Unconstrained online handwritten Uyghur word recognition based on recurrent neural networks and connectionist temporal classification

Mayire Ibrayim, Wujiahematiti Simayi and Askar Hamdulla*

Institute of Information Science and Engineering,
Xinjiang University, China
Email: ingiaoling@sina.com
Email: Xfhu123@outlook.com
Email: Askar_H@21cn.com
*Corresponding author

Abstract: This paper conducts the first experiments applying recurrent neural networks-RNN accompanied with connectionist temporal classification (CTC) to build end-to-end online Uyghur handwriting word recognition system. The traced pen-tip trajectory is fed to network without conducting segmentation and feature extraction. The network is trained to transcribe handwritten word trajectory to a string of characters in alphabet which has total 128 character forms. In order to avoid overfitting during training and improve generalisation of the model, dropout technique is implemented. An online handwritten word dataset has been established and used for model training and evaluation in writer independent manner. Recognition results are evaluated by calculating the Levenshtein-edit distance and 14.73% character error rate CER on test set of 3,600 samples for 900 word classes has been observed without help of any lexicon search and language model.

Keywords: online handwriting recognition; recurrent neural networks; connectionist temporal classification; CTC; dropout; Uyghur words.

Reference to this paper should be made as follows: Ibrayim, M., Simayi, W. and Hamdulla, A. (2021) 'Unconstrained online handwritten Uyghur word recognition based on recurrent neural networks and connectionist temporal classification', *Int. J. Biometrics*, Vol. 13, No. 1, pp.51–63.

Biographical notes: Mayire Ibrayim received her PhD in Communication and information systems from Wuhan University in 2013. And she engaged in Post-doctoral research in computer science and technology at the school of Information Science and Engineering, Xinjiang University from 2013 to 2016. She is currently an Associate Professor at the school of Information Science and Engineering, Xinjiang University in China. Her research interests include document image analysis, handwriting recognition, and pattern recognition.

Wujiahematiti Simayi is a PhD candidate in the Institute of Information Science and Engineering of Xinjiang University. He is currently working on unconstrained handwriting recognition studies. His research interests include pattern recognition, multilingual handwriting recognition and deep learning technologies.

Askar Hamdulla received his BE in 1996, ME in 1999 and PhD in 2003, all in Information Science and Engineering, from University of Electronic Science and Technology of China. Currently, he is a Professor in the Institute of Information Science and Engineering, Xinjiang University. His research interests include image processing and natural language processing. He is a senior member of CCF and an affiliate member of IEEE.

1 Introduction

Handwriting recognition is a main branch of pattern recognition and hot research topic in artificial intelligence. Based on the representation of handwritten entries, there are two basic types that handwriting recognition is categorised, online and offline. Handwriting recognition technology based on the recorded trajectory with sequential time order is called online handwriting recognition, while offline handwriting recognition processes scanned images of handwritten shapes on natural objects (Liu et al., 2013; Su, 2013). Achievements on both online and offline handwriting recognition has been witnessed on well-investigated script kinds such as English and Chinese (Jaderberg et al., 2016; Wu et al., 2017). Several competitions have been held to improve handwriting recognition technology on theses scripts and many successful recognition systems were compared (Yin et al., 2013).General pattern recognition systems are showing great potential for many different applications. In handwriting recognition field, we can gratefully mention the integrated segmentation and recognition, HMMs, Neural networks as the most successful ones (El Abed and Märgner, 2011). Nevertheless, handwriting word recognition for alphabetic scripts is challenging because of large vocabulary to be recognised. More difficulties are to be addressed for cursive natured scripts and handwritten texts.

