Recognition Based Segmentation of Handwritten Alphanumeric Characters Entry on Tablet PC

Recognition Based Segmentation of Handwritten Alphanumeric Characters Entry on Tablet PC

Myat Thiri Wai[a)], Thi Thi Zin[b)], Mitsuhiro YOKOTA[b)], Khin Than Mya[c)]

Abstract

Portable tablet PC are very useful in relevant industry of this age because tablets are elegant in appearance and convenient to use. Important things are noted on tablet by handwriting easily in respective industry. Recognition of handwritten characters automatically on tablet like human's brain is also necessary to be more convenient. To split each character of different handwritten styles is very difficult and it is the main challenging of handwritten character recognition. The previous handwritten character segmentation approaches are still continuing in different problems because of different handwritten styles. The combination of sliding windows, region of interest (ROI) box and convolutional neural network (CNN) are used to execute recognition based segmentation (implicit) of handwritten characters. This system is intended to perform both segmentation and recognition of tablet based application input handwritten characters. Handwritten data are collected from 24 members of our laboratory using three tablets PC models to perform the experiments.

Keywords: Handwritten character segmentation, Recognition, Sliding window, Region of interest (ROI), Convolutional neural network (CNN)

1. INTRODUCTION

Nowadays, handwritten character recognition is challenging in Optical Character Recognition (OCR) environment. OCR is a process of recognition of machine printed or handwritten characters from images. Machine printed character recognition has been developed whereas handwritten character recognition is difficult and still continuing with different problems in OCR environment. It is very difficult to segment and recognize different handwritten format of different people. Most of handwritten recognition systems have still the problems in preprocessing to recognize correctly. To segment and recognize tablet input handwritten characters is more difficult because there are many noisy and different handwriting styles. Researchers in [1, 2] accepted that segmentation is the most important part in recognition of handwritten character because bad segmentation errors lead to the recognition errors of the system. Segmentation of unconstraint handwritten characters of different people tend to unexpected difficult challenging. The method used to segment the handwritten character is the important role in this system.

One of the main objectives of this system is to introduce new handwritten characters segmentation. In this system, we intended to segment handwritten words and two digits number into each character and recognize them to test what characters are write on tablet. This system is proposed to use in touch-screen tablet based application for kindergarten education like the application in [3].

2. SOME RELATED WORKS

Various methods of segmentation in the literature can be categorized into three, explicit, implicit, and holistic. Pure segmentation (explicit) can be found in [4-5], segmentation free (holistic) recognition is in [6], and the segmentation of [7] is recognition based segmentation (implicit). Our approach is recognition based segmentation by combination of convolutional neural network, sliding window and finding region of interest.

In the region-based segmentation [4], the segmentation path is determined to split the only two touched characters into two sub images by defining the three zones, left, right, and middle. Locating the segmentation points based on the analysis of the character's geometric features is shown in [5] but it

a) Master DDP Student, Energy and Electronics Course, Graduate School of Engineering, University of Miyazaki (Master Student, University of Computer Studies, Yangon (UCSY), Myanmar)
b) Professor, Dept. of Electrical and Systems Engineering, Faculty of Engineering, University of Miyazaki
c) Professor, Dept. of Hardware, Faculty of Computer Systems and Technologies, University of Computer Studies, Yangon (UCSY), Myanmar

can be done after thinning the word image to get the stroke width of a single pixel. An offline handwritten word recognition system that combines NN and HMM was presented in [6] by dividing the handwritten word images into vertical slices.

3. PROPOSED METHOD

An overview of this proposed system is shown in Fig. 1. Handwritten input image taken using touch-screen tablet is loaded as input image to this system. Firstly, the input image is pre-processed to get the slant corrected binary image. Then, split each character clearly from pre-processed image in recognition based character segmentation stage. This system classify each character correctly, after segmenting the pre-processed tablet input image.

