A Structural method on Grapheme Segmentation of Hangul Characters for OCR

Hwi Hwa Jung and Jin-Young Ha
NHN Corp., Kangwon National University
hwihwa.jung@nhncorp.com, jyha@kangwon.ac.kr

Abstract

In this paper, we propose Hangul grapheme segmentation method by structural approach, which is developed on the machine printed characters with the widely known fonts such as Myunjo, Gulim and Gothic and applied to much more deformed fonts. The process is composed of two steps. One is a structural grapheme segmentation to the characters classified into 20 types, more a reasonable classification than 6 types in that the algorithm of the grapheme segmentation can be simpler and more effective by the intensified common features of 20 types. Furthermore, it is quite easy for 20 type classified characters to be postprocessed using the connected components separated by the boundary information. With the proposed method, we got 99% correct segmentation rate with very high execution speed

1. Introduction

While the research of Optical Character Recognition(OCR) has been continued over the decades, the off-line Hangul (Korean character) OCR hasn’t has such successful results as to satisfy the general OCR users because it has various fonts, sizes and two-dimensional structures compared with the one-dimensional structure of other languages(ex. English, French, etc.). Fortunately, Hangul can be classified in view of the well-defined set of construction rules based on 24 characters, 14 consonants and 10 vowels. We propose to classify the characters as 20 types instead of extant 6 types in order to intensify the common property per type and reduce the failure rate to segment the grapheme correctly. Moreover, it helps to propose the novice optimal algorithm of the grapheme segmentation and the post process we propose. In the result, the proposed method shows a high performance in terms of segmentation accuracy and speed.

2. Related Work

Hangul recognition has largely two different conventional methods of character and grapheme recognition according to the recognition unit.

The character recognition[1] method is simple and less time consuming compared to the grapheme recognition method. But it is hard to find features, due to a large number of target classes and to get a firm performance under the font variation.

The grapheme recognition method[2] separates grapheme in a character and uses it to recognize, so it reduces the number of target classes and maintains a strong performance under font variation. But its weak point is noise and is dependant to the initial type classification method. With this reason, it hasn’t been used practically yet.

3. Proposed grapheme segmentation method of printed Hangul characters

3.1. Precondition

Type classification must be done before grapheme segmentation and the extant studies have used the 6-type classification. In this paper, we classify characters more minutely to 20 types so that the common feature of each type is more obvious and grapheme segmentation can be done more or less easily and successfully. We will restrict our research to grapheme segmentation after completing 20-type classification, so

978-0-7695-3546-3/08 $25.00 © 2008 IEEE
DOI 10.1109/FGCNS.2008.21

we will not discuss about 20-type classification any more.

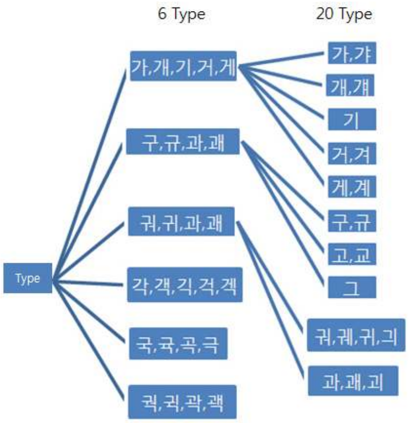

Figure 1. Type classification

In this paper, we defined essential terms which are used in segmentation processing, or in structural elements in characters.

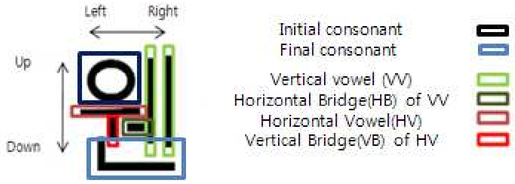

Figure 2. Definition

3.2. Method to the grapheme segmentation

3.2.1. Step 1. We suggest a new method to find the basis point to separate the vowel. Because each type of 20 types has the common character of the vowel that is widely distributed in the right or down side on a character image, we can trace the basis point easily and correctly by passing through the nearest part of the vowel from the outside boundary of the image. Therefore, it can be the leftmost or uppermost point of the vowel as shown in Figure 3.

3.2.2. Step 2. The basis point defined in step 1 is the starting point to trace the boundary of the vowel in the direction of up and down, or left and right according to the type as shown in Figure 3. We can save the execution time by tracing the boundary. When tracing the boundary, we save the position of the corner point which has a large curvature (the contact point of the main vowel and the bridge) because it means that there is large possibility to separate as two different grapheme elements.

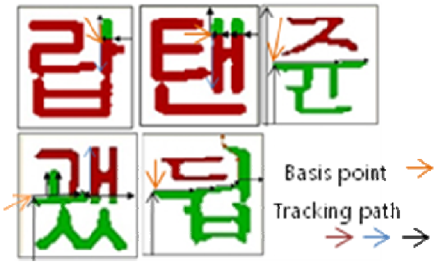

Figure 3. Tracking Method to the MC

3.2.3. Step 3. The bridge located at the saved corner position defined on step 2 is classified as a grapheme element by the information on the number and the position of the corners. This step must be done carefully because an incorrectly separated part will create a noise. By classifying 20 types, we can predict the number and position of the bridge and this leads to correct grapheme segmentation.

