Proceedings of the 2007 International Conference on Wavelet Analysis and Pattern Recognition, Beijing, China, 2-4 Nov. 2007

HANDWRITTEN TEXT LINE EXTRACTION BASED ON MINIMUM SPANNING TREE CLUSTERING

FEI YIN, CHENG-LIN LIU

National Laboratory of Pattern Recognition (NLPR), Institute of Automation, Chinese Academy of Sciences
P.O. Box 2728, Beijing 100080, P.R. China
E- MAIL: {fyin, liucl}@nlpr.ia.ac.cn

Abstract

Text line extraction from unconstrained handwritten documents is a challenge because the text lines are often skewed and curved and the space between lines is not obvious. To solve this problem, we propose an approach based on minimum spanning tree (MST) clustering with new distance measures. First, the connected components of the document image are grouped into a tree by MST clustering with a new distance measure. The edges of the tree are then dynamically cut to form text lines by using a new objective function for finding the number of clusters. This approach is totally parameter-free and can apply to various documents with multi-skewed and curved lines. Experiments on handwritten Chinese documents demonstrate the effectiveness of the approach.

Keywords: OCR; Handwritten text line extraction; Connected component labeling; MST clustering; Multi-skewed document

1. Introduction

Document image layout analysis and text line extraction are important tasks in optical character recognition (OCR). Unlike that printed documents consist mostly of rectangular blocks and straight lines, the text lines in unconstrained handwritten documents are often multi-skewed and curved, and the space between lines is not obvious. Moreover, some text lines in handwritten documents touch with each other. So, the techniques used in printed document analysis, such as projection analysis, Hough transform and recursive X-Y cut, do not perform well on handwritten documents.

As the techniques of handwritten character and word recognition advances, the analysis of real-life handwritten documents is receiving increasing attention. Some effective methods have proposed for handwritten text line extraction and have shown success in less challenging applications such as historical document analysis [1,2].

Likforman-Sulem, et al. categorize the methods of text line extraction into six types [1]: projection based, smearing based, Hough transform based, grouping based, repulsive-attractive network, and stochastic based. The former four methods have been widely followed.

The projection based method used to apply to printed documents, where the straight and parallel text lines can be separated by the valleys of the projection profile. Manmatha and Rothefeder apply the projection method to gray-level historical document images [2]. To separate curved and non-parallel text lines in handwritten documents, the document image is usually partitioned into vertical strips [3,4]. Local text lines extracted from different strips according to horizontal projection profiles are connected using heuristic rules.

By the smearing based method, the black pixels along the horizontal axis are smeared such that the bounding boxes of connected components enclose text lines. Similarly, Shi, et al. use an adaptive local connectivity map (ALCM), in which the value of each pixel is the sum of all pixels in the original image within a specified horizontal distance [5]. After thresholding the smeared image, the connected components then represent probable regions of text lines. The smearing based method involves some parameters to be tuned artificially, however.

The Hough transform is a powerful tool for detecting straight lines in images. It has been applied to extract text lines in printed and handwritten documents, usually with the gravity centers [6] or minima points [7] of connected components as the points to be fitted. Sometimes, the connected components are split into equally spaced blocks to be voted in the Hough domain [8].

The grouping based method forms text lines by aggregating units in a bottom-up manner, and so, is deemed to be more robust against skewed and curved lines. The units may be pixels, connected components, or other features such as main strokes. A clustering algorithm, such as nearest neighbor clustering [9] and minimum spanning tree (MST) [10,11] can group the units into text lines. However, the design of an appropriate distance measure between the units is influential, and the clustered text lines need careful post-processing.

1-4244-1066-5/07/$25.00 ©2007 IEEE

In this paper, we propose a new approach based on clustering for extracting text lines in unconstrained handwritten documents. We use the connected components as the units to be clustered using MST with a new distance measure. After MST clustering, the edges of the tree are dynamically cut to form text likes. We use the reduced hypervolume as a criterion for finding the edge to cut and use a novel objective function based on the characteristic of documents to determine the final number of clusters. Our new approach is free of artificial parameters, and can apply to various documents of multi-skewed and curved text lines.

In the rest of this paper, we explain the rationale of our approach in Section 2 and describe the details of the algorithm in Section 3. Experimental results are presented in Section 4 and concluding remarks are offered in Section 5.