Uyghur is an alphabetic script which uses many adapted Arabic letters and some unique ones to form a word and sentences. Due to its large vocabulary and cursive nature, Uyghur handwriting recognition is always found challenging. Despite the interesting results, there are still some deadly drawbacks for Uyghur Handwritten word recognition. Previous studies either based on a holistic recognition approach or analytic recognition methods which apply implicit or explicit segmentation techniques to find out the specific characters within a word. Holistic approach becomes inconvenient or inapplicable for large vocabulary set to be recognised. Analytic approaches are highly relied on the carefully designed segmentation techniques. The early experiments for Uyghur handwritten word recognition were based on carefully written handwritten samples in which the component characters can be easily segmented from the overall word shape. However, real-world Uyghur handwriting always misses some characters from a handwritten word shape. The highly cursive nature of Uyghur handwritten word shapes makes analytic approaches and HMMs weak on the natural handwritten samples which are often casual in shape. Therefore, an unconstrained recognition system which does not rely on the case-based segmentation is needed for real-world applications.

Recurrent neural networks (RNNs) have been showing capability to analyse sequenced information and is being applied in many research fields including speech and handwriting recognition (Graves et al., 2007). Connectionist temporal classification (CTC) is gaining admirations for its sequential labelling on un-segmented input sequence

(Graves et al., 2006). The integration of RNNs and CTC can deal with the segmentation problem conveniently and learn very different handwritten styles simultaneously. By implementing the integration of RNNs and CTC, this paper presents an unconstrained end-to-end online Uyghur handwriting word recognition system which can output word recognition result without character segmentation. As the first unconstrained word recognition study, the experiment is conducted without help of lexicon search and external language model.

The remaining content is arranged in several sections where Section 2 reviews some studies on handwriting recognition; Section 3 details the implemented recurrent network model structure; characteristics of Uyghur words and established dataset are described in Section 4. Experiment results are discussed in Section 5 and Section 6 draws conclusion.

2 Related studies

Studies on Uyghur handwriting recognition have also implemented the classic methods for character and word level recognitions task. HMM based and integrated segmentation-recognition approaches have been the most common word recognition systems (Simayi et al., 2016). For example, Chen (2013) reported 91.7% accuracy on 2,000 words using HMM; 93.71% recognition rate on 1058 word classes was achieved by Pi (2012). Then Liu (2012) put their 97% recognition rate of HMM based word recognition on 1066 words. A combination of HMM and GMM by Reyiman and Wushour (2017) produced 99% accuracy on 1000 words. However, strict constraints on writing while collecting samples and the manual segmenting of characters restrict the universality of the proposed models. However, almost all studies have been reported with help of finite lexicon set or using holistic approaches, therefore, an unconstrained handwritten word recognition result has not been reported yet.

Integrated segmentation-recognition method is promising for unconstrained analytical word recognition (Wushour, 2013).

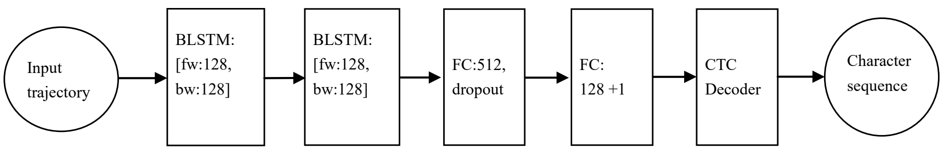
Ibrayim and Hamdulla (2015) achieved preliminary result using DTW based over-segmentation approach and reported 94.85% accuracy on 1,460 words. Xu (2013) also used two stages DTW for the integrated segmentation and recognition. The experiments based on the feedback mechanism obtained 94.93% recognition rate on 500 words. Although the segmentation based studies showed potential for developing unconstrained handwriting word recognition system, they were suffered from the highly cursive nature of Uyghur handwriting.

Recently, neural networks and deep learning techniques are updating state-of-arts in many fields including handwriting recognition. Convolutional neural networks let Simayi et al. (2017) have a very high word recognition accuracy, 99.31%, on 2344 words using holistic method. Li et al. (2016) performed text-line recognition experiments on printed Uyghur texts using deep RNNs with CTC. The best performance was observed 9.9% character error rate (CER) on the new collected printed text-line image dataset THOCR. The best CER on smaller subset was reported as 1.29%. Since printed texts are much regular than handwritten ones, as compared in Figure 2, handwriting recognition would be much more challenging

3 Unconstrained handwriting recognition system

A deep neural network including two bidirectional recurrent layers and several fully connected layers is proposed to build a first unconstrained online handwriting Uyghur word recognition system in this paper. The overall architecture of the applied recurrent network is given in Figure 1. Handwritten trajectory is fed into the first recurrent layer with simple pre-processing. The variable length handwritten trajectory is transformed or encoded to fixed length output sequence via feature learning through the first recurrent layer. Considering the subtlety to gradient vanishing of recurrent networks, Long Short Term Memory-LSTM is applied for each cell or unit of the recurrent layers.

