2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

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

Jun Du, Zi-Rui Wang and Jian-Fang Zhai
University of Science and Technology of China
Hefei, Anhui, P. R. China
Email: jundu@ustc.edu.cn, {cs211,zhaijf13}@mail.ustc.edu.cn

Jin-Shui Hu
iFlytek Research
Hefei, Anhui, P. R. China
Email: jshu@iflytek.com

Abstract—**This paper proposes a novel segmentation-free approach using deep neural network based hidden Markov model (DNN-HMM) for offline handwritten Chinese text recognition. In the general Bayesian framework, three key issues are comprehensively investigated, namely feature extraction, character modeling, and language modeling. First, as for the feature extraction on the basis of each frame or sliding window, the gradient-based features are extracted for the DNN-based classifier. Second, the text line is sequentially modeled by HMMs with each representing one character class. Meanwhile the DNN-based classifier is adopted to calculate the posterior probability of all HMM states. Finally, the character n-gram language model is integrated with the DNN-HMM character model for the Bayesian decision. The experiments on the ICDAR 2013 competition task of CASIA-HWDB database show that the proposed approach can achieve the best published recognition results to our knowledge, yielding a character error rate (CER) of 6.50%, which significantly outperforms the previously best reported oversegmentation approach (with a CER of 9.25%) and the segmentation-free approach using multidimensional long-short term memory recurrent neural network (MDLSTM-RNN) approach (with a CER of 10.6%).**

I. Introduction

With the new wave of artificial intelligence in the current mobile internet era, the robust recognition of handwritten Chinese characters in an unconstraint manner has become one critical issue in real applications. For the past several decades, many techniques [1], [2] have been proposed to handle this 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, due to the lack of ink trajectory information and the free writing style, the offline handwritten Chinese text recognition is the most challenging task, which is also the topic of this study.

To model the handwritten Chinese character, the prototype [5] and modified quadratic discriminant function [6] based classifiers are the conventional off-the-shelf solutions widely used in many applications. Recently, with the emergence of deep learning techniques, e.g. deep neural network (DNN) for speech recognition [7], convolutional neural network (CNN) for image classification [8], [9], the new milestone is also created for Chinese handwriting recognition. According to the ICDAR 2013 competition [4], the best systems of many tasks adopt the neural network based approaches. For online isolated character recognition task, the top1 system from University of Warwick uses the signature features and a CNN classifier [10]. The system from Fujitsu R&D Center employs multiple CNNs and yields the best results on the offline isolated character recognition task [11] while the researchers from the Dalle Molle Institute for Artificial Intelligence (IDSIA) claim that their system using multi-column DNNs [12] can achieve a better performance after the competition. More recently, several improved CNN approaches [13], [14], [15] are proposed for the isolated Chinese character recognition tasks. As for online handwritten Chinese text recognition task, the systems from Vision Object using character oversegmentation and neural network based classifier significantly outperform others. All the abovementioned systems with neural network based techniques for different tasks can achieve character error rates (CERs) close to or below 5%. However, only for offline handwritten Chinese text recognition, above 10% CER is obtained by the best submitted system from Harbin Institute of Technology (HIT) and the neural network is not adopted. This performance gap compared with other tasks is a further demonstration of its challenge.

Based on the above discussion, the key issue for offline handwritten Chinese text recognition is how to sequentially model the text line, which seems more critical than the design the character classifier. Accordingly, all existing techniques belong to two broad classes: oversegmentation-based and segmentation-free approaches. The former one [16], [17], [18], [19] often builds several modules first including character oversegmentation, character classification, modeling the linguistic and geometric contexts. Then they are integrated to calculate the score for path search. Among these approaches, the most effective one in [19] is to integrate multiple contexts for recognition of handwritten Chinese text line, which achieves the best recognition performance on the ICDAR 2013 competition task. In contrast to the diversified oversegmentation-based approaches, there are not many segmentation-free approaches. One early approach adopts the Gaussian mixture model based hidden Markov model (GMM-HMM) for the text line modeling [20]. Another recent approach [21] uses multidimensional long-short term memory recurrent neural network (MDLSTM-RNN), which is inspired by well verified LSTM-RNN approaches [22] for the recognition of

978-1-5090-4847-2/16/$31.00 ©2016 IEEE

handwritten western languages with a small set of character classes. The MDLSTM-RNN approach is quite flexible as connectionist temporal classification (CTC) technique [23] is adopted to avoid the explicit segmentation. Furthermore, it can achieve the comparable recognition accuracy with the best oversegmentation-based approach [19] on the ICDAR 2013 competition task.