2. Rationale

A document image can be viewed as a hierarchical structure as in Fig. 1: it consists of text lines, each text line consists of connected components (CCs), and a CC is made of black runs or pixels. Equivalently, a text line can be viewed as a cluster of CCs or pixels with irregular boundaries. The minimum spanning tree (MST) clustering algorithm is known to be computationally efficient and capable of detecting clusters with irregular boundaries. To cluster the CCs is even more computationally efficient because the number of CCs is much smaller than that of black pixels and the pixels in a CC mostly belong to the same text line. To deal with the few occasions of touching components from different lines, we can detect such components, exclude from clustering, and finally split into different lines after clustering. Currently, we have not yet considered such touching components.

In clustering-based text line extraction, two important issues are the distance measure between the units to be clustered and the criterion for determining the number of clusters. Under a good distance measure, the tree generated by MST algorithm has the neighboring components of the same line connected and each line corresponds to a subtree (Fig. 2). Anyway, the branches (paths between terminal and branching nodes) do not correspond to text lines perfectly due to the variability of layout of text lines. We hence use a second-stage clustering algorithm to dynamically cut the edges of the tree into groups corresponding to text lines.

The criterion to select the edge to cut and the criterion to stop cutting (to determine the number of clusters) are important in the second stage. Simply deleting the shortest edge does not work because the edges between different lines (red lines in Fig. 2) are not always longer than those within the same line. The published methods on generic MST-based clustering [12,13] cannot solve well the special problem of text lines grouping. In our approach, we select the edge to cut such that the sum of hypevolumes of clusters is reduced most, and stop clustering when the objective of text lines reaches a maximum.

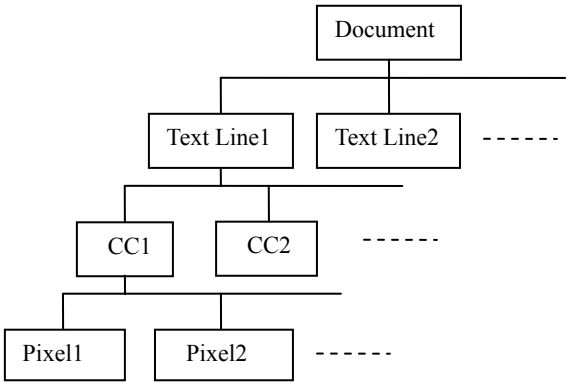


Fig. 1 Structure of a document

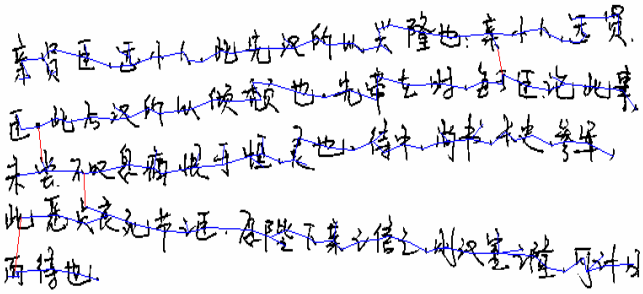


Fig. 2 MST of a document image

3. Algorithm Description

We start with a binary image of handwritten document. First, all the connected components in the document are labeled using a recent fast algorithm based on contour tracing [14]. Small components with few black pixels are considered as noises and are removed. Each component is then viewed as a node in a graph (document graph). Each pair of nodes is linked by an edge with the distance between them as the weight. The distance measure is designed to encourage within-line links and discourage between-line links. From the weighted document graph, a minimum spanning tree (MST) is built using Kruskal's algorithm [15]. In the tree, most edges correspond to within-line links and some correspond to between-line links. In a second stage, we dynamically delete edges (hopefully between-line links) such that the remaining subtrees form text lines.

Since the Kruskal's MST algorithm is well known and can be easily found in the literature, we will not give its details in this paper. Instead, we focus on the distance measure between connected components. For the second stage, we focus on the criterion for selecting edge to cut and the objective function for determining the number of clusters.

3.1. Distance Measure for MST

The definition of the distance between connected components is the key to make the generated minimum

spanning tree have the connected components of the same line in a subtree and those of different lines in different subtrees. We assume the text lines are nearly horizontal, and accordingly, design the distance measure to encourage the connection of horizontally aligned components and discourage the connection of vertically aligned components. Our approach, however, are tolerable to skewed text lines to a large degree. For vertical text lines, we can similarly make the distance measure encourage vertical linking.

