

Recognition of Printed Mathematical Formula Symbols Based on Convolutional Neural Network

Zi-long WAN¹, Kai FAN^{2,*}, Qi-hui WANG² and Sheng-nan ZHANG¹

¹School of Software, Shenyang University of Technology, Shenyang, 110870, China

²School of Information Science and Engineering, Shenyang University of Technology, Shenyang, 110870, China

*Corresponding author

Key words: Convolutional neural network, Formula symbol recognition, Deep learning.

Abstract. Printed mathematical formula recognition is a topic research region in OCR. But the diversity of fonts and sizes of mathematical symbols, as well as incorrect segmentation and stroke destruction of touching symbols lead to the difficulty of feature extraction and low recognition rate. In this paper, we constructed a convolutional neural network to recognize formula symbols, and determined the optimal parameters of the network through a large number of comparative experiments. Two convolution layers and sampling layers deepened the number of network layers and improved the recognition rate to a certain extent. Convolution kernels with fixed size extracted gradient information effectively, and ReLU activation function and dropout connection mode reduced the degree of over-fitting and gained the better generalization ability of the network. The experimental results show that the presented method can improve the recognition of printed formula symbols.

Introduction

The recognition of printed mathematics formula is an important research problem in the field of OCR. It uses image processing and pattern recognition technology to transform formulas inputted as images into editable symbols, and then realizes formula reuse. Although the current OCR technology has reached a high level in general text recognition, the recognition effect of mathematical formula is not ideal. The reasons involve many aspects, such as the complex structure of mathematical formulas, various fonts, segmentation for touching symbols and symbol recognition. This paper mainly focuses on the problem of formula symbol recognition.

The traditional recognition methods for formula symbols are mainly based on manual feature extraction, namely, the symbol recognition is achieved by selecting features with common, stable and good classification performance in a class of symbols, for example, the commonly used statistical features such as number of holes, pixel ratio, intersection features, etc.

Deep learning is a machine learning method widely used in speech and image recognition in recent years, and its advantage is that can overcome the shortcomings of manual feature extraction[1-2]. By building a deep network model with multiple hidden layers and using a large number of training data, the more useful features may be learned, and then the accuracy of symbol recognition can be improved [3]. Convolutional neural network (CNN), which is an important model in depth learning, has a wide range of applications, including image super-resolution[4], image recognition[5], style migration[6] and other aspects. In this paper, CNN is applied to recognize formula symbols.

Constructing CNN for Mathematical Symbol Recognition

CNN is a multi-layer neural network, and it generally consists of an input layer, an output layer, alternate multiple convolution layers and sampling layers. The CNN constructed for formula symbol recognition in this paper is shown in Fig.1, which has two convolution layers and two sampling layers. Wherein, the input layer corresponds to the symbol image to be recognized, and

considering the size of most formula symbols in printed documents, that of the symbol image is assumed to be 28×28 . The convolution layer is composed of several feature maps, and each feature map corresponds to one convolution kernel, which is responsible for extracting a kind of feature. The sampling layer is used to subsample the features obtained from the previous convolution layer, so that the extracted features have scaling invariant. The output layer is a fully connected layer adjacent to the last sampling layer, and the number of output nodes is that of targets to be classified.

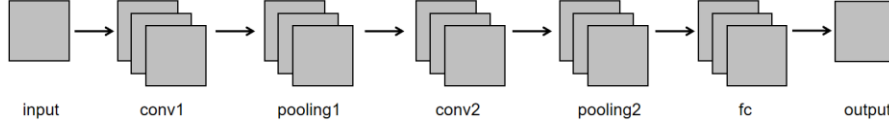


Figure 1. CNN for formula symbol recognition.