In this study, we propose a novel segmentation-free approach via DNN-based HMM (DNN-HMM). In the general Bayesian framework, three main components, namely feature extraction, character and language modeling, are comprehensively investigated. For feature extraction, on the basis of each frame represented by a left-to-right sliding window along with the text line, the gradient-based features are extracted for the DNN-based classifier. Then the text line is sequentially modeled by HMMs with each representing one character class. Meanwhile the DNN is adopted to model the posterior probability of all HMM states. Finally, the character n-gram language model (LM) is integrated with the DNN-HMM character model for the Bayesian decision. In comparison to other segmentation-free approaches [20], [21], GMM-HMM can be considered as an initialization model to boost the DNN-HMM. There are two main differences between DNN-HMM and MDLSTM-RNN. The first one is only the conventional DNN is adopted in DNN-HMM rather than the more complicated architecture with both convolutional and recurrent layers used in MDLSTM-RNN. The second one is to sequentially model the text line, HMM is used in DNN-HMM while CTC is adopted in MDLSTM-RNN. On the ICDAR 2013 competition task, our proposed approach can significantly outperform both MDLSTM-RNN and the best oversegmentation-based approach [19], yielding a relative CER reduction of 29.7% over the best reported results.

The remainder of the paper is organized as follows. In Section II, we first give an overview of the system framework. In Section III, the feature extraction is elaborated. In Section IV and V, we describe the character and language modeling in detail. In Section VI, we report experimental results. Finally we summarize our findings in Section VII.

II. System Overview

The main principle of our proposed new framework is based on the Bayesian formula:

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

where $\mathbf{X}$ is the extracted feature sequence of a handwritten text line and $\mathbf{C}$ is the underlying character sequence. $p(\mathbf{X} \mid \mathbf{C})$ is the conditional probability of $\mathbf{X}$ given $\mathbf{C}$, which corresponds to a sequence of HMMs. Each HMM represents one character class, which is also named as the character model. Meanwhile $P(\mathbf{C})$ is the probability of $\mathbf{C}$ which is also called the language model. Finally, $\hat{\mathbf{C}}$ can be generated according to the Bayesian decision theory.

As one implementation of this Bayesian framework, the overall flowchart of our proposed system is illustrated in Fig. 1.

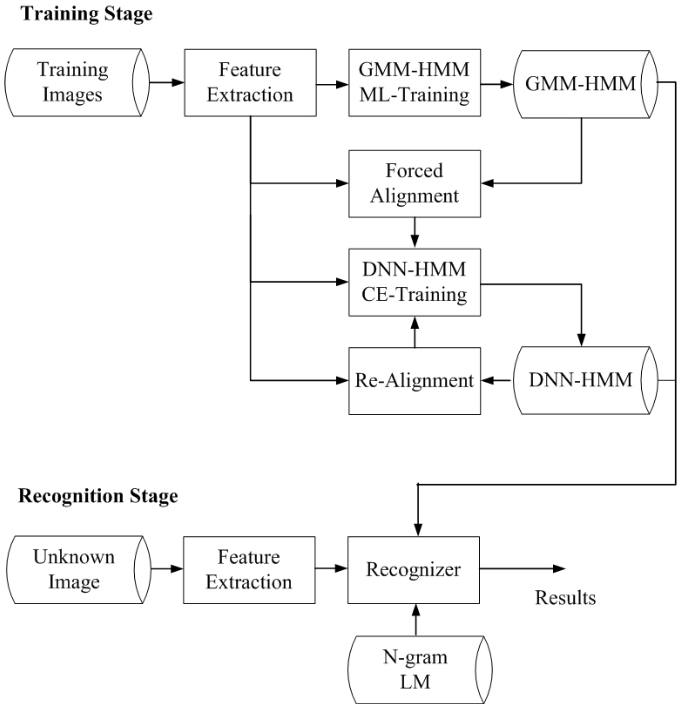

Fig. 1. Overall development flow and architecture.