A few published works have investigated into the distance measure between connected components, but primarily for separating text lines into words [10,16,17]. Some measures, such as minimum run-length (MRL) distance, minimum Euclidean distance and convex hull distance have shown good performance in these works. Among them, the MRL is computationally simple. We combine the MRL and the Euclidean distance between gravity centers (ECC) to make our distance measure. The MRL computes the horizontal run-length between vertically overlapping (horizontally aligned) connected components, wherein the minimum horizontal distance between black runs is taken as the distance measure. The definitions of MRL and ECC are shown in Fig. 3.

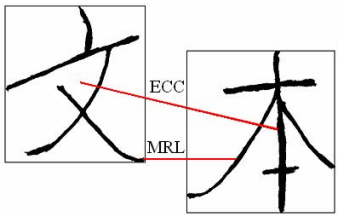


Fig. 3 Example of MRL and ECC

Considering the fact that connected components in the same text line mostly overlap vertically (align horizontally), we combine the two distances MRL and ECC incorporating the degree of vertical overlap. Following [18], the normalized overlap degree is computed as

$$novlp = \frac{1}{2}\left(\frac{ovlp}{h_1} + \frac{ovlp}{h_2}\right) - \frac{vdc}{span},$$

where *ovlp* is the overlapping height of two bounding boxes, h_1 and h_2 are the heights of bounding boxes, *vdc* is the vertical distance between the centers of bounding boxes, and *span* is the spanning height of two bounding boxes. We use $w=(novlp+1)/2$ as a weight for combing MRL and ECC. Denote the MRL and ECC between two connected components CC_i and CC_j as $MRL(i,j)$ and $ECC(i,j)$, the combined distance is computed as

$$D(i,j) = w * MRL(i,j) + (1-w) * ECC(i,j).$$

$D(i,j)$ is used as the weight of edges in the document graph for computing the MST.

3.2. Text Lines Grouping

The above distance measure encourages the connected components in the same text line to be connected in a subtree, but some components from different lines are also connected. To detect and delete such between-line edges is important to partition the document into text lines. The between-line edges are not obvious because their lengths (distances between the linked components) are not necessarily longer than the within-line edge lengths. Fortunately, a criterion based on hypervolume can deal with this problem well [19]. We use the sum of hypervolumes of the clusters of connected components for evaluating the partition:

$$F_v = \sum_{i=1}^{k} [\det(C_i)]^{1/2},$$

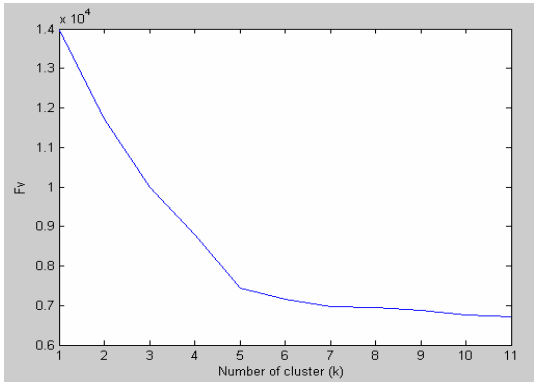
where $\det(C_i)$ denotes the determinant of the covariance matrix C_i of cluster i, which is computed from the constituent black pixels of the connected components in the cluster.

Initially, all the components in the MST are considered as a single cluster, and every edge is deleted tentatively to split the cluster into two clusters. The reduction from the F_v measure of the original cluster and the F_v measure of the split new clusters is used to evaluate the edge. The edge with the most reduction of F_v measure is selected and deleted such that the total F_v measure of the document is minimized. The selection and deletion of edges is repeated until the number of clusters reaches an appropriate value.

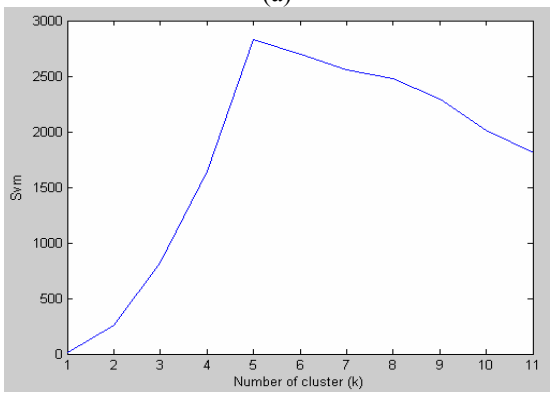
To determine the number of clusters is not trial since we do not know a priori the number of text lines in the document. The F_v measure cannot evaluate the number of clusters since it always decreases as the number of clusters increases. Fortunately, it is reasonable to assume rectangular shapes for the text lines (if a text line is curvilinear, it can be divided it into sublines that are approximately straight). We conjecture that when the number of clusters (partitioned text lines) is appropriate, a measure of the straightness of the text lines reaches a maximum. We compute the total straightness measure as

