A biologically inspired approach for recovering the trajectory of off-line handwriting

Rosa Senatore (rsenatore@nitesrl.com)
University of Salerno: Universita degli Studi di Salerno https://orcid.org/0000-0003-0858-3397
Adolfo Santoro
Natural Intelligent Technologies
Antonio Parziale
University of Salerno: Universita degli Studi di Salerno
Angelo Marcelli
University of Salerno: Universita degli Studi di Salerno

Research Article

Keywords: Handwriting trajectory recovery, Multi-stroke handwriting, Handwriting learning, Graph-based approach, Off-line handwriting recognition

Posted Date: July 1st, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1443088/v1

A biologically inspired approach for recovering the trajectory of off-line handwriting

Rosa Senatore[1,2*], Adolfo Santoro[1], Antonio Parziale[2] and Angelo Marcelli[2]

[1] Natural Intelligent Technologies, Ltd, Piazza Vittorio Emanuele, 10, Penta, Fisciano, 84084, Italy.
[2]Department of Electrical and Information Engineering and Applied Mathematics, Università degli Studi di Salerno, Via Giovanni Paolo II, 132, Fisciano, 84084, Italy .

*Corresponding author(s). E-mail(s): rsenatore@nitesrl.com;
Contributing authors: asantoro@nitesrl.com; anparziale@unisa.it; amarcelli@unisa.it;

Abstract

Background: Reconstructing the trajectory from the static image of handwritten ink traces is useful in many practical applications envisaging handwriting analysis and recognition from off-line data, as it allows to use methods, algorithms and tools that deal with on-line data, achieving better results than those achieved on off-line data. **Methods:** In this work we addressed the trajectory recovery by proposing an approach inspired by the processes involved in human learning of motor skills to perform voluntary and complex movements. As humans learn motor skills by a trial-and-error process driven by the performance and the consumption of metabolic energy, our approach generates a trajectory and estimates the consumption of metabolic energy needed to execute it, and in case it is deemed too energy demanding, a new one is generated and its energy consumption is evaluated. Eventually the one corresponding to the minimum energy consumption among the extant ones is selected as the actual one.

Results and Conclusions: The effectiveness of the proposed approach has been quantitatively and extensively evaluated on a large and publicly available dataset, containing multi-stroke words. The experimental results show that our approach outperforms the existing ones in terms of Root Mean Square Error and Dynamic Time Warping distance between the recovered trajectories and the actual ones. Furthermore, an on-line recognition system provided with the trajectory recovered from off-line samples showed an overall reduction of about **1%** with respect to the recognition rate achieved by the system when provided with on-line data.

Keywords: Handwriting trajectory recovery, Multi-stroke handwriting, Handwriting learning, Graph-based approach, Off-line handwriting recognition

1 Introduction

Knowledge of the dynamic information associated to the static images of handwritten traces is useful in many practical applications such as handwriting recognition, writer identification, signature verification and handwriting analysis for diagnostic purposes. On-line handwriting systems (which exploit both static and dynamic information) reach better performance than off-line systems (which exploit only the static information). However, in a real scenario, most of the available data consists of handwriting drawn on sheets of paper. Recovering the trajectory from static images is therefore crucial for improving the performance of applications dealing with this type of data. Several approaches have been proposed for retrieving the pen tip trajectory from the static image of handwriting (for a review see Nguyen and Blumenstein (2010); Noubigh and Kheralla (2017)), but it still remains a challenging and open issue. Indeed, recovering the trajectory from the ink trace seems an impossible task when the pen-tip is lifted from the paper, so that such missing information must be inferred from the existing trace. To fill this gap, we believe that may be beneficial to consider how humans learn and execute handwriting movements.

Handwriting is produced through a perception/action cycle involving brain areas implicated in attentive vision, learning, control and coordination of several motor subsystems Grossberg and Paine (2000); Senatore and Marcelli (2012). It results from the concatenation of elementary movements, or strokes, aimed at reaching time-varying spatial target, requiring the activation of two neuro-muscular systems (agonist/antagonist) with two synchronous input commands Plamondon and Djioua (2006). It has also been shown that handwriting movements are encoded in the brain through a limb-independent coding, which is mainly related to the spatial representation of the trajectory Rijntjes et al (1999). These studies suggest that producing a handwritten word requires to retrieve the desired trajectory and translating it into a series of muscular contractions to be sent to the specific effector for producing the articular movement required to execute the sequence of strokes composing the desired movement. Furthermore, according to Sparrow and Newell (1998), movement patterns emerge as a function of the organism propensity to conserve metabolic energy, so that humans, and other organisms as well, learn a coordination and control solution that is effective in producing the desired movement and economical in terms of metabolic energy consumption.