In the training stage, two HMM systems, namely GMM-HMM and DNN-HMM, are built. First, the gradient-based features on a frame basis are extracted, which is followed by principal component analysis (PCA) transformation to obtain a lower dimensional feature vector. Then the parameters of GMM-HMMs for all character classes are learned using maximum likelihood estimation. With the well-trained GMM-HMMs, state-level forced-alignment are performed to obtain the frame-level labels used for the subsequent DNN training. As for the DNN-HMM system, the DNN model is trained using the cross-entropy (CE) criterion. To refine the labels initialized by GMM-HMM system, the DNN-HMM system can be adopted for the re-alignment in a manner of multi-pass training. In the recognition stage, after the feature extraction of the unknown handwritten text line, the final recognition results can be generated via a weighted finite-state transducer (WFST) [24], [25] based decoder by integrating both character model and language model. The details of feature extraction, character modeling, and language modeling are elaborated in the following sections.

III. Feature Extraction

The procedure of feature extraction, illustrated in Fig. 2, is described as follows:

- **Step1**: **Text Line Preprocessing**
  First, the height of the text line will be 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. Finally, the center line is calculated.

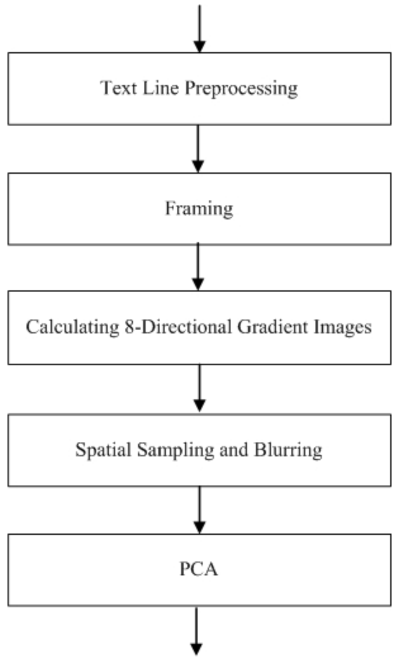

Fig. 2. The procedure of feature extraction.

- **Step2**: **Framing**
  Along the centre line, each frame, represented by a 64×32 sliding window from the left to right, with a frame shift of 3 pixels, is scanned across the text line.
- **Step3**: **Calculating 8-Directional Gradient Images**
  In each frame, the gradient features via the Sobel operator are calculated and 8-directional gradient images can be generated according to [26], [27].
- **Step4**: **Spatial Sampling and Blurring**
  Each gradient image is divided uniformly into 8×4 grids with the corresponding centers treated as locations of 8×4 spatial sampling points. Then the blurring via the Gaussian filters is applied as in [28]. By considering all 8 gradient images, a feature vector with 256 dimensions is formed.
- **Step5**: **PCA**
  In the last step, PCA transformation [29] is adopted to obtain a lower 50-dimensional feature vector fed to the subsequent character modeling.

IV. Character Modeling

The offline handwritten Chinese text line is a sequence of Chinese characters. In this study, we adopt a left-to-right HMM to model one character class as a basic unit. Accordingly, the text line is modeled by a sequence of HMMs. In the following subsections, the details of the HMM and its output distribution characterized by the DNN are introduced.

