Channel attention convolutional recurrent neural network on street view symbol recognition

Jinke Li[a], Chunyue Wang[b], Runheng Cai[c]

Jilin University, Jilin, China.

[a]lijk2019@ mails.jlu.edu.cn; [b]cchunyue@jlu.edu.cn; [c]airh2019@ mails.jlu.edu.cn

Abstract. Character recognition has always played an important role in the area of image processing. In our paper, we focus on the recognition problem of street view symbol, and propose a novel network architecture called Channel-Attention-based Convolutional Recurrent Neural Network (CACRNN) for character detection and character recognition. Different from CRNN, our method is based on convolutional neural network (CNN) structure with the help of attention guidance, which is a new way of combining CNN, recurrent neural network, and SE block. CRNN can be learned directly from sequence labels without detailed annotations, and the recognized objects are not limited in length. In the network, CNN plays the role of text detection, and recurrent neural network have a significant effect on text recognition. Among them, we use the long short-term memory (LSTM) network as the recurrent neural network. In addition, we introduce SE block and propose an effective attention layer, so that the model is able to dynamically pay attention to certain parts that contribute to the existing task. Then our model can determine the most relevant aspects, and achieve better effect of filtering useless information and removing noise. Finally, the entire network is optimized using the CTC loss function. Through experiments, we found that the recognition accuracy of CACRNN processing characters is higher than other methods.

Keywords: Street View Notation, Convolutional Recurrent Neural Network, CTC Loss Function.

1. Introduction

License plate recognition and sign recognition are both important parts of autonomous driving technology. Emerging technologies such as smart cities require house number recognition. Trademark recognition technology in e-commerce platforms is also very important. These challengeable jobs all rely on character recognition.

Nowadays researchers in related fields have done a large number of works on this technology, but there are still many problems with the recognition of characters while capturing images in real scenes. For example, external factors such as light, wear, tilt, and occlusion will interfere with character information. These will cause details to be blurred and distorted, even affect the accuracy of target positioning, then ultimately affect the recognition rate. Prior to this, many studies have attempted to enhance image brightness [1], control color deviation [2], suppress amplified noise [3], preserve details and texture information [4], and restore blurred edges [5]. However, images taken in low light often suffer from poor visibility, low contrast, color distortion, and severe noise, which can have many effects on character recognition.

Conventional character recognition methods mainly include two key parts: feature extraction and character recognition. Traditional algorithms such as edge-based detection belong to one feature extraction method. For example, De Campos *et al* [6] created an image database using multiple handcrafted features, but case sensitivity leads to lower performance. Zhang, Honggang *et al* [7] proposed edge-based detection methods, stroke width transformation methods and so on. The methods are relatively simple, but the complex background makes it difficult to segment and verify text strokes. However, when extracting features in this step, many samples need to be processed, and the process is cumbersome and will cost a large amount of time.

For the purpose of solving above problems, learning-based method has been effectively applied to the street-view symbol recognition problem. Because it has greater advantages in terms of time cost and accuracy compared with conventional methods. Note that neural networks simulate the neural connection structure of the human brain by stimulating the visual perception part of the human brain,

so they can own higher robustness and accuracy. Besides they can perform multiple levels of abstraction, analysis, and characterize data, then performs interpreting. Moreover, these networks can automatically learn features and can be trained through massive data to obtain more essential features of the data, thereby improving the accuracy of recognition.

For street view character recognition, the method based on machine learning [8] is to use learning-based methods such as convolutional neural networks to directly recognize and classify characters. CRNN [9] structure integrated the advantages of deep convolutional neural network and recurrent neural network, combined CNN, LSTM, and CTC as three methods for text recognition in images. Researchers [10] also redefined the layer as the learned residual function of the reference layer input and then obtained a residual learning framework. On the ImagaeNet test set， they simplified training of deeper networks than previously used, with a 3.57% error.

In order to solve the problem of feature extraction and recognition of characters in images, our work design a convolutional neural network based on the SE block-attention mechanism. Overall, our contributions are in three folds:

1. We propose a SE block-guided recurrent convolutional neural network. The CNN part is a simplified residual network called Resnet and the learned parameter is the residual. Shortcut connections can simplify the training of deeper networks so that residual networks are easier to optimize and can improve accuracy.

2. In order to make the network structure better, we choose LSTM and GRU to replace the conventional RNN, and compared the accuracy obtained by the two networks. In addition, we introduce the SE block-attention mechanism, in which the SE block makes full use of the information flow.

