

Correction of Text Character Skeleton for Effective Trajectory Recovery

Hoai Nam Vu, In Seop Na*, Soo Hyung Kim

School of Electronics and Computer Engineering

Chonnam National University, 77 Yongbong-ro, Buk-gu, Gwangju 500-757, Korea

ABSTRACT

One of the biggest problems of skeletonization is the occurrence of distortions at the junction point of the final binary image. At the junction area, a single point usually becomes a small stroke, and the corresponding trajectory task, as well as the OCR, consequently becomes more complicated. We therefore propose an adaptive post-processing method that uses an adaptive threshold technique to correct the distortions. Our proposed method transforms the distorted segments into a single point so that they are as similar to the original image as possible, and this improves the static handwriting images after the skeletonization process. Further, we attained promising results regarding the usage of the enhanced skeletonized images in other applications, thereby proving the expediency and efficiency of the proposed method.

Key words: Skeleton, Offline Handwriting, Thinning, Post-processing.

1. INTRODUCTION

Artificial analysis of offline handwriting can be classified into two regions: recognition of offline handwriting and trajectory recovery of offline handwriting for the purpose of authorship verification. Though the algorithms implemented in these regions have many similarities, there are also considerable differences. Both regions use features extracted from offline handwriting samples. In recognition of offline handwriting, application features are used for differentiating characters from the others or their combinations and thus needing to find out dissimilarities between characters and similarities of the same character written by various people. In trajectory recovery of offline handwriting, the features are used for recovering drawing order of writer when writing characters without pen-tip movement information, such as pen-tip coordinates, pressure, velocity, acceleration, pen-up and pen-down, thus needing to emphasize the correction of skeletonization stage, the very first stage in trajectory recovery of offline handwriting system. For both regions, many of those features are extracted from skeleton image of offline handwriting characters.

The number of algorithms was proposed in literature to obtain skeletonized pattern of binary handwriting samples. Pertinence of individual algorithm depends on requirement of particular application. For trajectory recovery of offline handwriting character purpose [1]-[4], the length of the character should be preserved from skeletonization process, thus thinning algorithms which conserve the length of the

character are used very popular in regions of trajectory recovery of offline handwriting character. Nevertheless, thinning results usually include unexpected area around intersection point of binary image. This makes thinning results of offline handwriting character become far different from human handwriting. In the comprehensive survey [5], the general idea of trajectory recovery problem always includes the ambiguous zone detection step which deal with distortion around the junction points of the original character. In order to solve this problem, most of the approaches [6]-[8], start at correct junction point from skeleton character. The authors of [1] proposed a machine learning based approach which uses a large of instance image in order to train on training dataset. Therefore they do not face the problem of the distortion of skeleton images. However, their method depends heavily on the quality of their dataset, and the results will change over the different dataset. In [3] Qiao et. al. proposed a method which builds a graph model of the skeleton topology of offline handwriting characters. This method depends heavily on how the skeleton topology is created, because their concept Edge Continuity Relation (ECR) is deeply affected by quality of the skeleton result. In addition, their maximum weighted matching of general graph collects input value from the junction point of the skeleton topology. However, the skeleton topology of most well-known algorithms always is resulted with a lot of ambiguous regions around the junction point. Therefore, in the field of trajectory recovery, some authors [6], [9]-[11] have tried to overcome problem of skeleton by creating specialized skeleton result for their own use. Nevertheless, this process makes their algorithm cost more processing time and complex to follow. There are various types of skeleton as well as thinning algorithms which are well-defined and explained in literature, they are also proven to be efficient in time

* Corresponding author, Email: ypencil@daum.net

Manuscript received Feb. 24, 2015; revised Aug. 03, 2015; accepted Aug. 10, 2015

consuming. All of these observations motivated us to conduct a post processing method after thinning algorithm in order to exactly extract feature point around the ambiguous region of intersection point in the original character, especially for off-line handwriting drawing order recovery purpose.

The problem of thinning extraction has been widely studied for many years. Most of those thinning algorithms [12]-[14] base on the repeated erosion of pattern boundaries while keeping the connectivity of pattern unchanged. All are significant impressible to spurious small hole appearing around intersection area and some other noises in binary image. Post-processing stage needs implementing to avoid defects around the intersection regions as much as possible and make the skeleton image more useful for the real applications which take the skeleton image as input.