$$S_{vm} = \sum_{i=1}^{k} \frac{\lambda_{i1}}{\lambda_{i2}},$$

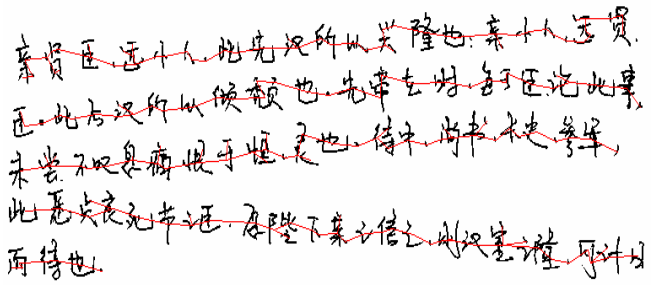
where k is the number of clusters, λ_{i1} and λ_{i2} are the eigenvectors of the covariance matrix of each cluster and $\lambda_{i1} \geq \lambda_{i2}$.



(a)



(b)



(c)

Fig. 4 Criteria for the document in Fig. 2.

(a) F_V as a function of the number of clusters; (b) S_{VM} as a function of number of clusters;

(c) The grouped text lines.

Our experiments demonstrate that the S_{vm} measure performs well in finding the number of clusters: the number of maximum S_{vm} fits well the actual number of text lines. An example is shown in Fig. 4. By iteratively deleting edges according to F_v, the total F_V measure and S_{vm} measure with increasing number of clusters are shown in Fig. 4 (a) and (b), respectively. We can see that $k = 5$, corresponding to the maximum of S_{vm}, gives a preferable partition of text lines.

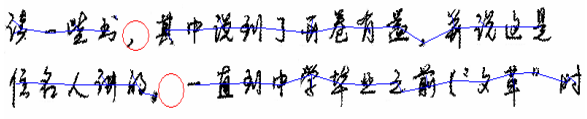
4. Experimental result

To evaluate the performance of our method, we have collected a small database of 20 handwritten documents from various sources, which contain 320 text lines in total. The documents were digitized using a scanner at a resolution of 300dpi. The proposed algorithm has been implemented in C++ codes on a personal computer of Pentium 4-3.6 GHz. We processed the document images to partition the connected components into text lines.

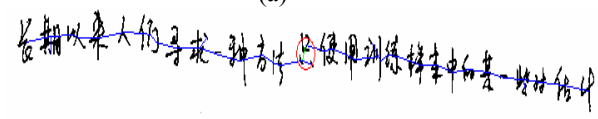
A real text line is considered to be correctly detected when its connected components are contained in a single cluster. We define two types of errors: (1) Error line splitting (ELS): a real text line is split into two or more lines (corresponding to multiple clusters); (2) Error line merging (ELM): two or more real text lines are merged into a single cluster. The percentages of correct detection and error are obtained by dividing the number of correct detection and errors by the total number of real text lines.

On the 20 handwritten documents with 320 text lines, we achieved a correct rate of line detection 88.4%. The rate of ELS is 7.81% and the rate of ELM is 3.43%.

We observed that the errors of line splitting (ELS) are caused by two reasons: (1) Apparently large blanks between the connected components within a text line, such as those in Fig. 5(a); (2) Curved text lines, such as that in Fig. 5(b). Nevertheless, these errors can be easily rectified in post-processing.



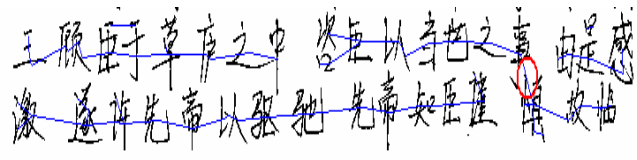
(a)



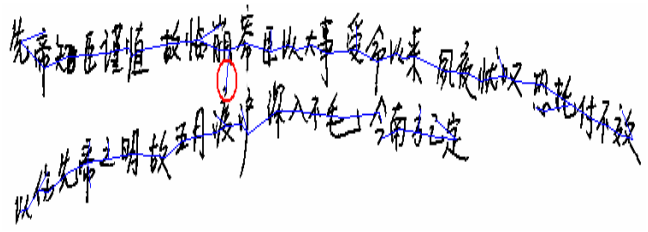
(b)

Fig. 5 Two causes of ELS

Our experiments observed two types of error line merging (ELM): (1) Two or more lines are merged because the edge between lines is not detected to be deleted, as shown in Fig. 6(a); (2) The small component in a line is mis-connected to another line, as shown in Fig. 6(b).