3.1 Preprocessing

There are two processes in pre-processing stage: binarization, and slant correction. In binarization, the input image is converted into a binary image by using Otsu's method from Eq. 1.

$$\sigma_w^2(t)=W_b(t)\sigma_b^2(t)+W_f(t)\sigma_f^2(t) \qquad (1)$$

where t means the global thresholding, σ_w^2 is the within-class variance, W_b and σ_b^2 are the background weight and the background variance respectively, W_f and σ_f^2 denote the foreground weight and the foreground variance respectively.

The slant correction is performed by employing the horizontal shear transformation of Eq. 2 on the whole word image.

$$\begin{pmatrix}x'\\y'\end{pmatrix}=\begin{pmatrix}x+sy\\y\end{pmatrix}=\begin{pmatrix}1&s\\0&1\end{pmatrix}\begin{pmatrix}x\\y\end{pmatrix} \qquad (2)$$

where, s denotes shearing factor which is specified by computing the slant of each character, x and y are pixel coordinates of slanted image and, x' and y' are coordinates of slant corrected images.

3.2 Recognition Based Character Segmentation

The combination of sliding windows, ROI box and CNN are used in segmentation of handwritten characters [8]. There are two main steps in segmentation: outer ROI segmentation and sliding window segmentation. Pre-processed image is segmented by applying outer ROI segmentation as shown in Fig. 2. In outer ROI segmentation, ROI boxes are defined in pre-processed image and each outer ROI is determined a valid character or not by using trained classifier. The valid character outer ROI is marked as a qualified Outer ROI. Unqualified outer ROI is inserted to sliding window segmentation process as shown in Fig. 3.

1) Outer ROI is defined by applying the bounding rectangle of all connected black pixels by labelling based on the binary image. In characters, '*i*' and '*j*', there are two ROIs for small dot and main part. Those two ROIs are needed to combine to get the whole character, '*i*' or '*j*' as shown in Fig. 4. If a very small ROI is detected, a narrow width ROI will be searched from its neighboring ROIs. If a small dot ROI is adjacent with a narrow width ROI, that two ROIs are combined to get the whole character. If there is any narrow width ROI surrounding of a very small ROI, that small dot noise will be removed.

