Deep Convolutional Neural Network Based Hidden Markov Model for Offline Handwritten Chinese Text Recognition

Zi-Rui Wang, Jun Du
University of Science and Technology of China
Hefei, Anhui, P. R. China
cs211@mail.ustc.edu.cn, jundu@ustc.edu.cn

Jin-Shui Hu, Yu-Long Hu
iFlytek Research
Hefei, Anhui, P. R. China
{jshu, ylhu3}@iflytek.com

***Abstract*—Recently, an effective segmentation-free approach via deep neural network based hidden Markov model (DNN-HMM) was proposed and successfully applied to offline handwritten Chinese text recognition. In this study, to further improve the modeling capability, we adopt deep convolutional neural networks (DCNN) to calculate the HMM state posteriors. First, on the frame basis, the DCNN-HMM can automatically learn the features from the raw image of the handwritten text line via the convolutional architecture rather than the handcrafted gradient features using in the DNN-HMM. Second, we examine several important factors of DCNN to the recognition performance, namely the kernel size, the number of blocks and convolutional layers. We also improve the language modeling by using more text data and high-order N-gram. Tested on ICDAR 2013 competition task of CASIA-HWDB database, the proposed DCNN-HMM could achieve a character error rate (CER) of 4.07%, yielding a relative CER reduction of 30.8% over the DNN-HMM approach. To the best of our knowledge, this is the best published result of the segmentation-free approaches. Furthermore, we explain why DCNN-HMM is more effective than DNN-HMM via the visualization of feature learning and the error pattern analysis.**

I. Introduction

With the new wave of artificial intelligence, handwritten Chinese character recognition is becoming more and more important in real applications, which has been intensively studied for many years [1], [2]. Due to the large vocabulary and different writing styles, it is a challenging problem. In terms of the task complexity, all the research efforts can be divided into four categories [3], [4], namely online/offline isolated Chinese character recognition, and online/offline handwritten Chinese text recognition. Obviously, the offline handwritten Chinese text recognition (OHCTR) is the most challenging task due to the lack of trajectory information and the free writing style, which is also the topic of this study.

For OHCTR, most existing techniques can be classified into two classes: oversegmentation-based and segmentation-free approaches. Oversegmentation-based approaches [5], [6], [7] often need to explicitly segment text line into a sequence of primitive image patches and then merge them to form a candidate lattice. Finally, character classification, linguistic and geometric contexts are integrated to calculate the score for path search. With the emergence of deep learning techniques [10], the new progress has also been made for oversegmentation-based approaches. Wang et al. [8] used positive and negative samples to train the so called heterogeneous convolutional neural network (CNN) [17] as character classifier. Wu et al. [9] adopted three different CNN models to replace conventional character classifier, over-segmentation and geometric models respectively. which were combined with the neural network language model (NNLM) under the general integrated segmentation-and-recognition framework. In contrast to the oversegmentation-based approaches, segmentation-free approaches do not require the explicit segmentation for text line. An early attempt in [11], the authors adopted the Gaussian mixture model based hidden Markov model (GMM-HMM) for the text line modeling. Messina et al. [12] successfully used multidimensional long-short term memory recurrent neural network (MDLSTM-RNN) [14] with connectionist temporal classification (CTC) [13] for OHCTR. Another recent work [15] utilized a CNN followed by a LSTM neural network under the HMM framework to obtain a significant improvement when compared with LSTM-HMM model. In [16], the authors proposed the deep neural network based HMM (DNN-HMM) for text recognition and three key issues, namely feature extraction, character modeling, and language modeling are comprehensively investigated under the general Bayesian framework. And this approach achieved promising results on the ICDAR 2013 competition [4].

On the other hand, recently a great progress has been made to design the new architectures of CNN in computer vision area [18], [19], [20], [21], [22]. In [21], the authors used small 3×3 convolution filters throughout the whole net and the depth of the net could be designed as 19 layers. Szegedy et al. [20] proposed the inception module to increase both the depth and width of the net without extra computation budget. He et al. [22] used a series of shortcut connections to combine the different layers of the net, which could lead to a very deep model. All these recent works indicate that deeper architectures could yield better recognition performance.