This paper is organized as follows: In section II, we present a preprocessing stage. A detailed implementation of proposed post-processing algorithm is presented in section III. In section IV, we show experimental results and discussion. Concluding of the paper is given in section V.

2. PRE-PROCESSING

The most considerable aspect in offline handwriting recognition system as well as trajectory recovery of offline handwriting is that the feature points need extracting correctly from the offline handwriting input images. These feature points of offline handwriting images can be simply extracted from their thinning results. Nevertheless, thinning algorithms generally encounter the problem that they are sensitive to some kind of noise in binary image. It makes the thinning results far different from expected one, thus needing to implement a pre-processing stage before the thinning procedure to remove noise and make the quality of the thinning results much better.

By observing the differences between the thinning images and their corresponding binary images, it can be certainly shown that the isolated areas (hole, dot) in binary images of Figure 1 create a number of spurious strokes in the thinning image as seen in Fig. 2.

In this paper, some pre-processing techniques are given for removing these defects before thinning steps.

- We define a suitable mask for removing spurious hole in the binary image to avoid unexpected stroke which may appear in the thinning results.
- For the case of spurious isolated pixel, firstly we calculate average number of pixel of connected component t . Secondly we define a threshold t_1 which base on t . After that, the spurious isolated pixel area with the number of pixel below t_1 will be removed.
- We use the algorithms presented in [15], [16] to eliminate others spurious stroke may appear in the thinning results.



Fig. 1. Spurious hole in binary image

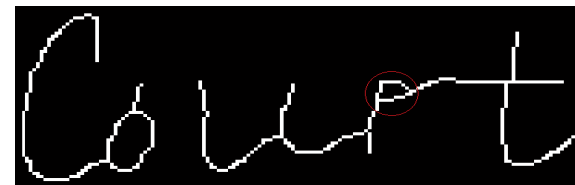


Fig 2. Spurious hole in binary image

3. PROPOSED POST-PROCESSING METHOD

There are two types of segment detected from skeleton of a handwriting image, i.e. a real segment corresponding to part of a real stroke, and a spurious segment resulting from the skeleton process and never exists in an original handwriting image. To distinguish these two types in the post-processing stage plays a vital role in skeleton-based applications. The recognition of spurious segment has a great effect on the performance of skeleton-based applications. Besides, due to a reduction in the number of segments, the application processing time will be shortened. The adaptive threshold method whose flowchart is shown in Figure 3 will be used as follows.

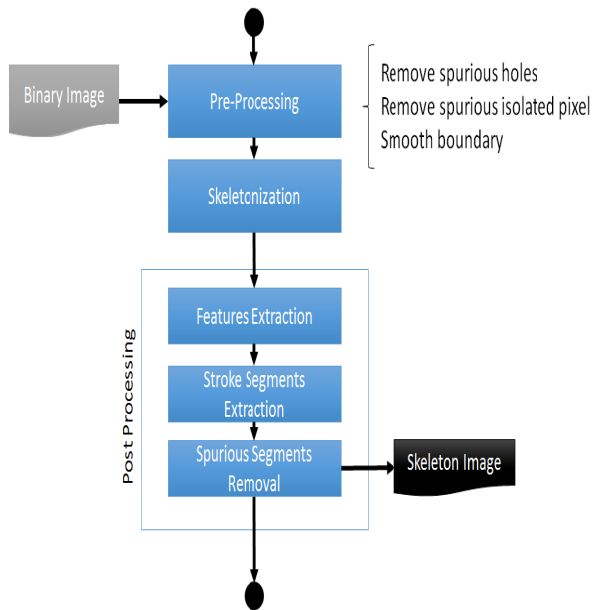


Fig. 3. Flowchart of proposed method

3.1 Stroke width estimation

To classify two types of segment, it is necessary to calculate the average stroke width information (w) of the static handwriting character. In this paper, we have implemented the stroke width estimation in [16].

3.2 Loop segment detection

The segment connected to loop segment is the most probably the spurious segment. As a result, the adaptive method should be used at the intersection related to a loop segment and a normal intersection separately.

3.3 Adaptive threshold method

1) Identify every feature point in skeleton image by gauging the number of neighbor of pixel (n) and the cross number (r). Junction point is the pixel that abides by the following condition [17].

$$n \geq 4 \text{ or } r \geq 3 \quad (1)$$

2) Fix feature point detection in case of inaccurate detection of junction points according to the method used in step 1. As a result, a predefined mask needs to be used to iterate whole skeleton image in the fixing process. If identifying five junction points (white pixel) fitting mask L, we will replace them with only one point the center of mask L. If a pixel and their neighbor 8 fits mask H1, H2, H3, H4 which shown in Figure 4, it will be marked to be deleted.