In the CNN shown in Fig.1, two convolution layers are similar and their core work is to perform convolution operation and non-linear transformation on the feature maps outputted from the previous layer, and then obtain the feature maps of the current layer. Taking the third layer as an example, the size of the convolution kernel is 5×5 , the number of feature maps is 40, and its convolution operation is shown in Eq.1.

$$F_3(Y) = W_3 * F_2(Y) + B_3 \quad (1)$$

Where $F_2(Y)$ and $F_3(Y)$ represent the outputs of previous layer and current layer respectively. W_3 are 40 convolution kernels with 5×5 , which can be understood that W_3 convolutes the image 40 times. B_3 is a bias value, and it is also a 40-dimensional vector.

After convolution operation, the results are inputted into ReLU function for non-linear transformation. ReLU is a modified linear unit, which is a common activation function in artificial neural networks. It can improve the efficiency of gradient descent and reverse propagation, the gradient explosion and disappearance are further avoided. The formula of ReLU is shown in Eq.2.

$$F_3(Y_1) = \max(0, F_3(Y)) \quad (2)$$

Where $F_3(Y)$ and $F_3(Y_1)$ represent the input and the output of activation function respectively.

The two sampling layers in Fig.1 are also similar, and they all use the maximum pooling method for sampling. Experiments show that 2×2 sampling kernel of maximum pooling is the best. Compared with the mean pooling, maximum pooling can reduce the deviation of the estimated mean caused by the error of convolution layer parameters, and retain more information so as to improve the recognition rate.

The fifth layer is a fully connected layer, and its calculation formula is similar to Eq.1. The fully connected layer is related to the symbol category to be classified by deep learning. In this paper, the output layer is constructed in the simplest way, namely, for the i th category, only the output of the i th unit of the output layer is 1, and others are 0.

The output layer adopts dropout connection mode. Dropout randomly makes weights of some hidden nodes in the network not work during the network training. The weights of non-working nodes are not updated for the time being, but they are still retained. Its purpose is to decrease the over-fitting of the network.

The learning method of the network adopts the stochastic gradient descent algorithm, and it can minimize the loss of the sample and obtain the global optimal solution.

Optimizing Network Parameters

In order to explore the influence of different network parameters on the recognition rate of formula symbols, such as network depth, number and size of convolution kernels. This paper uses the deep learning framework Caffe to train the above network model on different data sets.

Data Set

At present, image databases of formula symbols are still rare, so the training and testing of the presented CNN are carried out on the self-made data set. The data set contains 134 categories and a total of 6,300 formula symbol images, which covers commonly used mathematical symbols. Before the image is inputted into the network, a series of image preprocessing such as binarization, filtering, refinement, and normalization are performed. Partial examples of the data set are shown in Fig.2.

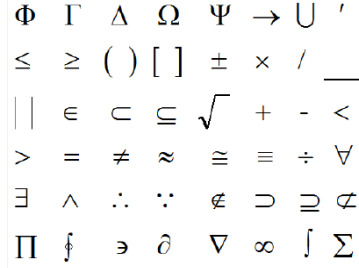


Figure 2. Some formula symbols.

Determination of Network Structure Parameters

Aiming at the influence of network structure parameters on recognition rate, we have carried out relevant tests in this paper. The first is the experiment on the size and number of convolution kernels, and the results are shown in Table 1.

Table 1. Experiments on the size and number of convolution kernels.

No.	Conv1		Sampling1	Conv2		Sampling2	Fc	Output	Accuracy
	Size of kernel	Number of kernel	Size of kernel	Size of kernel	Number of kernel	Size of kernel	Number of neurons	Number of categories	
1	3×3	20	2×2			2×2	128	134	90.17%
2	5×5	20	2×2			2×2	128	134	92.26%
3	9×9	20	2×2			2×2	128	134	92.71%
4	3×3	20	2×2	5×5	40	2×2	128	134	92.23%
5	5×5	20	2×2	5×5	40	2×2	128	134	94.15%
6	9×9	20	2×2	5×5	40	2×2	128	134	92.41%

The results show that when the number of classifications is 134, and the size of the input image is 28×28, the most suitable structure is that two convolution layers and two sampling layers are alternately connected, and the size of the convolution kernel and the sampling kernel is 5 × 5 and 2 × 2 respectively.

