Training an End-to-End Model for Offline Handwritten Japanese Text Recognition by Generated Synthetic Patterns

Nam Tuan Ly, Cuong Tuan Nguyen, Masaki Nakagawa
Department of Computer and Information Sciences
Tokyo University of Agriculture and Technology
2-24-16 Naka-cho, Koganei-shi, Tokyo, 184-8588 Japan
{namlytuan, ntcuong2103}@gmail.com, nakagawa@cc.tuat.ac.jp

***Abstract* – This paper presents an end-to-end model of Deep Convolutional Recurrent Network (DCRN) for recognizing offline handwritten Japanese text lines. The end-to-end DCRN model has three parts: a convolutional feature extractor using Deep Convolutional Neural Network (DCNN) to extract a feature sequence from a text line image; recurrent layers employing a Deep Bidirectional LSTM to predict pre-frame from the feature sequence; and a transcription layer using Connectionist Temporal Classification (CTC) to convert the pre-frame predictions into the label sequence. Since our end-to-end model requires a large data for training, we synthesize handwritten text line images from sentences in corpora and handwritten character patterns in the Nakayosi and Kuchibue database with elastic distortions. In the experiment, we evaluate the performance of the end-to-end model and the effectiveness of the synthetic data generation method on the test set of the TUAT Kondate database. The results of the experiments show that our end-to-end model achieves higher than the state-of-the-art recognition accuracy on the test set of TUAT Kondate with 96.35% and 98.05% character level recognition accuracies without and with the generated synthetic data, respectively.**

Keywords – Handwritten Japanese Text Recognition, End-to-End Model, CNN, BLSTM, Synthetic Image Generation

I. INTRODUCTION

The offline handwritten Japanese recognition is a still big challenging problem because of the large character set; varieties of characters mixed of thousands of Kanji characters of Chinese origin, two sets of phonetic characters, alphabets, numerals, symbols, etc.; the fact that Kanji radicals are often characters as themselves; diversity of writing styles and multiple-touches between characters. Most of the traditional offline handwritten Japanese/Chinese text recognition methods [1, 2, 3] use some pre-segmentation of text lines before individually recognizing each character and integrating linguistic and geometric context. However, pre-segmentation of text lines is quite costly and the errors due to this process directly affect the performance of the whole system.

In recent years, Deep Neural Networks have been proven to be very powerful models and achieve the state-of-the-art accuracies on many computer vision tasks such as Convolutional Neural Network (CNN) for image recognition [4], Long Short-Term Memory (LSTM) for sequence prediction and labeling tasks [5]. Graves et al. [6] introduced Connectionist Temporal Classification (CTC) for labeling unsegmented sequence data. They also combined Bidirectional LSTM and CTC to build a connectionist system for unconstrained handwriting recognition [7]. Base on Deep Neural Network and CTC, many segmentation-free methods [8, 9, 10] have been proposed and proven to be very powerful for handwriting recognition tasks. R. Messina and J. Louradour [9] combined Multi-Dimensional LSTM (MDLSTM) and CTC to build an end-to-end trainable model for offline handwritten Chinese text recognition. However, the Multi-Dimensional LSTM networks are quite computationally expensive and J. Puigcerver [11] provided multiple evidences that Multi-Dimensional LSTM may not be necessary to achieve good accuracy for handwriting recognition tasks. In this paper, we propose an end-to-end model of DCRN for offline handwritten Japanese text recognition. It consists of 3 components, the convolutional feature extractor, the recurrent layers, and a transcription layer.

Deep Neural Networks, especially end-to-end models typically require a large data for training. However, for many handwriting datasets, especially handwritten Japanese character and text datasets, the number of data is not enough, so that it is necessary to apply data argumentation. Many data argumentation methods for handwriting datasets have been proposed by modifying the original data such as affine transformations [12, 13], nonlinear combinations [13, 14] and Random warp grid distortion [15]. However, such method just modifies the original data, can't gain the real text line image. In this work, we propose a synthetic pattern generation method which synthesize handwritten text line images from sentences in corpora and handwritten character patterns in the Nakayosi and Kuchibue [16] database with elastic distortions.

