Research on Handwritten Chinese Character Recognition Based on Improved Convolutional Neural Network

WANG Jian-hua[1], ZHANG Ya-qi[1], XIAO Bo-huai[2], LI Ben-jian[1*]
(1.*College of Arts*, *Guilin University of Technology*, *Guilin* 541006, *China*; 2. *School of Information Science and Engineering*, *Guilin University of Technology*, *Guilin* 541006, *China*)

Abstract Handwritten Chinese character recognition is one of the important research areas of intelligent information technology, which is crucial in handwritten input, document entry and other related fields. However, due to the wide variety and complex structure of Chinese characters, the usual deep learning methods have poor feature extraction ability and large computational effort for Chinese characters, and the accuracy of Chinese character recognition cannot reach a satisfactory level. To solve the above problems, in this study, an improved deep convolutional neural network model was proposed by adding a batch normalization layer to the structure of the model, adding Dropout and regularization methods to the loss function, and adding an RMSprop optimizer to the training process. To confirm the effectiveness of the proposed model, the experiments were conducted on the dataset of CASIA-HWDB1.1 and IAHCC-UCAS2016. The overall experiments of the proposed model were conducted, grouped with other deep learning models, and the ablation experiments of the batch normalization layer and RMSprop optimizer were conducted to verify the high accuracy and high running speed of the proposed model from an all-round perspective.
Key words Handwritten Chinese character recognition; Convolutional neural network; Deep learning

基于改进卷积神经网络的手写汉字识别研究

王建华[1]，张雅祺[1]，肖博怀[2]，李本建[1*]
（1.桂林理工大学 艺术学院，桂林 541006；2.桂林理工大学 信息技术与工程学院，桂林 541006）

摘要 手写汉字识别是智能信息化的重要研究领域之一，在手写输入、文件录入等相关领域至关重要。然而，由于汉字的种类繁多、结构复杂等问题，通常的深度学习方法对汉字的特征提取能力差、计算量大，汉字识别的准确度无法达到令人满意的程度。为解决上述问题，本研究提出了一种改进的深层卷积神经网络模型，在模型的结构上添加了批标准化层，在损失函数上加入了Dropout和正则化方法，在训练过程中加入了RMSprop优化器。为了证实提出的模型的有效性，在CASIA-HWDB1.1和IAHCC-UCAS2016的数据集上进行实验。对提出的模型进行了整体实验，与其他深度学习模型进行了分组对比实验，对批标准化层和RMSprop

收稿日期：2022-06-20　　修回日期：2022-09-29　　*为通讯作者
本文引用格式：WANG Jian-hua, ZHANG Ya-qi, XIAO Bo-huai, et al. Research on Handwritten Chinese Character Recognition Based on Improved Convolutional Neural Network [J]. Printing and Digital Media Technology Study, 2023, (1): 45-56.

优化器进行了消融实验，全方面验证了本研究模型的高准确率和高运行速度。

关键词 手写汉字识别；卷积神经网络；深度学习

中图分类号 TS801.8; TP391.1 **文献标识码** A **文章编号** 2097-2474(2023)01-45-12

DOI 10.19370/j.cnki.cn10-1886/ts.2023.01.006

0 Introduction

In the current era of rapid information development, handwritten Chinese character recognition has broad application prospects in the fields of checks, printing, manuscripts, and handwritten text input devices. However, the traditional character recognition technology has been unable to meet the needs of people who want to recognize Chinese characters quickly and accurately due to the characteristics and similarity of Chinese character structures.

Therefore, how to recognize handwritten Chinese characters with high efficiency and low cost through computer and artificial intelligence technology is one of the major difficulties in pattern recognition[1-4].

With the emergence of deep learning, handwritten Chinese characters have achieved breakthroughs and development in stages. The results of handwritten Chinese characters based on Convolutional Neural Network (CNN) are the most remarkable, and the recognition performance is far greater than that of traditional methods. Ciresan et al[5] won the first place in the 2011 ICDAR competition with a CNN-based handwritten Chinese character recognition method; in 2012, Ciresan et al proposed an end-to-end multi-column CNN model; Yin et al[6] won the championship at the 2013 ICDAR competition by using the model.

With the continuous development of various fields of deep learning, the handwritten Chinese character method based on CNN has been further improved in the recent years. Chen et al[7] proposed an adaptive local receptive field CNN model for handwritten Chinese character recognition, which introduced a maximum pooling layer to improve the feature extraction ability; Yang et al[8] proposed an iterative refinement module, which is implemented by an attention-based recurrent neural network, through which small differences between certain characters can be overcome and the overall accuracy of the model can be improved; Bi et al[9] proposed a CNN model based on GoogleLeNet, which improved the width and depth of CNN while could reduce the number of parameters and improve the accuracy of recognition.