After determining the size of convolution kernel, the number of convolution layers and sampling layers, the second experiment is about the relationship between the number of convolution kernels and the number of classifications, the results are shown in Table 2.

Table 2. Experiments on the number of f convolution kernels and classifications.

No.	Conv1		sampling1	Conv2		sampling2	Fc	Output	Accuracy
	Size of kernel	Number of kernel	Size of kernel	Size of kernel	Number of kernel	Size of kernel	Number of neurons	Number of categories	
1	5×5	16	2×2	5×5	24	2×2	128	134	92.41%
2	5×5	16	2×2	5×5	32	2×2	128	134	92.15%
3	5×5	18	2×2	5×5	36	2×2	256	134	92.56%
4	5×5	20	2×2	5×5	40	2×2	256	134	94.25%
5	5×5	24	2×2	5×5	48	2×2	256	134	91.76%
6	5×5	40	2×2	5×5	80	2×2	256	134	93.30%

Based on the structural parameters determined in Table 1, Table 2 shows that if the number of classifications is 134, in order to improve the recognition rate, the number of convolution kernels and neurons in each layer must be increased accordingly. However, with the increase of the number of parameters, convergence time of network training becomes longer correspondingly, when the

parameters are increased to a certain extent, the recognition rate will decrease. In Table 2, the best recognition rate can be obtained by using 20 and 40 convolution kernels respectively in two convolutional layers.

In the network training process, the back propagation algorithm was applied to update the network parameters. At the same time, in order to shorten convergence time, we divided the training set into 168 batches containing 32 samples, and each update only used a batch of data instead of the entire training set.

Selection of Connected Mode

Dropout is a widely used network connection mode, and it can temporarily stop updating the weights of neurons in the fully connected layer with a certain probability during training. In this way, the updating of weights no longer depends on the interaction of hidden nodes with fixed relationships, and the situation, in which certain features are effective only under other specific features, can be avoided. Thus, the over-fitting problem is effectively suppressed and the generalization ability of the network is enhanced. Keeping other network parameters unchanged, contrast experiments with and without dropout connection are shown in Table 3. Among them, the number of iterations of the network is 10,000, and the node loss rate of dropout is set to 50%.

Table 3. Effect of connection method on accuracy.

No.	Dropout connection	Recognition rate	Mean square error
1	no	90.67%	0.125
2	yes	94.01%	0.280

Experimental Results and Analysis

In this paper, we use deep learning framework Caffe to train network. After 10,000 iterations, the results are shown in Fig.3- Fig.6. Fig.3 and Fig.4 show the change of accuracy and loss with the number of iterations during the network training. It can be seen that the accuracy and loss gradually became stable with the increase of iteration times, and finally reached a certain height or achieved convergence. Fig.5 and Fig.6 show the change of learning rate with the number of iterations and training time, and learning rate decreased steadily until the end of the iteration. The above results indicate that the network parameters are set reasonably, the experimental results are reliable, and the optimal performance under the above network parameters has been achieved.

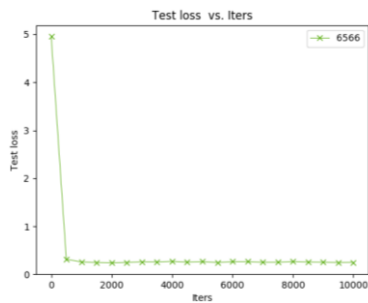


Figure 3. Change of loss with iterations.

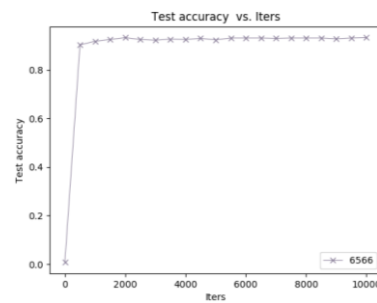


Figure 4. Change of accuracy with iterations.