The rest of this paper is organized as follows: Session II presents the overview of the end-to-end DCRN model. Session III describes the synthetic pattern generation method. Session IV reports our experimental results and analysis. Session V draws conclusions.

II. OVERVIEW OF THE END-TO-END DCRN MODEL

We propose an end-to-end model of Deep Convolutional Recurrent Network (DCRN) for recognizing offline handwritten Japanese text lines. Our end-to-end model consists of 3

978-1-5386-5875-8/18/$31.00 ©2018 IEEE
DOI 10.1109/ICFHR-2018.2018.00022

components, the convolutional feature extractor, the recurrent layers, and a transcription layer, from bottom to top as shown in Fig. 1.

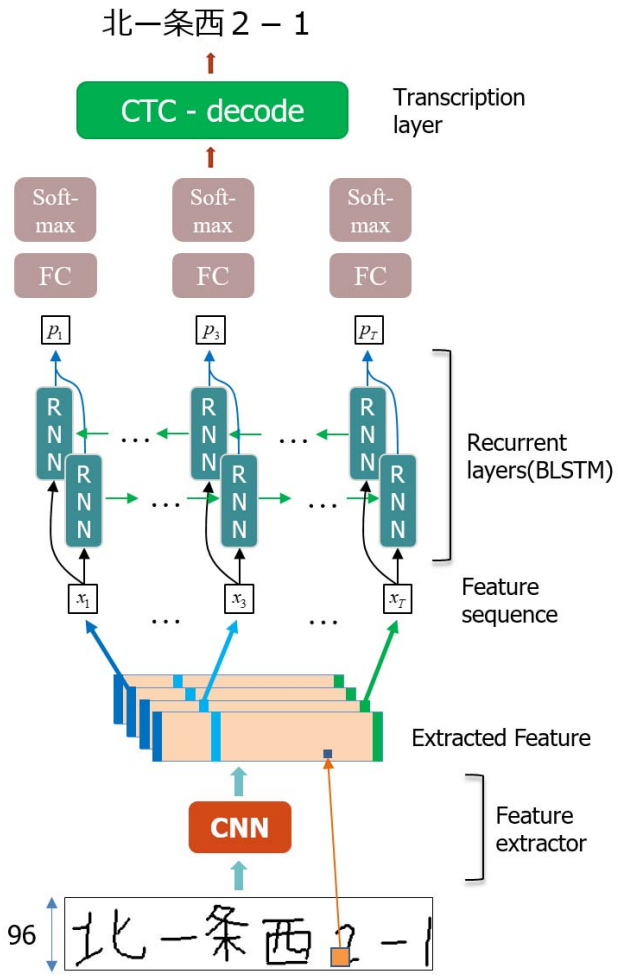


Fig. 1. Network architecture of end-to-end DCRN model. The network consists of three components: 1) Convolutional feature extractor; 2) Recurrent layers; 3) Transcription layer.

A. Preprocessing

Firstly, all of the text line images are scaled to the same height of size before recognized by the end-to-end DCRN model. This is necessary because in our model the feature dimension of feature sequence which extracted by the convolutional feature extractor is constant since deep BLSTM expects a fixed-size feature dimension. After resizing, in order to manage the noisy and complicated background, the text line images are converted into binary images by Otsu thresholding algorithm [17].

B. Convolutional feature extractor

Convolutional neural networks (CNNs) have been proven to be very powerful visual models and achieve the state-of-the-art accuracies on some tasks of computer vision such as image recognition [18] and feature representation [19]. S. Ioffe and C. Szegedy [20] introduced a technique called Batch normalization which normalizes the summed input to a neuron over a mini-batch of training cases by calculating a mean and a variance from the distribution of the summed input to that neuron. This technique is demonstrated to significantly reduce the training time in feed-forward neural networks.