Figure 1 Architecture of unconstrained handwritten word recognitions system



Notes: BLSTM means bidirectional LSTM layer, fw and bw means forward and backward recurrent layers, respectively. FC means fully connected layer. The number of units for each layer is given correspondingly.

The output of the first recurrent layers is directly sent to the second recurrent layer to obtain more generalised sequential features. The features from two successive LSTM layers are flattened and sent to the fully connected layers. The fully connected layers are assumed to generalise the learned features effectively and perform classification. We applied dropout regularisation on fully connected layers to avoid overfitting, because dense connectivity in fully connected layers adds large number of variables to the network. The number of neurons in the last fully connected layer is set by the number of characters/letters in alphabet and a Blank label especially designed for CTC decoding. The last output string (character sequence) is obtained by CTC decoder with beam search algorithm.

3.1 Input

- *Pre-processing*. Long sequences are hard to be modelled and make training very difficult and extremely slow. The raw handwritten trajectory goes through very simple pre-processing including noise and duplication removing, and turn-point selection procedures before it is sent to model as input (Zhang et al., 2018). On each stroke of handwritten word trajectory, the removing thresholds are set based on average neighbour point distance. If the distance from a point to its previous neighbour is greater than three times of average neighbour-point distance, this point is treated as noise and removed. 1/2 of the average neighbour point distance is used as duplication removing threshold. If the distance from a point to its previous neighbour is smaller than the duplication removing threshold, this point is removed from the trajectory. A point in the trajectory is detected as critical turn point if direction change θ exceeds the threshold of $\pi/12$ in this paper. Direction change θ at a point in the trajectory is obtained by its previous and next neighbour points using equation (3). a, b, c are the Euclidian distances between the three successive points, respectively.

$$\theta = \pi - \arccos\left(\frac{b^2 + c^2 + a^2}{2bc}\right) \tag{1}$$

- *Representation*. In order to enrich informative content of the input, direction vector (Δx, Δy) and a two dimensional pen-state vector are put together with min-max normalised (x, y) coordinates of each point. Thus, each point in input sequence is in shape of [x, y, Δx, Δy, PS[0], PS[1]] which is of 6-dimension. The temporal direction vector is easily calculated using equation (2) where (x_t, y_t) and (x_{t-1}, y_{t-1}) are the coordinates of current and previous points in trajectory.

  $$\Delta x = x_t - x_{t-1} \qquad \Delta y = y_t - y_{t-1} \tag{2}$$

  The pen-state vector represents pen-up and pen-down states of pen-tip movement on handwriting table screen and thus has real significance. The pen-state values use two dimensional vector which are [1, 0] for pen-down and [0, 1] for pen-up states. The pen-state values can be obtained by stroke end marks or stroke order changes.

3.2 Bidirectional recurrent layer (LSTM cells)

Observing handwritten trajectory from both right-left and left-right directions is more demonstrative and fit for the nature of online handwriting process. So, bidirectional LSTM layer is adopted for each recurrent layer in the proposed system. A bidirectional recurrent layer consists two uni-directional sub recurrent layers. Input sequence is fed to one uni-directional recurrent layer in original order while another one receives the input sequence in reverse order. Input, output and state values in and LSTM cell are controlled by gate mechanism, as given in equations (3)–(7).

$$i_t = sigm\left(W_i x_t + U_i h_{t-1} + b_i\right) \tag{3}$$

$$f_t = sigm\left(W_f x_t + U_f h_{t-1} + b_f\right) \tag{4}$$

$$o_t = sigm\left(W_o x_t + U_o h_{t-1} + b_o\right) \tag{5}$$

$$c_t = f_t \odot c_{t-1} + \tanh\left(W_o x_t + U_o h_{t-1} + b_o\right) \tag{6}$$

$$h_t = o_t \odot \tanh\left(c_t\right) \tag{7}$$

where W_i, W_f, W_o are the input-hidden weight matrix, U_i, U_f, U_o are the state-state weight matrix and b_i, b_f, b_o are bias vectors, respectively. i_i, i_f, i_o are the activation values at the input, forget and output gates, while c_t and h_t are the state and output values of the cell.

