CVPR),Boston,2015，1-9

[50] G.S.Kamaledin.Competitive Cross-Entropy Loss: A Study on Training Single-Layer Neural Networks for Solving Nonlinearly Separable Classification Problems [].Neural Processing Letters, 2019, 50(2):1115-1122

[51] A.Krizhevsky, I. Sutskever, G.Hinton. ImageNet Classification with Deep Convolutional Neural Networks [J].Communications with the ACM, 2017,

60(6):84-90

[52] Zhu Hong. The foundation and application of digital image processing [M]. Beijing: Tsinghua University Press, 2013: 1.79.