In the End-to-end DCRN model, we employ a CNN network to build a convolutional feature extractor. The CNN network is constructed by taking the convolutional, max-pooling layers from a standard CNN model (fully connected and softmax layers are removed). The Leaky ReLu [21] activation is applied in all convolutional layers. Batch normalization is applied between convolutional layer and Leaky ReLu activation. We apply this convolutional feature extractor to an input image of size $w \times h \times c$ (where c is the color channels of image), resulting in a multi-channel output of dimension $w' \times h' \times k$, where k is the number of feature maps in last convolutional layer, w' and h' depend on the w and h of input images and the amount of pooling layers in the CNN network. Then we pass the w' features of dimension $h' \times k$ to the recurrent layers. Since the height of input images is fixed, the dimension $h' \times k$ of each feature is the same.

C. Recurrent layers

Recurrent neural networks (RNNs) are connectionist models containing a self-connected hidden layer. The benefits of RNNs are allowing information of previous inputs to remain in the network's internal states and the ability to make use of previous context. In the traditional RNNs, however, the vanishing gradient problem was recognized [22]. Long Short-Term Memory (LSTM) [5] is a special kind of the RNN architecture designed to address the vanishing gradient problem which is capable of learning long-term dependencies. The standard LSTM can only use past contextual information in one direction. For many tasks such as handwritten recognition, however, it is useful to have access to future as well as past contextual information in both directions. This can be overcome by using Bidirectional LSTM (BLSTM [5]) that is able to access context in both directions along the input sequence.

In our end-to-end DCRN model, the recurrent layers are built on top of the convolutional feature extractor to predict a label distribution for each frame of the feature sequence extracted from the previous component. The recurrent layers consist of the Deep Bidirectional LSTM layers which take the feature sequence from the convolutional feature extractor as the input. In the last LSTM layer, each time step of feature sequence is followed by a fully connected linear layer which converts the output feature dimension to the size of the total character set (plus 1 for CTC blank character). Finally, a softmax layer is placed at the end to generate the label probability vector at each time step.

D. Transcription layer

At the top of our end-to-end DCRN model, the transcription layer decodes the pre-frame predictions made by the recurrent layers into the final label sequence. Mathematically, decoding is to find the label sequence with the highest probability conditioned on the pre-frame predictions. To obtain the

conditional probability, we employ a CTC [6] layer as the transcription layer.

For decoding, we apply the CTC beam search [23] with 100 for the beam width combined with a linguistic context to obtain the final label sequence with the highest probability conditioned. In this work, we employ the tri-gram probability [24] as the linguistic context. The tri-gram probability $P(C_i|C_{i-2}, C_{i-1})$ is calculated from the corpus. It is reduced to unigram or bi-gram when C_i is the first or second character.

III. Text Line Image Generation

A. Synthetic Data Generations

Since the end-to-end model requires large data for training, we propose a synthetic pattern generation method which synthesizes handwritten text line images from sentences in corpora and handwritten character patterns in the Nakayosi and Kuchibue [16] database with local elastic distortion and global elastic distortion model.

We generate the synthetic handwritten text line dataset by taking the following 6 steps:

1. Get a sentence from the listed sentences of corpus.
2. Randomly choose a writer from the listed writers of the handwritten character pattern database.
3. For each character of the sentence in the step 1, a handwritten image of this character is randomly chosen from the writer selected in the step 2.
4. Apply a local elastic distortion to each handwritten pattern in the step 3.
5. Synthesize a handwritten text line image from the sentence selected in the step 1 and elastically distorted handwritten character images in the step 4 with random spacing between each character image.
6. Apply a global elastic distortion to the generated synthetic text line image.

B. Local Elastic Distortion

The local elastic distortion model performs an affine transformation on each handwritten character image before concatenating them into a synthetic text line image. In the local elastic distortion model, we employ shearing, rotation, scaling, translation transformations.