Based on the above discussion, in this study, we propose a segmentation-free approach via deep convolution neural network based HMM (DCNN-HMM) to improve the recently proposed DNN-HMM approach [16]. First, on the frame basis, the DCNN-HMM can automatically learn the features from

2327-0985/17 $31.00 © 2017 IEEE
DOI 10.1109/ACPR.2017.65

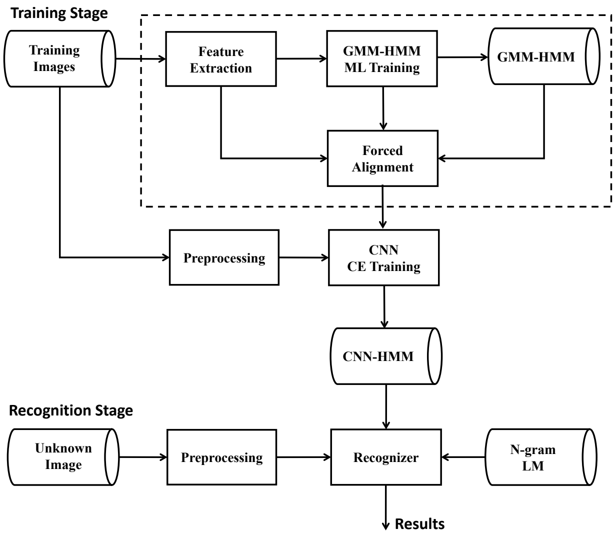

Fig. 1: A block diagram of the proposed system.

the raw image of the handwritten text line via the deep convolutional architecture rather than the handcrafted gradient features using in the DNN-HMM. Second, we examine several important factors of DCNN to the recognition performance, namely the kernel size, the number of blocks and convolutional layers. We also improve the language modeling by using more text data and high-order N-gram. Tested on ICDAR 2013 competition task of CASIA-HWDB database, the proposed DCNN-HMM could achieve a character error rate (CER) of 4.07%, yielding a relative CER reduction of 30.8% over the DNN-HMM approach. To the best of our knowledge, this is the best published result of the segmentation-free approaches. Furthermore, we explain why DCNN-HMM is more effective than DNN-HMM via the visualization of feature learning and the error pattern analysis.

The remainder of the paper is organized as follows. In Section II, we first give an overview of the system framework. In Section III and IV, we describe DCNN-HMM based character modeling and N-gram based language modeling. Then we report experimental results and analysis in Section V. Finally we summarize our work and discuss the future work in Section VI.

II. System Overview

The proposed framework aims to search the optimal character sequence $\mathbf{C}$ for a given extracted feature sequence $\mathbf{X}$ of a text line, which can be formulated according to the Bayesian decision theory as follows:

$$\hat{\mathbf{C}} = \arg\max_{\mathbf{C}} p(\mathbf{C} \mid \mathbf{X}) = \arg\max_{\mathbf{C}} p(\mathbf{X} \mid \mathbf{C}) P(\mathbf{C}) \qquad (1)$$

where $p(\mathbf{X} \mid \mathbf{C})$ is the conditional probability of $\mathbf{X}$ given $\mathbf{C}$ which is named as the character model. Meanwhile $P(\mathbf{C})$ is the prior probability of $\mathbf{C}$ which is named as the language model.

As one implementation of this Bayesian framework, we use an HMM [32] to model one character class. Accordingly a text line is modeled by a sequence of HMMs. An HMM has a set of states and each frame is supposed to be assigned to one underlying state. For each state, an output distribution describes the statistical property of the observed frame. With HMMs, we rewrite the $p(\mathbf{X} \mid \mathbf{C})$ in Eq. (1):

$$\begin{aligned} p(\mathbf{X} \mid \mathbf{C}) &= \Sigma_S \left[p(\mathbf{X}, S \mid \mathbf{C}) \right] \\ &= \Sigma_S \left[\pi(s_0) \prod_{t=1}^{T} a_{s_{t-1}s_t} p(\mathbf{x}_t | s_t) \right] \qquad (2) \\ &= \Sigma_S \left[\pi(s_0) \prod_{t=1}^{T} a_{s_{t-1}s_t} \frac{p(s_t|\mathbf{x}_t) p(\mathbf{x}_t)}{p(s_t)} \right] \qquad (3) \end{aligned}$$

$S = \{s_0, s_1, ..., s_T\}$ is one underlying state sequence of $\mathbf{C}$ to represent $\mathbf{X}$. $\pi(s_0)$ is the prior probability of the initial state s_0 and $a_{s_{t-1}s_t}$ is the transition probability from state s_{t-1} at the $(t-1)^{\text{th}}$ frame to state s_t at the t^{th} frame. $p(\mathbf{x}_t|s_t)$ is the emission probability, which can be directly calculated (e.g., GMM-HMM) or indirectly obtained via the calculation of state posterior probability $p(s_t|\mathbf{x}_t)$ (e.g. DNN-HMM).