Although the above methods have improved the recognition of handwritten Chinese characters based on CNN, they only consider the adjustment and modification of the structure of CNN, and do not consider the problems of gradient and fitting. The test results may be too different from the training results. At the same time, conducting experiments on larger datasets increases the time cost. Therefore, a Chinese character recognition method based on an improved CNN is proposed in this paper. A batch normalization layer is added to the designed deep CNN to reduce training time. Adding regularization and RMSprop gradient optimizer in training effectively prevents overfitting while preventing gradients from disappearing and exploding, thereby improving the recognition accuracy and speed of handwritten Chinese characters.

1 Theoretical Foundation

1.1 Handwritten Chinese Character Recognition

The traditional handwritten recognition method framework is shown in Fig.1, which mainly includes three parts: data input, feature extraction and classification processing.

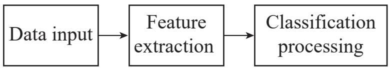


Fig.1 Traditional handwritten Chinese character recognition framework
图1 传统手写汉字识别框架

Data input includes data preprocessing. Data preprocessing determines the quality of the input data, which will directly affect the effect of feature extraction and classification processing. Data preprocessing suitable for images usually adopts methods such as sample normalization, smoothing and denoising, and image binarization[10]. Feature extraction is the most important step, and the way of feature extraction determines the final recognition performance. Feature extraction suitable for handwritten Chinese characters is mainly divided into structure-based feature extraction and statistics-based feature extraction[11]. The classification process usually adopts a classifier, and the classifiers suitable for the recognition of handwritten Chinese characters mainly include Naive Bayes, quadratic discriminant function, Softmax and so on.

1.2 Convolutional Neural Network

CNN is the LeNet network proposed by Yann LeCun[12]. It performs local feature extraction by convolving the input data with a filter, then sampling through the resulting local features and passing it to the next layer. The classical CNN model includes input layer, convolutional layers, pooling layers, fully connected layer and output layer[13], as shown in Fig.2.

The input layer feeds data into the CNN model. In the process of handwritten Chinese character recognition, the input data is the image pixel matrix after preprocessing. The length and width of the matrix are the size of the handwritten Chinese character image, and the depth is the color channel of the image. The convolutional layer is a feature extraction layer, which

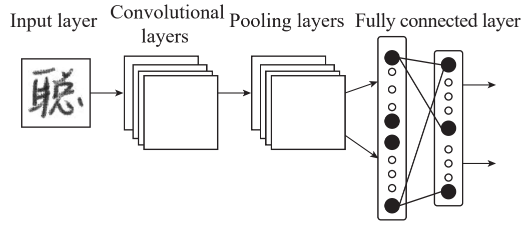


Fig.2 Classical CNN structure
图2 经典卷积神经网络结构

extracts the original feature information on the initial convolutional layer. As the number of layers increases, advanced and complex features can be extracted from the original feature information. Once one feature is extracted, the location mapping relationship between this feature and other features is also determined. In the process of convolution calculation, each convolution kernel has a filter with a manually set size, and the depth of the filter must be consistent with the depth of the input layer. Pooling layers are generally divided into max pooling and average pooling. By taking the maximum value or the average value of the convolution results, the dimensionality reduction is achieved while reducing the amount of calculation and the number of parameters. The fully connected layer acts as a classifier. After the model performs continuous convolution and pooling, the fully connected layer links the nodes of this layer with the points of the previous layer, and outputs through the Softmax function or other functions.

2 Handwritten Chinese Character Recognition Based on Improved Convolutional Neural Network

It is required to train and extract features from batches of Chinese characters. Therefore, a deep CNN is designed consisting of five convolutional layers, four pooling layers, three batch normalization layers and one regression layer. The network constructed by convolutional layer, batch normalization layer, pooling layer and dropout layer is used for feature extraction of

Chinese character images, and the regression layer is used for classification processing. The model structure is shown in Fig.3.

2.1 Data Preprocessing and Input

Most of the existing public datasets of Chinese characters are of the image type. The preprocessing operation is performed before data input in order to facilitate model processing and speed up model processing. Normalization is the most common preprocessing operation and is an effective means used to solve gradient disappearance and gradient explosion, so in this study, the normalization operation is used to control the input value distribution of the neural network in the interval of [0, 1]. Specifically, as shown in Formula (1), the size of the Chinese character image is set to a uniform size, and the pixel matrix of the image is controlled within the interval of [0, 1] through normalization processing.

$$\begin{gathered} Uniformsize(A) = [m \times n] \\ X = \frac{A'}{255},\quad X \in R^{m \times n} \end{gathered} \tag{1}$$

A is the image of the original data set, the height of the unified size is m, and the width is n; A' is the image of pixel normalization after the unified size. After preprocessing, each normalized image $X=[x_1, x_2, ..., x_i]$ of $m \times n$ is used as input to the model.