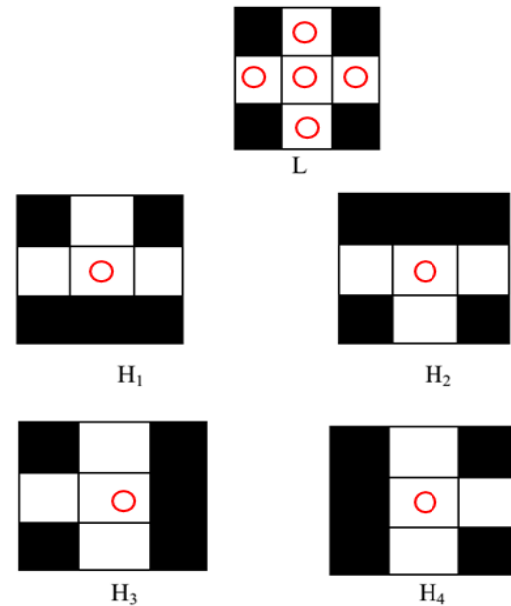


Fig. 4. Predefined mask for correcting feature detection (white pixel is foreground and black pixel is background)

3) Average length of loop segment needs computing. For the segment which is not connected to any loop segment, we set threshold as $t_1=w$. For the segment which is connected to a loop segment, we set threshold as follows:

$$t_2 = \frac{w \cdot l}{l} \quad (2)$$

4) Firstly, each segment is identified whether it is connected to any loop segment for setting a suitable threshold. Secondly, if the length of segment is smaller than the threshold, the segment is classified as a spurious segment. Otherwise the segment is classified as a real segment. The threshold is chosen based on the stroke width because of two reasons. Firstly, the junction point in the original character is the intersection of two strokes or segments; therefore the length of spurious segment of junction area will be proportioned to the stroke width. That means, the more the stroke width is, the more the length of spurious segment is. Secondly, the threshold is not uniform for all character images since they have different writing style and size of input character. However these various aspects are not significant, therefore the use of stroke width in this case is acceptable. In addition, during experimentation we observed the changing of the length of the spurious segment depending on the stroke width in order to set suitable threshold.

5) By observation the characters which are written by various writers, it can be easily seen that a spurious segment can appear only between two junction points. Therefore, if the spurious segment candidate detected in step 3 is between a junction point and an end point or between two end points, it should be considered a real segment.

6) After all spurious segments are identified; we have to cluster these spurious segments if they are connected together. Two types of spurious segment so-called single spurious segment and group spurious segment will be transformed into a new intersection point. For single spurious segment, the new

intersection point is considered as middle point of the spurious segment. For the group of the spurious segment, the new intersection point is calculated by using set of middle points of the segments which belong to that group.

7) After having all new intersection points, a reconstruction process is utilized at these spurious segments as well as group of spurious segment by deleting spurious segment and group these spurious segments and then connecting real segment to corresponding the new intersection point which related to the real segment.

Algorithm :

Finding all feature points in the input image, try to find set of the junction points using equation (1).

Initializing: calculate total number of segment (real segment and spurious segment) in the input image (n), calculate average length of loop segment (als), stroke width (w)

```

for i := 1 to n step 1 do
  if segment is loop segment then
    threshold =  $w * \text{length of entire loop segment} / als$ ;
  else if
    threshold =  $w$ ;
  end
  if length of segment is smaller than threshold and the
  segment is between two junction points then
    segment is classified as spurious segment;
  end
end

```

Building adjacent graph [18] of the segments

```

for i := 1 to n step 1 do
  if segment is spurious segment then
    do grouping based on the adjacent graph
  end
end

```

Doing reconstruction process
return new skeleton image

All intermediate images of whole process are shown in following figures. Fig. 5 shows the input binary image of handwritten character “western”. Then, equation (1) is applied to skeleton image to extract the feature point such as end points (green point) and junction points (red point) in Fig. 6. After having all feature points extracted, all segments of character are examined to figure out which is the spurious segment as shown in Fig. 7. Lastly, clustering process is implemented to turn spurious segment into only one feature point in Fig. 8. At this point of view, a skeleton pattern of character contains only real segment which model exactly a handwritten character without any extraordinary unexpected pattern.