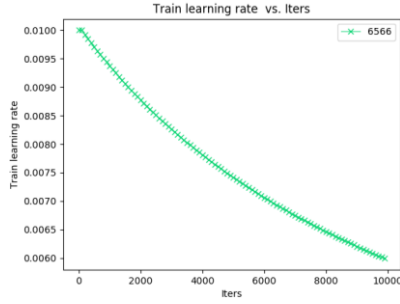


Figure 5. Change of learning rate with iterations.

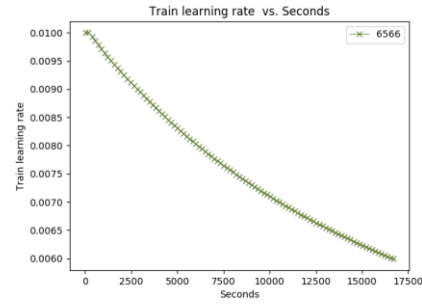


Figure 6. Change of learning rate with training time.

Using the above network to test the data set, we also find that the following images are classified incorrectly, and some images are selected to show in Fig.7.

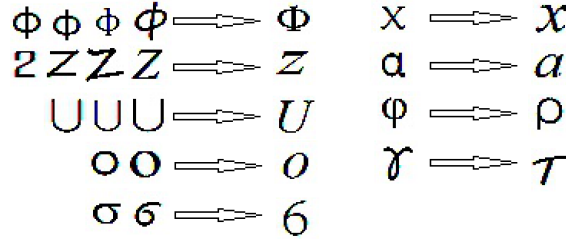


Figure 7. Partial images with misclassification

In Fig.7, the recognition for symbols with very similar shapes is weak, but the symbols with distinct differences are better identified. The reason is that the presented algorithm cannot understand the meaning of the symbol in the context, and classification and recognition for single formula symbol are only carried out purely in terms of shapes.

Conclusion

Taking advantage of characteristics of convolutional neural network in weight sharing and less learning parameters, this paper presents a 5-layer convolutional neural network for recognizing formula symbols. Compared with traditional methods, the accuracy has been significantly changed. Due to the use of ReLU activation function and dropout connection, the over-fitting of the function is effectively avoided and the generalization ability of network is improved. However, it should be pointed out that although the method can accurately classify most of the printed formula symbols, it has no more advantages for symbols without obvious shape distinction or with too serious distortion. How to distinguish formula symbols with subtle differences through the analysis of context semantics is the focus of future work.

References

- [1] Lingzheng Bu, Hongdong Wang, Meiqiang Zhu, et al, Multi-source digital recognition based on improved convolutional neural network, J. Journal Computer Applications. 38(2018)55-60 (in Chinese).
- [2] Zhifeng Xie, Jiaping Wu, Shuhan Zhang, et al, Intelligent recognition method of smoke Code based on depth Neural Network, J. Journal Computer-Aided Design & Computer Graphics. 31(2019)111-117 (in Chinese).
- [3] Shengnan Zhang, Qihui Wang, Lianqian Niu, et al, Handwritten symbol recognition of data form image based on convolutional neural network, J. ICIC Express Letters, Part B: Applications. 7(2016)1637-1643.
- [4] Dong C, Loy C C, He K M, et al, Image super-resolution using deep convolutional networks, J. IEEE Transactions on Pattern Analysis and Machine Intelligence. 38(2016)295-307.

- [5] Lianqiang Niu, Xiangzhen Chen, Shengnan Zhang, et al, Model construction and performance analysis for deep consecutive convolutional neural network, J. Journal of Shenyang University of Technology. 38(2016)662-666 (in Chinese).
- [6] Honglin Liu, Renjun Shuai, Method of Fast Neural Style Transfer with Spatial Constraint, J. Computer Science. 46(2019)283-286(in Chinese).