01 Jan 1998

Substroke Matching by Segmenting and Merging for Online Korean Cursive Character Recognition

Chang-Soo Kim
Missouri University of Science and Technology, ckim@mst.edu

Kang Ryoung Park

Byung Hwan Jun

Jaihie Kim

Substroke Matching by Segmenting and Merging for On-Line Korean Cursive Character Recognition

Chang Soo Kim, Kang Ryoung Park, Byung Hwan Jun*, and Jaihie Kim
School of Electrical and Mechanical Eng., Yonsei University, Seoul, Korea
*Dept. of Computer Science, Kongju National University, Chungnam, Korea
E-mail : hammer@troll.yonsei.ac.kr
Address : A.I. Lab., Dept. of Electronic Eng., Yonsei University, Seoul 120-749, Korea.

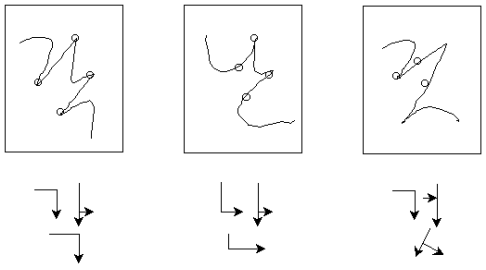
Abstract

The Korean character is composed of several alphabets in two-dimensional formation and the total number of Korean characters exceeds eleven thousand. Therefore, the previous approaches to Korean cursive characters pay most of their attention to segmenting a character into alphabets accurately. However, it is difficult because the boundaries of alphabets are not apparent in most cases. In this paper, we propose a new alphabet-based method without assuming accurate alphabet segmentation. In the proposed method, cursive character is segmented into substrokes by a set of segmenting conditions. Then it is matched with the reference substrokes generated from alphabet models and ligatures by segmenting and merging in the process of recognition. Among substrokes, a certain substroke can be either an alphabet itself, a part of alphabet or a composite of the alphabet and ligature. We applied the proposed method to 5,000 Korean characters and got the result of 83.4% for the first rank and 89.2% for the top 5 result candidates with the speed of 0.17 seconds on average per character on a PC which uses Intel Pentium 90Mhz CPU.

key words : *Korean cursive character recognition, segmenting and merging*

1. Introduction

Recent advances in wireless communication and pen-based hardware such as HPC(Handheld PC) and PDA(Personal Digital Assistant) have been influential factors in the renewed interest in on-line handwriting recognition. The Korean character[1] is composed of two or three letters in two-dimensional formation, as shown in Fig. 1 and the total number of Korean characters exceeds eleven thousand. The approach of recognizing Korean characters as a whole is inevitably limited to a small set of characters in comparison with the entire characters. Therefore, the approach of segmenting a Korean character into letters is suitable to the recognition of Korean cursive characters. We will call this approach as a letter-based approach in this paper.



$Character := C \cdot L_{cv} \cdot V \cdot L_{vc} \cdot C$
where
$(C)onsonant :=$ { ㄱ,ㄴ,ㄷ,ㄹ,ㅁ,ㅂ,ㅅ,$\cdots$,ㅎ },
$(V)owel :=$ { ㅏ, ㅑ, ㅓ, ㅕ,$\cdots$, ㅣ },
L_{cv} and L_{vc} are $(L)igature$ between letters, and the circles of handwriting scripts within the box mean the desired points of letter segmentation

Figure 1. The formations of Korean characters

In the letter-based approach, Korean characters recognition is the reverse process of handwriting, that is, a task to segment/recognize head consonant, ligature, vowel, ligature, and bottom consonant in succession. The previous letter-based approaches to Korean cursive characters pay

[1]The Korean character is composed of several alphabets. Henceforth, we will use the term "Korean character" in the sense of English word and the term "letter" in place of alphabet.

most of their attention to segmenting a character into letters accurately [2] [1]. However, accurately segmenting letter is sometimes very difficult due to co-articulation (the influence of one letter on another) as shown in Fig. 1(b) and 1(c). In Fig. 1c, the ligature between vowel and bottom consonant is omitted. For that reason, recognition errors of the previous approaches are caused mainly due to failures in letter segmentation. Therefore, we proposed a new letter-based method without assuming accurate letter segmentation. The proposed method consists of a couple of stages: The first is segmenting the Korean cursive character into substrokes among which a certain substroke may contain a part of letter and ligature, and the second stage is matching them with the reference substrokes generated from letter and ligature models by segmenting and merging in the process of recognition.

2. Outline of the recognition process

Fig. 2 first shows how a cursive input script of a Korean character '것' is segmented into substrokes. And it also shows how reference substrokes are generated from letter and ligature models, and how they are matched. In Fig. 2, the desired points of letter segmentation are on the second and the third line substrokes as shown in Fig. 1c, but it is difficult to find them by a particular segmentation algorithm. Therefore, it should be solved in the process of recognition. All Korean characters do not have the same problems as it is in Fig. 2 Some of them can be easily segmented into letters. However, not assuming accurate letter segmentation is more general in Korean characters.