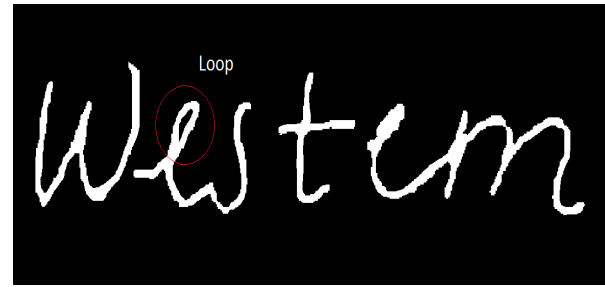


Fig. 5. Binary image

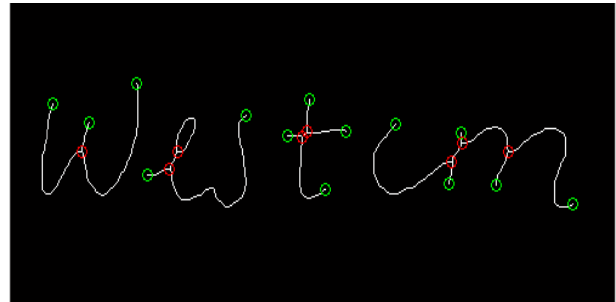


Fig. 6. Feature point extraction
(green point is end point, red point is junction point)

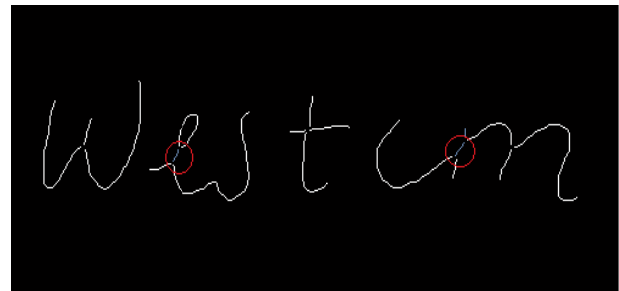


Fig. 7. Spurious segment detection



Fig. 8. Spurious segment clustering

4. EXPERIMENTAL RESULTS

4.1 Dataset

In this experiment, we used set of the hand drawn scripts which include 9 English documents with over 1000 character in different style as shown in Fig. 9.

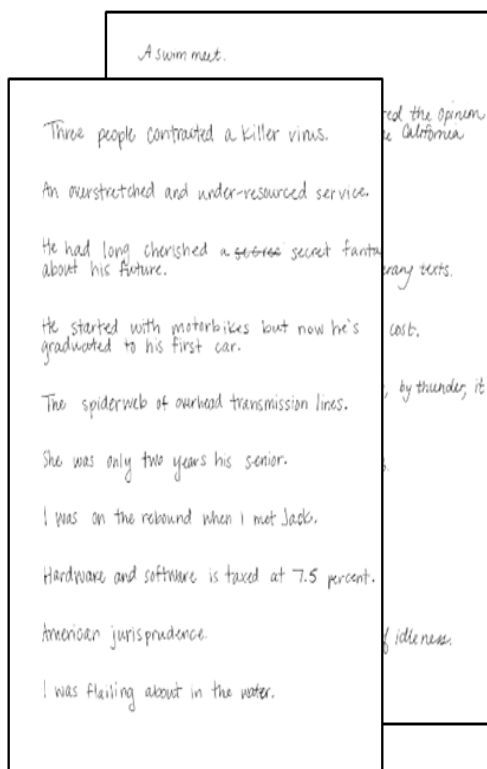


Fig. 9. Dataset

In addition, we have used the public dataset from [19] in order to evaluate the proposed system. Experimental result of our proposed system on this dataset is given in Table 2. In our evaluation scheme, first of all, original skeleton algorithm such as Zhang-Suen algorithm is applied to the binary image. Then our proposed method is implemented on these results to get the final corrected skeleton image. Lastly, we have reconstructed the binary images from the result of the skeleton image to make a comparison between our proposed method and original skeleton algorithm. The reconstructed image is calculated by using the stroke width of the binary character (w) from the previous section. Because there are many distortions existing in the skeleton image, the reconstructed image of original skeleton algorithm considerably different from the original binary image, and the reconstructed binary image of corrected skeleton image is similar to the original binary image. Based on these observations, we proposed a method in order to evaluate the efficiency of our proposed system by using the matching rate which is calculated by the reconstructed binary images and the original binary images

4.2 Evaluation by matching rate

For directly evaluating the results of proposed method, we reconstructed a binary image based on the skeleton after proposed method to calculate matching rate between the original binary image and the reconstructed binary image. The matching rate is calculated using following equation:

$$matching_rate = 100 \frac{t_1}{t_2} \tag{3}$$

Where t_1, t_2 are the number of foreground pixel in the original binary image and the reconstructed binary image respectively.

