Proceedings of the 2nd International Conference on Computing Innovation and Applied Physics
DOI: 10.54254/2753-8818/18/20230456

Network architecture exploration for Chinese character recognition with densenet

Ruilin Dai[1]

[1] Mathematic and Applied Mathematic, Jinan University, Panyu District, Guangzhou City, Guangdong Province, 511486, China

drl20096@163.con

Abstract. Chinese character recognition can be widely used in many fields. Though Resnet and Densenet are both used in this area already, using these two networks and making a comparison on training performances between them is a ground that has not been explored. In this paper, these two methods are built and compared. Firstly, a dataset of Chinese character including 5,772 images with 28*28 size will be introduced. Next, Resnet and Densenet model pre-trained on the dataset is selected. Then fine-tuning is done to improve the accuracy of networks. After 50 epochs of training, the final result shows that Densenet is more stable compared with Resnet but less efficient with more epoches to perform well.

Keywords: convolutional neural network, DenseNet, ResNet, Chinese character recognition.

1. Introduction

Nowadays, recognition of Chinese character has been an important topic of CNN (Convolutional Neural Networks) for a long time. Resnet risen a revolution in the field of CNN. Densenet, published in 2017, performed even better than Resnet, has been used in many different research areas.

This paper will introduce some image data of the abbreviation of Chinese provinces. Based on these existing data, this article will construct densenet and resnet networks and use them for recognition training of the existing data. The accuracy and the loss rate will be output finally, and the results of these two networks will be compared.

2. Related works

Resnet has been used in Chinese character recognition. Sun, Li and Wu constructed improved inception-ResNet for recognition of Handwritten Ancient Chinese Character [1]. Huang and Zhang applied it on skew correction of handwritten Chinese character [2]. It sometimes applied in researches for comparisons. For instance, it was compared with the deep CNN method in Devenagari character recognition [3], and it was contrasted with GoogleNet in the field of malware detection [4].

Densenet also has been used in Chinese character recognition. A research refer to fuzzy attentional-based densenet in Chinese image captioning was published in 2021 [5]. Jalali and Lee combine densenet with inception models and use them for the traditional Asian character recognition [6]. But such research is rare. Not to mention a comparison of densenet and resnet in Chinese character recognition.

3. Dataset

In China, the license plate is combined with a Chinese abbreviation of the province and five English characters or numbers. This usually requires an image recognition of the Chinese abbreviation of the province.

In this paper, we use a dataset including 31 kinds of different characters (see Figure 1), all of which are different representations of provinces in China, such as "京, 冀, 沪, 浙, 粤, 苏, 鄂, 闽, 黑". The dataset includes 4251 images for training and 1521 images for testing. All of the images are grey-scale images. Each picture was 20 by 20 at first. For convenience, all of them are changed into 28 by 28. The images finally change into the shape of numpy array.

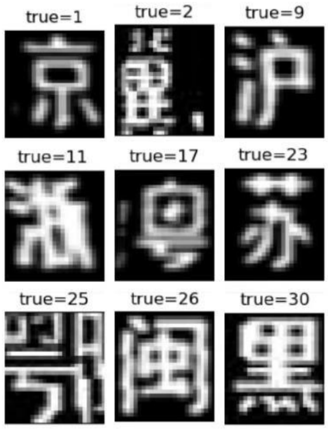

Figure 1. Preview of dataset.

4. Networks

4.1. ResNet

4.1.1. Introduction of ResNet. CNN (Convolutional Neural Networks) is a hot topic in the field of pattern recognition, especially in the field of image recognition. Published in 2015, ResNet (Residual Networks) raised a revolution in CNN by its high accuracy.

Such a high accuracy of getting feature of images of ResNet is obtained by introducing the residual unit with an identity mapping. Identity mapping allows the deep layers to directly learn the data received by the shallow layers, which helps to decline the difficulty of network convergence to a certain extent. As a result, ResNet has a better learning ability [7].

4.1.2. Method of Resnet. Instead of using every few stacked layers straightforwardly fit underlying mapping, the innovation of Resnet lies in it makes these layers fit a residual mapping. For instance, denoting the desired underlying mapping as $H(x)$. Denote $F(x)$ as path that fit residuals, called residual paths, and x as path that being the identity mapping called "shortcut" (see Figure 2).

DOI: 10.54254/2753-8818/18/20230456

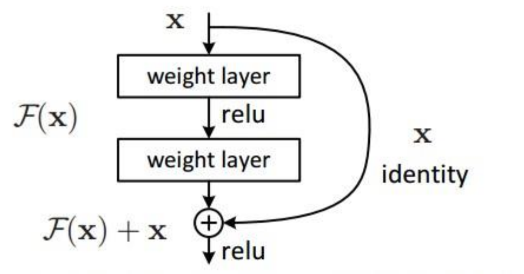

Figure 2. Structure of Residual block [8].