The block diagram of proposed system is illustrated in Fig. 1. To make a fair comparison with the DNN-HMM approach in [16], we still use the state-level labels via the GMM-HMM system for each frame. Namely, the processing within the dotted box of Fig. 1 is the same as [16]. As for the preprocess module of DCNN-HMM, first the height of the text line is estimated, followed by the size normalization while keeping the aspect ratio. Then the margin is extended to accommodate the text area for all the sliding windows in the next step. Along the centre line, each frame, represented by a 40×80 sliding window from the left to right, with a frame shift of 3 pixels, is scanned across the text line. Moreover each frame is normalized to 40×40 by the bilinear interpolation. Finally, each frame is extended to 48×48 by adding the margin, which is fed to DCNN model. With the frame-level label of each frame, the DCNN model is trained using the cross-entropy (CE) criterion. In the recognition stage, the DCNN is adopted to calculate the state posterior probability $p(s_t|\mathbf{x}_t)$ in Eq. (3) and the final recognition results can be generated via a weighted finite-state transducer (WFST) [38], [39] based decoder by integrating both DCNN-HMM based character model and N-gram based language model (LM). The details of character and language modeling are elaborated in the following sections.

III. DCNN-HMM Based Character Modeling

The conventional convolutional neural network [17] successively consists of stacked convolutional layers optionally followed by spatial pooling, one or more fully-connected layers and a softmax layer. For the convolutional and the pooling layers, each layer of them is a three-dimensional tensor organized by a set of planes called feature maps while the fully-connected layer and the softmax layer are the same as the conventional DNN. Inspired by the locally-sensitive,

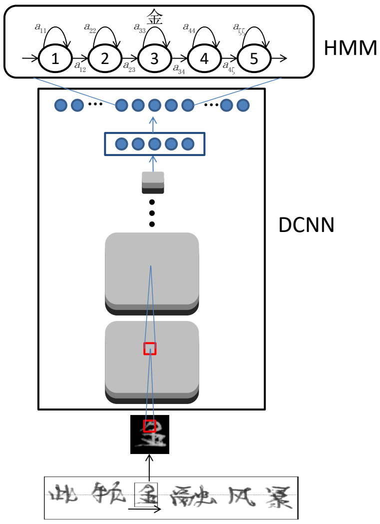

Fig. 2: The architecture of DCNN-HMM.

orientation-selective neurons in the visual system of cats [35], each unit in a feature map is constrained to connect a local region in the previous layer, which is called the local receptive field. Two contiguous local receptive fields are usually s pixels (referred as stride) shifted along a certain direction. All units in the same feature map of a convolutional layer share a set of weights, each computing a dot product between its weights and local receptive field in the previous layer and then followed by nonlinear activation functions (e.g., rectifier). Meanwhile the units in a pooling layer perform a spatial average or max operation for their local receptive field to reduce the spatial resolution and the noise interferences. Accordingly, the key information for identifying the pattern is retained. We formalize the convolution operation in a convolutional layer as:

$$Y_{i,j,k} = \sum_{m,n,l} X_{(i-1)\times s+m,(j-1)\times s+n,l} K_{m,n,k,l} \tag{4}$$

where $X_{i,j,k}$ is the value of the input unit in feature map k at row i and column j while $Y_{i,j,k}$ is corresponding to the output unit. $K_{m,n,k,l}$ is the connection weight between a unit in feature map k of the output and a unit in channel l of the input, with an offset of m rows and n columns between the output unit and the input unit. Similarly, the pooling operation can be conducted by using a max operation in this study.