The output of the two sub-recurrent layers are concatenated into longer sequence, as in equations (8)–(10).Using concatenation sub-layer outputs performed much better in our experiments than using averaged outputs.

$$Y_{forward} = Y_{right-left} = \left[y_{r1}, y_{r2}, \ldots\ldots y_{rn} \right] \tag{8}$$

$$Y_{backward} = Y_{left-right} = \left[y_{l1}, y_{l2}, \ldots\ldots y_{\ln} \right] \tag{9}$$

$$\mathrm{Y} = concat\left(Y_{forward}, Y_{backward} \right) \tag{10}$$

where y_{rN} represents the output of the N^{th} node of in right-left sub-recurrent layer, y_{lN} represents the output of the N^{th} node of in reverse sub-recurrent layer, $Y_{forward}$ and $Y_{backward}$ are the outputs of the two inverse left-right and right-left sub-layers and Y is the last output of the bi-directional recurrent layer.

3.3 Output unit

Different from holistic or lexicon based handwriting word recognition, word labels should cover whole vocabulary in the language for unconstrained recognition task. 128 character forms based word transcriptions are used as labels in our dataset. So, the number of nodes in the last fully connection-FC layer which its output is decoded into character string is set by the number of basic character shapes, 128, with adding the Blank label used in CTC decoding. Each node in the last FC layer reflects the probability of each character form to be the in the input handwritten trajectory by softmax classification as given in equations (11)–(12).

$$Z_k = W_k \cdot x + b \tag{11}$$

$$P_k = P\left(y = k / x \right) = \frac{e^{Z_k}}{\sum_{k=1}^{K} e^{Z_k}} \tag{12}$$

where Z_k is the output of the nodes in the classification layer of the network with given input x and weights to the node W_k and bias b; P_k or P(y = k/x) represents the output probability of character for k given input x.

3.4 Connectionist temporal classification

CTC has been being admired for its capability to align variable length input-target sequence couples (El Abed and Märgner, 2011). With its help, various un-segmented sequence labelling systems including handwriting recognition are being realised. It can generate character sequence outputs of recurrent networks to character or label strings with variable length. This is the most expected function to build lexicon independent, unconstrained handwriting recognition system for alphabetic scripts. CTC tries to find the most possible input-label sequence match. The probability of the input-output sequence path π is the product of all output probabilities at every time step, see equation (13)

$$P\left(\pi | x \right) = \prod_{t=1}^{T} P\left(k \mid t, x \right) \tag{13}$$

The probability of an output label sequence $P(l|x)$ can be obtained by summing all probabilities of the possible paths π, after removing all blanks and repeated labels, denoted as $B^{-1}(l)$ as in equation (14).

$$P(l|x) = \sum_{\pi \in B^{-1}(l)} P(\pi|x) \tag{14}$$

Analysing all possible paths to the same output label sequence makes CTC able to be applied directly on un-segmented sequence input. With the help of CTC, RNNs can be trained to minimise the objective function, see equations (15)–(16), with given target sequence.

$$CTC(x) = -\log P(y \mid x) \tag{15}$$

$$y^* = B(\pi^*), \qquad \pi^* = \arg\max_\pi P(\pi \mid x) \tag{16}$$

The most probable character string is produced as word recognition output with help of beam search algorithm. It always works together with CTC in most of the Deep learning frameworks (Li et al., 2015). Beam search algorithm computes the all possible character combinations to find the most possible output sequence for the input trajectory.