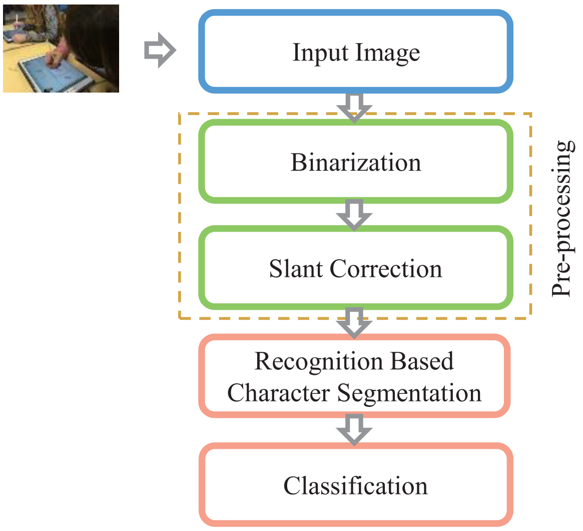

Fig. 1. Overview of the proposed system

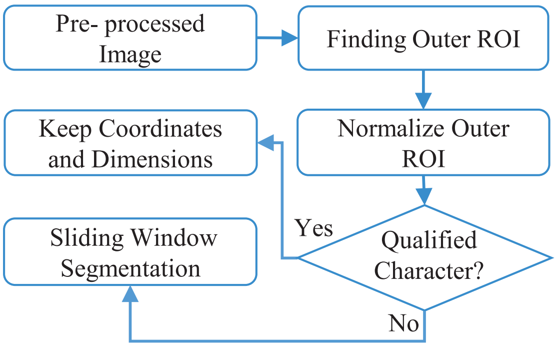

Fig. 2. Overview of outer ROI segmentation

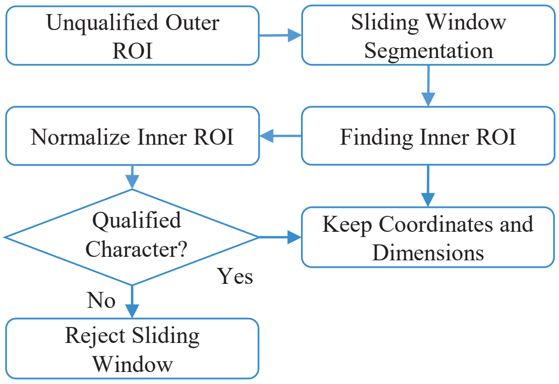

Fig. 3. Overview of sliding window segmentation

2) To normalize the ROI, the outer ROI is padded based on row and column of that ROI to be a square image. Then, the padding image is resized into 28x28 size to insert the classifier. The outer ROI processing and normalization is shown in Fig. 5.

3) The image we obtained from normalization step is fixed to the trained classifier, convolutional neural network (CNN) to determine the character based on the accuracy threshold and probability. If the highest output confidence is greater than the confidence threshold parameter, Outer ROI's coordinates and dimensions are keep as a candidate. If the highest confidence is less than the threshold parameter, the sliding window segmentation is applied to that outer ROI. Top five output probabilities of sample qualified outer ROI is shown in Fig. 6.

4) Initial window size is set to the height of outer ROI and 20% of the width of outer ROI. Inner ROI is specified on the window and normalization is perform on that inner ROI to insert to the classifier as shown in Fig. 7. Moving distance of sliding window is 5% of the window width to set the overlap ratio to 1/20. Before exceed the bounding of outer ROI, stop the sliding and increase window width by adding 50% of the initial width to repeat the process until the window width is reached to the ROI width. The processing of sliding window segmentation is shown in Fig. 8.

Inner ROI specification is finding the bounding rectangle by applying uppermost, lowermost, leftmost and rightmost of all foreground pixels found within the sliding window. The cropped image by using inner ROI is padded based on row and column of that inner ROI. Then, the padded image is resized into 28x28 and inserted to the classifier.

The normalized inner ROI is classified by using trained CNN classifier. If the highest output confidence is greater than the confidence threshold parameter, Inner ROI's coordinates and dimensions are keep as a candidate for representing a valid letter. If overlap of one candidate over the other is smaller than overlap threshold parameter, both candidates are qualified.

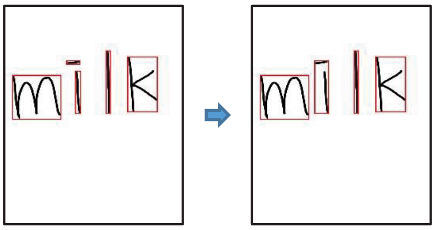

Fig. 4. Region combining for character '*i*'