The DCNN-HMM in this work, illustrated in Fig. 2, is designed with the following innovations. First, we investigate on creating a DCNN architecture suitable for the specific OHCTR task, which is experimentally verified to be more effective than the widely used standard CNN models [21], [22] for image recognition tasks. Second, the proposed DCNN can model the character with a high-resolution by using many hidden states in the HMM framework, which is the key to achieving the promising recognition results. In our experiments, the size of DCNN output layer is close to 20,000 corresponding to the total number of states, which should be the largest among the existing publications for OHCTR, to the best of our knowledge. Finally, in comparison to the DNN-HMM approach in [16], the modeling capability is significantly improved by using deep convolutional layers to automatically learn more useful information from the input.

IV. N-GRAM BASED LANGUAGE MODELING

The language modeling is quite an effective way to improve the recognition accuracy by incorporating more context information during the decoding. Suppose the sequence $\mathbf{C}$ contains m characters. $P(\mathbf{C})$ in Eq. (1) can be written as:

$$p(\mathbf{C}) = \prod_{i=1}^{m} p(c_i|\mathbf{h}_1^{i-1}) \tag{5}$$

where $\mathbf{h}_1^{i-1} = (c_1, ..., c_{i-1})$. Statistical N-gram LM is adopted in this study as an approximation of Eq. (5), where only $n-1$ history characters of c_i in $\mathbf{h}_1^{i-1}$ are considered:

$$p(\mathbf{C}) \approx \prod_{i=1}^{m} p(c_i|\mathbf{h}_{i-n+1}^{i-1}) \tag{6}$$

where n is the order of N-gram LM. To calculate those conditional probabilities in Eq. (6) and alleviate the data sparse problem, back-off n-gram language models (BLMs) [42] are widely used. Specifically, we adopt a conventional BLM approach with Katz smoothing [43]. The SRILM toolkit [36] is employed to generate Katz n-gram LM and then used to generate the WFST-based decoder [38], [39] by Kaldi [37].

V. EXPERIMENTS

A. Experimental setup

The experiments are conducted on the public CASIA-HWDB database [40]. The training set consists of HWDB1.0, HWDB1.1, HWDB2.0, HWDB2.1, and HWDB2.2 datasets. HWDB1.0 and HWDB1.1 are offline isolated handwritten Chinese character datasets while HWDB2.0-HWDB2.2 are offline handwritten Chinese text datasets. Almost all the data samples of these datasets are used for training. In total, there are 3,980 classes (Chinese characters, symbols, garbage) with 4,091,599 samples. Here "garbage" classes represent the short blank model between characters and the long blank model at the beginning or end of the text line. As for the evaluation set, the ICDAR 2013 competition set with 60 writers is adopted [4], which consists of 3432 text lines.

Five DCNN architectures are investigated in Table I. Similar to [21] , we design CNN1-5 to evaluate the effect of receptive field size, network width and depth. For CNN1, there are 7 weights layers, including 5 convolutional (conv) and 2 fully connected (FC) layers. And the number of channels starts from 100 in the first conv layer to 400 in the last conv layer. For CNN5, there are 16 weights layers (13 conv and 2 FC layers) and the number of channel can reach to 700 in the last conv layer. In CNN1, CNN3-5, the image patch of each frame is passed through a stack of 3×3 conv layers. After the last max pooling layer, a 1×1 conv layer is used to increase the

TABLE I: Five DCNN architectures. The convolutional layer are denoted as "conv(receptive field size)-(feature map size)".

CNN Configuration				
CNN1	CNN2	CNN3	CNN4	CNN5
7 weights layers	10 weights layers	10 weights layers	13weights layers	16 weights layers
Input 48 × 48 Gray Image				
conv3-100	conv2-100	conv3-100	conv3-100	conv3-100
maxpool				
conv3-200 Pa1	conv2-100 conv2-200	conv3-100 Pa1 conv3-200 Pa1	conv3-100 Pa1 conv3-200 Pa1 conv3-300 Pa1	conv3-100 Pa1 conv3-200 Pa1 conv3-300 Pa1 conv3-300 Pa1
maxpool	maxpool Pa1	maxpool	maxpool	maxpool
conv3-300 Pa1	conv2-200 conv2-300	conv3-200 Pa1 conv3-300 Pa1	conv3-300 Pa1 conv3-400 Pa1 conv3-500 Pa1	conv3-300 Pa1 conv3-400 Pa1 conv3-500 Pa1 conv3-500 Pa1
maxpool	maxpool Pa1	maxpool	maxpool	maxpool
conv3-400 Pa1	conv2-300 conv2-400	conv3-300 Pa1 conv3-400 Pa1	conv3-500 Pa1 conv3-600 Pa1 conv3-700 Pa1	conv3-500 Pa1 conv3-600 Pa1 conv3-700 Pa1 conv3-700 Pa1
maxpool	maxpool Pa1	maxpool	maxpool	maxpool
conv1-400	conv1-400	conv1-400	conv1-700	conv1-700
FC500				
FC19900				
Soft-max 19900				