A. HMM

As shown in Fig. 3, we list an HMM [30] example for one handwritten character class with a set of hidden states. For the feature sequence extracted from one handwritten character sample, each frame is supposed to be assigned to one underlying state. For each state, an output distribution describes the statistical property of the observed feature vector. With HMMs, the $p(\mathbf{X} \mid \mathbf{C})$ in Eq. (1) can be decomposed as:

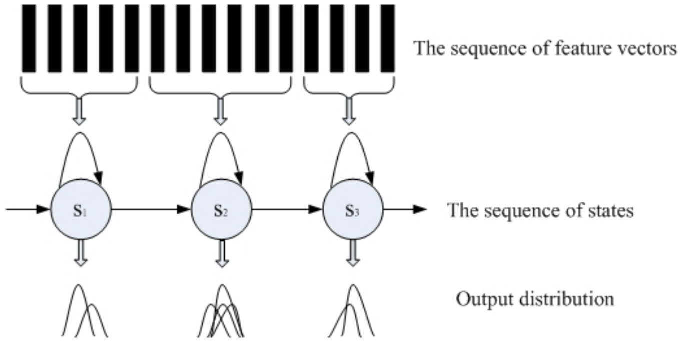

Fig. 3. Illustration of an HMM representing one character class.

$$\begin{aligned} p(\mathbf{X} \mid \mathbf{C}) &= \Sigma_S \left[p(\mathbf{X} \mid S, \mathbf{C}) p(S \mid \mathbf{C}) \right] \\ &= \Sigma_S \left[\pi(s_0) \prod_{t=1}^{T} a_{s_{t-1}s_t} \prod_{t=0}^{T} p(\mathbf{x}_t | s_t) \right] \end{aligned} \quad (2)$$

where $\pi(s_0)$ 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. $S = \{s_0, s_1, ..., s_T\}$ is one underlying state sequence of $\mathbf{C}$ to represent the $\mathbf{X}$. The key item in Eq. (2) is $p(\mathbf{x}_t | s_t)$ denoting the observation probability which can be rewritten as:

$$p(\mathbf{x}_t | s_t) = \frac{p(s_t | \mathbf{x}_t) p(\mathbf{x}_t)}{p(s_t)} \quad (3)$$

where $p(s_t | \mathbf{x}_t)$ is the state posterior probability, $p(s_t)$ is the prior probability of each state estimated from the training set, and $p(\mathbf{x}_t)$ is independent of the character sequence and thus can be ignored in the optimization of Eq. (1). In the traditional GMM-HMM system [20], $p(\mathbf{x}_t | s_t)$ is directly characterized by the GMM. However, in our proposed DNN-HMM system, DNN is adopted to calculate the state posterior probability $p(s_t | \mathbf{x}_t)$, which can be used to estimate $p(\mathbf{x}_t | s_t)$ via Eq. (3).

The DNN-HMMs can be trained using the embedded Viterbi algorithm with the following steps:

- **Step1**: **Building GMM-HMMs**
  The GMM-HMMs for all character classes are first built according to the maximum likelihood estimation [31]. By adopting the expectation-maximization (EM) and Baum-Welch algorithms, all parameters, including the state prior probabilities and transition probabilities of HMMs, and the weight/mean/varaiance parameters of GMMs, can be effectively estimated in an iterative manner and the Gaussian mixtures of all character classes can be progressively learned and increased from scratch.
- **Step2**: **Constructing DNN-HMM Topology**
  Before touching the hardcore of DNN-HMM system, the topology of HMM is directly copied from that of GMM-HMM system. And the corresponding state prior

