Expert Systems With Applications 64 (2016) 352–364

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Recovery of drawing order from multi-stroke English handwritten images based on graph models and ambiguous zone analysis

Minh Dinh, Hyung-Jeong Yang*, Guee-Sang Lee, Soo-Hyung Kim, Luu-Ngoc Do

School of Electronic and Computer Science, Chonnam National University, 300 Yongbong-dong, Buk-gu, Gwangju 500-757, South Korea

A R T I C L E I N F O

Article history:
Received 21 August 2015
Revised 24 March 2016
Accepted 2 August 2016
Available online 3 August 2016

Keywords:
Multi-stroke handwriting
Static handwritten image
Drawing order
Trajectory recovery
Smoothest path

A B S T R A C T

Recovery of the drawing order of strokes in a handwritten image can be seen as searching for the smoothest path for each stroke on an undirected graph that is constructed from the skeleton of the handwritten image. However, this requires correcting for separating strokes, and detecting starting points. Moreover, ambiguousness at junction points increases the complexity of finding the smoothest paths. In order to resolve these issues, an effective approach that can simultaneously detect the points to separate strokes and find the optimal path for each stroke is proposed. To reduce the complexity of the problem, the skeleton graph of the handwritten image is used, and touching characters or crossing strokes are separated. Touches or crossings of stroke parts at ambiguous zones are detected and the smoothness values are adjusted to improve the accuracy. The greedy algorithm and Dijkstra'salgorithm with a well-defined function of smoothness are applied in searching the optimal path. The nature of the recovery is increased when the optimal path is split into many strokes by using the curvatures of the edges, the un-smoothness between edges and the appearance of double-traced edges. Finally, pixel sequences of strokes are extracted and ordered by using rules of handwriting. The effectiveness of the proposed method is demonstrated through low error rates of pixel sequence comparison and high accuracy of on-line recognition.

1. Introduction

Recovering dynamic information from static word images has a large number of potential applications such as writer identification, signature verification, handwriting recognition, etc. (Nguyen & Blumenstein, 2010). Offline handwriting has only a connected mass of pixels in a static 2D image and its time information is unknown. The time information is the order of pixel sequences or stroke segments drawn continuously to form characters (Berthold, 1982; Simon, 1992; Tappert, Suen, & Wakahara, 1988). If this time information is recovered, this bitmap characters are able to be transformed into digital ink data which are usually generated by using on-line drawing or writing devices. Therefore, many applications with camera captured documents from notebooks or white boards after recovering trajectories can be feasible to be edited for further usages.

Recovering information of the order from a static image is very difficult because this is an ill-posed problem (Qiao, Nishiara, & Yasuhara, 2006). It means that there are many possible solutions, particularly in the cases where two parts of a stroke or two different strokes touch or cross each other. The only thing that can be relied on is human nature. Naturally, people always do things in a way that saves energy (Jaeger, 1998). Thus, recovering the time information is equivalent to finding the order of the segments that would minimize the energy that was used to draw them as movements of a human. The energy-spending depends on the smoothness of the stroke, especially at junction points. The smoother, the more energy is saved. This is our fundamental principle for resolving the problem.

Even though this is a difficult problem, there are many approaches that have been proposed in recent years but the results are still not ideal (Nguyen & Blumenstein, 2010). Trajectory recovery techniques found in the literature are categorized as being either contour-based approaches or skeleton-based approaches. Techniques based on contours, such as Baati, Charfi, Alimi, and Ennaji (2005); Lallican, Viard-Gaudin, and Knerr (2000); Plamondon and Privitera (1999); Steinherz, Doermann, Rivlin, and Intrator (2009) have the common problem of being complex and slow in computational time. Steinherz et al. (2009) perform a deep investigation on offline loops. They propose many models for different types of loops and perform a good analysis based on the contour of the handwriting to improve the loop investigation. However, they do not present experiments that demonstrate

* Corresponding author.
E-mail addresses: minh.dinh.1086@gmail.com (M. Dinh), hjyang@jnu.ac.kr (H.-J. Yang), gslee@jnu.ac.kr (G.-S. Lee), shkim@jnu.ac.kr (S.-H. Kim), doluungoc@gmail.com (L.-N. Do).

http://dx.doi.org/10.1016/j.eswa.2016.08.017
0957-4174/

the direct impact of this investigation for handwriting recognition. Therefore, advantages of their work are not clear in practice.

On the other hand, techniques based on skeletons, such as Kato and Yasuhara (2000); Qiao and Yasuhara (2006); Qiao et al., (2006); Rousseau, Camillerapp, and Anquetil (2006) give comparable results and faster than those based on contours. Kato and Yasuhara (2000) propose a two-phase method that consists of globally analyzing the skeleton graph and determining the types of edges at first, and then tracing the graph from the start vertex. Even though this method gives a high performance, it is limited with many constraints: the script must be single stroke and a clear start point and end point should be presented. These constraints are not practical since handwritings usually include many strokes, such as in the character ‘t’, or a hidden start point, such as in the character ‘d’.