Based on these studies we have developed a novel approach for handwriting trajectory recovering that is inspired by human learning of handwriting. The proposed method generates an initial trajectory, which is then decomposed into a sequence of strokes and whose energy consumption is eventually estimated. If the estimated trajectory contains added movements, and/or the sequence of strokes is not deemed as the least expensive, and there are alternatives, new trajectories are generated and their energy consumption estimated. The final trajectory is chosen as the least energy demanding among the extant ones.

The paper is organized as follows: Section 2 briefly describes related work, Section 3 presents the overall architecture of the proposed method, describes the implementation of the ancillary preprocessing module, with special emphasis on the ad-hoc procedure used to obtain a faithful representation of the shape of the ink traces from their skeletons, and finally provides a detailed description of each of the other modules, explaining the rationale for their design and the actual implementation. Section 4 describes the experiments we have designed and performed to assess the quality of the recovered trajectory, as well as to compare the end-to-end evaluation of a fully developed, publicly available, on-line handwriting recognition system when the input is the on-line trajectory vs the trajectory recovered by our method from the off-line handwriting, while Section 5 discusses the experimental results. Eventually, the Conclusions paragraph summarizes the main features of the proposed method, its performance when used for off-line handwriting recognition by means of an off-the-shelf on-line recognition system, and eventually outlines possible further developments to deal with non-Latin alphabets.

2 Related Work

Several approaches have been proposed for recovering the dynamics from static handwritten traces, some exploiting global or local characteristics of the ink shape, (Boccignone et al (1993); Doermann and Rosenfeld (1995); Plamondon and Privitera (1999)), others mapping somehow the ink to a graph and reformulate the trajectory recovery as a graph traversal one (Bunke et al (1997); Dinh et al (2016); Jager (1996); Kato and Yasuhara (2000); Nagoya and Fujioka (2012); Phan et al (2015); Qiao et al (2006); Diaz et al (2021)). Hidden Markov Model (Viard-Gaudin et al (2005)), genetic algorithms (Elbaati et al (2009)) and convolutional neural networks (Zhao et al (2018); Elbaati et al (2019);

Kumarbhunia et al (2018); Nguyen et al (2020); Rabhi et al (2021); Sumi et al (2019); Zhang et al (2019)) have also been proposed.

Due to the differences in writing styles and missing information when the pen is lifted from the paper (in-air trajectory), obtained results are still not ideal, although several methods have been proposed for tackling this problem (Nguyen and Blumenstein (2010); Noubigh and Kheralla (2017)).

In Doermann and Rosenfeld (1995) the authors provided a taxonomy of the features to be analyzed for the recovery of the dynamic information, but their work did not report an experimental validation. Other proposed techniques have been evaluated only on single-stroke traces, i.e. traces performed without lifting the pen from the paper(Kato and Yasuhara (2000); Nagoya and Fujioka (2011); Qiao et al (2006)), or on single characters (Jager (1996); Pervouchine et al (2005); Zhao et al (2018); Rousseau et al (2005); Elbaati et al (2009); Kumarbhunia et al (2018); Nguyen et al (2020); Rabhi et al (2021); Sumi et al (2019); Zhang et al (2019)). Other approaches, applied on multi-stroke traces, have been evaluated just on few examples (Boccignone et al (1993); Kato and Yasuhara (1999); Nagoya and Fujioka (2012); Yu Qiao and Yasuhara (2006)). In the last years, methods based on deep learning approaches have been proposed for recovering the writing order Elbaati et al (2019); Kumarbhunia et al (2018); Nguyen et al (2020); Sumi et al (2019); Zhang et al (2019); Zhao et al (2018) and both the writing order of the trajectory and the pen velocity profile Rabhi et al (2021). These methods, however, require a lot of data for training the neural networks and, up to now, they have been tested only on single characters and digits Elbaati et al (2019); Kumarbhunia et al (2018); Rabhi et al (2021); Sumi et al (2019) or multi-stroke symbols like Chinese and Japanese characters Nguyen et al (2020); Zhang et al (2019); Zhao et al (2018), but never on words or signatures.

In Plamondon and Privitera (1999) the authors have exploited the existing knowledge about the cognitive-behavioral processes used by human subjects to recover dynamic information from the image of a handwritten word, and have defined a segmentation system that retrieves the stroke sequence belonging to the word while preserving the original motor-temporal information. The approach is based on human visual system processes and perceptive-motor constraints, but it did not take into account the proprioceptive feedback, responsible of action refinement, that is integrated in our method. Furthermore, the results of the method is performed just by visual inspection.