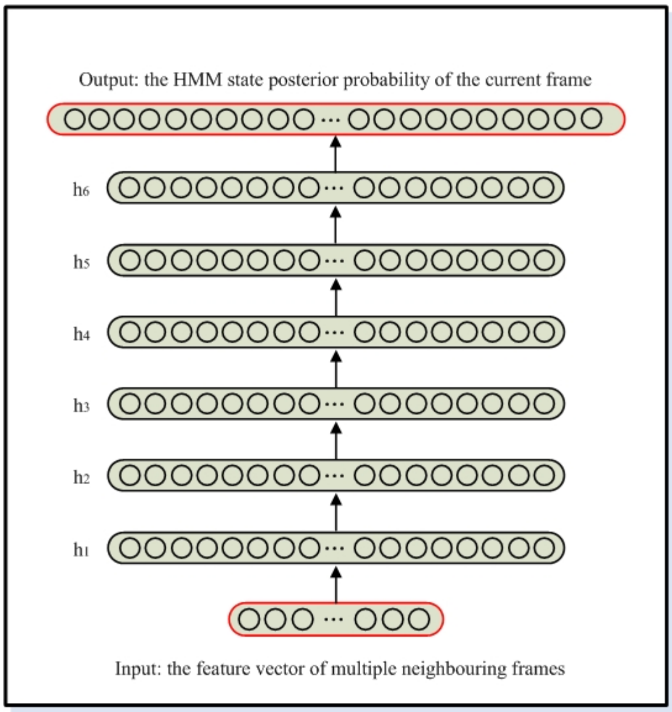

Fig. 4. DNN architecture.

probabilities and transition probabilities remain the same as those of GMM-HMM system.

- **Step3**: **Forced-Alignment**
  As the role of DNN is to predict the state posterior probability, the output labels should be prepared before the DNN training. This set of labels are initially generated by performing the state-level forced-alignment for all the training data with the text-level labels.
- **Step4**: **DNN Training**
  With the input features and the output labels, the parameters of DNN can be learned, which is elaborated in the next subsection.
- **Step5**: **Re-Alignment**
  To refine the state-level labels, the DNN trained in **Step4** is used for the re-alignment of all training data samples.
- **Step6**: Go to **Step4** for several times.

B. DNN Training

A DNN is a feed-forward, artificial neural network with more than one layer of hidden units between its input and output [7], as shown in Fig. 4. In this study, DNN aims to model the posterior probability of the HMM states using the input feature vector of multiple neighbouring frames. The activation function of each hidden layer is logistic sigmoid function. After the random initialization of all the weight parameters, a supervised fine-tuning of the parameters is conducted. For this multiclass classification task, output unit j converts its total input z_j into a class probability p_j by using

TABLE I
THE INFORMATION OF CASIA-HWDB DATABASES.

#	Class	Writer	Text Line	Character Sample
HWDB1.0	3,837	420	-	1,592,978
HWDB1.1	3,834	300	-	1,145,074
HWDB2.0	1,222	419	20,495	540,468
HWDB2.1	2,310	300	17,292	429,926
HWDB2.2	1,331	300	14,443	383,153

the “softmax” non-linearity:

$$p_j = \frac{\exp(z_j)}{\sum_k \exp(z_k)} \tag{4}$$

where k is an index over all classes. Then the objective function is the cross-entropy between the target probabilities $\bar{p}$ and the outputs of the softmax p:

$$C = -\sum_j \bar{p}_j \log p_j \tag{5}$$

where the target probabilities, taking values of one or zero, are the supervised information provided to train the DNN classifier. As our task involves large training samples, the objective function is optimized using back-propagation procedure with stochastic gradient descent in mini-batch mode. Please note that the number of the classes (or output layer nodes) corresponding to the number of states of all character classes, is much larger (about 20000 in our experiments) than that in previous work, e.g. 1379 in [21].

V. Language Modeling

The language modeling is quite an effective way to improve the recognition accuracy by incorporating more context information during the decoding. Statistical n-gram LM is widely used for speech recognition [32] is also adopted here. In this work, the character n-gram LM is considered rather than word n-gram LM. The SRILM toolkit [33] is employed to generate n-gram LM with different orders. Finally, integrated with character models using Eq. (1), the WFST-based decoder [24], [25] is implemented to generate the final recognition results using the Kaldi toolkit [34].

VI. Experiments

The experiments are conducted on the public database released by the Institute of Automation of Chinese Academy of Sciences (CASIA) [35]. The training set consists of the data from 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. The detailed information can refer to Table I. 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 is adopted

TABLE II
THE CER (IN %) COMPARISON OF THE PROPOSED DNN-HMM APPROACHES ON THE EVALUATION SET WITH DIFFERENT MODEL COMPLEXITIES (N_U IS THE NUMBER OF HIDDEN UNITS AND N_L IS THE NUMBER OF HIDDEN LAYERS).