Shear is a transformation that slants the shape of an object. There are two shear transformations include X-shear and Y-shear (vertical and horizontal shear). They are calculated by eq. (1) and eq. (2).

Translation is a transformation that moves an object to a different position without rotation. Scaling is a transformation that changes the size of an object. The translation and scaling transformations are shown in eq. (3) and eq. (4).

Rotation is a transformation that rotates the object at particular angle α from its origin. The rotation transformation is shown in eq. (5).

$$\begin{cases} x' = x + y\tan\alpha \\ y' = y \end{cases} \quad (1) \qquad \begin{cases} x' = x \\ y' = y + x\tan\alpha \end{cases} \quad (2)$$

$$\begin{cases} x' = x + t_x \\ y' = y + t_y \end{cases} \quad (3)$$

$$\begin{cases} x' = kx \\ y' = ky \end{cases} \quad (4)$$

$$\begin{cases} x' = x\cos\alpha - y\sin\alpha \\ y' = x\sin\alpha + y\cos\alpha \end{cases} \quad (5)$$

Here, (x', y') is the new coordinate of a point (x, y) transformed by any transformation model, α is the angle of the shear and rotation transformations, k is the scaling factor of the scaling transformation, the pair (t_x, t_y) is the shift vector of the translation transformation. The parameters of the local elastic distortion model is presented by $[(p_{SH}, \alpha), (p_T, t_x, t_y), (p_{SC}, k), (p_R, \alpha)]$, where p_{SH}, p_T, p_{SC} and p_R are the probabilities of applying the shearing, translation, scaling and rotation transformations, respectively, α is from -8^o to 8^o with a step of 0.1, t_x and t_y are from 3 to 5 pixels with a step of 1, and k is from 0.8 to 1.2 with a step of 0.01.

Fig. 2 show examples of the local elastic distortion model with $\alpha = 8°$, k=0.9 and $t_x = t_y = 3$.

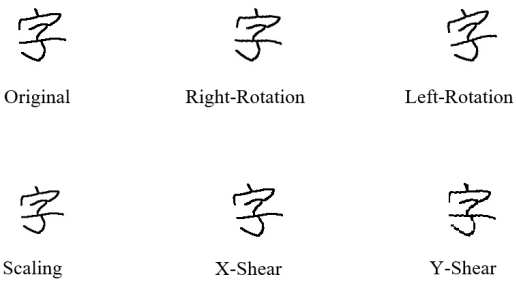
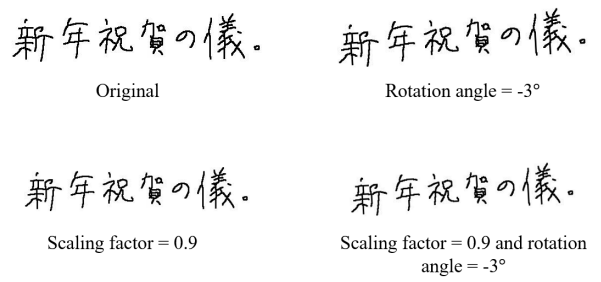


Fig. 2. Examples of local elastic distortion by shearing, rotation and scaling transformations.

C. Global Elastic Distortion

Global elastic distortion model performs affine transformation on a whole text line image generated by concatenating isolated handwritten character images. In the global elastic distortion, we employ the rotation and scaling transformations. The rotation and scaling transformations is similar to the local elastic distortion. The parameters of the global elastic distortion are presented by $[(p_{SC}, k), (p_R, \alpha)]$, where p_{SC} and p_R are the probabilities of applying the scaling and rotation transformations, respectively, k is the scaling factor and from 0.8 to 1.2 with a step of 0.01, and α is the angle of the global rotation transformation and from -5^o to 5^o with a step of 0.1.

Fig. 3 show examples of the global elastic distortion by the scaling and rotation transformations.