The comparing results which given in Table.1 shows that when our proposed method is applied, the better results are achieved.

Table 1. Matching rate for our dataset

	Zhang-Suen algorithm [13]	Distance Transform-based algorithm[15]	After proposed method
Document. 1	89.23%	87.52%	91.41%
Document. 2	92.1%	91.06%	93.68%
Document. 3	90.1%	89.72%	93.17%
Document. 4	92.2%	91.15%	92.23%
Document. 5	89.6%	89.71%	91.64%
Document. 6	91.5%	90.63%	93.56%
Document. 7	91.23%	90.17%	94.87%
Document. 8	92.32%	91.25%	93.76%
Document. 9	92.64%	92.2%	94.1%
Average	91.21%	90.38%	93.16%

Table 2. Matching rate for dataset [19]

	Zhang-Suen algorithm [13]	Distance Transform-based algorithm[15]	After proposed method
Average rate	89.84%	88.52%	92.12%

4.3 Evaluation by trajectory recovery application

Trajectory recovery of handwriting character is a technique to estimate the drawing order of a given handwritten character image. For “x” a correct stroke recovery result is “>” → “<” (where strokes are written from top to bottom). Similarly, for χ , the correct result should be “>” → “>”. Trajectory recovery of handwriting character depends on behavior of writer and we also do not have information about pen-tip movements, such as pen-tip coordinate, pressure, velocity, acceleration, pen-up, pen-down, therefore it is difficult to obtain correct results. The above examples of two similar characters clearly show how the problem is difficult. In this paper, we utilize an algorithm where trajectory of character can be estimated using graph based method to searching the optimal path. Through the optimal path searching, the multi-strokes character can be decomposed into single strokes to achieve exactly trajectory of character.

In the trajectory recovery results shown in Fig. 10, the different color arrow symbol shows the direction of the different stroke respectively. For example in the character “co”, the yellow arrow shows the direction of character “c” and the pink arrow shows the direction of the character “o”. And the Fig. 11 shows some others results of trajectory recovery application after our method. It is clearly seen that the results of the trajectory recovery after proposed method are much more similar to the human handwriting than the original skeleton.

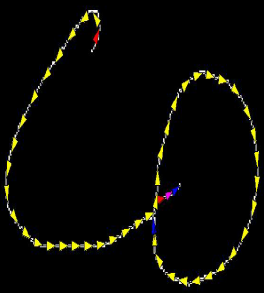
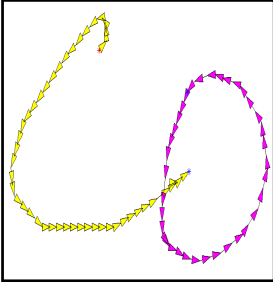
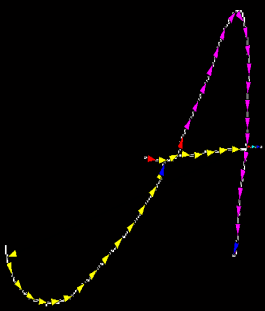
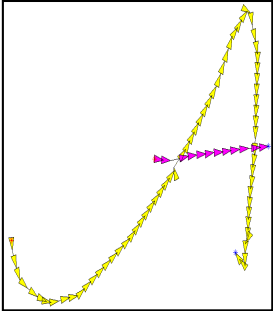
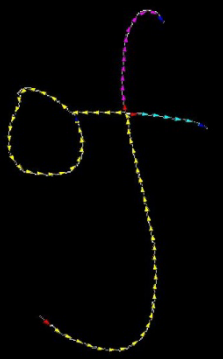
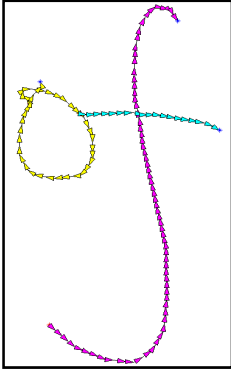
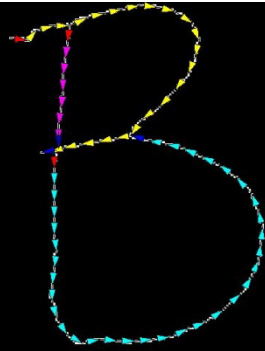
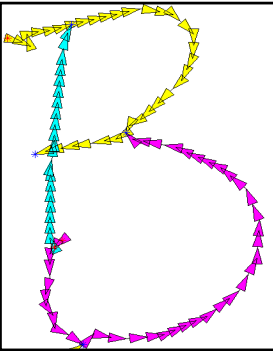
Zhang-Suen skeletonization	Skeletonization after proposed method
	