2.2 Convolutional Layer

Preliminary feature extraction is performed on the input Chinese character image through the convolutional layer, the convolutional kernel is multiplied with the input Chinese character image and a bias function is added. Finally, the convolutional data features are obtained through the activation function. As shown in Formula (2).

$$x_j^l = f\left(\sum_{i \in M_j} x_i^{l-1} * k_{ij}^l + b_{1j}^l\right) \tag{2}$$

x_i^{l-1} is the i-th feature map of the l-1-th layer, k_{ij}^l is the feature map of the i-th filter in the l-th layer connecting the i-th filter in the previous layer, b_{1j}^l is the bias function of the convolution, * is the convolution operation, $f(\cdot)$ is the activation function.

First, during the convolution process, each pixel point of the Chinese character image is involved in the operation of features through the convolutional kernel, which can make the extracted features all involved in the classification. Secondly, the convolutional neuron single extracts the features inside the filter, which can effectively take into account the local feature information, while the final fused output features

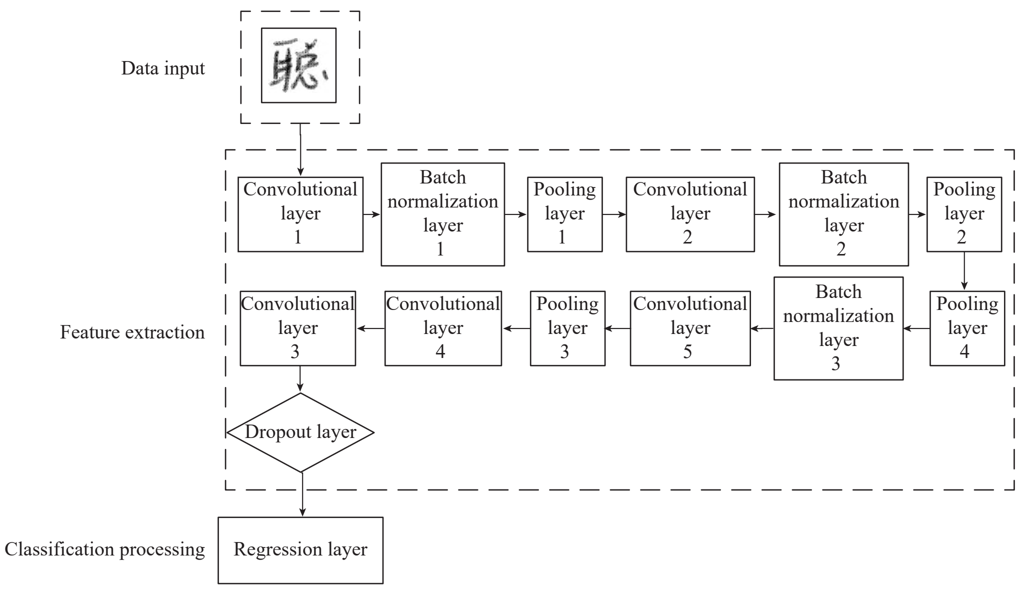


Fig.3 Improved convolutional neural network structure
图3 改进卷积神经网络结构

combine the local information to obtain the global feature information. Finally, by setting convolutional kernels with the different parameters, the convolutional neural network is made to extract the different features on Chinese character images, which can effectively avoid the appearance of a priori knowledge misconceptions caused by manual extraction.

Particularly, for more efficient gradient descent and backpropagation, a random rectified linear unit (Randomized Leaky ReLU, RReLU) is used as the activation function of the model, as shown in Formula (3). Compared with ordinary activation functions, RReLU is easier to determine gradient parameters in the testing process and effectively avoid gradient disappearance.

$$f(x)=\begin{cases}x \text{ if } x>0\\ \lambda x \text{ if } x\leqslant 0\end{cases} \tag{3}$$

λ is a random variable taken from the continuous uniform distribution $U(l, u)$ probability model, $l<u$ and $l, u\in[0, 1)$.

2.3 Batch Normalization Layer

A deep neural network is used for training in order to better train the model. However, when the network is very deep, since the input preprocessed Chinese character image data is in the interval [0, 1], the data becomes smaller and smaller during forward propagation, and the gradient may disappear during back propagation. Therefore, batch normalization is used to optimize the node characteristics of each layer of the network. Batch normalization refers to the normalization of the input data in terms of batch samples in stochastic gradient descent, so that its probability distribution in each dimension becomes a stable probability distribution with a mean of 0 and a standard deviation of 1. In order to avoid the destruction of the features learned in normalization, it is necessary to introduce learnable parameters γ and β to transform and reconstruct the data, and the flow of the batch normalization algorithm is shown below.