Original | Rotation angle = -3° | Scaling factor = 0.9 | Scaling factor = 0.9 and rotation angle = -3°

Fig. 3. Examples of global elastic distortion by scaling and rotation transformations.

D. Synthetic Handwritten Text Line Dataset

We employ the sentences of Nikkei newspaper corpus and Asahi newspaper corpus and the handwritten character database, Nakayosi and Kuchibue [16] to generate the Synthetic Handwritten Text Line Dataset (SHTL). Nikkei corpus consists of about 1.1 million sentences collected from Nikkei News and Asahi corpus consists of about 1.14 million sentences collected Asahi News. We randomly choose 30,000 sentences which contain less than 30 characters from each corpus. Since it make sure that the end-to-end model can be trainable by SHTL. SHTL consists about 60,000 of synthetic handwritten text line images. Fig. 4 show samples of generated synthetic text line image in the SHTL dataset.

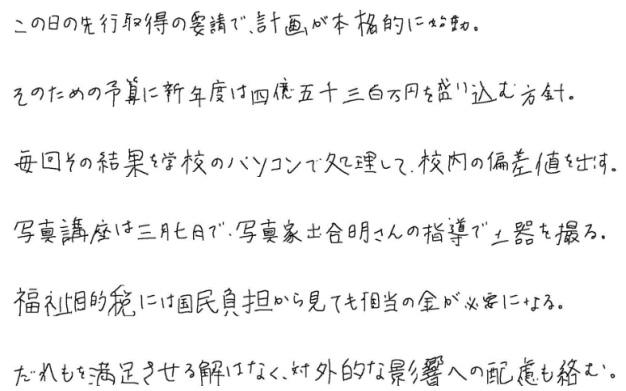


Fig. 4. Samples of generated synthetic data.

IV. Experiments

To evaluate the performance of the proposed end-to-end DCRN model and the effectiveness of the synthetic data generation method, we conducted experiments on standard benchmarks for offline handwritten Japanese text recognition. The information of handwritten Japanese text databases is given in Sec. A, the implementation details are described in Sec B, the results of the experiments are presented in Sec. C and the correctly recognized and misrecognized samples are shown in Sec. D.

A. Offline Handwritten Japanese Text Databases

TUAT Kondate database [25] is a database of online handwritten patterns mixed of text, figures, tables, maps, diagrams and so on. It was turned to offline patterns by thickening strokes by constant width. The Japanese portion of Kondate was collected from 100 Japanese writers and the horizontal Japanese text lines stored in Kondate were used in our experiments. 13,685 horizontal Japanese text lines were split into three parts: first one consisting of 11,487 text line images collected from 84 Japanese writers were used as the training set, the second one consisting of 800 text line images collected from 6 Japanese writers were used as the validation set, the last one consisting of 1,398 text line images collected from 10 Japanese writers were used as the test set. They are summarized in Table I.

TABLE I. The detail of information of Kondate database.

	Kondate		
	Train set	**Valid set**	**Test set**
Number of writers	84	6	10
Number of samples	11,487	800	1,398

B. Implementation Details

The architecture of our CNN model used in the convolutional feature extractor is shown in Table II. It consists of 8 convolutional layers. Batch normalization is applied after the 2nd, 4th, 6th and 8th convolutional layers followed by Max-Pooling layers. The Leaky ReLu [21] activation function is applied in all convolutional layers.

TABLE II. Network configuration of our CNN model. 'maps', 'k', 's' and 'p' denote the number of kernels, kernel size, stride and padding size of convolutional layers respectively.