	
	
	

Fig. 10. Comparison results between proposed method and Zhang-Suen skeletonization

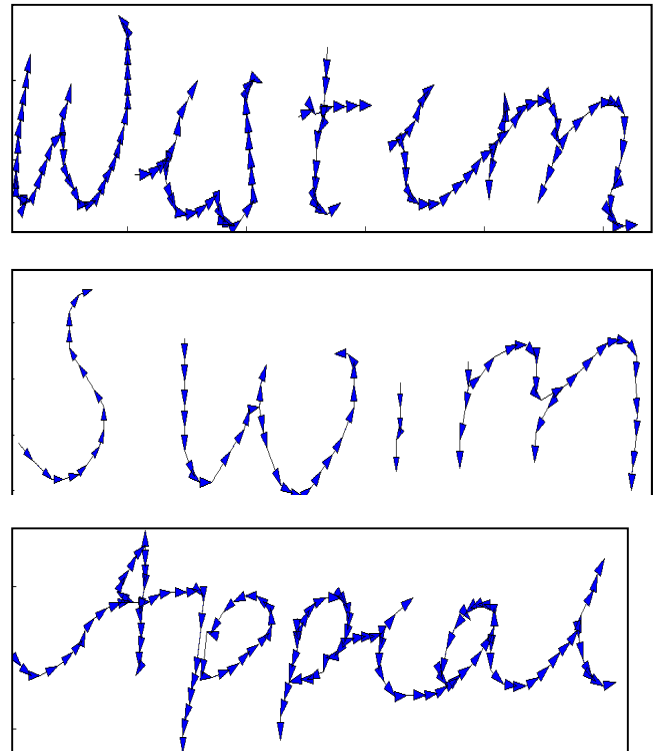


Fig. 11. Results from trajectory recovery application

5. CONCLUSION

In this paper, several efficient post-processing techniques are given to increase the quality of skeletonized image of offline handwriting character for feature extraction, an important step in recognition systems as well as trajectory recovery applications by using an adaptive method to distinguish a real stroke from a spurious stroke in the skeleton image. However, in some terrible cases, when the spurious segment length is very long, our method has failed to detect these spurious segments. The higher value of the matching rate as well as the success rate in trajectory recovery system has illustrated the promising performance of our proposed method.

ACKNOWLEDGEMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2015-018993) and this research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2015R1C1A1A02036495).

REFERENCES

- [1] Y. Iwakiri, S. Shiraishi, Feng Yaokai, and S. Uchida, "On the possibility of instance-based stroke recovery," *Frontiers*