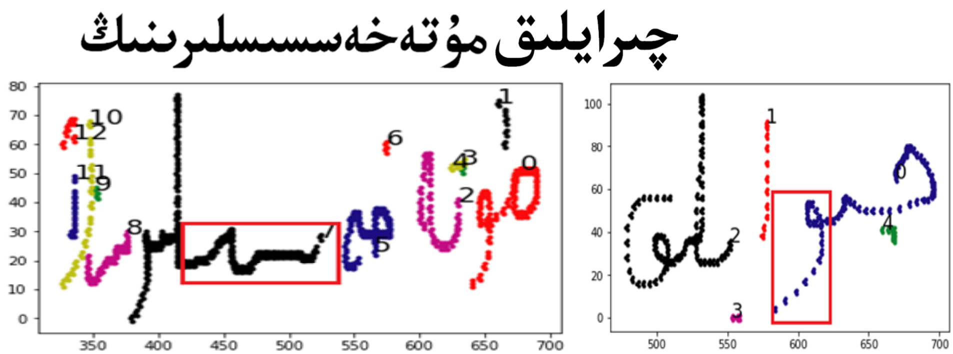
4 Data collection

Modern Uyghur is written from right to left and from top to down orientation. There are 32 basic Uyghur characters containing 128 writing forms according the position of character in a word. A dataset has been established by collecting online handwritten word samples from 26 different writers. The dataset contains 900 word classes and each word is recorded with associate Unicode based word transcription. Each writer is asked to write all word classes and recorded handwritten word trajectories of each writer are saved in separate binary files, with POTEX extension. An online handwritten word trajectory consist sequentially recorded pen-tip (x, y) coordinates. A stroke in handwritten trajectory is separated from its neighbour by special stroke-end mark and complete word trajectory is ended by another word-end mark. A handwritten word trajectory in binary files is put together with its two kinds of Unicode word transcriptions respectively based on character types and character forms. The overall trajectory information put in sample information includes total number of strokes and points, word order and writer index etc. The collected 23,400 handwritten word samples are divided into training and test sets with respect to writers to conduct writer independent word recognition experiments. 19,800 samples from 22 writers are put in training set while the remained 3,600 samples from other four writers are used as test set. The dataset used in this paper is temporally noted as UY-OLHW2.1 which is a pre-built subset of a larger dataset we are preparing.

A handwritten word, especially in cursive natured scripts, always misses some character forms because of joining with neighbour characters or casual continues handwriting, as shown in Figure 2. Strokes are drawn with different colours and a numerical notation is given at the beginning of each stroke to represent the original time order of the stroke during handwriting. Since Uyghur has obvious cursive style both in printing and handwriting cases, a stroke always consists several character forms. The handwritten word sample in Figure 2(a) has missed some character forms, and

Figure 2(b) shows a handwritten word with false written character forms. We found there are many such careless written samples in the dataset. However, they are regarded as readable in whole view and still kept in the train and test sets.

Figure 2 Some handwritten word samples with printed shapes, (a) missed or connected (b) false written (see online version for colours)



(a) (b)

5 Experiments and results

5.1 Configuration

In pre-processing, a point in the trajectory is removed if its distance to previous neighbour is less than half of the average neighbour distance or larger than three times of it. If a point has larger direction change from a point to its previous neighbour point, than the threshold of $\pi/6$, is judged as turn-point and retained as informative trajectory point.

The proposed model generates output sequence of specific character shape labels directly. The decoded output character string is used as word recognition result without help of any lexicon search and external language models. Thus, the system can be regarded as unconstrained handwritten word recogniser. Model performance is evaluated using character error rate-CER (Sun et al., 2016) which is calculated by equation (17). This can be transformed to character accurate rate-CAR using equation (18).