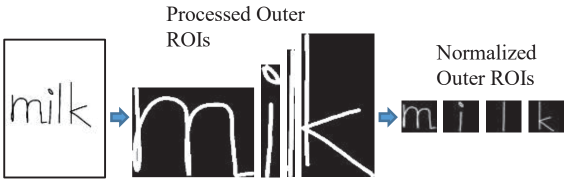

Fig. 5. Processing and normalization of outer ROI

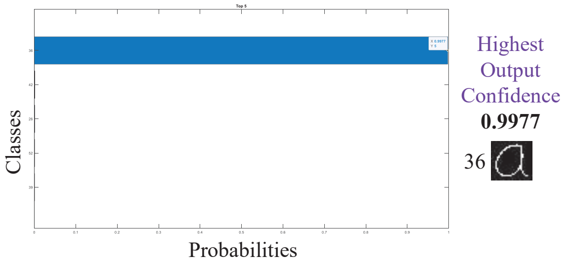

Fig. 6. Top five output probabilities outer ROI

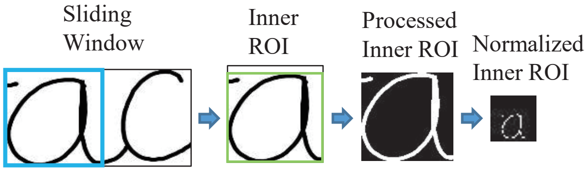

Fig. 7. Finding inner ROI and normalization

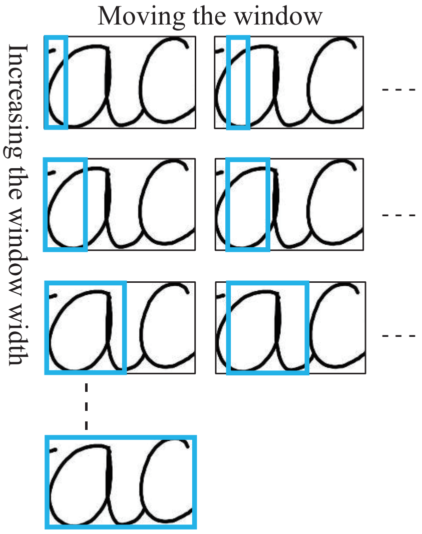

Fig. 8. Sliding window processing

3.3 Noise Removal

There are some noises in segmentation results, because small pieces of neighboring characters are containing in segmented region of character. So, these unnecessary noises are removed to make the classifiable character for the best recognition. The area open operation of morphological operations is used to remove these noises by detecting the area of unnecessary parts as shown in Fig. 9.

3.4 Classification

To recognize the segmented classifiable character, this system uses convolutional neural network (CNN) that is widely accepted classifier in image classification. Firstly, training and testing data are prepared and labeled using EMNIST database [9]. After that, the network is created and trained using prepared database. The layers of the CNN implemented in MATLAB, MatConvNet [10] are listed in Table 1 and the architecture of the CNN is shown in Fig. 10.

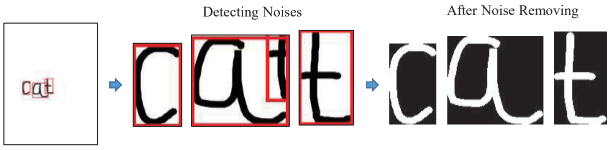

Fig. 9. Detecting and removing the noises

Table 1. Layer structure of CNN

Layer No.	Layers	Input Size	Filter Size	Stride	Output
Layer 1	Conv	28x28	5x5	1	32@ 24x24
Layer 2	Max Pooling	24x24	2x2	2	32@ 12x12
Layer 3	Conv	12x12	5x5	1	32@ 8x8
Layer 4	Max Pooling	8x8	2x2	2	32@ 4x4
Layer 5	Conv	4x4	4x4	1	64@ 1x1
Layer 6	FC	1x1	-	-	256@ 1x1
Layer 7	FC	1x1	-	-	62 classes

Conv = Convolution
FC = Fully-connected

4. EXPERIMENTAL RESULTS

In this system, handwriting of different people are collected on tablet to build our own dataset. There may be unexpected ligatures problems and noises in character images written on tablet PC. This system we intended is not only to over the challenging of unconstraint handwritten character segmentation but also to recognize each segmented character correctly. Our approach is recognition based segmentation by combination of CNN, sliding window segmentation and finding ROI. The performance accuracy of segmentation and recognition are described in Table 2. The system correctly segmented 1445 images from the total images of 1603 and 5632 character images are correctly recognized from 6734 character images. Final segmentation result and normalization is shown in Fig. 11. Wrong segmentation of character '*w*' that is split into two ROIs, '*v*', is shown in Fig. 12 and unconnected character '*5*' is segmented correctly as shown in Fig. 13.

Our own dataset is constructed by using three tablet PC models. Table 3 describes the detail of tablets used for this system. The dataset consists of 7 categories as described in Table 4.

Table 2. Performance of segmentation and recognition

	Total Images	Correct Image	Accuracy (%)
Segmentation	1,603	1,445	90.19
Recognition	6,734	5632	83.635