Then we let the myriad nonlinear layers fit another mapping :

$$F(x) = H(x) - x \tag{1}$$

So we recast the original mapping into:

$$H(x) = F(x) + x \tag{2}$$

In the assumption, the residual mapping is much easier to optimize than the original, disordered mapping [8].

DOI: 10.54254/2753-8818/18/20230456

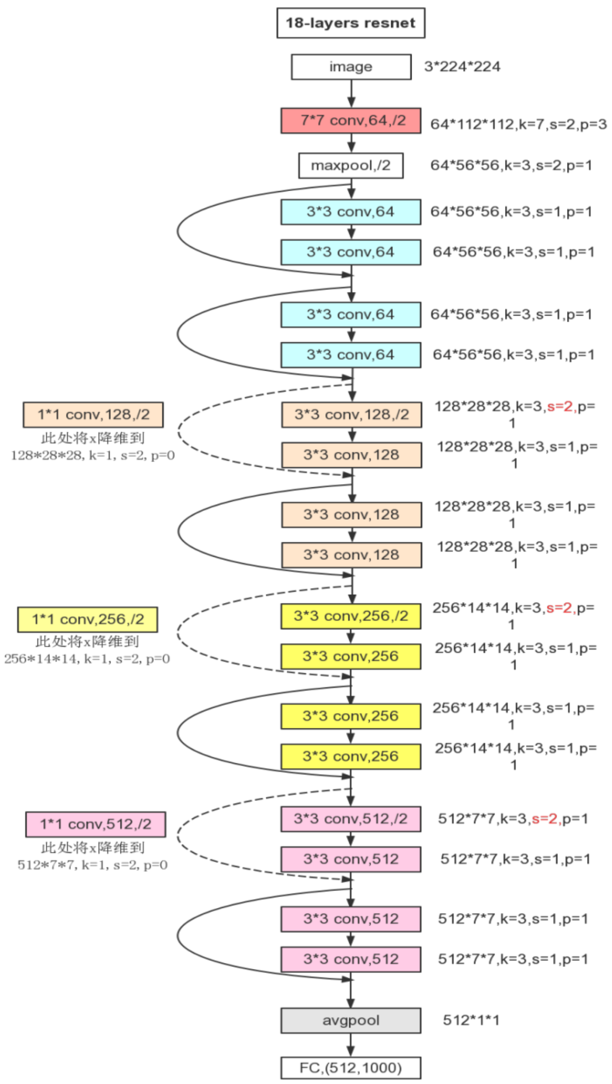

Figure 3. Structure of Resnet 18.

4.1.3. Construction of Resnet. Resnet is built following the instructions in Keras and the paper [8].

The block achieving $F(x) + x$ is called Residual block. A residual block contains two mappings, $F(x)$ and x. $F(x)$is form by Convolutional layer, Batch- normalization layer and activation layer with Relu. The paper use ‘if’ to construct a structure, making sure the output shape of $F(x)$ is the same as x. If the output shape is different, then the output will pass through a Convolutional layer with 1×1 kernel to make sure the following operations can be continued.

The structure of Resnet can be seen in Figure 3. Resnet in this paper consists of four residual blocks. For convenience, ‘for’ loop is used to simplify the code.

4.2. DenseNet

4.2.1. Introduction of Densenet. The Densely Connected Network (Densenet) is an improvement of Resnet. The crucial part been improved is called Dense Connections. This property concatenates all the outputs of preceding layers and thus improve the performance by encouraging feature reuse to the extreme [9].

4.2.2. Method of Densenet. The main difference between Densenet and Resnet is, specifically, in Resnet, layers just connected only to the layer before it, however, in Densenet (see Figure 4), layers are connected to all layers before it. The benefit of this is to realize feature reuse and to improve efficiency.

To be more specific, this paper will express this by nonlinear formular. In Resnet, here, ℓ represents the layer, x_ℓ represents the identity output of layer ℓ, H_ℓ represents a nonlinear transformation. The nonlinear formular is:

$$x_\ell = H_\ell(x_{\ell-1}) + x_{\ell-1} \tag{3}$$

In Densenet, $[x_0, x_1, x_2, \ldots\ldots, x_{\ell-1}]$ indicates the coalescence of output feature from layer 0 to layer $\ell - 1$, merging the channels from layer 0 to layer $\ell - 1$, which is different from just adding them up as Resnet. The nonlinear formular is:

$$x_\ell = H_\ell([x_0, x_1, x_2, \ldots\ldots, x_{\ell-1}]) \tag{4}$$

The main body of densenet consists of Dense Block and Transition. The implementation details of them are specified below.