Algorithm 1: Batch Normalization Algorithm

Input：Characteristics of m nodes in a layer neural network $x=[x_1, x_2, ..., x_m]$，parameters to be learned γ and β

Output：Characteristics of m nodes after standardization $\tilde{x}=[\tilde{x}_1, \tilde{x}_2, \cdots, \tilde{x}_m]$

Initializes the parameters to be learned γ

Initializes the parameters to be learned β

for i=1 to m do

$\mu=\mu+\frac{1}{m}x_i$←Average value；

$\sigma^2=\sigma^2+\frac{1}{m}(x_i-\mu)^2$←Mean Square error；

for i=1 to m do

if μ and σ^2 to help do

$x_{i\ \text{norm}}=\frac{x_i-\mu}{\sqrt{\sigma^2+\varepsilon}}$←Initial standardized node；

$\tilde{x}_i=\gamma x_{i\ \text{norm}}+\beta$； ←Standardized node with extended parameters；

else do

$\gamma=\sqrt{\sigma^2+\varepsilon}$， $\beta=\mu$ ← Offset standardization with parameters when not in effect；

return $\tilde{x}=[\tilde{x}_1, \tilde{x}_2, \cdots, \tilde{x}_m]$；

In CNN, a feature map can be treated as one neuron because of the shared weights, i.e., the mean and variance of all neurons of a feature map are obtained, and then the neurons of this feature map are normalized.

2.4 Pooling Layer

The dimensionality reduction operation is performed on the extracted convolutional feature map through the pooling layer, which reduces the hidden complexity of the neural network, thereby speeding up the training speed of the model. As is shown in Formula (4).

$$y_j^l=f(\lambda_j^l down(\tilde{x}_j^{l-1})+b_{2j}^l) \tag{4}$$

$\tilde{x}_i^{l-1}$ is the standardized feature map of the i-th batch of the l-1-th layer, $down(\cdot)$ and y_j^l are the downsampling functions and coefficients, b_{2j}^l is the bias function, and $f(\cdot)$ uses RReLU as the activation function like the convolutional layer.

2.5 Dropout Layer and Regression Layer

To prevent overfitting, a dropout layer is added at the end of the forward pass. Dropout makes the model randomly discard some neurons during the training process, reducing time cost and computing resources, so as to achieve the purpose of diversifying output features and effectively achieve overfitting. Dropout makes the neural network stop working with a certain probability when it propagates forward, which can increase generalization and speed up the overall training speed while effectively achieving overfitting.

The regression layer adopts the Softmax classifier, as shown in Formula (5). The classifier performs better for datasets with many categories and large numbers of Chinese characters.

$$S_i = \frac{\exp(y_i)}{\sum_i^m \exp(y_i)} \tag{5}$$

$y=[y_1, y_2, ..., y_m]$ is the output after pooling, and $S=[S_1, S_2, ..., S_i]$ is the output after classifier regression processing.

2.6 Cost Function

The cross-entropy cost function is used to calculate the amount of loss that occurs during model training in order to measure the gap between the model's prediction and the actual situation. As is shown in Formula (6).

$$Loss = \frac{-\sum_i x_i \log(S_i)}{m} \tag{6}$$

x_i is the actual correct label of the dataset sample, S_i is the output after the classifier regression processing, $-\sum_i x_i \log(S_i)$ is the cross-entropy value between the actual label and the output after the classifier regression processing, and $Loss$ is the final loss value.

2.7 Model Optimization

2.7.1 Cost function optimization

In the process of training the CNN model, the results of the test set and the training set are too different due to the excessive training parameters and the high model complexity, and the phenomenon of overfitting occurs[14]. In order to prevent this phenomenon, the cost function is optimized by means of regularization and Dropout. The Dropout is shown in the Dropout layer in section 2.5, while the regularization uses the L_2 regular term, which is a new regular term added to the loss function to significantly reduce the variance of the model while slightly increasing the bias, thus reducing the auxiliary and instability of the model, speeding up the convergence of the model, and avoiding the overfitting phenomenon. The optimized cost function is shown in Formula (7).

$$Loss\_L = Loss + \frac{\eta}{2m}\sum_j w_j^2 \tag{7}$$

$Loss\_L$ is the cost function added to the regular term, $Loss$ is the original cost function, η is the regularization parameter, and w_j is the parameter of each weight.

2.7.2 Gradient descent optimization

In the process of training the CNN model, an optimizer is required to continuously train parameters, so that the model can perform effective gradient descent to achieve an optimal situation. Therefore, the RMSprop optimizer is used to optimize the parameters of the model, and the algorithm flow of the optimizer is as follows.

Algorithm 2: RMSprop Algorithm

Input: Characteristics of m nodes in a layer neural network $x=[x_1, x_2, ..., x_m]$ and the corresponding output category $z=[z_1, z_2, ..., z_m]$, global learning rate ε, numerical stability parameter $\delta=10^{-6}$, parameters to be learned σ, decay rate ρ and gradient accumulation γ

Output: Parameter of gradient optimal solution

Initiating decay rate ρ

Initial gradient cumulant γ

Initializes the parameters to be learned β

While the optimal solution is not reached do

$g = \frac{1}{m}\nabla_\theta \sum_i L(f(x_i;\theta), z_i)$ ←Calculated gradient；

$\gamma = \rho \cdot \lambda + (1-\rho) \cdot g \odot g$ ←Cumulative Square gradient；