(a)



(b)

Fig. 6 Two types of ELM

Overall, our proposed method performs well on handwritten documents with multi-skewed and curved text lines. An example of fully correct line extraction by the proposed method is shown in Fig. 7. The algorithm is somewhat computationally demanding, however, because it involves repeated criterion computation on the large number of connected components or edges. The overall processing speed, about 300 connected components per second, is nevertheless acceptable.

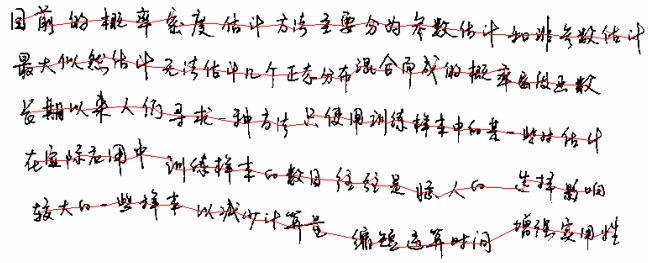


Fig. 7 An example of fully correct line extraction

A common database is not available to compare the performance of our method with previous methods. A recent work reported correct detection rates of 90.34% and 91.44% for Bengali and English documents, respectively [20]. However, the performance of the method in [20] is sensitive to two artificially tuned parameters, whereas our method is totally parameter-free. The text lines detected by our method can be improved by heuristic post-processing.

Our proposed method is not dependent on the language and content of document. Here we give an example to show that our method also works well on English document. The handwritten English handwritten document in Fig. 8 is clustered into text lines by our method with only three errors of line splitting (labeled by blue circles), which can be easily rectified by simple post-processing.

Chinese documents are characteristic of large number of connected components because many Chinese characters consist of multiple components. In cursive English scripts, a word is often written in one connected component. For Chinese documents, merging some components locally before MST clustering may help improve both the speech and the accuracy of text line extraction

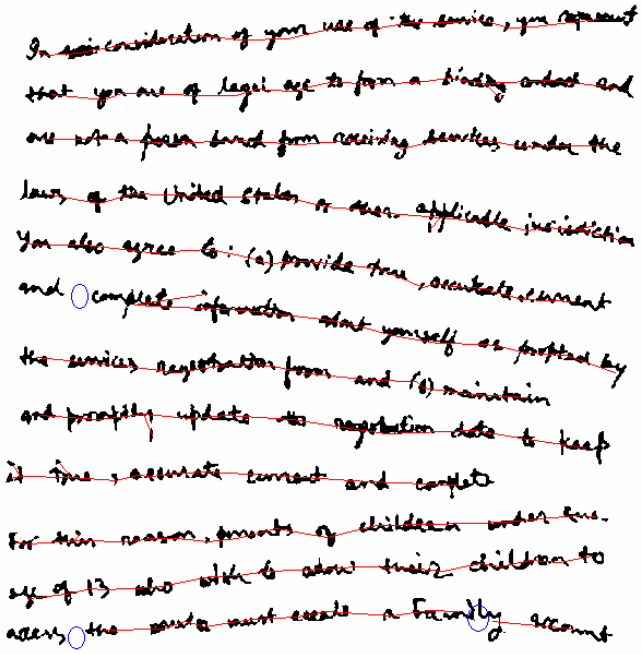


Fig. 8 An example of text line extraction in handwritten English document

5. Conclusions and future work

We propose a new method for text line extraction from unconstrained handwritten documents based on minimum spanning tree (MST) clustering. The connected components of the document are grouped into a tree by MST algorithm, then selected edges are dynamically deleted such that the remaining subtrees correspond to text lines. We design appropriately the distance measure between connected components, the criteria for selecting edge to delete and for determining the number of clusters. Experiments on handwritten Chinese documents demonstrate the promise of our method.

In our future works, we will evaluate the performance of text line extraction on a larger database of handwritten document images, improve the computation speed by pre-processing of connected components and fine implementation of clustering algorithm, and improve the line detection accuracy by post-processing using heuristic rules.