Qiao and Yasuhara (2006) address the recovery problem so as to find the smoothest path to cover all the edges in the graph representation of a handwritten image. They introduce the direction context to calculate the smoothness between edges and the smoothest path is found by solving the optimal Euler path. However, this work is also on single-stroke handwriting. Rousseau et al. (2006) give an approach that proposes several starting and ending points by using handwriting knowledge. Then, several paths are generated and the best one is chosen. Their approach can process multi-stroke letters and they give a good test methodology by using an online recognition system. However, their research is just on letters not words where letters can be connected as one stroke or touch together.

In another work, Qiao et al. (2006) propose a sophisticated 3-phase approach within the framework of the Edge Continuity Relation (ECR). First, they obtain possible ECRs at an even-degree node by using a neural network for the node of degree four and by introducing certain assumptions for the nodes of degree higher than four. Second, they identify double-traced lines by employing maximum weighted matching and transform the problem of obtaining possible ECRs at an odd-degree node to that at an even-degree node. Finally, they find the smoothest path on a graph model. Even though they show a good result of accuracy, their work is only on single-stroke handwriting. In the most recent, Phan, Na, Kim, Lee, and Yang (2015) propose a new approach of the polygonal contour skeletonization of the handwritten image to improve the result of the thinning process. Then, a graph model is applied to search the optimal path with an assumption that there is no double trace. This assumption is not genuine in practice, given that the double trace is a common way to save the energy of handwriting.

In order to tackle these issues, a new four-stage approach that is effective for separating and recovering the drawing order of strokes is proposed in this paper. In the first stage, the skeleton of the handwritten image is extracted. Then, the skeleton graph in which its nodes represent end points and junction points, and its edges are segments connecting nodes is constructed. Nodes of spurious segments are merged, touching characters are separated, and crossing strokes are also split. In the second stage, the smoothest path is searched on the line graph of the skeleton graph by using greedy algorithm (Cormen, Leiserson, Rivest, & Stein, 2009b). In order to search this path, some potential starting nodes in the line graph, that are correspondingly the potential starting edges in the skeleton graph, are proposed. A minimum (smoothest) path is then searched for each of those and the best one is chosen. Touches and crossings of stroke parts at ambiguous zones are detected and smoothness values are adjusted to improve the accuracy of searching. In the third stage, redundant segments (edges) in this path are detected and removed by using rules. Strokes are then separated based on the curvature of edges, the appearance of double-traced edges and the un-smoothness of two continuous edges in the path. Finally, pixel sequences of strokes are extracted by going along the paths and are ordered as written by the human.

The proposed method has the advantages that include multi stroke processing, graph model, ambiguous zone analysis and short processing time. Correctly splitting strokes for a handwritten word by merging and splitting in the proposed method increases the nature of the recovery result. The application of the graph model with the simple greedy algorithm and Dijstra’s algorithm to search the optimal path avoids the NP problem and has the short processing time. The smoothest pre-Hamiltonian path that allows the appearance of doubled-traced segments is found instead of the Euler path that the doubled-traced segments have to be treated first. The support of ambiguous zone analysis boosts the quality of the search result of the optimal path. Considering double-traced segments, many candidates of the starting edge and ambiguous zone analysis make the result of the proposed method more similar to the original trajectory than other methods. We conducted experiments with various datasets comparing to conventional methods.

Section 2 of this paper explains the details of the proposed method which uses a graph model with ambiguous zone analysis in four stages such as building skeleton graphs, searching for the smoothest path, separating the smoothest path, and ordering pixels. Section 3 presents experiments with single and multi-stroke images and discusses the results. Finally, conclusions are offered in Section 4.

2. Proposed method

In order to resolve the problem of recovering the drawing order in an English handwritten word image, a novel graph model and an ambiguous zone analysis are proposed as in the flow-chart in Fig. 1. The first stage is to build a skeleton graph. In the second stage, the smoothest path is found. The third stage separates the smoothest path into many strokes. The last stage extracts the pixel sequences of the strokes and orders them.

2.1. Building a skeleton graph

In this section we introduce a skeleton graph which represents end and junction points as nodes and segments connecting those points as edges. Spurious segments that are unwanted results of skeletonization are detected. Their adjacent nodes are grouped into new nodes to improve the accuracy of smoothness estimation. Then, touching characters and crossing strokes are detected and separated by using knowledge of bridges and terminal-edges. The smoothness between two adjacent edges that is used to search the smoothest path is estimated by a function based on the angle between two edges.

2.1.1. Extracting nodes and edges

From a 2D handwritten binarized image, such as capital character *“A”* shown in Fig. 2(a), at first, its skeleton is extracted, as shown in Fig. 2(b). Skeletons of characters are images of strokes thinned to one-pixel width. The skeleton graph is built with nodes and edges; nodes are terminal pixels or joint pixels and edges represent a continuous sequence of pixels connecting two nodes. In the skeleton with one-pixel width, a terminal pixel can be easily detected as a black pixel with only one 8-connected neighbor and a joint pixel is defined if there are three or more neighbors for a pixel. Then, from each node, its adjacent edges are extracted by going through (black) pixel by pixel until meeting another node.

The skeleton from the thinning process often includes artifacts or spurious outputs which are unwanted. The thinning procedure iteratively deletes black pixels in the original image until