$$CER = \frac{D_e + S_e + I_e}{N_t} \tag{17}$$

$$CAR = 1 - \frac{D_e + S_e + I_e}{N_t} \tag{18}$$

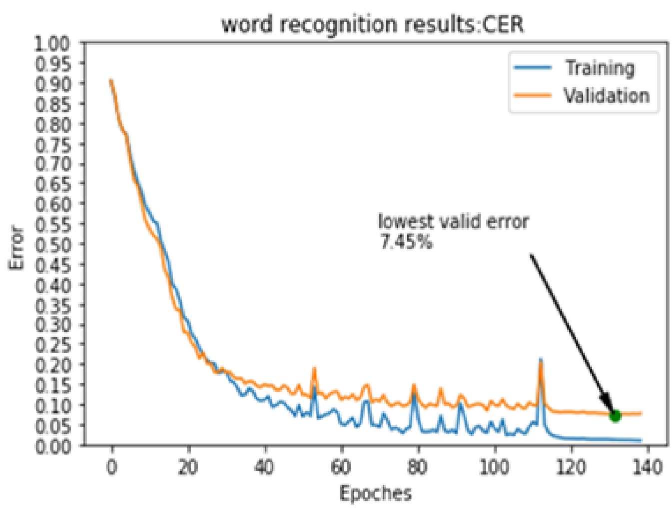
where N_t is the total characters in the reference text. S_e, D_e and I_e denote substitution errors, deletion errors and insertion errors, respectively. In fact, these are just the minimum edit distance which is needed to align the output sequence to ground truth. This is exactly the Levenshtein distance and calculated by dynamic programming.

The codes are written in Python within Tensor flow deep learning framework. The models are trained using one Titan X GPU with 12G RAM for acceleration and evaluated on Intel Core 3.1GHZ I5 CPU and with 4G RAM. One of most favoured self-adaptive optimisers-Adam with initial learning of 0.001 is used in all experiments (Kingma and Ba, 2015). Samples from one writer in training set are temporarily used for performance validation during training and remained samples from 21 writers are used to update network parameters. Train samples are shuffled in each epoch and put 64 samples in a mini-batch. Dropout rate is set as 0.5 and global learning rate is lowered by decreasing factor of 0.5 when no improvement seen in successive three epochs on validation set. The training is stopped by early stopping mechanism with ten successive epochs cannot see any benefit of training. The generalisation ability of the trained model is evaluated on the test set which contains 3,600 samples from new four writers.

5.2 Results and discussion

Training RNN on long sequence inputs is difficult and become more challenging when it is accompanied with CTC. The recorded training procedure in Figure 3 illustrates the results against per epoch. The results in Figure 3 are the character error rates on ten batches from train and validation sets, respectively, in order to save training time.

Figure 3 RNN-CTC training records for online handwritten word recognition (see online version for colours)



Note: The results are based on ten batches from train and validation sets.

Huge number of variables in the implemented network are wanted be adapted to their optimal values for task the networks handles. The implemented network model treats with around 2 million parameters during training and evaluation as well. Thanks to GPU acceleration that it spent only around 5.2 minutes for each epoch of training. This acceleration is also attributed to the pre-processing which shortened input trajectory, too.

Observing the training and validation result curve can find that the implemented self-adaptive optimiser produced very steady decline in error at the beginning phase of training. Model reached to substantially low error rate at around 60 epochs. Although training has last for another 60 epochs, it didn’t give much improvement in model generalisation which is expressed as validation errors. But model learned the samples in train set quite well without occurring obvious over-fitting due to dropout regularisation. This also demonstrates the great learning ability of neural networks. No cross-validation is adopted for model training since training is very costly.

After finishing whole training process, model is evaluated on all samples of train set and test set. The word recognition results from experiments using the proposed model are given Table 1. The recorded information in Table 1 includes CER results on train and test sets, as well as the character accurate rate on test set. The total number of epochs and the time per epoch, average recognition time per sample are also given in Table 1.

Table 1 Handwriting word recognition training records and test results

Model	*No. vars*	*Model size (M)*	*No. ep*	*T/ep (min)*	*Tr_CER (%)*	*Te_CER (%)*	*Te_CAR (%)*	*Av-recT (s)*
Char128	1,994,117	7.8	150	5.2	0.99	14.73	85.27	0.035

Notes: Where Tr_CER and Te_CER are the character error rates on train and test sets, respectively, Te_CAR means character accurate rate on test set, No.ep and T/ep the number of epochs that training stopped and average time spent per epoch. Av-recT means average recognition time per sample.