$\Delta\theta = -\frac{\varepsilon}{\sigma + \sqrt{\gamma}} \odot g$ ←Calculation parameter update；

$\theta = \theta + \Delta\theta$ ←Update parameter；

end while

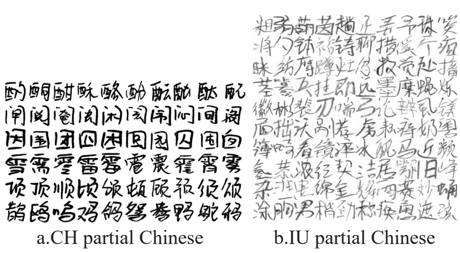
return σ；

3 Experiment and Data Analysis

3.1 Datasets and Preprocessing

The handwritten Chinese characters datasets CASIA-HBDB1.1 (CH) and the datasets IAHCC-UCAS2016 (IU) were used for experiments. The CH datasets contain 3755 GB2312 first-class Chinese characters, which are handwritten by 300 different people, and the IU datasets contain 2811 GB2312 first-class Chinese characters, which are handwritten by 115 different people. In order to validate the improved model proposed in this paper, 15 sets of similar data samples of datasets CH are randomly selected, each set contains 10 similar samples, and each sample pool contains 240 training sets and 60 test sets; meanwhile, 15 sets of similar data samples of datasets IU are randomly selected as the validation set. The sample datasets of CH partial Chinese characters are shown in Fig.4a, the sample datasets of IU partial Chinese characters are shown in Fig.4b.

In data preprocessing, the training, testing and validation datasets are first randomly shuffled. Then, the dataset in GNT format is converted to an image dataset in PNG format and extracts the labels. Finally, by Formula (1), the size of each image is normalized to 64×64, and the pixels are normalized to the interval of [0,1].



a.CH partial Chinese character sample b.IU partial Chinese character sample

Fig.4 Sample of some Chinese characters in the datasets
图4 数据集部分相似汉字样本

3.2 Experimental

3.2.1 Contrast method

To demonstrate the effectiveness of the proposed improved CNN model, a comparison is made with the following baselines.

Baselines can be divided into two categories: typical deep learning methods (MQDF and MCDNN), and improved deep learning methods (HSP-DCNN, DropSample-DCNN).

MQDF[15]: extracts gradient direction features from binary images and grayscale images, extracts stroke direction features online from the pen-down trajectory and pen-lift trajectory, and uses the modified quadratic discriminant function for classification.

MCDNN[16]: uses GPU to train multiple CNNs, and then performs a simple average integration of all CNN outputs, and the final result is the recognition result.

HSP-DCNN[17]: performs discriminative learning on a modified quadratic discriminant function and proposes a new dynamic marginal regularization, with excellent training speed and accuracy on large-scale handwritten digits datasets.

DropSample-DCNN[18]: associates each training sample with a quota function that is dynamically adjusted according to the output of the Softmax. Furthermore, a domain knowledge layer is added on top of traditional CNNs.

3.2.2 Evaluation metrics

The essence of handwritten Chinese character recognition based on CNN is to construct classifiers for Chinese character images by means of deep learning. The input Chinese character images are subjected to layer-by-layer feature extraction by CNNs to obtain the probabilities of various types of Chinese characters. The classification with the highest probability is subsequently selected for the target output, i.e., the recognition result. Therefore, the accuracy rate is adopted as the evaluation index of the model, as shown in Formula (8). The accuracy rate $R_{precision}$ can reflect the reliability of the prediction results of the CNN, and the higher the accuracy rate, the better the recognition performance of the model.

$$R_{precision} = \frac{S_{right}}{S_{right} + S_{error}} \tag{8}$$

S_{right} is the number of correctly identified samples in the given test set, and S_{error} is the number of incorrectly identified samples in the given test set.

3.2.3 Parameter settings

In setting the parameters of the CNN, all convolution kernels and pooling window sizes are fixed to 3 × 3, pooling is performed on pixel windows with stride 2, and linear activations are used for all convolutions on the activation function of the layer. The specific parameter settings are shown in Tab.1.

Continuous pre-training is performed to adjust the batch size and learning rate except the above fixed parameters in order to better improve the accuracy of the model. Specifically, with other parameters fixed, make the batch size in {64, 128, 256, 512}, and make the learning rate in {0.0005, 0.001, 0.002, 0.0025}, and arrange them. The group settings are as follows shown in Tab.2. Pre-training is performed for each group, and the accuracy is shown in Fig.5. When the batch size is 512 and the learning rate is 0.002, the model can achieve the highest accuracy. Therefore, in the following experiments, the improved model proposed in this study selects a batch size of 512 and a learning rate of 0.002. The rest of the parameters are shown in Tab.1.