3. We evaluate our network on the SVHN dataset, and the experimental results show that our structure is able to increase the accuracy of character recognition.

2. Related works

In the past few years, researchers in this field have adopted a number of effective methods for character recognition. According to the algorithm principle, the methods of character recognition is able to be roughly divided into two categories: one is conventional methods, the other is deep learning methods.

2.1 Conventional methods

The license number image recognition method is mainly on account of the traditional feature extraction recognition method. De Campos *et al* [6] created a database of annotated images, leveraging multiple handcrafted features, but still have some problems with lower case sensitivity.

The proposed edge-based detection method applied edge detectors and texture-based methods. And the text was extracted through methods. Combined with a connected component-based approach, components were grouped into larger components until all regions were identified. Finally, there was the stroke width transformation method [7], which is relatively simple, but the complex background made it difficult to segment and verify text strokes.

2.2 Learning-based method

In 1989, LeCun *et al* proposed the LeNet-5 model on the basis of CNN. This is the first attempt to apply CNN to handwritten digit recognition. This method [11] achieved a recognition rate of 99.1% on the MNIST dataset of handwritten digits. In the 1990s, the model was applied to bank handwritten check recognition with success.

Created by Pierre *et al*, a house number recognition method based on a convolutional neural network (CNN) [8] revealed that Lp-pooling and multi-stage features can lead to good results. The results reveal that the recognition rate in the SVHN dataset was 95.1%. But there were some

drawbacks, if the image contains some areas that are too dark, halo artifacts and partial shadows will appear.

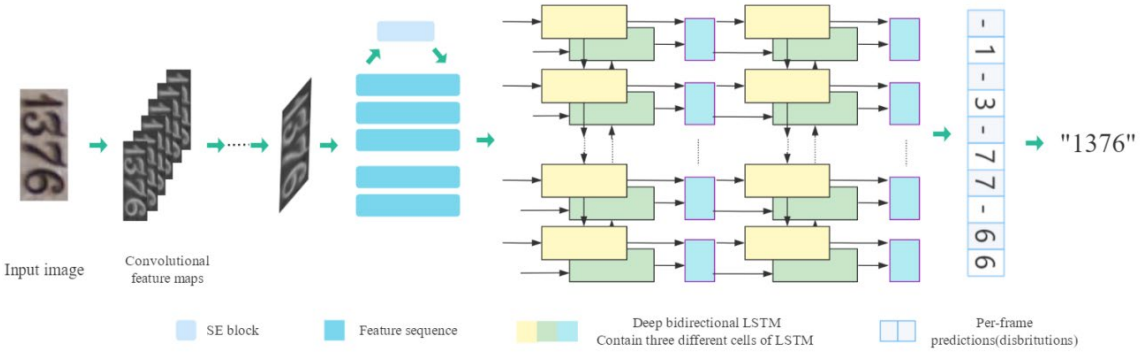

Figure 1. Overview of CACRNN to solve character recognition problem. The images containing characters are firstly extracted by CNN, and the information flow is filtered by SE Block. Then the RNN is used to identify word phrases by using the correlation between characters. Finally, the letters are spliced by CTC. ("-" is a class of characters added by CTC to represent whitespace)

The label sequence of the feature vector was obtained by Hidden Markov Model (HMM), and according to the change of the provided sequence information, a CNN-HMM hybrid model was constructed.

The recognition rate of this model [12] in the SVHN dataset is 81.07%. Because the convolutional neural network can effectively and automatically extract features and classify them, it saves a lot of time for manual feature design. A method combining multi-level pooling and warped sample techniques to apply multi-pooling and data augmentation with nonlinear transformations to CNN for multi-font PCCR was proposed. A multi-pooling layer was added on top of the final convolutional layer, using a warped sample generation technique to warp the local density of image-based Chines strokes by applying a nonlinear warping function along with the original font image. Experiment results on the SCUT-SPCCI database show that their model [13] achieve a recognition rate of 94.38% and 99.74%, applying for 3755 classes of Chinese characters in 280 fonts and 120 selected fonts, respectively.