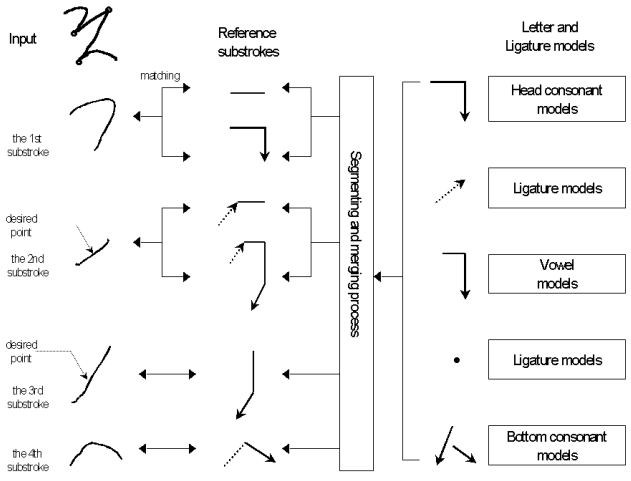


Figure 2. An example of the recognition process

3. Segmentation

As long as one of the below segmenting conditions is satisfied, taking place of segmentation is possible.

1. point of sudden change in direction : Where a change in direction over 90 degrees occurs.
2. point of inflection : Where there is a sign change in the angle of curvature.
3. point of excess rotation : Where the angle of curvature exceeds $|360|$ degrees. ($|\cdot|$ means absolute value)

As an example, Fig. 3 shows how the Korean cursive character is segmented into four substrokes.

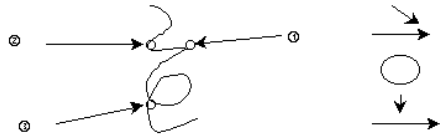


Figure 3. Segmentation of an input script

4. Recognition

In this section, a detailed description on Fig. 2 is provided; the description on letter models and ligature models, on the process of segmenting and merging, and on the features and the distance function used in matching, as follows.

4.1. Letter models and ligature models

We use three letter models and two ligature models. The letter models consist of the printed style templates which have real x-y coordinates. The templates for letter models can be obtained from any person. When it comes to ligature models, ligatures at letter boundaries are the dominant source of shape variability and the main hindrance of letter segmentation in cursive script. So, we classify ligatures into a set of groups according to the ending direction of preceding letters and the starting direction of following letters. An example of ligature models is shown in Fig. 4 The dot connection is used to determine whether the ligature can be omitted as shown in Fig. 2.

4.2. Segmenting and Merging

The templates for letter models are segmented by using the same algorithm as it was used to segment the input script except that segmentation is not restricted within letters. Rather, segmentation including ligature or the stroke

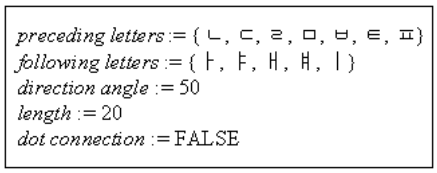
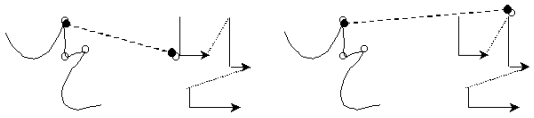


Figure 4. An example of ligature models

of following letters is allowed. This way is possible because all handwritten Latin words or composite characters like Korean characters can be modelled into finite state network(FSN), That is,

$$\text{Handwriting} := \text{Letter} \cdot \{\text{Ligature} \cdot \text{Letter}\}^*$$

where "*" means repetition. Owing to the finite state network, we do know the following letters and relationship(or ligature) between the current letters and the following letters. Therefore, we can segment beyond letter boundaries. We also use finite state network that is composed of head consonant, ligature, vowel, ligature, and bottom consonant. However, segmenting beyond letter boundaries will lead to another problem; where to stop segmentation. For that reason, the previous approaches have mostly paid attention to accurate letter segmentation algorithm. In this paper, we segment the templates for letter models and ligatures into substrokes irrespective of letter boundaries and use segmenting and merging conditions as a guide to the answer of where to stop segmentation. Fig. 5a shows how an input script is segmented by either "(1) point of sudden change in direction" or "(2) point of inflection". And it also shows how the templates for letter models and ligatures are segmented by "(1) point of sudden change in direction" alone.



(a) (b)

Figure 5. An example of segmenting and merging process

As shown in Fig. 5a, the correct matching between an input script and models cannot be carried out only by segmenting. Usually, the detected points by segmenting are reasonable, but some of them are unnecessary sometimes. Therefore, to cope with these problems, the possibility that there may be unnecessary detected points by segmenting should always be considered. The consideration can be solved by merging. The merging condition used in this paper is as follows:

merging condition : any previous segmented points can be ignored, until either "(2) point of inflection" or "(3) point of excessive rotation" occurs.