(N_U, N_L)	(512, 6)	(1024, 6)	(2048, 6)	(2048, 5)	(2048, 4)
No LM	19.66	17.78	**17.01**	17.37	17.36
2-gram	9.83	8.80	8.59	**8.54**	8.72
3-gram	7.68	6.91	**6.73**	6.78	6.95

TABLE III
THE CER (IN %) COMPARISON OF THE PROPOSED DNN-HMM APPROACHES WITH DIFFERENT RE-ALIGNMENTS ON THE EVALUATION SET.

	DNN Baseline	ReFA-1	ReFA-2	ReFA-3	ReFA-4
No LM	17.01	16.31	16.20	**16.11**	16.25
2-gram	8.59	8.15	**7.80**	7.98	8.03
3-gram	6.73	**6.37**	6.45	6.50	6.48

[4]. The gradient-based feature extracted from one frame of the text line is a 256-dimensional vector, followed by PCA to obtain a 50-dimensional feature vector. This feature vector is directly used for GMM-HMM system while an augmented version of 7 frames is fed to DNN-HMM system.

For GMM-HMM system, each character class is modeled by a left-to-right HMM with 5 states. For each state, a GMM with 40 Gaussian mixtures is used. So there should be $3,980 * 5 * 40 = 796,000$ Gaussians in all. For DNN-HMM system, the input size of DNN is 350 while the output size is $3,980 * 5 = 19,900$ corresponding to the number of states of all classes. The mini-batch size is 256. The initial step size is set to 0.008 which is halved after each iteration if the loss of cross-validation set is reduced. 16 iterations are conducted.

As for the n-gram 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, 115MB texts of People's Daily between 2000 and 2004, 129MB texts of other newspapers, and 93MB texts of Sina News. For all experiments, the Kaldi tool [34] is adopted for both training and decoding. The evaluation measure is the character error rate, which is ratio between the total number of substitution/insertion/deletion errors and the total number of character samples in the evaluation set.

First, we list a CER comparison of the proposed DNN-HMM approaches on the evaluation set with different model complexities in Table II. With the increase of hidden units, the CER was significantly reduced, e.g. from 7.68% ($N_\mathrm{U} = 512$) to 6.73% ($N_\mathrm{U} = 2048$). And the recognition performance was saturated when using a maximum of 6 hidden layers. All these observations were consistent for different LMs. For the following experiments, the configuration of $(2048, 6)$ was used as default. From Table II, the use of LM was crucial to the recognition accuracy. A relative CER reduction of 60.4% was achieved by the system using 3-gram over the system with no LM under the setting of $(2048, 6)$.

Second, we show a CER comparison of the proposed DNN-HMM approaches with different re-alignments on the evaluation set in Table III. For different LMs, the re-alignment could always improve the recognition performance. One interesting observation was that the weak LM tends to need more passes of re-alignment than the strong LM to achieve the best performance. For example, three passes of re-alignment (ReFA-3) were in demand for the system with no LM while only one pass re-alignment (ReFA-1) was enough for 3-gram system. This was reasonable as the re-alignment of the frame-level labels also improved the context information to a certain extent, which was with a similar role to the use of a strong LM.

TABLE IV
THE CER (IN %) COMPARISON OF THE DIFFERENT APPROACHES ON THE EVALUATION SET.