Type	**Configurations**
Input	96×w image
Conv1 - LReLu	#maps:32, k:3×3, s:1, p:1
Conv2 - Batch Norm - LReLu	#maps:32, k:3×3, s:1, p:1
MaxPooling1	#window:2×2, s:2
Conv3 - LReLu	#maps:64, k:3×3, s:1, p:1
Conv4 - Batch Norm - LReLu	#maps:64, k:3×3, s:1, p:1
MaxPooling2	#window:2×2, s:2
Conv5 - LReLu	#maps:128, k:3×3, s:1, p:1
Conv6 - Batch Norm - LReLu	#maps:128, k:3×3, s:1, p:1
MaxPooling3	#window:2×2, s:2
Conv7 - LReLu	#maps:256, k:3×3, s:1, p:1
Conv8 - Batch Norm - LReLu	#maps:256, k:3×3, s:1, p:1
MaxPooling4	#window:2×2, s:2

At the recurrent layers, we employ Deep BLSTM network with 128 hidden nodes of three layers. In order to prevent overfitting when training the model, the dropout (dropout rate=0.8) is also applied in each layer of Deep BLSTM. A fully connected layer and a softmax layer with the node size equal to the character set size (n=3347) are applied after each time step of Deep BLSTM network.

The end-to-end DCRN model is trained using stochastic gradient descent with the learning rate of 0.001 and the momentum of 0.9. The training process stops when the

recognition accuracy of validation set do not gain after 10 epochs. The end-to-end DCRN model is trained by two datasets; the first is the training set of TUAT Kondate and the second is the training set of TUAT Kondate combining the SHTL Dataset. We call the former End-to-End and the latter End-to-End_SHTL. We use the validation set and test set of TUAT Kondate to validate and test the performance of End-to-End and End-to-End_SHTL.

C. Results of Experiments

In order to evaluate the performance of the end-to-end DCRN model and the effectiveness of the synthetic data generation method, we employ the terms of Label Error Rate (LER) [6] and Sequence Error Rate (SER) [6] that are defined as follows:

$$LER(h,S') = \frac{1}{Z}\sum_{(x,z)\in S'} ED(h(x),z)$$

$$SER(h,S') = \frac{100}{|S'|}\sum_{(x,z)\in S'} \begin{cases} 0 & if \quad h(x)=z \\ 1 & otherwise \end{cases}$$

where Z is the total number of target labels in S' and ED(p, q) is the edit distance between two sequences p and q.

The first experiment evaluated the performance of the end-to-end DCRN model and the effectiveness of the synthetic data generation method without using the linguistic context. Table III shows the recognition rate on the validation and test sets. End-to-End obtained LER of 3.65% and SER of 17.24% on the test set. The results imply that the end-to-end DCRN model substantially outperforms the state-of-the-art recognition accuracy in the previous model DCRN-s [8]. End-to End_SHTL achieved LER of 1.95% and SER of 14.02%. These results show that the recognition accuracy is further improved when we use the SHTL dataset to train the end-to-end DCRN model.

TABLE III. Label Error Rate (LER) and Sequence Error Rate (SER) on Kondate.

Model	LER		SER	
	Valid set	Test set	Valid set	Test set
DCRN-f&s [8]	11.74%	6.95%	39.33%	28.04%
DCRN-s [8]	11.01%	6.44%	37.38%	25.89%
End-to-End	5.22%	3.65%	24.47%	17.24%
End-to-End_SHTL	3.62%	1.95%	21.87%	14.02%

Secondly, we evaluated the performance of the end-to-end DCRN model and the effectiveness of the synthetic data generation method with the linguistic context [24]. Table IV shows the recognition rate of the end-to-end DCRN model on the test set when combined with the linguistic context. It is compared with the previous segmentation based method [1] and the previous models DCRN-s and DCRN-f&s [8] with the linguistic context. The results show that the end-to-end DCRN model is superior to the segmentation based method and its recognition accuracy is further improved when the linguistic context is integrated.

TABLE IV. Label Error Rate (LER) and Sequence Error Rate (SER) on test set of Kondate when combined with the linguistic context.