Simonyan K *et al* [14] utilized a VGG Network as an encoder to extract visual features from the input. It allows the network to be trained on unsegmented sequence data. and outperforms state-of-the-art HMM-based systems with more than 40% reduction in relative error. Chung J *et al* [15] processed images based on a gated recurrent unit (GRU) decoder and then extracts features.
In order to generate descriptions for a given picture, an attention-based encoder-decoder model [16] was created with a deep convolutional neural network instead of a deep recurrent neural network to extract features in the image and then extend the attention mechanism to two-dimensional image features. The decoder [17] works on the principle of a long short-term memory network.

Baoguang *et al* [9] designed the CRNN structure, which integrated feature extraction, sequence modeling, and transcription into a unified framework. First, the image convolution features are extracted by CNN, and then the sequence features in the image convolution features are further extracted by LSTM. Finally, the problem of unaligned characters during training is solved with the help of CTC.

Taking the residual as the learning target, the residual learning network is proposed in [10].

3. Methodology

Our proposed method adopts resnet18 in CNN and then optimizes by using LSTM and GRU distributions in RNN. In addition, the attention mechanism is also added to this model for improving its robustness of the model.

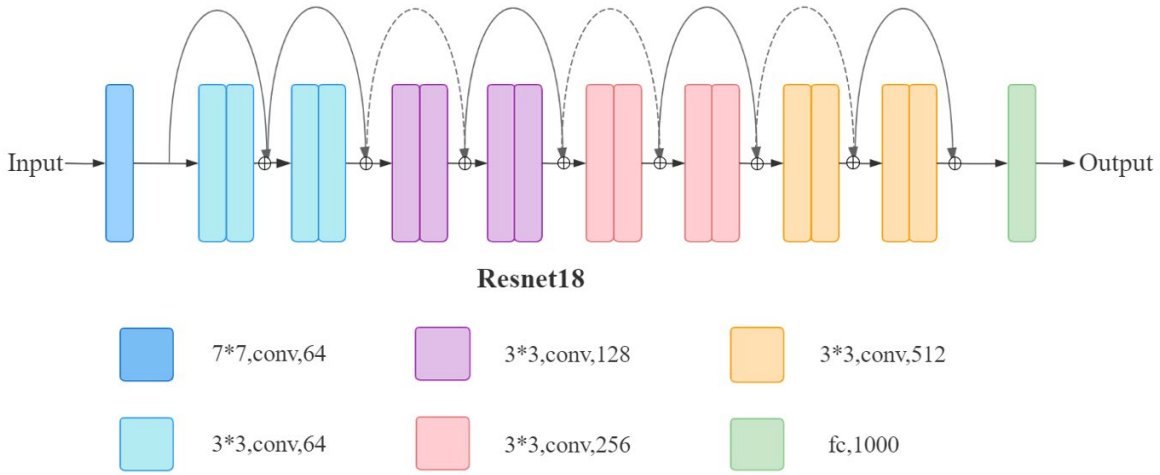

Figure 2. Structure of Resnet18: consists of 17 convolutional layers and a fully connected layer. Each convolutional layer contains a ReLU activation function.

3.1 Resnet

Neural networks that are too deep to train. When the number of the layers increases, the network become difficult to train. Therefore, we use residual learning to simplify the training of deeper networks, unlike CNNs. Here the layers are redefined to learn residual functions for reference layer inputs instead of learning unreferenced functions. These kind of networks [10] are easier to optimize and can improve model performance.

Resnet is composed of several residual blocks. The structure of each residual block is shown in the Figure2. Researchers define a building block as formula (1):

$$y = F_{Resnet}(x, \{W_i\}) + x \tag{1}$$

Here x and $F_{Resnet} + x$ is the input and output vector. The learned parameter is the residual. After a shortcut and the next ReLU, we get the output y. The function $F_{Resnet}(x, \{W_i\})$ represents the residual map that needs to be learned, i.e., the residual function. If there are only two layers, then $F_{Resnet} = W_2\sigma(W_1x)$, where σ is ReLU [29]. Besides biases are omitted for simplicity of annotation. The operation $F_{Resnet} + x$ is performed by shortcut connection and element addition. The dimensions of x andF_{Resnet} must be equal in formula (1).

A residual block consists of two building blocks, and a building block consists of two convolutional layers, four "blocks" or 16 layers. Then plus the average pool and the fully connected layer. The process is also shown in Figure 2, where we use four residual blocks and one fully-connect layer. Note that Resnet18 is lightweight [18] so it is easier to optimize than deep networks.