nonlinearity of the net without more computation and memory compared to other larger receptive fields. Although 3×3 conv layer has proven to be very effective [21], [29], [9], we still design CNN2 with a stack of 2×2 conv layers to study the effect of smaller receptive field. All conv layers in Table I, are followed by the rectification non-linearity (ReLU [18]) and the stride is 1 while the stride of all max pooling layers is 2 with 3×3 pixel window. For some conv and max pooling layers, we use the padding operation (denoted as "Pa1" in Table I) to preserve the spatial resolution. Because each character class is modeled by a left-to-right HMM with 5 states, the output size of DCNN is $3,980 \times 5 = 19,900$ corresponding to the number of states of all classes. For training the DCNN model, the mini-batch size is 400, while the momentum is 0.9 and the weight decay is 0.0001. 1.8 million iterations are conducted. The learning rate is initially set to 0.01 and then decreased by 0.92 after every 10,000 iterations. The DCNN model is initialized with Xavier method [44] using Caffe toolkit [41].

As for LM, in addition to the transcriptions in the CASIA database, other corpora are used as supplementary sources, including 208MB texts of Guangming Daily between 1994 and 1998, 178MB texts of People's Daily 2000-2004 and 2011, 129MB texts of other newspapers, and 93MB texts of Sina News. A 4-gram LM with Katz smoothing was trained by the SRILM toolkit with the default parameters.

B. Recognition performance

TABLE II: The CER and model size comparison of the different NN architectures under the same HMM framework on the ICDAR 2013 competition set.

System	CNN1	CNN2	CNN3	CNN4	CNN5	DNN
CER	4.88%	4.87%	4.22%	4.09%	**4.07**%	5.88%
Model Size	49MB	47MB	53MB	96MB	107MB	238MB

Table II lists the CER and model size comparison of different NN architectures under the same HMM framework on the ICDAR 2013 competition set. The DNN structure was the same as in [16]. Obviously, all CNN models significantly outperformed the DNN model in terms of both CER and model size. The CER of DNN model was 5.88% while the best CNN5 model could achieve a promising CER of 4.07%, yielding a relative CER reduction of 30.8% over the DNN model. And the model size of CNN5 was 107 mega bytes (MB), which was less than half of that for DNN model. From CNN1 to CNN5, the CER was significantly reduced (from 4.88% to 4.07%) and saturated at CNN5 with the increase of both depth and width. By comparing CNN2 with CNN3, we could conclude that 3×3 receptive field is the smallest size to cover the notions of left, right, up and down and guarantee a good recognition performance for the OHCTR task.

TABLE III: The CER comparison of the different approaches on the ICDAR 2013 competition set.

System	CER
CNN5-HMM (Our best system)	4.07%
CNN + LSTM [15]	16.50%
MDLSTM + CTC [12]	10.6%
Heterogeneous CNN [8]	5.88%
CNN shape models + NNLM [9]	3.80%

Table III shows a CER comparison of the best CNN5-HMM system with other approaches on the ICDAR 2013 competition set. Our approach is similar to [15] in terms of using HMM framework. However our recognition result was much better with about 12% absolute CER reduction, which might be explained as: 1) we used a deeper CNN architecture (16 weight layers vs. 5 weight layers); 2) we achieved a higher resolution for modeling the posterior probability of hidden states by using more output nodes (19,900 vs. 7,356); 3) actually we adopted a lower order N-gram LM (4-gram vs. 10-gram); 4) other details, such as the preprocessing, GMM-HMM design for alignment, different text data for language modeling, etc. In comparison to the method using MDLSTM+CTC [12], our approach could yield a relative CER reduction of 61.6%, demonstrating that the higher resolution HMM model was much more effective than CTC technique (similar to 1-state HMM model) for modeling the handwritten text line. To the best of our knowledge, the proposed CNN5-HMM should achieve the best published result of segmentation-free approaches. As for the comparison with over-segmentation approaches, CNN5-HMM outperformed the recent work using heterogeneous CNN [8] and slightly underperformed the approach in [9]. However, three different CNN models should be used in [9] to design the character classifier, over-segmentation and geometric models while our proposed approach only adopted one CNN model.