The trained model gave good word recognition result without any lexicon search or language model on both train and test set. Model performance on train set is very pleasing that only 0.99% CER has been reached. Generalisation ability of the trained model is evaluated on 3,600 samples for 900 words classed from four new writers, and 14.73% character error rate has been gifted, which equals to 85.27% character accurate rate. Sending too much samples to model at once easily overloads memory, batch size of 512 is used when evaluation, and measured average recognition time per sample as around 0.035 s. This speed is applicable for real-world application.

Since this is the first unconstrained handwritten word recognition system which is different from the previous works either conducted in holistic manner or within finite lexicon set, we found ourselves unable to make a fair comparison of the reported results on online Uyghur handwritten word recognition.

5.3 Comparison of the fundamentals

We do not think the experiment result in this paper can be compared with the ones from the previous studies. Meanwhile, the experiment results from the previous studies are not comparable, too. Up to date, there is not a public database has been released for Uyghur handwriting recognition and we are working on it.

We also did not compare analytic approaches or HMMs on this database. A simple analysis on the fundamentals can lead to the superiority of the proposed system over analytic and HMM based approaches.

1 Analytic approaches are applicable for the character separable strings, such as number strings, CJK (Chinese, Japanese and Korean) scripts etc. However, analytic approach does not work well on highly cursive scripts like Uyghur.

2 HMM takes just one previous state (dot or segment) into consideration to get the current state. Thus, HMMs also require well-formed handwritten word samples for experiments.

3 RNNs analysed multiple previous states to decide the current state, thus have much strong learning ability that can tackle a word sample with some missed character shapes in it. The following points are believed the superiority of the proposed recognition system over other traditional methods.

- The proposed system can make character level outputs directly from an Uyghur handwritten word trajectory, without character segmentation which is extremely challenging for Uyghur.
- The proposed system provides the character-level recognition result directly without lexicon search which is very common in previous studies.
- The proposed system can learn different handwriting styles during training. This introduces robust and easily adapted recognisers.

Therefore, the proposed system has great potential to develop real-world applications for general use. However, the analytic or HMM based approaches are restricted by careless handwriting and not expected to provide robust recognition system for Uyghur script. We will make our database public for research issues to make a benchmark. More in-depth research activities are still needed and the results in this paper can be the first baseline for later studies on unconstrained handwriting word recognition. This model can also be applied for text line recognition tasks, too.

6 Conclusions

In this paper, a first unconstrained online handwriting word recognition study is performed using RNN model and CTC. Handwritten word trajectory is sent to bidirectional recurrent networks without segmentation. The most possible character strings are obtained as word recognition result from the model directly without any lexicon set or integrated language model. A dataset has been established that the collected 23,400 samples are split into standard training and test sets in writer-specific manner. The character error rates on train and test sets have been measured as 0.99% and 14.73%, respectively. The promising results reached in this paper encourages to more intensive studies for further improving the model generalisation.

Acknowledgements

This work is supported by Youth Fund for scientific research program of Autonomous Region (XJEDU2019Y007); National Science Foundation of China (NSFC) under grant number 61462081 and 61263038. The second author is grateful to National Laboratory of Pattern Recognition of CASIA (Institute of Automation, Chinese Academy of Sciences) for providing the excellent study and experiment environment.

References

Chen, X.J. (2013) *HMM Based Recognition Research on Online Handwriting Uyghur*, XinJiang University.

El Abed, H. and Märgner, V. (2011) 'ICDAR 2009-Arabic handwriting recognition competition', *International Journal on Document Analysis and Recognition (IJDAR)*, Vol. 14, No. 1, pp.3–13.

Graves, A., Fernández, S., Gomez, F. and Schmidhuber, J. (2006) 'Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks', in *Proceedings of the 23rd International Conference on Machine Learning*, ACM, Pittsburgh, Pennsylvania, USA, pp.369–376.

Graves, A., Liwicki, M., Bunke, H., Schmidhuber, J. and Fernández, S. (2007) 'Unconstrained on-line handwriting recognition with recurrent neural networks', in *Conference on Neural Information Processing Systems*, DBLP, Vancouver, Canada, pp.458–464.