4.2.3. Construction of Densenet. The Densenet is built following the instructions in Keras and the paper [10].

Dense Block do the job of concating the features of all previous layers together. It consists of a 'for' loop including the Dense layer, which is composed of bottleneck and composite function. Bottleneck can simplify the complexity of calculation and the composite function is mainly used to concate the features. There are 4 dense blocks, and the number of layers, in the order is 6,12,32, and 32, all of which include 1*1 and 3*3 convolutional layers.

The growth rate is set to be 12 after several times of fine-tuning. As mentioned, every dense layer receives features from all the previous layers. For instance, assume the number of channels connected to the input layer is k_0, then the number of channels in layer L is $k_0 + k_{L-1}$

The computation complexity is quite high without bottleneck. Each layer has feature maps joined with the previous layers in the number of k. So, to reduce computation, the bottleneck is formed by batch-normalization, ReLU and convolutional layer.

Transition layer is used to connect the Dense blocks. It is designed mainly to avoid overfitting and to compress the model. In this paper, It is achieved by batch-normalization, ReLU, convolutional layer with 1*1 convolution block and an average pooling layer to simplify the calculation. The parameter 'reduction' (called compression in code) is set to be 0.5, meaning that half of the channels are compressed before getting into the next Dense block.

In this paper, the Densenet include 4 dense blocks. Each of them with a dropout rate of 0.2. Three transition layers are used to connect dense blocks. The output goes through a batch-normalization layer and a global average pool. Then it finally outputs a number between 1 and 31 to represent the type of Chinese character.

The loss function of both Resnet and Densenet are chosen to be cross-entropy, which is commonly used in training of classification model. Cross-entropy can show the distance between the data and a particular class. It is in the following form:

$$\text{loss} = -\sum \mathrm{y\_} * \ln \mathrm{y} \tag{1}$$

where $\mathrm{y\_}$ and y represent two different matrix of probability distribution.

5. Experiment

DOI: 10.54254/2753-8818/18/20230456

5.1. *Implementation*

Growth rate of Densenet is set to be 12 after multiple times of experiments. Dropout rate is set to be 0.2. Considering there are 31 types of data, the training batch is set to be 128, and the epoch is set to be 50.

5.2. *Results and Discussion*

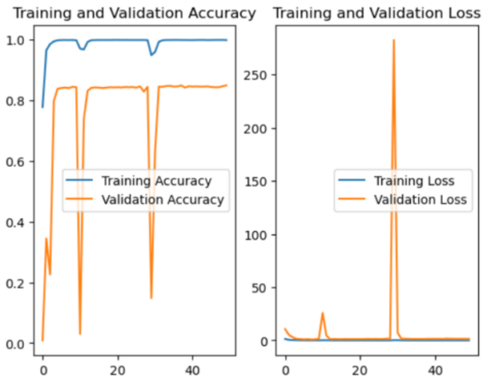

Figure 4. Accuracy and loss of Resnet.

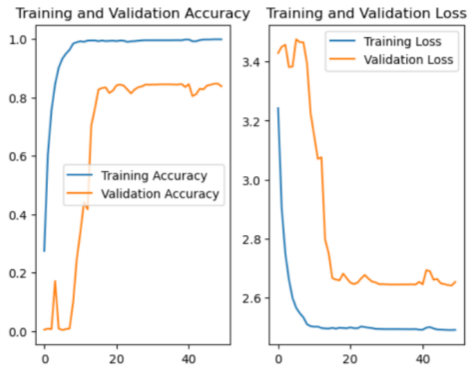

Figure 5. Accuracy and loss of Densenet.

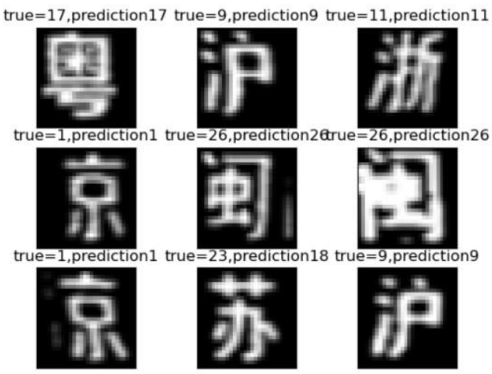

Figure 6. Random test on Resnet.