Model	Test set	
	LER	SER
Segmentation based [1]	11.2%	48.53%
DCRN-f&s [8]	6.68%	26.97%
DCRN-s [8]	6.10%	24.39%
End-to-End	3.52%	16.67%
End-to-End_SHTL	1.87%	13.81%

For the convergence of training, our end-to-end DCRN model achieves convergence after 39 epochs for End-to-End and 31 epochs for End-to-End_SHTL compared with 110 epochs for CNN and about 50 epochs for BLSTM&CTC of DCRN-s in previous work [8]. Fig. 5 shows the label error rate achieved after each epoch when training the End-to-End, End-to-End_SHTL, DRCN-s [8] and DRCN-f&s [8].

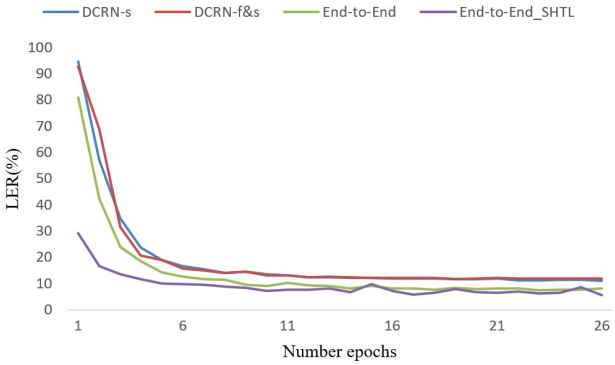
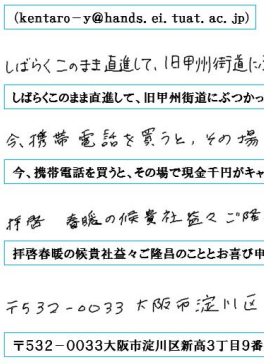
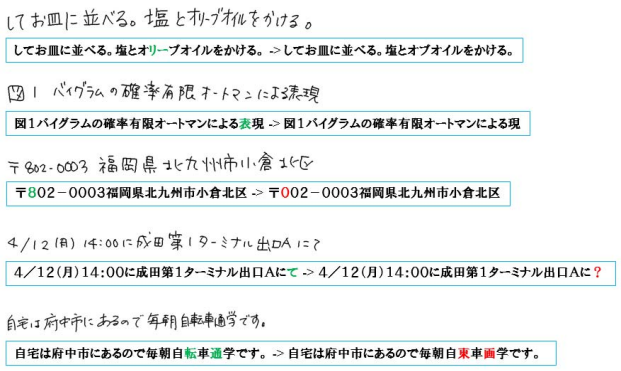


Fig. 5. LER on the validation set after each epoch when training DCRN-s [8], DCRN-f&s [8], End-to-End and End-to-End_SHTL.

D. Correctly recognized and misrecognized samples



a). Correctly recognized samples.



a). Misrecognized samples.

Fig. 6. Correctly recognized and misrecognized samples by End-to-End_SHTL.

Fig. 6 shows some correctly recognized and misrecognized samples by End-to-End_SHTL whose SER is about 14.02%. For each misrecognized sample, the upper image is an input handwritten text line image and the text bounded by the lower blue rectangular shows the ground-truth followed by "->" and the recognition resulted. There are a total of 196 misrecognized samples among 1398 samples in the test set. Most of them are missing some characters in the ground-truth.

V. Conclusion

In this paper, we presented the end-to-end DCRN model for recognizing offline handwritten Japanese text lines. We proposed the method of synthesizing handwritten character images combining local and global elastic distortion models for generating handwritten text line images. Following the experiments on the test set of TUAT Kondate, the end-to-end DCRN model archived the 96.35% and 98.05% character level recognition accuracy without and with the SHTL dataset, respectively. The following conclusions are drawn: 1) the end-to-end DCRN model substantially outperforms the previous model DCRN-s and the traditional segmentation-based method [1, 8]; 2) the recognition accuracy is improved by using the SHTL dataset to training the end-to-end DCRN model; 3) the recognition rate is further improved when combined with the linguistic context.