3.2 SE block

The visual attention mechanism is similar to the brain's signal processing. Human vision can scan the global image to obtain the target that needs to be paid attention to. To select the information that is more critical to the current task goal from a vast information. Jie Hu *et al* [21] proposed the idea of channel attention, which re-adjusts the channel-wise weights through global information, and then introduced Squeeze-and-Excitation Networks (SENet) [21]. SE Net adds processing between two adjacent layers, making it possible to exchange information between channels. The method pays more attention to the channels that need information further improving the effectiveness of feature extraction, suppressing the transmission of useless information, and improving the efficiency of useful information interaction.

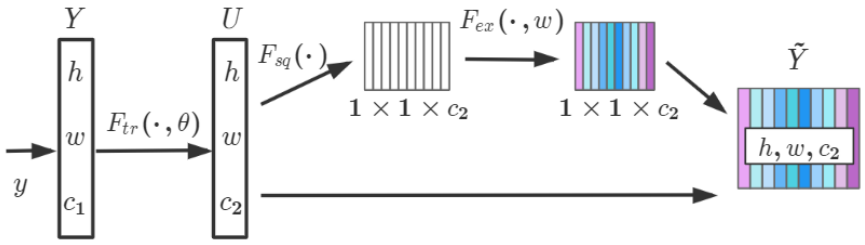

Figure 3. The role of SE Block includes two important steps: squeeze operation and excitation operation. The squeeze operation is to perform feature extraction on the entire image, but keep the channel as one. The excitation operation is to multiply the obtained attention map and the original feature.

Assuming the input is y, the number of feature channels is c_1, and the feature with c_2 feature channels is obtained after general transformation such as convolution. H and W represent the spatial dimension, and C represents the number of channels.

First, map the input F_{tr} as $Y \to U$, $Y \in R^{H' \times W' \times C'}, U \in R^{H \times W \times C}$. The convolution kernel is $V = [v_1, v_2, \dots, v_c]$, v_c represents the cth convolution kernel, $U = [u_1, u_2, \dots, u_c]$.

The convolution operation is shown as formula (3):

$$u_c = v_c * Y = \sum_{s=1}^{C'} v_C^s * y^s \tag{2}$$

The previously obtained features are then rescaled by means of three operations.

(1) Squeeze operation [22]: Feature compression by spatial dimension, turning each two-dimensional feature channel into weights. The result of squeeze operation is defined as:

$$b_c = F_{sq}(u_c) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} u_c(i,j) \tag{3}$$

(2) Excitation operation: A weight is generated for each feature channel through the parameter w, which is used to explicitly exchange the information between feature channels.

$$s = F_{ex}(b, W) = \sigma\big(g(b, W)\big) = \sigma\big(W_2 \delta(W_1 b)\big) \tag{4}$$

First, we reduce the dimension to r (r is a ratio set by yourself) through a fully connected layer and obtain the nonlinear relationship through ReLU. Then it is restored to the C dimension through the second fully connected layer. Finally, the output is limited in [0, 1] through the Sigmoid function.

(3) Reweight operation: Taking the weight of the output of Excitation as the importance of each feature channel after feature selection. The original features are rescaled in the channel dimension by weighting the previous features channel by channel through multiplication.

$$\widetilde{Y_C} = F_{scale}(u_c, s_c) \tag{5}$$

Here F_{scale} is the channel-wise multiplication.

3.3 LSTM and GRU

3.3.1 LSTM

Long short-term memory (LSTM) is a special RNN network. Compared with ordinary RNN, LSTM can perform better in longer sequences. The structure of the basic LSTM unit is shown in Figure 3. Note that stacking multiple bidirectional LSTMs can produce a deep bidirectional LSTM.

$$\begin{cases} z_1 = F_{LSTM0}(y) \\ z_2 = F_{LSTM1}(y) \\ \quad \dots \\ z_t = F_{LSTMt-1}(y) \end{cases} \tag{6}$$

The detailed computation process is shown in formula (2). Besides, as shown in figure 4, suppose the input is y_t, the output is z_t, c_{t-1} denotes the previous state, c_t represents the current state. Note that the current input content is represented