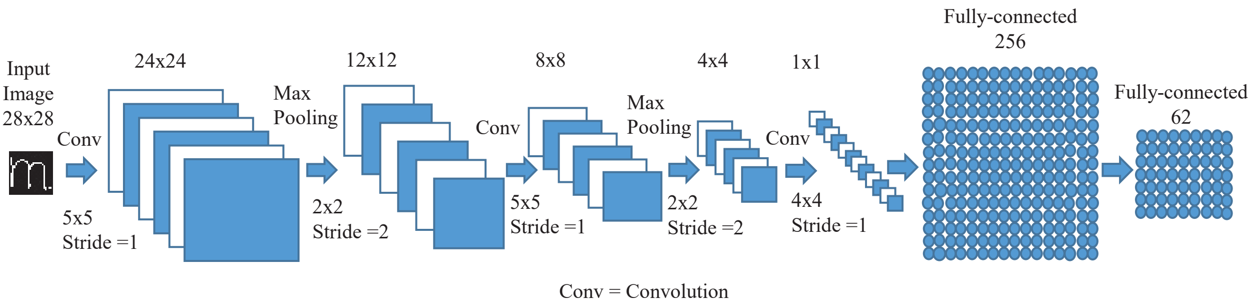

Fig. 10. Architecture of convolutional neural network (CNN)

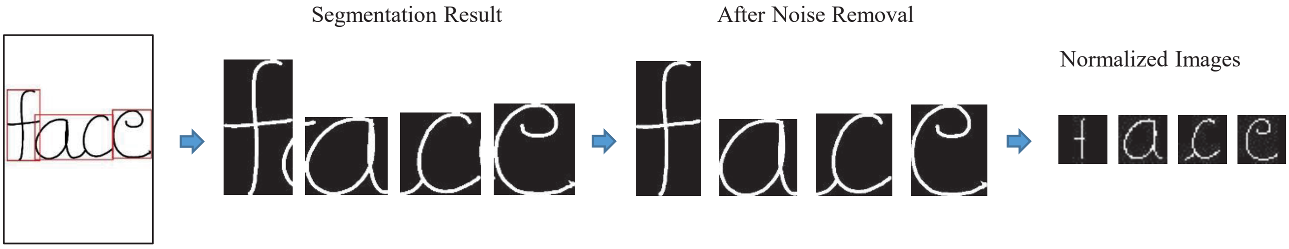

Fig. 11. Final segmentation result and normalization

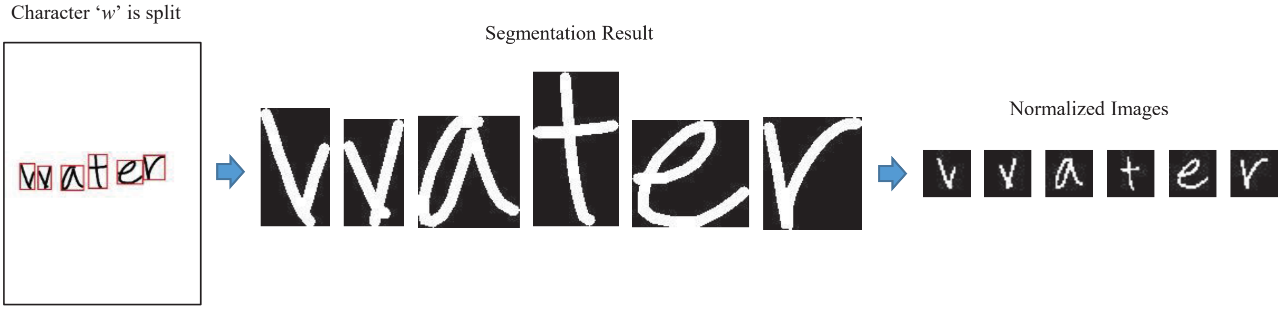

Fig. 12. Wrong segmentation for split character '*w*'

Table 3. Details of tablets

Model	**ASUS ZENPAD 10 SPECS**	**HUAWEI MediaPad T310**	**Gecoo Tablet A1**
OS	Android 7.0	Android 7.0	Android 5.1.1
Storage	16GB	16GB	8 GB
Memory	2GB	2GB	2GB
Display	10.1 inches	9.6 inches	8.0 inches
Resolution	1280 x 800	1280 x800	19201200
Image Type	<.jpg>	<.jpg>	<.jpg>

5. CONCLUSION

If characters are not touched to each other in the input image, this system can exactly segment each character from that image. Our experiments accomplish the outstanding results of segmentation although this system still need to achieve the goal. In this study, we focused on segmentation process for better segmentation accuracy of handwritten character recognition. In the future, we will execute more experiments on more cursive handwritten data and will try to recognize the whole word on the application.