References

[1] K. C. Nguyen and M. Nakagawa, "Text-Line Character Segmentation for Offline Recognition of Handwritten Japanese Text," IEICE Technical Report, BioX2015-50, PRMU2015-173, 2016.

[2] 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.

[3] S. N. Srihari, X. Yang, and G. R. Ball, "Offline Chinese handwriting recognition: an assessment of current technology," Frontiers of Computer Science in China, vol. 1, no. 2, pp. 137–155, 2007.

[4] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv preprint arXiv:1409.1556, 2014.

[5] S. Hochreiter, J. Schmidhuber, "Long Short-term Memory," Neural Computation, vol. 9, no. 8, pp. 1735-1780, 1997.

[6] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," In Proceedings of the 23rd international conference on Machine learning, pp. 369-376. ACM, 2006.

[7] A. Graves, M. Liwicki, S. Fern´andez, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 5, pp. 855-868, 2009.

[8] N. T. Ly, C. T. Nguyen, K. C. Nguyen and M. Nakagawa, "Deep Convolutional Recurrent Network for Segmentation-free Offline Handwritten Japanese Text Recognition," Proc. MOCR2017, 2017.

[9] R. Messina and J. Louradour, "Segmentation-free handwritten chinese text recognition with lstm-rnn," 13th International Conference on Document Analysis and Recognition (ICDAR), pp. 171-175, 2015.

[10] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," CoRR, vol. abs/1507.05717, 2015.

[11] J. Puigcerver, "Are Multidimensional Recurrent Layers Really Necessary for Handwritten Text Recognition?," in: Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, 2017, pp. 67–72.

[12] A. Poznanski and L. Wolf, "Cnn-n-gram for handwriting word recognition," *Proc. CVPR*, 2016.

[13] B. Chen, B. Zhu, M. Nakagawa, "Training of an on-line handwritten japanese character recognizer by artificial patterns," Pattern Recognition Letters, vol. 35, no. 178–185, 2014.

[14] K. C. Leung, C. H. Leung, "Recognition of handwritten Chinese characters by combining regularization Fisher's discriminant and distorted sample generation," Proc. 10th Int. Conf. Document Analysis and Recognition, pp. 1026-1030, 2009.

[15] Curtis Wigington, Seth Stewart, Brian Davis, Bill Barrett, Brian Price, Scott Cohen, "Data Augmentation for Recognition of Handwritten Words and Lines Using a CNN-LSTM Network," Proc. ICDAR2017, 2017.

[16] M. Nakagawa, K. Matsumoto, "Collection of on-line handwritten Japanese character pattern databases and their analysis," Int. J. Document Anal. Recognit., vol. 7, no. 1, pp. 69-81, 2004.

[17] N. Otsu, "A threshold selection method from gray level histograms," IEEE Trans. systems. Man. and Cybernetics, 9:62–66, 1979.

[18] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," arXiv preprint arXiv:1409.1556, 2014.

[19] Ben Athiwaratkun and Keegan Kang, "Feature Representation In Convolutional Neural Networks," arXiv:1507.02313v1 [cs.CV] 8 Jul 2015.

[20] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," In ICML, 2015.

[21] AL Maas, AY Hannun, AY Ng, "Rectifier Nonlinearities Improve Neural Network Acoustic Models," Proc. ICML, 2013.

[22] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 6(02): 107–116, 1998.

[23] Alex Graves, Navdeep Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," In Proc. ICML, 2016.

[24] B. Zhu, X.-D. Zhou, C.-L. Liu, M. Nakagawa, "A Robust Model for Online Handwritten Japanese Text recognition," IJDAR, 13(2), pp. 121-131, 2010.

[25] T. Matsushita and M. Nakagawa, "A Database of On-line Handwritten Mixed Objects named "Kondate"," 14th International Conference on Frontiers in Handwriting Recognition (ICFHR), pp. 369-374 , 2014.