

# Chinese Font Recognition Based on Convolution Neural Network

Yifan Chang

University of Electronic Science and Technology of China, Chengdu 610000, China

changyifan123@qq.com

**Abstract.** In the traditional OCR text recognition system, recognition rate is low with many errors. Also, it can not effectively identify the different text fonts. It's a long-standing problem. Modern text recognition system should be for different fonts within accuracy to identify the text and font information. The traditional methods are mostly based on feature extraction, including local features extraction and global features extraction. Proposes a Convolution Neural Network[1] based on deep learning to deal with Chinese font recognition. Compared with the previous methods, this method has high recognition rate and high speed and is suitable for complex applications.

**Keywords:** Font Recognition; Image Processing; CNN

## 1. Introduction

There is an increasing demand for the modern office to convert paper documents to electronic documents. The current system for this is the OCR system, which is called optical character recognition system. However, the OCR system [2] mainly focuses on the identification of texts and does not provide a corresponding solution to the resumption of the layout of the publications. The main difficulty for layout recovery is the recovery of text fonts. There are mainly several methods for Chinese character font recognition, but all are based on the method of artificial feature extraction. The past methods are mainly divided into two categories. The first is holistic analysis. These methods all have one important flaw: if the text contains different font information, this kind of scenario can not use holistic analysis. The second is individual analysis. Individual analysis method is for single character image. This method needs to acquire the image of a single Chinese character. Obviously this method can solve the case if the text contains some characters using different fonts. It can be seen from the point of view that the individual method is obviously more flexible and suitable for many situations. This article will use the individual method to achieve Chinese font recognition. However, it should be noted that the traditional methods are basically methods of image processing, and the time complexity is high. In summary, we can see the lack of traditional Chinese font recognition method. Because of the complexity of Chinese characters, feature extraction methods can not handle the changing Chinese shapes. Feature extraction methods require human

expert define important feature locations. Moreover, no uniform standard can be given for which feature points are important. Therefore, this paper proposes a convolution neural network font recognition method. The CNN network can automatically generate features based on the convolution kernel and the pooling layer. This avoids instability and blind spots in the extraction by artifacts. We generate experimental data through computer embedded fonts, and then train convolution neural network to achieve the final result. This system shows high stability and accuracy.

## 2.Principle of font recognition based on CNN

Unlike traditional feature extraction methods, convolutional neural networks extract features via convolution kernels. Each neuron is connected to a local receptivity field[3] of the previous layer and the local features are computed by the convolution kernel. The model generates the feature plane with the movement of the convolution window, and each feature plane shares a convolution kernel, so as to share the weight and reduce the weight. CNN is mainly used to identify two-dimensional images. Sharing weights are obtained using supervised learning and backpropagation algorithms, eliminating the need to manually extract features. Therefore, CNN has the advantage of learning parameters from training data. Convolution neural network is usually divided into multiple layers. In addition to the fully connected layers, the convolution layer and the pooling layer are the most important structures in CNN, which are used for feature extraction and feature parameter processing respectively.

As shown in Figure 1, the input data is a  $28 \times 28$  pixel picture. Convolution layer with  $5 \times 5$  convolution kernel. Translating the convolution kernel yields a  $24 \times 24$  feature map. Three convolution kernels are used here, resulting in three different features. The pooling layer[4] uses the output of the first layer as the input of the pooling layer, and uses  $2 \times 2$  averaging methods to obtain three  $12 \times 12$  pooling layer data. The full connection layer, like the traditional neural network, uses the loss function to calculate the loss, and then uses the back propagation algorithm to correct the weights to achieve the purpose of training the neural network.