Tab.1 CNN parameter settings for each layer
表1 卷积神经网络各层参数设置

Layer class	Parameter configuration/ step size	Channel number
Convolution layer1	3 × 3/1	16
Batch standardization layer 1	4 × 4	None
Pooling layer 1	3 × 3/2	16
Convolution layer 2	3 × 3/1	16
Batch standardization layer 2	4 × 4	None
Pooling layer 2	3 × 3/2	32
Convolution layer 3	3 × 3/1	64
Convolution layer 4	3 × 3/1	128
Pooling layer 3	3 × 3/2	64
Convolution layer 5	3 × 3/1	128
Batch standardization layer 3	4 × 4	None
Pooling layer 4	3 × 3/2	128
Dropout layer	0.6	None
Regression layer	None	128

Tab.2 Group setting
表2 组别设置

Group \ Parameters	Batch size	Learning rate	Group \ Parameters	Batch size	Learning rate
1	64	0.0005	9	256	0.0005
2	64	0.001	10	256	0.001
3	64	0.002	11	256	0.002
4	64	0.0025	12	256	0.0025
5	128	0.0005	13	512	0.0005
6	128	0.001	14	512	0.001
7	128	0.002	15	512	0.002
8	128	0.0025	16	512	0.0025

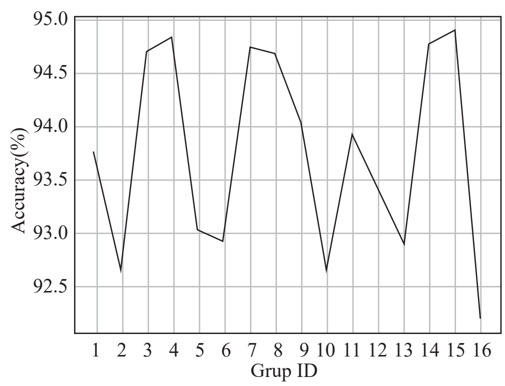
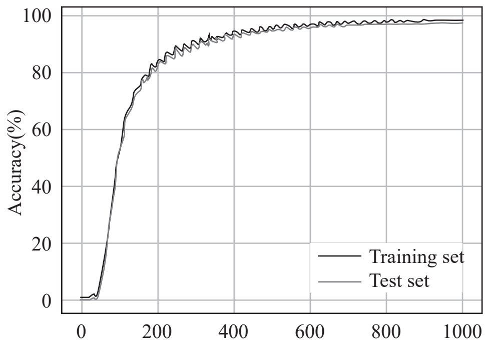


Fig.5 Results of batch size and learning rate tuning
图5 批大小和学习率调参的结果

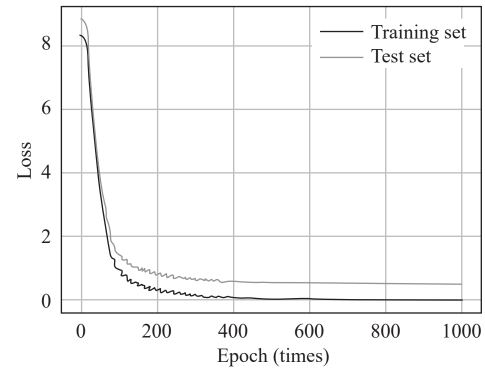
3.3 Experimental Results and Comparison

3.3.1 Overall performance

The 15 datasets of the CH datasets are taken as a population, and the debugged model parameters are input into the designed model for iterative training, and the number of iterations is set to 1000 times. The Softmax is used for activation, and then the training output range is converted to the interval [0, 1], and the output value is obtained as the probability of each classification sample. After the forward propagation is over, backpropagation is performed with the average loss of a batch of data, the accuracy is output, and the regularized cross-entropy function is used as the loss indicator of the model. In addition, the learning rate is added to the RMSprop optimizer to optimize the model to avoid overfitting of the model. The overall training and testing accuracy and loss values of the model are shown in Fig.6a and Fig.6b, respectively. With the increase of the number of iterations, the accuracy of the model gradually increased and finally stabilized, around 98%; the loss value gradually decreased and stabilized, around 0.01. At the same time, under the multi-iteration training of the model, there is no overfitting between the test results and the training results. The order of the sample datasets is randomly shuffled. Therefore, some level of volatility is within the normal range. In terms of overall performance, the improved CNN proposed in this study has a good effect on Chinese character recognition.



a.Accuracy



b.Loss value

Fig.6 Overall model accuracy and loss values
图6 模型整体准确率和损失值

3.3.2 Comparative experiments

The improved CNN model proposed in this study is named I-CNN, and the results under the test of CH datasets are compared with the four models in 3.2.1. During the identification process, five groups of 15 datasets in the CH datasets are randomly selected for comparative experiments, and the locally optimal parameters are adjusted according to the corresponding literature. The obtained accuracy and test time results of each tests set are shown in Tab.3.