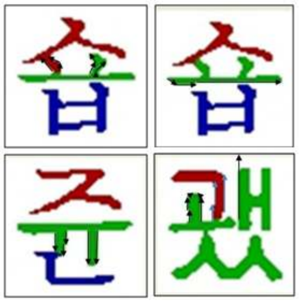

Figure 4. Tracking Method to the Bridge of the MC

3.2.4. Step 4. The final consonant is located on the lowest portion of the syllable from the corner or the end of the vertical vowel, which can be a basis point to separate the final consonant. After tracing to meet black pixel from the basis point and checking the direction to following the boundary from this pixel, we can find out which grapheme element has this boundary. That is, a clockwise direction means the boundary is a part of the initial consonant, and a count clockwise means the final consonant. With this step, the grapheme segmentation is done.

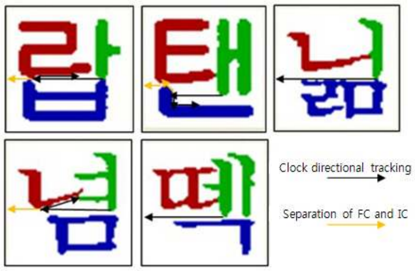

Figure 5. Method of FC separation

3.3. Method to the grapheme segmentation

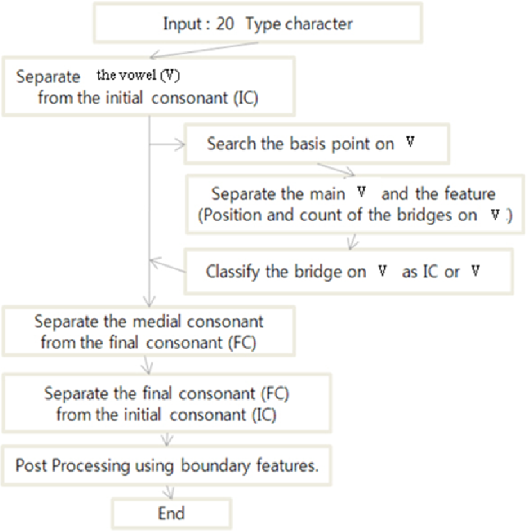
Input : 20 Type character
Separate the vowel (V) from the initial consonant (IC)
Search the basis point on V
Separate the main V and the feature (Position and count of the bridges on V)
Classify the bridge on V as IC or V
Separate the medial consonant from the final consonant (FC)
Separate the final consonant (FC) from the initial consonant (IC)
Post Processing using boundary features.
End

Figure 6. Process of the grapheme segmentation

4. Post processing of the grapheme segmentation

For the post process, the position, the number, and the boundary information (ex. the contact grapheme component element) of the connected components is found by tracing the boundary of the initial consonant, final consonant and the vowel respectively. When the number of the connected components in a consonant or a vowel is more than one, it can be defined as a noise if the size is small, or the position is not desirable. At last, the noise can be changed to the proper consonant or vowel by the information of the position and the boundary.

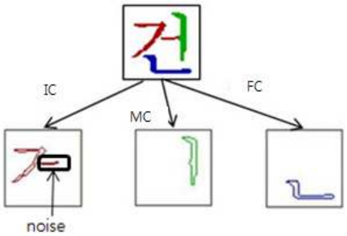

Figure 7. Connected components by tracking the boundary of consonants and vowel

This method also has the effect to correct the wrong grapheme segmentations caused on the wrong type classification.

5. Experimental results

5.1. Experimental circumstance

The proposed structural method of the grapheme segmentation is performed on the computer with Intel Core™ 2 Duo CPU E6570 2.66GHz. Using the scanner with 300 DPI and 400 DPI resolution, we developed the proposed algorithm at the basis of *Myunjo*(*Batang*, *Sinmyunjo*), *Gothic*(*Dodwom*, *Jungothic*), and *Gulim*. Full Korean font sets of 2,350 characters per each font are trained and tested to get the performance. In detail, the train data has 14700 characters and the test data has two hundred thousand characters. And the diverse fonts of general books are used to test and influence the performance in order to check the extensibility of our algorithm and with this purpose, the quality of printed document is also various levels including the image with the disconnected stroke or line.

5.2. Performance of the proposed grapheme segmentation algorithm

The successful segmentation rate of our grapheme is over 98.5% on average about the two hundred thousand characters without the post process.

One of the distinguishing marks of this algorithm is the invariant performance in various fonts. *Myunjo* only has a curved stroke in the vowel, *Gulim* is relatively thick, and the fonts in general books are deformed to some degree. In testing these diverse fonts that have not been used in developing process of our algorithm, we still see the consistently high performance levels to show the strong reliance on the invariance of fonts.