Acknowledgements

This work is supported by the National Natural Science Foundation of China (NSFC) under grants no.60543004 and no.60121302.

References

[1] L Likforman-Sulem, A. Zahour, and B Taconet, Text Line Segmentation of Historical Documents: A Survey, International Journal of Document Analysis and Recognition, Vol. 9, 123-138, 2007

[2] R. Manmatha and J. L. Rothfeder, A Scale Space Approach for Automatically Segmenting Words from Historical handwritten Documents, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.27, No.8, 1212-1225, 2005

[3] U. Pal and S. Datta, Segmentation of Bangla unconstrained handwritten text, Proc. 7th International Conference on Document Analysis and Recognition, Vol.2, 1128-1132, 2003

[4] M. Arivazhagan, H. Srinivasan, and S. N. Srihari, A Statistical Approach to Handwritten Line Segmentation, Document Recognition and Retrieval XIV, 6500T-1 to 6500T-11, 2007

[5] Z. Shi, S. Setlur, and V. Govindaraju, Text extraction from Gray Scale Historical Document Image Using Adaptive Local Connectivity Map, Proc. 8th International Conference on Document Analysis and Recognition, Vol.2, 794-798, 2005

[6] Y. Pu and Z. Shi, A Natural Learning Algorithm Based on Hough Transform for Text Lines Extraction in Handwritten Document, Proc. 6th International Workshop on Frontiers in Handwriting Recognition, 637-646, 1998

[7] L Likforman-Sulem, A. Hanimyan, and C. Faure, A Hough Based Algorithm for Extracting Text Lines in Handwritten Documents, Proc. 3th International Conference on Document Analysis and Recognition, 774-777, 1995

[8] G. Louloudis, B. Gatos, I. Pratikakis, and K. Halatis, A Block-Based Hough Transform Mapping for Text Line Detection in Handwritten Document, Proc. 10th International Workshop on Frontiers in Handwriting Recognition, 515-520, 2006

[9] L O'Gorman, The Document Spectrum for Page Layout Analysis, IEEE Trans, Pattern Analysis and Machine Intelligence, Vol.15, No.11, 1162-1173, 1993

[10] I.S.I. Abuhaiba, S. Datta, and M.J.J. Holt, Line Extraction and Stroke Ordering of Text Pages, Proc. 3th International Conference on Document Analysis and Recognition, Vol.1, 390-393, 1995

[11] A. Simon, J.-C. Pret, and A.P. Johnson, A Fast Algorithm for Bottom-Up Document Layout Analysis, IEEE Trans, Pattern Analysis and Machine Intelligence, Vol.19, No.3, 273-277, 1997

[12] Li Yujian, A Clustering Algorithm Based on Maximal θ-Distance Subtrees, Pattern Recognition, Vol. 40, 1425-1431, 2006

[13] Y. He and L. Chen, A Threshold Criterion, Auto-Detection and Its Use in MST-Based Clustering, Intelligence Data Analysis, Vol.9, 253-271, 2005

[14] F. Chang, C.-J. Chen, and C.-J. Lu, A Linear-Time Component-Labeling Algorithm Using Contour Tracing Technique, Computer Vision and Image Understanding, Vol.93, 206-220, 2004

[15] A.V. Aho, J.E. Hopcroft, and J.D. Ullman, Data Structures and Algorithms, Reading, Mass, Addison-Wesley, 1983

[16] G. Seni and E. Cohen, External Word Segmentation of off-line Handwritten Text Lines, Pattern Recognition, Vol.17, 41-52, 1994

[17] S.H. Kim, S. Jeong, G.-S. Lee, and Ching Y. Suen, Word Segmentation in Handwritten Korean Text Lines Based on Gap Clustering Technique, Proc. 6th ICDAR, 189-193, 2001

[18] C.-L. Liu, M. Koga, and H. Fujisawa, Lexicon-Driven Segmentation and Recognition of Handwritten Character Strings for Japanese Address Reading, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.24, 1425-1437, 2002

[19] I. Gath and A. B. Genv, Unsupervised optimal Fuzzy Clustering, Vol. 11, 773-781, 1989

[20] S. Basu, C. Chaudhuri, M. Kundu, M. Nasipuri, and D.K. Basu, Text line extraction from multi-skewd handwritten documents, Pattern Recognition, Vol. 40, 1825-1839, 2007