C. Result Analysis

To give readers a better understanding why the DCNN-HMM model could be more effective than DNN-HMM model, an example is given in Fig. 3, where DNN-HMM system generates one substitution error (marked red) while CNN5-HMM system produces the correct results as the ground truth.

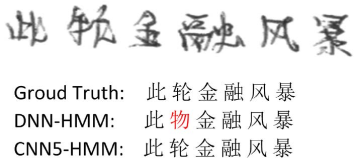

Fig. 3: The recognition results comparison of an example between DNN-HMM and CNN5-HMM.

We conduct the result analysis from two aspects. First, we analyze five selected kernel outputs (with kernel ID in Fig. 4) of the first convolutional layer of CNN5. Based on the visualization results, each kernel processed the input image patch from different aspects, e.g., the contrast enhancement, stroke deletion on a certain direction, smoothing, sharpening, and the profile extraction. All these kernels working together could comprehensively extract the useful information and remove the interferences, which made CNN5-HMM more powerful than DNN-HMM as a recognizer. Second, in Fig. 5, we analyze the output layer of DNN and CNN5, namely the state posterior probability of the frames for the reference character class which is misclassified as the character with the red color by DNN-HMM in Fig. 3. The posterior probability of each frame in Fig. 5 was the maximum value among all the states of the reference character class. It was clear that CNN5 could achieve much higher posterior probability of the reference character class than DNN among the central frames of the handwritten character (from 10 to 20), which implied that CNN5 could better predict the reference character than DNN.

VI. Conclusion

In this study, we investigate on the offline handwritten Chinese text recognition by using the DCNN-HMM, yielding significant performance improvements over the conventional DNN-HMM. As for the future work, we aim to further improve the DCNN-HMM with other techniques, e.g., discriminative training.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grants 61671422 and U1613211, in part by the Strategic Priority Research Program of the Chinese Academy of Sciences under Grant XDB02070006, and in part by the National Key Research and Development Program of China under Grant 2017YFB1002200

References

[1] R.-W. Dai, C.-L. Liu, and B.-H. Xiao, "Chinese character recognition: history, status and prospects," *Frontiers of Computer Science in China* Vol. 1, No. 2, pp.126-136, 2007.

[2] H. Fujisawa, "Forty years of research in character and document recognition–an industrial perspective," *Pattern Recognition*, Vol. 41, No. 8, pp.2435-2446, 2008.

[3] C.-L. Liu, F. Yin, Q.-F. Wang, and D.-H. Wang, "ICDAR 2011 Chinese handwriting recognition competition," *Proc. ICDAR*, 2011, pp.1464-1469.

[4] F. Yin, Q.-F. Wang, X.-Y. Zhang, and C.-L. Liu, "ICDAR 2013 Chinese handwriting recognition competition," *Proc. ICDAR*, 2013, pp.1464-1470.

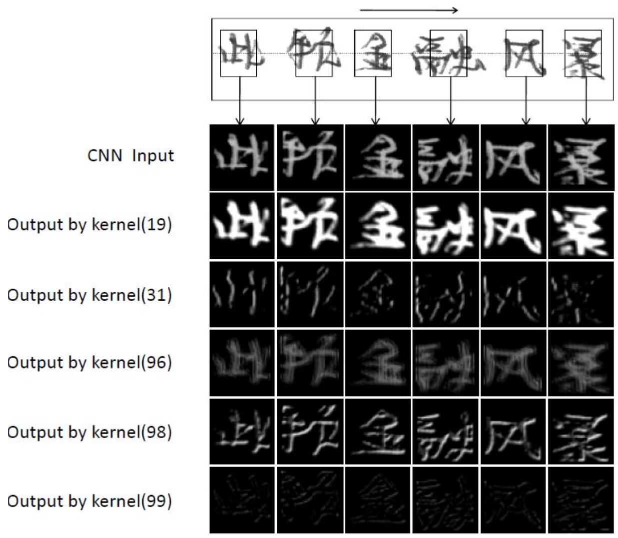

Fig. 4: Visualization of the selected kernel output of the first convolutional layer of CNN5.