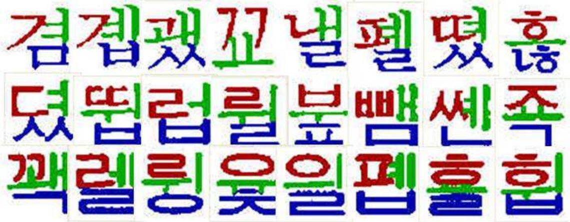

Figure 8. Grapheme Segmentation with small deformation or noise

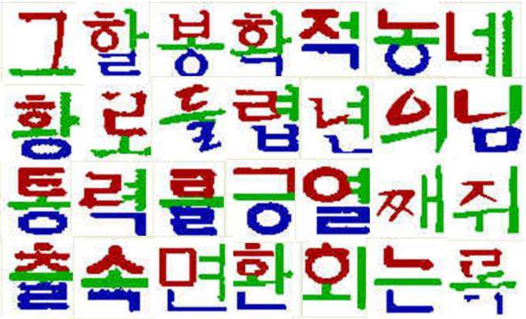

Figure 9. Grapheme Segmentation with large deformation or noise

After post processing the performance of the grapheme segmentation has improved to 99% on average from 98.5% on average in grapheme segmentation. The point of the post process is that the connected components separated by our rules can be a noise, and the position and the size of the components tell if they are noise or not.

The noise is caused by two cases. One occurs from the wrong segmentation of grapheme, and the other from the wrong type classification. The noise located on the specific position can be very important obstacle not to recognize the character correctly. Therefore, the proposed post process takes a role in checking the core part of characters and correcting it after the grapheme segmentation.

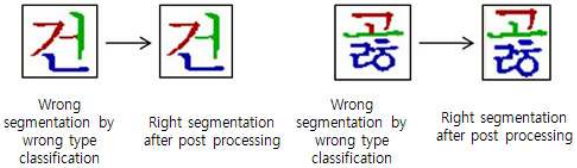
Wrong segmentation by wrong type classification
Right segmentation after post processing
Wrong segmentation by wrong type classification
Right segmentation after post processing

Figure 10. Post processing for wrong type classification

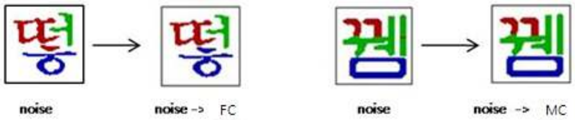

Figure 11. Post processing for correcting noise

The segmentation rates of 20 types after post processing are individually slightly different from each other. So, each type has the 99% segmentation rate on average. It means that the performance is irrespective to the complexity of the character.

The speed of the grapheme segmentation is so fast that it takes only a second per a thousand characters to segment graphemes.

6. Conclusion

The proposed structural method of grapheme segmentation has a strong point to be applicable to the characters with high complexity and deformation which is not expected in character recognition. We propose the simple and solid method to define and search the basis point of the vowel. And by tracing boundary and considering the point with high curvature as the significant separable of the grapheme, we reach the high efficiency in terms of correctness and speed.

Additionally, the proposed post process investigates and improves the performance by checking the boundary of the separated significant components in a grapheme.

Henceforth, we will test many more deformed and various characters and tune the algorithm by correcting and analyzing the trivial errors. And the limit of the post process that can correct when the noise is separated from the meaningful component will be improved.

10. References

[1] Seong-Whan Lee, "An Optimal Tree Classifier for High-Speed Recognition of Large-Set Multi-Font and Multi-Size Hangul", *Journal of the Korea Information Science Society*, v.20, n.8, Korea, 1993, pp.1083-1092.

[2] Lee Jin Soo, Oh-Jun Kwon, Sung-Yang Bang, "Highly Accurate Recognition of Printed Korean Characters through an Improved Grapheme Identification Method", *Pattern Recognition*, 1999, pp. 1935-1945.

[3] F.-H. Cheng and W.-H. Hsu, "Research on Chinese OCR in Taiwan", *Int. J. Pattern Recogn. Artificial Intell.,* v. 5, n. 1, 1991, pp. 139-164.

[4] Kil Taek Lim, Ho Yon Kim, "A Study on Machine Printed Character Recognition Based on Character Type Classification", *Journal of the Institute of electronics engineers of Korea*, v.40, n.5, 2003, pp.26~39.

[5] C.H. Teh and R.T. Chin, "On the detection of dominant points on digital curves", *IEEE Tr. On Pattern Analysis and Machine Intelligence*, v.11, n.8, 1989, pp.859-872.

[6] Y. Tseng. et al., "Recognition-based handwritten Chinese character segmentation using a probabilistic Viterbi algorithm", *Pattern Recognition*, 1999, pp. 791-806.

[7] S. Kahan, T. Pavlidis and H.S. Baird, "On the Recognition of Printed Characters of Any Font and Any Size", *IEEE Trans. on PAMI*, v.PANMI-9, n. 2, 1987, pp. 274-288.