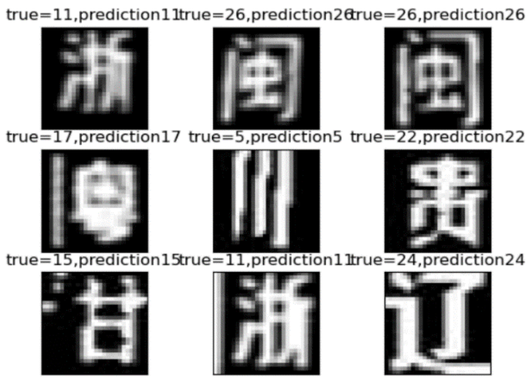

Figure 7. Random test on Densenet.

After the experiment, the final accuracy of Resnet is 0.8494 and the accuracy of Densenet is 0.8383. The loss of Resnet is 1.5015 and the loss of Densenet is 2.6525.

As it is shown in Figure 4 and Figure 5, Resnet learns faster than Densenet on the dataset, taking several epochs reaching the peak in accuracy and the lowest loss. While Densenet takes about 15 epochs.

However, it also faces a higher risk to encounter a shift change in testing. Densenet seems to not contain this weakness, performing more stable though with a bit more fluctuation than Resnet.

Totally, if the object of training is to construct a model as fast as possible with high accuracy, Resnet is a better and more efficient choice for us, and if a more stable model without suddenly increasing loss is needed, then Densenet is a better choice.

Randomly draw 9 images from the dataset and test it on the well-trained model. It can see from Figure 6 and Figure 7 that both models perform well on the dataset.

6. Conclusion

In conclusion, not surprising, Densenet has completed the goal with ideal results close to or even better than that of Resnet. Although the accuracy of Resnet model is slightly higher than that of densenet, it may be the reason of not-enough epoches. In the future, if the researchers want to apply it to the recognition of Chinese characters, increasing the types of training data is the most critical point. One way is to input more manually labeled images, and the other is to consider using cross-validation on the dataset to increase its number.

DOI: 10.54254/2753-8818/18/20230456

References

[1] Sun, Jinhu Li, Peng Wu, Xiaojun, Handwritten Ancient Chinese Character Recognition Algorithm Based on Improved Inception-ResNet and Attention Mechanism, 2nd IEEE International Conference on Software Engineering and Artificial Intelligence, SEAI 2022, 2022

[2] Huang, Zetao Zhang, Qian ,Skew correction of handwritten Chinese character based on resnet, 2019 International Conference on High Performance Big Data and Intelligent Systems, HPBD and IS 2019, 2019

[3] Patnaik, Suprava Kumari, Saloni Das Mahapatra, Shreya , Comparison of deep CNN and ResNet for Handwritten Devanagari Character Recognition, 1st IEEE International Conference for Convergence in Engineering, ICCE 2020, KIIT and School of Electronics Bhubaneswar India, 2020

[4] Khan, Riaz Ullah Zhang, Xiaosong Kumar, Rajesh , Analysis of ResNet and GoogleNet models for malware detection, JOURNAL IN COMPUTER VIROLOGY, Volume 15, Issue 1,2019, p29-p37

[5] Lu, Huimin Yang, Rui Deng, Zhenrong Zhang, Yonglin Gao, Guangwei Lan, Rushi, Chinese Image Captioning via Fuzzy Attention-based DenseNet-BiLSTM, ACM TRANSACTIONS ON MULTIMEDIA COMPUTING, COMMUNICATIONS AND APPLICATIONS, Volume 17, Number 1, 2021, p1-p18

[6] Jalali, Amin Lee, Minho , High cursive traditional Asian character recognition using integrated adaptive constraints in ensemble of DenseNet and Inception models, PATTERN RECOGNITION LETTERS, Volume 131, as of 2020, p172-p177

[7] Miao, Jun Xu, Shaowu Zou, Baixian Qiao, Yuanhua ResNet based on feature-inspired gating strategy,MULTIMEDIA TOOLS AND APPLICATIONS,Volume 81,Issue 14,2022, p19283-p19300

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun,Deep Residual Learning for Image Recognition, 10 Dec 2015 , arXiv:1512.03385

[9] Yu, Dawei Yang, Jie Zhang, Yun Yu, Shujuan Additive DenseNet: Dense connections based on simple addition operations, JOURNAL OF INTELLIGENT AND FUZZY SYSTEMS, Volume 40, Issue 3, 2021, p5015-p5025

[10] Gao Huang, Zhuang Liu, Laurens van der Maaten, Kilian Q. Weinberger ,Densely Connected Convolutional Networks, 25 Aug 2016, arXiv:1608.06993