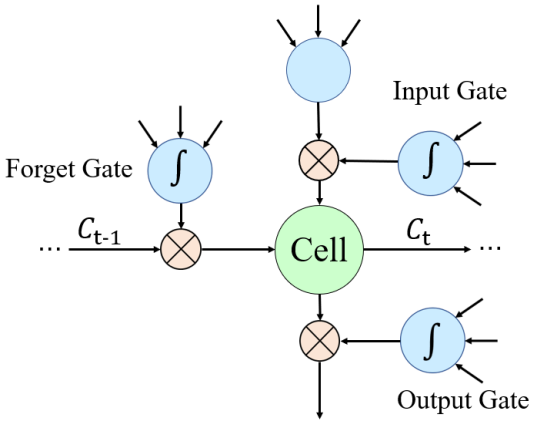

Figure 4. The structure of a basic LSTM unit. An LSTM cell consists of a cell module and input gates, output gates, and forget gates.

by d that is calculated in previous stage. The selected gating signal is controlled by h_i, m_f, m_i, and m_o, which are all multiplied by the weight matrix by the splicing vector. Then they are converted into a value between 0 and 1 through a sigmoid activation function as a gated state. Moreover, m is defined to convert the result to a value between -1 and 1 through a $tanh$ activation function.
There are three main stages inside LSTM, which is summarized as follows:

1. Forgetting stage: Selectively forget the input passed in from the previous node. m_f acts as a forget gate, which controls which information is forgotten by c_{t-1} in the previous state.
2. Select memory stage: We selectively remember y_t, which aims to forget less important information. Then d and h_i is added to our model to get the c_t transmitted for the next state.
3. Output stage: This part determines the output of the current state and scales the c_0 obtained in the previous stage. The process is varied by a tanh activation function [17].

3.3.2 GRU

GRU [19] (Gate Recurrent Unit) is also proposed to solve problems such as long-term memory and gradient in backpropagation. Its experimental effect is similar to LSTM, but it is easier to train and easier to compute than LSTM.

3.4 CTC

CTC (Connectionist Temporal Classification) is a way to avoid manual alignment of input and output. It is suitable for speech recognition and character recognition and other scenarios. CTC converts the predictions made by the LSTM for each feature vector into a sequence of labels. In addition, CTC has the effect of de-redundancy, first removing consecutive repeated characters from the character sequence, and then removing all "-" characters from the path.

The input sequence and output sequence of recurrent neural network are y and d. $\mathrm{d}_{\pi_t}^t$ represents the probability that π_t is observed at time t. $\pi \epsilon L'^T$. T is the sequence length. $l \in L^{\leq T}$. S is training data set. Y is input space. Z is target space, which is a set of sequences consisting of L. L is the character set of the output. $L = \{0,1,2,\cdots,9\}$. $L' = L \cup \{blank\}$. $B: L'^T \rightarrow L^{\leq T}$. B transformation is to deduplicate consecutive identical characters and remove empty characters. $\alpha_t(s)$ is the forward recurrence probability and $\beta_t(s)$ is the backward recursive probability. The relationship between $P(l|z)$ and the recursive formula of forward and backward can be obtained as the following formula:

$$P(l|y) = \sum_{s=1}^{|l'|} \frac{\alpha_t(\mathrm{s})\beta_t(\mathrm{s})}{d_{l'_s}^t} \tag{7}$$

Here CTC Loss function is:

$$L(\mathrm{S}) = -ln \prod_{(y,z)\in S} p(z|y) = -\sum_{(y,z)\in S} ln \sum_{\pi \in B^{-1}(z)} (z) \prod_{t=1}^{T} d_{\pi_t}^t, \forall \pi \epsilon L'^T \tag{8}$$

The advantage of CTC model is that there is no mandatory alignment of labels and the labels can be of variable length. Besides, only the input sequence and the supervised label sequence can be trained, which enables the reliability of our model.

4. Experiment

4.1 Dataset and metric

We adopt the house number data set (SVNH data set) in Google Street View images to evaluate the performance of our method. In this dataset, the training set data includes 30000 images, the validation set data includes 10000 photos, each photo includes a color image and the corresponding coding category and specific location. The resolution size is 256x512. CTC function is selected as the loss function.

4.2 Experiment Details

The entire network is conducted on P40 and PyTorch framework. During the training, we scale the image, then perform a radiographic transformation on the image, another data augmentation method is periodic translation flipping. What's more, we implemented our network with the CTC loss function.

Based on the optimized affection of the CRNN network, we can conclude that CNN has a better application effect than Resnet18. The reason behind this fact is that LSTM and GRU are selected to combine with the SENet attention mechanism for optimization, which can effectively improve the SE block feature extracting ability. Besides, the evaluation index is accurate, which means dividing the number of correct identifications by the number of images in the test set. The Adam is selected as optimizer and the learning rate is set 1×10^{-5}, the epoch is 300, and the batch size is 64.