Table 4. Categorization of our dataset

Categories	Types	Count
Animals	ant	20
	bird	33
	cat	33
	cow	33
	dog	39
	fish	33
	fox	30
	zebra	30
Body Parts	ear	33
	eye	30
	face	34
	head	36
	mouth	29
Education	book	21
	classroom	32
	homework	33
	school	33
	student	33
	teacher	33
Fruits	apple	42
	avocado	35
	banana	39
	blueberries	21
	cherries	19
	coconut	25
	durian	27
	grapes	21
	guava	28
	lemon	34
	mango	35
	orange	33
Drinks	coffee	38
	juice	33
	milk	33
	tea	36
	water	32
Numbers	18	27
	24	27
	35	33
	60	26
	97	26
Others	bell	21
	birthday	21
	blue	20
	box	30
	candy	22
	girl	24
	good	23
	hill	21
	quarter	30
	queen	33
	vegetable	30
	yard	30
	zoo	30
Total	**54**	**1603**

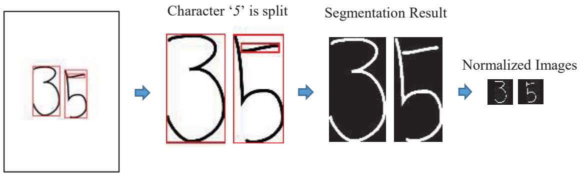

Fig. 13. Correct segmentation for split character ‘5’

ACKNOWLEDGEMENTS

We would like to express our appreciation to Mr. Ogino Tsuhunobu, CEO of KJS Company LTD. for collaboration in this joint research.

REFERENCES

1) H. A. Al Hamad, R. A. Zitar, “Development of an Efficient Neural-based Segmentation Technique for Arabic Handwriting Recognition,Pattern Recognition”, Vol. 43, No, 8, pp. 2773-2798, August 2010.

2) T. Saba, A. Rehman, G. Sulong, “Cursive Script Segmentation with Neural Confidence”, International Journal of Innovative Computing, Information and Control, Vol. 7, No 8, pp. 4955-4964, 2011.

3) L. C. Lin, et al., “Using iPad Apps to Enhance Literacy Skills of English Language Learners with Special Needs”, International Journal of Languages and Literatures, USA, Vol. 2, No. 1, pp. 21-29, March, 2014.

4) F. Kurniawan, et al., “Region-based Touched Character Segmentation in Handwritten Words”, International Journal of Innovative Computing Information and control, Vol. 7, No. 6, pp. 3107-3120, June 2011.

5) A. Choudhary, et al., “A New Character Segmentation Approach for Off-Line Cursive Handwritten Words”, International Conference on Information Technology and Quantitative Management, India, 2013.

6) Y. H. Tay, et al., “An Offline Cursive Handwritten Word Recognition System”, IEEE Region 10 International Conference Electrical and Electronic Technology (TENCON), vol. 2, pp. 519–524, 2001.

7) A. Vetzislav, “Using Critical Points in Contours for Segmentation of Touching Characters”, Proc. of the 5th International Conference on Computer Systems and Technologies, Rousse, Bulgaria, 2004.

8) Myat Thiri Wai, Thi Thi Zin, Mitsuhiro Yokota, Khin Than Mya, “Handwritten Character Segmentation in Tablet Based Application”, (Submitted to the 8th IEEE Global Conference on Consumer Electronics, Osaka, Japan, 2019).

9) G. Cohen, S. Afshar, J. Tapson, A. V. Schaik, “EMNIST: Extending MNIST to handwritten letters”, International Joint Conference on Neural Networks (IJCNN), pp. 2921-2926, 2017.

10) A. Vedaldi, K. Lenc, A. Gupta, “MatConvNet Convolutional Neural Networks for MATLAB”, 23rd ACM International Conderence on Multimedia, Brisbane, Australia, October 26-30, 2015.