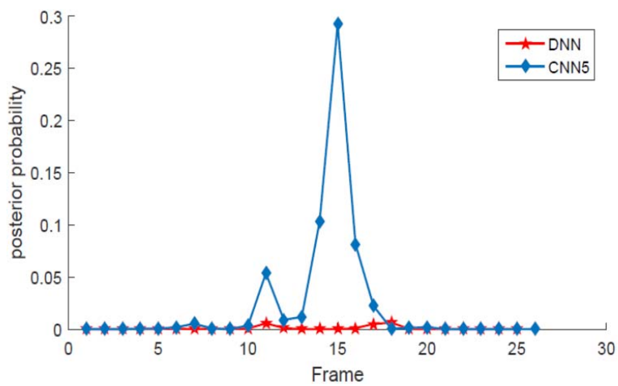

Fig. 5: The analysis of the state posterior probability of the frames for one reference character class.

[5] Q. Fu, X.-Q. Ding, T. Liu, Y. Jiang, and Z. Ren, "A novel segmentation and recognition algorithm for Chinese handwritten address character strings," *Proc. ICPR*, 2006, pp.974-977.

[6] N.-X. Li and L.-W. Jin, "A Bayesian-based probabilistic model for unconstrained handwritten offline Chinese text line recognition,?*Proc. IEEE SMC*, 2010, pp.3664-3668.

[7] Q.-F. Wang, F. Yin, and C.-L. Liu, "Handwritten Chinese text recognition by integrating multiple contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 34, No. 8, pp.1469-1481, 2012.

[8] S. Wang, L. Chen, L. Xu, W. Fan, J. Sun, and S. Naoi, "Deep Knowledge Training and Heterogeneous CNN for Handwritten Chinese Text Recognition," *Proc. ICFHR*, 2016, pp.84-89.

[9] Y.-C. Wu, F. Yin, and C.-L. Liu, "Improving handwritten Chinese text recognition using neural network language models and convolutional neural network shape models," *Pattern Recognition*, Vol. 65, pp.251-264, 2017.

[10] LeCun, Y., Bengio,and G. Hinton, "Deep learning," *Nature*, Vol. 521, No. 7553, pp.436-444, 2015.

[11] T.-H. Su, T.-W. Zhang, D.-J. Guan, and H.-J. Huang, "Off-line recognition of realistic Chinese handwriting using segmentation-free strategy," *Pattern Recognition*, Vol. 42, No. 1, pp.167-182, 2009.

[12] R. Messina and J. Louradour, "Segmentation-free handwritten Chinese text recognition with LSTM-RNN," *Proc. ICDAR*, 2015, pp.171-175.

[13] A. Graves, S. Fernandez, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," *Proc. ICML*, 2006, pp.369-376.

[14] A. Graves, and J. Schmidhuber, "Offline handwriting recognition with multidimensional recurrent neural networks," *Proc. NIPS*, 2009, pp.545-552.

[15] D. Suryani, P. Doetsch, and H. Ney, "On the benefits of convolutional neural network combinations in offline handwriting recognition," *Proc. ICFHR*, 2016.

[16] J. Du, Z.-R. Wang, J.-F. Zhai, and J.-S. Hu, "Deep neural network based hidden markov model for offline handwritten Chinese text recognition," *Submitted to Proc. ICPR 2016*.

[17] L. Yann, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, Vol. 11, pp.2278-2324, 1998.

[18] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," *NIPS*, 2012, pp.1097-1105.

[19] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and F.-F. L, "Imagenet: A large-scale hierarchical image database. In Computer Vision and Pattern Recognition," *CVPR*, 2009, pp.248-255.

[20] C. Szegedy, W. Liu, Y.-Q. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *CVPR*, 2015, pp.1-9.

[21] K. Simonyan, and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv:1409.1556*, 2014.