4.3 Result

Table 1. The comparison test results are as follows:

Method	CRNN	Resnet+LSTM	Resnet+GRU	SENet	**Resnet+LSTM+SEblock**
Accuracy (%)	82.04	85.28	84.77	88.91	**89.63**

From Table 1, we can find that our proposed model can obtain the best performance than other models. Besides, we can find the positive effect of the Resnet model and SE block model, which can improve the 0.04% and 0.01% accuracy, respectively. Compared with the traditional CRNN, our model can obtain more satisfactory robustness.

4.4 Ablation study

To evaluate the effectiveness of each part of our network, we conducted several comparative experiments by adding or changing blocks step by step and compared the experimental results.

Our CNN network selects Resnet18, and the loss function is CTC. RNN part uses LSTM and GRU, respectively. The third experiment introduces an attention mechanism, that is, adding a SE block. The evaluation index is the accuracy rate and the results are shown in Table 1. Comparing the experimental results, it can be concluded that SE block can significantly improve the accuracy, which has a positive effect on obtaining correct recognition results. Comparing the experimental results, it can be concluded that all three results can get better experimental results. SE block can significantly improve the accuracy, which has a positive effect on obtaining correct recognition results.

5. Conclusion

We propose a novel network architecture called convolutional recurrent neural network with channel attention (CACRNN)that enable to solve the problem of feature extraction, feature

recognition and information flow filtering of characters in images. We also did comparative experiments on different blocks. Our network effectively improves the accuracy of character recognition. But there are still many problems to be solved, such as the effect of light adaptation on character recognition. In future work, we will continue to explore the effectiveness of CACRNN on other special conditions.

Acknowledgment

This work was supported by the Natural Science Foundation of Jilin Province(20200401122GX)

References

[1] Guo, Chunle, et al. "Zero-reference deep curve estimation for low-light image enhancement." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020.

[2] Bao, Xiaoli, Heming Jia, and Chunbo Lang. "A novel hybrid harris hawks optimization for color image multilevel thresholding segmentation." Ieee Access 7 (2019): 76529-76546.

[3] Wu, Qingbo, et al. "Accurate transmission estimation for removing haze and noise from a single image." IEEE transactions on image processing 29 (2019): 2583-2597.

[4] Ma, Jiayi, et al. "DDcGAN: A dual-discriminator conditional generative adversarial network for multi-resolution image fusion." IEEE Transactions on Image Processing 29 (2020): 4980-4995.

[5] Liu, Yu-Qi, et al. "Estimating generalized gaussian blur kernels for out-of-focus image deblurring." IEEE Transactions on circuits and systems for video technology 31.3 (2020): 829-843.

[6] De Campos, Teófilo Emídio, Bodla Rakesh Babu, and Manik Varma. "Character recognition in natural images." VISAPP (2) 7 (2009): 2.

[7] Zhang, Honggang, et al. "Text extraction from natural scene image: A survey." Neurocomputing 122 (2013): 310-323.

[8] Sermanet, Pierre, Soumith Chintala, and Yann LeCun. "Convolutional neural networks applied to house numbers digit classification." Proceedings of the 21st international conference on pattern recognition (ICPR2012). IEEE, 2012.

[9] Shi, Baoguang, Xiang Bai, and Cong Yao. "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition.

[10] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[11] LeCun, Yann, et al. "Backpropagation applied to handwritten zip code recognition." Neural computation 1.4 (1989): 541-551.

[12] Guo, Qiang, et al. "Hybrid CNN-HMM model for street view house number recognition." Asian Conference on Computer Vision. Springer, Cham, 2014.

[13] hong, Zhuoyao, Lianwen Jin, and Ziyong Feng. "Multi-font printed Chinese character recognition using multi-pooling convolutional neural network." 2015 13th International Conference on Document Analysis and Recognition (ICDAR). IEEE, 2015.

[14] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).

[15] Chung, Junyoung, et al. "Empirical evaluation of gated recurrent neural networks on sequence modeling." arXiv preprint arXiv:1412.3555 (2014).

[16] Xu, Kelvin, et al. "Show, attend and tell: Neural image caption generation with visual attention." International conference on machine learning. PMLR, 2015.

[17] Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." Neural computation 9.8 (1997): 1735-1780.