	DNN-HMM	MDLSTM-RNN [21]	[19]
No LM	**16.11**	16.5	22.66
With LM	**6.50**	10.6	9.25

Finally, we give a CER comparison of the different approaches on the evaluation set in Table IV. We could observe that without using LM both proposed DNN-HMM and MDLSTM-RNN approaches significantly outperformed the approach in [19] , which demonstrated that the segmentation-free approaches were much more effective than the oversegmentation approach especially without using more context information. Furthermore, our DNN-HMM approach achieved a comparable recognition performance with MDLSTM-RNN approach without using LM. This could be explained from two aspects. From the aspect of modeling units, the state-level resolution of DNN-HMM was higher than the character-level resolution of MDLSTM-RNN. From the other aspect of learning context information, the LSTM architecture with CTC technique in MDLSTM-RNN approach was superior to the frame-level optimization using CE criterion in DNN-HMM approach. So the comparable performances between DNN-HMM and MDLSTM-RNN should be the trade-offs between those two aspects. However, after using 3-gram LM, our DNN-HMM approach yielded a much better result than MDLSTM-RNN approach, which could be partly due to that MDLSTM-RNN already considered more context information than DNN-HMM. Overall, our proposed approach could achieve a relative CER reduction of 29.7% than the best reported results in [19].

VII. CONCLUSION

In this study, we investigate on the offline handwritten Chinese text recognition by using DNN-HMMs. With the well designed feature extraction, character modeling using DNN-HMM, and language modeling using n-gram, the proposed approach achieves the best published results on the ICDAR 2013 competition task, which also yields a large performance gain over the previously best results. As for the future work, more advanced techniques, e.g. CNN architecture or sequence training of DNN, will be integrated in the proposed HMM framework.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grants No. 61305002, National Key Technology Support Program under Grants No. 2014BAK15B05, the "Strategic Priority Research Program" of the Chinese Academy of Sciences under Grant No. XD-B02070006, MOE-Microsoft Key Laboratory of USTC.

References

[1] C.-L. Liu, S. Jaeger, and M. Nakagawa, "Online recognition of Chinese characters: the state-of-the-art," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 26, No. 2, pp.198-213, 2004.

[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.

[5] J. Du and Q. Huo, "Designing compact classifiers for rotation-free recognition of large vocabulary online handwritten Chinese characters," *Proc. ICASSP*, 2012, pp.1721-1724.

[6] C.-L. Liu, H. Sako, and H. Fujisawa, "Discriminative learning quadratic discriminant function for handwriting recognition," *IEEE Trans. on Neural Networks*, Vol. 15, No. 2, pp.430-444, 2004.

[7] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Processing Magazine*, Vol. 29, No. 6, pp.82-97, 2012.

[8] 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, 1986.

[9] O. Russakovsky et al., "ImageNet large scale visual recognition challenge," 2014.

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

[11] 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.

[12] D. C. Ciresan and J. Schmidhuber, "Multi-column deep neural networks for offline handwritten Chinese character classification," *Preprint arXiv:1309.0261*, 2013.

[13] 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.

[14] 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.

[15] 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.

[16] C.-H. Wang, Y. Hotta, M. Suwa, and S. Naoi, "Handwritten Chinese address recognition," *Proc. IWFHR*, 2004, pp.539-544.

[17] 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.

[18] 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.

[19] 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.

[20] 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.

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

[22] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for improved unconstrained handwriting recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 31, No. 5, pp.855-868, 2009.

[23] 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.

[24] 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.

[25] 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.

[26] C.-L. Liu, K. Nakashima, H. Sako, and H. Fujisawa, "Handwritten digit recognition: benchmarking of state-of-the-art techniques," *Pattern Recognition*, Vol. 36, No. 10, pp.2271-2285, 2003.

[27] C.-L. Liu, "Normalization-cooperated gradient feature extraction for handwritten character recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 29, No. 8, pp.1465-1469, 2007.

[28] Z.-L Bai and Q. Huo, "A study on the use of 8-directional features for online handwritten Chinese character recognition," *Proc. ICDAR*, 2005, pp.262-266.

[29] A. C. Rencher, *Methods of Multivariate Analysis*, Wiley, New York, 2002.

[30] 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.

[31] B. H. Juang, S. Levinson, M. Sondhi, "Maximum likelihood estimation for multivariate mixture observations of Markov chains," *IEEE Trans. on Information Theory*, Vol. 32, No. 2, pp.307-309, 1986.

[32] F. Jelinek, "The development of an experimental discrete dictation recognizer," *Proceedings of the IEEE*, Vol. 73, No. 11, pp.1616-1624, 1985.

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

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

[35] 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.