Tab.3 Comparison of the performance of the different models in the CH datasets
表3 CH数据库中不同模型的性能对比

Models / Datasets	MQDF		MCDNN		HSP-DCNN		DropSample-DCNN		I-CNN	
	Accuracy (%)	Test time (s)	Accuracy (%)	Test time (s)	Accuracy (%)	Test time (s)	Accuracy (%)	Test time (s)	Accuracy (%)	Test time (s)
Group 1	94.87	23860	94.46	24721	95.94	110922	97.28	100384	98.55	22524
Group 2	94.63	23750	94.92	24366	95.59	110968	96.96	11626	97.65	22120
Group 3	94.86	23854	94.07	24445	96.19	11744	97.05	11452	98.31	22830
Group 4	94.87	23271	94.14	24765	96.45	11942	96.31	11810	97.75	22456
Group 5	94.43	23099	94.05	24592	95.33	11186	96.89	11288	98.87	22894

MQDF extracts gradient features, and it is prone to gradient disappearance without adding an optimizer. Therefore, the accuracy rate is the lowest among the five models in most cases; the disadvantage of MCDNN is that it does not make good use of prior domain knowledge to transform the data, and domain knowledge cannot be trained by CNN, so the accuracy of the model cannot be improved to a high level; although HSP-DCNN and DropSample-DCNN combine prior domain knowledge to perform operations such as direction transformation on the data, the storage capacity is significantly increased, which greatly increases the test time; the I-CNN model proposed in this study combines the advantages of the previous models and achieves an excellent result in both accuracy and time in each set of tests.

Meanwhile, in order to better verify the effectiveness of the model, the IU datasets are subjected to handwritten Chinese character recognition experiments as a way to verify the recognition ability of the model. Since the IU datasets are slightly smaller than the CH datasets, the value of Dropout is modified to 0.5 in order to enhance the generalization ability, and the number of iterations is set to 1000 as in training and testing, and the results of the validation of the IU datasets are compared with the four models in 3.2.1, and the results are shown in Tab.4. It can be seen that the proposed I-CNN model has the highest recognition accuracy, which further validates the effectiveness of the model.

Tab.4 Comparison of the performance of the different models in the IU datasets
表4 IU数据集中不同模型的性能比较

	MQDF	MCDNN	HSP-DCNN	DropSample-DCNN	I-CNN
Accuracy(%)	97.44	97.96	98.21	98.46	99.07

3.3.3 Ablation analysis

To better understand the proposed improved model, the key parts of the further test model (batch standardization module and RMSprop optimizer) are applied to the experiment. Tab.5 shows the corresponding test results in the CH datasets, and Tab.6 shows the corresponding validation results in the IU datasets, where CNN refers to the neural network model with the batch normalization module and RMSprop optimizer removed in this study. CNN_{BN} refers to the neural network model using only the batch normalization module, and $CNN_{RMSprop}$ refers to the neural network model using only the RMSprop optimizer. It can be seen from the test results that using only the CNN model can achieve better accuracy than other typical deep learning models. This may be due to the use of a multi-layer deep CNN, which optimizes the feature extraction performance of the CNN from multiple layers. While the neural network model using only batch normalization module achieves better recognition. The reason is that the batch normalization module normalizes each batch of data in each layer during forward propagation, so that the data is passed more efficiently. Similarly, the neural network model using only the RMSprop optimizer achieves good recognition. This is thanks to the optimizer's use of gradient-averaged exponential smoothing, which

Tab.5 Comparison of the performance of the different models in the CH datasets
表5 CH数据集中不同模型的性能比较

	MQDF	MCDNN	HSP-DCNN	DropSample-DCNN	CNN	CNN_{BN}	$CNN_{RMSprop}$
Accuracy(%)	94.73	94.32	95.90	96.89	95.81	97.18	97.23

Tab.6 Comparison of the performance of the different models in the IU datasets
表6 IU数据集中不同模型的性能比较

	MQDF	MCDNN	HSP-DCNN	DropSample-DCNN	CNN	CNN_{BN}	$CNN_{RMSprop}$
Accuracy(%)	95.38	96.73	96.89	97.17	95.63	97.23	97.81

effectively solves the problem of very small late-stage gradients. Overall, all parts of the proposed improved model achieve better performance than others, both in the CH datasets, and in the IU datasets, demonstrating the effectiveness of the model with the addition of the batch normalization module and the RMSprop optimizer.

4 Conclusions

For the existing problems in the field of handwritten Chinese character recognition, such as low accuracy, slow running speed, easy disappearance of gradients, etc., an improved CNN model is designed in this study. A batch normalization processing layer is added to the designed deep CNN, regularization and RMSprop gradient optimizer are added to the training. After pre-training and selecting parameters, in the experiment, the overall experiment of the model, the comparison experiment with other deep learning models, and the ablation experiment of the batch normalization module and the RMSprop gradient optimizer are carried out, which confirms the effectiveness of the model from all aspects.