Ibrahim, M. (2013) *Key Technologies For Recognition of Online Handwritten Uyghur Characters and Words*, PhD, WuHan University.

Ibrayim, M. and Hamdulla, A. (2015) 'On-line Handwritten Uyghur Word Recognition Using Segmentation-Based Techniques', *International Journal of Signal Processing Image Processing & Pattern Recognition*, Vol. 8, No. 9, pp.256–265.

Jaderberg, A., Simonyan, K., Vedaldi, A. and Zisserman, A. (2016) 'Reading text in the wild with convolutional neural networks', *Int. J. Computer Vision*, Vol. 116,No. 1, pp.1–20.

Kingma, D.P. and Ba, J. (2015) 'Adam: a method for stochastic optimization', in *The 3rd International Conference for Learning Representations*, San Diego, http://arxiv.org/abs/1412.6980.

Li, J., Zhang, H., Cai, X. and Xu, B. (2015) Towards end-to-end speech recognition for Chinese mandarin using long short-term memory recurrent neural networks', *Proc. 6th Annual Conference of the International Speech Communication Association*.

Li, P., Zhu, J., Peng, L. and Guo, Y. (2016) 'RNN based Uyghur text line recognition and its training strategy', *2016 12th IAPR Workshop on Analysis Systems (DAS)*, IEEE, Santorini, Greece, pp.19–24.

Liu, C.L., Yin, F., Wang, D.H. and Wang, Q.F. (2013) 'Online and offline handwritten Chinese character recognition: benchmarking on new databases', *Pattern Recognition*, Vol. 46, No. 1, pp.155–162.

Liu, L.L. (2012) *Design and Building of the Handwriting Uyghur Words Recognition Using HTK*, Ms, XinJiang University.

Pi, G.L. (2012) *HMM Based Recognition Method Research On Online Handwriting Uyghur Word*, Ms, XinJiang University.

Reyiman, T. and Wushour, S. (2017) 'A Uyghur online handwriting recognition system', *Journal of Chinese Information Processing*, Vol. 28, No. 3, pp.112–115.

Simayi, W., Hamdulla, A. and Liu, C.L. (2017) 'Holistic handwritten Uyghur word recognition using convolutional neural networks', *2017 4th IAPR Asian Conference on Pattern Recognition (ACPR)*, IEEE, pp.846–851.

Simayi, W., Ibrayim, M., Tursun, D. and Hamdulla, A. (2016) 'A survey on the classifiers in on-line handwritten Uyghur character recognition system', *International Journal of Hybrid Information Technology*, Vol. 9, No. 3, pp.189–198.

Su, T. (2013) *Chinese Handwriting Recognition: An Algorithmic Perspective*, Springer Briefs in Electrical & Computer Engineering, Springer, Berlin, Heidelberg.

Sun, L., Su, T., Liu, C. and Wang, R. (2016) 'Deep LSTM networks for online Chinese handwriting recognition', in *2016 15th International Conference on IEEE Frontiers in Handwriting Recognition (ICFHR)*, Shenzhen, China, pp.271–276.

Wu. Y.C., Yin, F., Chen, Z. and Liu, C.L. (2017) 'Handwritten Chinese text recognition using separable multi-dimensional recurrent neural network', *Proc. 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, IEEE, Vol. 1, pp.79–84.

Xu, Y.M. (2013) *A Study of Key Technologies For Uyghur Handwriting Recognition*, PhD, XiDian University.

Yin, F., Wang, Q-F., Zhang, X-Y. and Liu, C-L. (2013) 'ICDAR 2013 Chinese handwriting recognition competition', in *2013 12th International Conference on Document Analysis and Recognition (ICDAR)*, IEEE, Washington, DC, USA, pp.1464–1470.

Zhang, X.Y., Yin, F., Zhang, Y.M., Liu, C.L. and Bengio, Y. (2018) 'Drawing and recognizing chinese characters with recurrent neural network', *IEEE Transactions on Pattern Analysis & Machine Intelligence*, Vol. 40, No. 4, pp.849–862.