[22] K.-M. He, X.-Y, Zhang, S.-Q. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv:1512.03385*, 2015.

[23] D. Ciresan, U. Meier, J. Masci, L. Gambardella, and J. Schmidhuber, "Flexible, high performance convolutional neural networks for image classification," *Proc. IJCAI*, 2011, pp.1237-1242.

[24] C. Wu, W. Fan, Y. He, J. Sun, and S. Naoi, "Handwritten character recognition by alternately trained relaxation convolutional neural network," *Proc. ICFHR*, 2014, pp.291-296.

[25] B. Graham, "Sparse arrays of signatures for online character recognition," *Technical Report*, University of Warwick, 2013.

[26] J. Du, J.-F. Zhai, J.-S. Hu, B. Zhu, S. Wei, and L.-R. Dai, "Writer adaptive feature extraction based on convolutional neural networks for online handwritten Chinese character recognition," *Proc. ICDAR*, 2015, pp.841-845.

[27] W. Yang, L. Jin, Z. Xie, and Z. Feng, "Improved deep convolutional neural network for online handwritten Chinese character recognition using domain-specific knowledge," *Proc. ICDAR*, 2015, pp.551-555.

[28] Z. Zhong, L. Jin, and Z. Xie, "High performance offline handwritten Chinese character recognition using GoogLeNet and directional feature maps," *Proc. ICDAR*, 2015, pp.846-850.

[29] X.-Y. Zhang, Y. Bengio, and C.-L. Liu, "Online and offline handwritten Chinese character recognition: A comprehensive study and new benchmark," *Pattern Recognition*, Vol. 61, pp.348-360, 2017.

[30] Z.-C. Xie, Z.-H. Sun, L.-W, Jin, Z.-Y. Feng, and S.-Y. Zhang, "Fully convolutional recurrent network for handwritten Chinese text recognition," *arXiv:1604.04953*, 2016.

[31] Z.-C Xie, Z.-H. Sun, L.-W. Jin, H. Ni, and T. Lyons, "Learning spatial-semantic contex with fully convolutional recurrent network for online handwritten chinese text recognition," *arXiv:1610.02616*, 2016.

[32] L. E. Baum, J. A. Eagon, "An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology," *Bulletin of American Mathematical Society*, Vol. 73, pp.360-363, 1967.

[33] M. Mohri, F. Pereira, and M. Riley, "Weighted finite-state transducers in speech recognition," *Computer Speech and Language*, Vol. 20, No. 1, pp.69-88, 2002.

[34] C. Allauzen, M. Riley, J. Schalkwyk, W. Skut, and M. Mohri, "OpenFst: a general and efficient weighted finite-state transducer library," *Proc. CIAA*, 2007, pp.11-23.

[35] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *Journal of Physiology*, Vol. 160 , pp.106-154, 1962.

[36] A. Stolcke, "SRILM-an extensible language modeling toolkit," *Proc. ICSLP*, 2002, pp.901-904.

[37] D. Povey, A. Ghoshal, et al., "The kaldi speech recognition toolkit," *Proc. ASRU*, 2011.

[38] M. Mohri, F. Pereira, and M. Riley, "Weighted finite-state transducers in speech recognition," *Computer Speech and Language*, Vol. 20, No. 1, pp.69-88, 2002.

[39] C. Allauzen, M. Riley, J. Schalkwyk, W. Skut, and M. Mohri, "OpenFst: a general and efficient weighted finite-state transducer library," *Proc. CIAA*, 2007, pp.11-23.

[40] C.-L. Liu, F. Yin, D.-H. Wang, and Q.-F. Wang, "CASIA online and offline Chinese handwriting databases," *Proc. ICDAR*, 2011, pp.37-41.

[41] Y.-Q. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *Proc. ACM*, 2014, pp.675-678.

[42] R. Kneser and H. Ney, ' Improved backing-off for m-gram language modeling," *Proc. ICASSP*, 1995, pp.181-184.

[43] S. Katz, "Estimation of probabilities from sparse data for the language model component of a speech recognizer," *IEEE Trans. Acoust. Speech Signal Processs.*, Vol. 35, No. 3, pp.400-401, 1987.

[44] Glorot, Xavier, and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," *Proc. Aistats*, 2010, pp.249-256.