Although the improved CNN used in this study has a good effect in the recognition of handwritten Chinese characters, there are still shortcomings. The following research is still needed: 1) How to reduce the parameters of the CNN while stabilizing the accuracy; 2) How to further improve the recognition speed without affecting the accuracy; 3) How to recognize the continuous handwritten text based on the global.

References

[1] ZHAO Y, ZHANG X, FU B, et al. Evaluation and Recognition of Handwritten Chinese Characters Based on Similarities [J]. Applied Sciences, 2022, 12(17): 8521.

[2] WANG Xue-jiao. Chinese Character Recognition Design in Guide Space [J]. Packaging Engineering, 2019, 40(8): 68-71, 129.
王雪娇. 导视空间中汉字识别设计[J]. 包装工程, 2019, 40(8): 68-71, 129.

[3] ZENG Zhen, LIAO Xiang, LV Xi. Exploration of Chinese Character Font Design Based on Machine Learning [J]. Packaging Engineering, 2020, 41(18): 48-52.
曾真, 廖祥, 吕曦. 基于机器学习的汉字字体设计探索[J]. 包装工程, 2020, 41(18): 48-52.

[4] LUO Xue-yang, CAI Jin-da. Research on Image Classification Algorithm Framework Based on Deep Learning [J]. Packaging Engineering, 2021, 42(21): 181-187.
罗雪阳, 蔡锦达. 基于深度学习的图像分类算法框架研究[J]. 包装工程, 2021, 42(21): 181-187.

[5] CHAN K H, IM S K, KE W. Multiple Classifier for Concatenate-Designed Neural Network [J]. Neural Computing and Applications, 2022, 34(2): 1359-1372.

[6] PENG D, JIN L, LIU Y, et al. PageNet: Towards End-to-End Weakly Supervised Page-Level Handwritten Chinese Text Recognition [J]. International Journal of Computer Vision, 2022, (130): 2623-2645.

[7] HE B, SHEN L, WANG H, et al. Finger Vein De-noising Algorithm Based on Custom Sample-Texture Conditional Generative Adversarial Nets [J]. Neural Processing Letters, 2021, 53(6): 4279-4292.

[8] YANG X, HE D, ZHOU Z, et al. Improving Offline Handwritten Chinese Character Recognition by Iterative Refinement [C]// Proceedings of 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). IEEE, 2017, 1: 5-10.

[9] BI N, CHEN J, TAN J. The Handwritten Chinese Character Recognition Uses Convolutional Neural Networks with the GoogLeNet [J]. International Journal of Pattern Recognition and Artificial Intelligence, 2019, 33(11): 1940016.

[10] SHI P, DUAN M, YANG L, et al. An Improved U-net Image Segmentation Method and Its Application for Metallic Grain Size Statistics [J]. Materials, 2022, 15(13): 4417.

[11] XIAO Z X. Research and Implementation of Offline Handwritten Chinese Character Recognition Algorithm Based on Deep Learning [D]. Chengdu: University of Electronic Science and Technology, 2021.
肖正欣. 基于深度学习的离线手写汉字识别算法研究与实现 [D]. 成都: 电子科技大学, 2021.

[12] GUO J, HAN K, WU H, et al. Cmt: Convolutional Neural Networks Meet Vision Transformers [C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022, (7): 12175-12185.

[13] KAUL A, RAINA S. Support Vector Machine Versus Convolutional Neural Network for Hyperspectral Image Classification: A Systematic Review [J]. Concurrency and Computation: Practice and Experience, 2022, 34(15): 6945.

[14] ABDOU M A. Efficient Deep Neural Networks Techniques for Medical Image Analysis [J]. Neural Computing and Applications, 2022, 7(34): 5791-5812.

[15] CHEN J, YU H, MA J, et al. Text Gestalt: Stroke-Aware Scene Text Image Super-Resolution [C]// Proceedings of the AAAI Conference on Artificial Intelligence. 2022, 36(1): 285-293.

[16] KABIR H M D, ABDAR M, KHOSRAVI A, et al. Spinalnet: Deep Neural Network With Gradual Input [J]. IEEE Transactions on Artificial Intelligence, 2022, (7): 1-13.

[17] LIU X, HU B, CHEN Q, et al. Stroke Sequence-Dependent Deep Convolutional Neural Network for Online Handwritten Chinese Character Recognition [J]. IEEE Transactions on Neural Networks and Learning Systems, 2020, 31(11): 4637-4648.

[18] XIE Z, SUN Z, JIN L, et al. Learning Spatial-Semantic Context with Fully Convolutional Recurrent Network for Online Handwritten Chinese Text Recognition [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 40(8): 1903-1917.

Main Author

WANG Jian-hua, born in 1987. He got the master degree and now is an associate professor. His main research interests are printing packaging and art design.
王建华（1987年-），硕士，副教授；主要研究方向为印刷包装与艺术设计。

Prof. LI Ben-jian, born in 1976. He got the master degree and his main research interest is digital media art.
李本建（1976年-），硕士，教授；主要研究方向为数字媒体艺术。