- in Handwriting Recognition (ICFHR), 2012 International Conference, Sep. 2012, p. 29, p. 34.
- [2] Yu Qiao, M. Nishiara, and Makoto Yasuhara, "A Framework Toward Restoration of Writing Order from Single-Stroke Handwriting Image," *Pattern Analysis and Machine Intelligence, IEEE Transactions*, vol. 28, no. 11, Nov. 2006, pp. 1724-1737.
- [3] T. Steinherz, D. Doermann, E. Rivlin, and N. Intrator, "Offline Loop Investigation for Handwriting Analysis," *Pattern Analysis and Machine Intelligence, IEEE Transactions*, vol. 31, no. 2, Feb. 2009, pp. 193-209.
- [4] L. P. Cordella, C. De Stefano, A. Marcelli, and A. Santoro, "Writing Order Recovery from Off-Line Handwriting by Graph Traversal," *Pattern Recognition (ICPR), 2010 20th International Conference*, Aug. 2010. pp. 1896-1899.
- [5] Vu Nguyen and Michael Blumenstein, 2010, Techniques for static handwriting trajectory recovery: a survey In *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems(DAS '10)*, ACM, New York, NY, USA, 463-470. DOI=10.1145/1815330.1815390 <http://doi.acm.org/10.1145/1815330.1815390>
- [6] V. Pervouchine, G. Leedham, and K. Melikhov, "Handwritten character skeletonization for forensic document analysis," in *ACM Symposium on Applied Computing*, 2005.
- [7] E. L'Homer, "Extraction of strokes in handwritten characters," *Pattern Recognition*, vol. 33, no. 7, 2000, pp. 1147-1160.
- [8] Y. Kato and M. Yasuhara, "Recovery of drawing order from single-stroke handwriting images," *PAMI, IEEE Transactions on*, vol. 22, no. 9, 2000, pp. 938-949.
- [9] E. M. Nel, J. du Preez, and B. Herbst, "A pseudoskeletonization algorithm for static handwritten scripts," *IJDAR*, vol. 12, no. 1, 2009, pp. 47-62.
- [10] A. Dawoud and M. Kamel, "New approach for the skeletonization of handwritten characters in gray-scale images," in *7 th ICDAR, Edinburgh, Scotland, 2003*, pp. 1233- 1237.
- [11] T. Steinherz, N. Intrator, and E. Rivlin, "A special skeletonization algorithm for cursive words," in *7 th IWFHR, Amsterdam, Netherland, 2000*, pp. 529-534.
- [12] Louisa Lam, Seong-Whan Lee, and Ching Y. Suen, "Thinning Methodologies-A Comprehensive Survey," *Pattern Analysis and Machine Intelligence,* *IEEE Transactions on*, vol. 14, no. 9, Sep. 1992, pp. 869-885.
- [13] T. Y. Zhang and C. Y. Suen, "A Fast Parallel Algorithm for Thinning Digital Patterns," *Communications of the ACM*, vol. 27, no. 3, Mar. 1984.
- [14] Z. Gou and Richard W. Hall, "Parallel Thinning with Two-Subiteration Algorithms," *Communications of ACM*, vol. 32, no. 3, Mar. 1989.
- [15] Sukmoon Chang, "Extracting Skeletons form Distance Maps," *IJCSNS International Journal of Computer Science and Network Security*, vol. 7, no. 7, Jul. 2007.
- [16] F. W. M. Stentiford and R. G. Mortimer, "Some New Heuristics for Thinning Binary Handprinted Characters for OCR," *IEEE Transactions on System, Man, Cybernetics*, vol. SMC-13, no. 1, 1983.
- [17] Ke Liu, Yea S. Huang, and Ching Y. Suen, "Identification of Fork Points on the Skeletons of Handwritten Chinese Characters," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 21, no. 10, Oct. 1999, pp. 1095-1100.
- [18] Hoang-Nam Bui, In-Seop Na, and Soo-Hyung Kim, "Staff Line Removal Using Line Adjacency Graph and Staff Line Skeleton for Camera-Based Printed Music Scores," *Proc. 22nd International Conference on Pattern Recognition, Stockholm, Sweden, Aug. 2014*, pp. 2787-2789.
- [19] <http://www.iam.unibe.ch/fki/database/iam-handwriting-database>.



Hoai Nam Vu

He received his B.E. degree in school of Electronics and Telecommunications from Ha Noi University of Science, Viet Nam and Technology in 2013. He has been taking M.S course in Electronics & Computer Engineering at Chonnam National University, Korea. His research interests are pattern recognition, machine learning and image processing.



In Seop Na

He received his B.S., M.S. and Ph.D. degree in Computer Science from Chonnam National University, Korea in 1997, 1999 and 2008, respectively. Since 2012, he has been a research professor in Department of Computer Science, Chonnam National University, Korea. His research interests are image processing, pattern recognition, character recognition and digital library.



Soo Hyung Kim

He received his B.S. degree in Computer Engineering from Seoul National University in 1986, and his M.S. and Ph.D degrees in Computer Science from Korea Advanced Institute of Science and Technology in 1988 and 1993, respectively. From 1990 to 1996, he was a senior member of research staff in Multimedia Research Center of Samsung Electronics Co., Korea. Since 1997, he has been a professor in the Department of Computer Science, Chonnam National University, Korea. His research interests are pattern recognition, document image processing, medical image processing, and ubiquitous computing.