The above merging condition is not a novel concept, however it is just union of the second and the third conditions among segmenting conditions. But, it has an effect of ignoring the unnecessary segmented points caused by the first condition, "(1) point of sudden change in direction". By merging, we can carry out the correct matching, as shown in Fig. 5b. The Fig. 5b shows the segmented point caused by "(2) point of inflection". However, it does not mean that the matching by merging is correct all the time. It is true that there are times where the matching by segmenting is correct and there are times where the matching by merging is correct. In conclusion, the process of recognition consists of several stages: The first is segmenting an input script into several substrokes by segmenting conditions. The second stage is generating many reference substrokes from letter models and ligatures by segmenting and merging conditions. In the third stage, a lot of matching are carried out. During this process so many recognition candidates are produced. The second and the third stages are repeated until the last substroke of an input script is reached. The character recognition is finished at the last substroke of an input script by choosing the best candidate.

4.3. Matching

Matching between a substroke and a candidate substroke is based on the euclidean distance of features. The features are shown in Fig. 6.

Curvature : made up of the accumulated angles between vectors toward each point from the center of gravity. The value of curvature carries signs according to clockwise(-) or anti-clockwise(+) direction.

DCH (Direction from Center to Head) : the direction of the vector from the center point of a substroke to the head point.

DCT (Direction from Center to Tail) : the direction of the vector from the center point of a substroke to the tail point.

DHT (Direction from Head to Tail) : the direction of the vector from the head point of a substroke to the head point.

In the last stage of the matching process, the distance function is used. The definition of distance function is expained in the below. Imagine that N substrokes are used in the recognition, distance funciton *Dist* of the character is

(a) (b) (c) (d)

Figure 6. Features for describing a substroke (a)Curvature (b)DCH (c)DCH (d)DHT

defined as the average for distance values of N substrokes in equation (1).

$$Dist = \sum_{i=0}^{N} d_e(d_i, r_i)/N \quad (1)$$

where $d_e(s_i, r_i)$ is Euclidean distance, s_i is a substroke of an input script, and r_i is a substoke from models.

5. Experiments

The proposed algorithm was implemented and tested on IBM PC compatible Pentium(90Mhz). In the process, WACOM tablet digitizer was used as an input device. The test data were written by 10 different writers without any constraints. These writers wrote 500 different characters that were well prepared by putting all the consideration of the frequency of usage and the combination of letters in Korean characters. The survey of Lexicographical Center in Yonsei University is referenced for the frequency of usage in Korean [3]. Fig. 7 shows some examples of the test data.

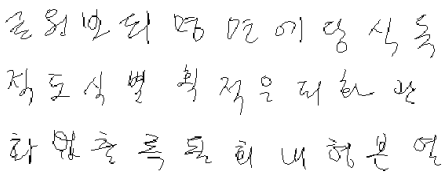
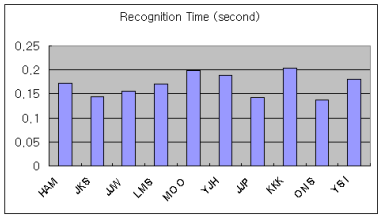


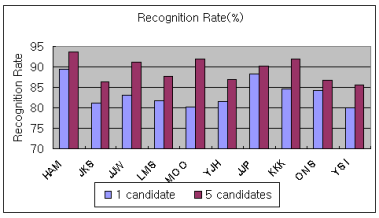
Figure 7. Some examples of the test data

The recognition rates and times for each writers is shown in Fig. 8

The main cause of recognition errors lies in the similarity of Korean characters itself such as (“ㄷ”) and (“ㄹ”). The difference between 1 candidate rate and 5 candidate rate comes from such kind of reason. In order to overcome the similarity, some pairwise distinction is needed. Other causes are excessive hooks which are misrecognized as the part of other letters, another is the habit of personal writing, and so on.



(a)



(b)

Figure 8. The recognition rates and times

6. Conclusion

The Korean character is composed several letters in which connections appears within and among letters. However, the boundaries of letters are not apparent in most cases. Therefore, we propose a new segmentation-based method without assuming accurate letter segmentation. The cursive characters which have ambiguous letter boundaries could be recognized in most case. In order to improve the recognition rate, it is necessary to augment cursive templates for letter models and it is also necessary to use some pairwise distinction for the similar patterns.

References

[1] O. H. Kwon, M. K. Kim, M. K. Park, and Y. B. Kwon. A cursive on-line hangul recognition system based on the combination of line segments. In *Proc. Int'l Conf. Document Analysis and Recognition*, pages 200–203, Tsukuba City, Japan, Oct. 1993.

[2] S. Lee and J. H. Kim. On-line cursive script recognition by a letter spotting techniques based on hmms. In *Proc. Second Workshop Character Recognition*, pages 93–104, Seoul, Korea, Sept. 1994.

[3] S. S. Lee. Accumulation of linguistic information and statistical information - accumulation of korean information. Technical report, KAIST Research Report, Lexicographical Center, Yonsei University, 1991.