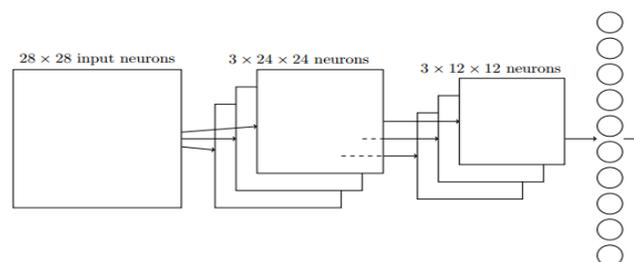


Figure 1

## 3.Generate training data

I use python and pygame library to generate Chinese image data. The core code is as follows.

```
#encoding: utf-8
import os
import pygame
chinese_dir = 'chinese'
if not os.path.exists(chinese_dir):
    os.mkdir(chinese_dir)
pygame.init()
start,end = (0x4E00, 0x9FA5)
for codepoint in range(int(start), int(end)):
    word = unichr(codepoint)
    font = pygame.font.Font("msyh.ttc", 64)
    rtext = font.render(word, True, (0, 0, 0), (255, 255, 255))
    pygame.image.save(rtext, os.path.join(chinese_dir, word + ".png"))
```

The generated data is shown in Figure 2 and the font is bold.



Figure 2

#### 4. Construct of CNN

I choose TensorFlow[5] to set up the convolution neural network. TensorFlow is an open source software library for numerical computation using data flow graphs. Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) communicated between them. The flexible architecture allows you to deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile device with a single API. TensorFlow was originally developed by researchers and engineers working on the Google Brain Team within Google's Machine Intelligence research organization for the purposes of conducting machine learning and deep neural networks research, but the system is general enough to be applicable in a wide variety of other domains as well.

The input data is a 64 \* 64 pixel map. The first layer uses 32 12 \* 12 convolution kernels to extract features with a sliding step of 2. The pooling layer uses a 2 \* 2 averaging method. CNN process shown in Figure 3.

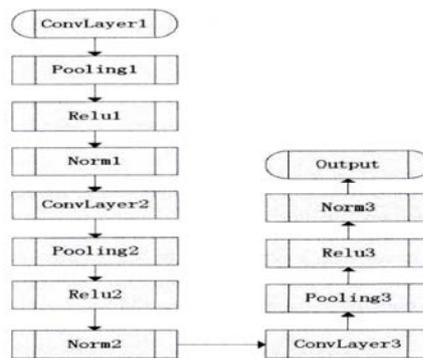


Figure 3

### 5.Experimental data and analysis

This experimental environment is the Intel i5-6500 processor, NVIDIA GTX1060 graphics processor, memory 16GB. Experimental data consists of 200,000 characters, of which 150000 as training data, 50000 as validation data. The CNN iterated 60,000 iterations, generating validation results of error rates and accuracy rates every 1,000 times, running under Ubuntu 16.04 for 8 hours. The error rate and the correct rate are shown in Figure 4. The horizontal axis shows the accuracy and the error rate every 1000 iterations. It can be seen from Figure 4 that CNN has a high recognition rate for man font.

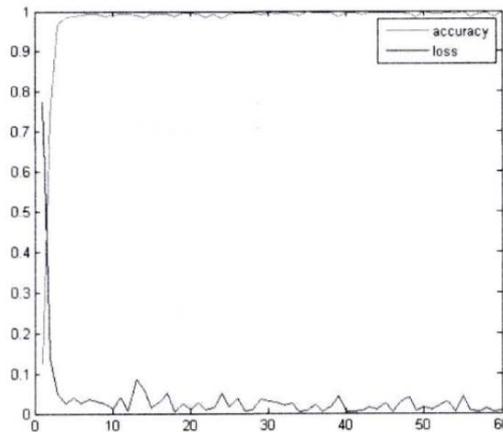


Figure 4

### 6.Conclusions

This paper presents a Chinese character font recognition method based on CNN. The construction of CNN and the method of generating training data are explained in detail. The method solves the problem of human-based extraction features in the past font recognition system. The method achieves a high recognition rate for the printed Chinese fonts, and has certain practical significance for solving the practical identification of Chinese fonts.

**References**

- [1] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017, 60(2):2012.
- [2] Xing X H, Guo-Hua G U. Method of Quickly Recognizing Vehicle Plate Based on Pattern Matching and Characteristic Dot Matching[J]. Optoelectronic Technology, 2003, 23(4):268-270.
- [3] Michael A.Nielsen, Neural Networks and DeepLearning, Determination Press,2015
- [4] Michael A.Nielsen, Neural Networks and DeepLearning, Determination Press,2015
- [5] Information on <https://www.tensorflow.org/>