Overall, the state-of-art reveals that the problem of recovering the writing order from static handwritten images is far from being solved. Methods based on a learning process, as for example the ones using HMM and neural networks, have been applied on single-stroke traces, as for example characters or digits, that are less complex than those one could encounter in a real scenario. Moreover, the method evaluation has been carried out with different approaches as for example by visual inspection Plamondon and Privitera (1999) or by comparing the recognition rates obtained by training a classifier with the actual and reconstructed trajectories Viard-Gaudin et al (2005) without providing any comparison between the actual and reconstructed trajectories. Eventually, in some cases performance are evaluated on a proprietary dataset Bunke et al (1997).

3 The method

The rationale behind the method arises from the analysis of the processes occurring during the learning of novel motor plan aimed at performing handwriting (Senatore and Marcelli (2012)). Recovering handwriting temporal information corresponds to estimate the sequence of target points reached by

the writer for drawing the desired ink trace. According to the sequence of target points, the writer performs the motor plan, that is the sequence of elementary movements for drawing the ink trace. Supposing that the handwriting is performed by skilled writers, the motor plan underlying the performed trajectory should be characterized by low energy consumption (Sparrow and Newell (1998)).

In seeking to emulate the biological processes involved in handwriting generation, we developed an approach made up of three modules connected through a loop: 1) the unfolding module generates a trajectory by estimating the sequence of points reached by the pen tip while producing the ink trace and eventually adding in-air movements, if needed to generate a plausible trajectory, i.e a trajectory from the starting point of the first movement until the ending point of the last movement executed to write the whole word; 2) the segmentation module estimates the target points and the corresponding sequence of movements, i.e. the strokes, drawn for producing the trajectory, and 3) the energy evaluation module computes the energy expenditure associated with the estimated motor plan and sends this information back to the unfolding module. This information is then exploited for changing the sequence of estimated target points in order to obtain lower energy consumption. The loop is repeated until the possible trajectories have been evaluated, and the one characterized by the lowest energy consumption is selected as the correct one. The scheme of the whole process is reported in figure 1. The proposed approach is applied to the static image of the word after a series of standard preprocessing steps, briefly described in the following subsection.

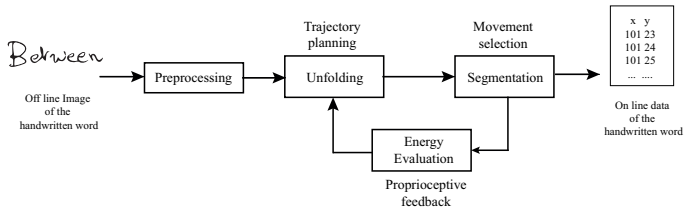

Fig. 1 Scheme of the proposed approach for trajectory retrieval.

3.1 The Preprocessing module

The preprocessing module aims at obtaining a one-dimensional curve representing the ink trace from the original grayscale image of it (figure 2). The image is converted into a binary one and the flaws due to the acquisition noise, such as salt and pepper noise, or small holes that can generate spurious loops in the skeleton, which can possibly lead to thinning errors, are removed using standard correction methods (Marcelli et al (2017)).

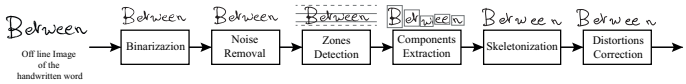

Fig. 2 Preprocessing Steps

After noise removal the image of the handwritten word is divided into zones (central, top and bottom). The boundaries are determined through the horizontal projections of black ink in the image: the central zone represents the zone in which is concentrated the largest amount of black pixels and its position is used for determining the position of the top and bottom zone, according to the amount of black pixels that are above and below the central zone. In order to reduce the complexity of the ink trace provided to the unfolding algorithm, the connected components are extracted. A connected component is defined as the set of foreground pixels having at least one foreground pixels in their 8-neighborhood. Therefore, they are naturally associated with ink traces produced without lifting the pen from the paper. However, when different

parts of the ink trace interact, handwriting embedding both on-paper and on-air movements may originate a single connected component, as it may be the case when the horizontal bar of the letter 't' is drawn after the vertical bar and an-on air movement occurs between the two on-paper ones. Anyway, the goal of the connected components extraction step is to reduce the number of pen lifts the unfolding algorithm has to deal with, since this reduces the amount of missing information, related to the on-air movement, that the unfolding algorithm has to infer.

The ink trace of each connected component is represented in the binary digital image as a ribbon, whose thickness depends on several aspects such as the scanner resolution, the writing instrument, the paper, and the pressure exerted through the writing instrument on the paper. We performed skeletonization for transforming the ribbon into a 8-connencted one pixel wide line, in order to eliminate the variability introduced by the just mentioned factors. The points belonging to the skeleton were classified as: a) end point (EP) if the corresponding pixel has only one adjacent pixel; b) normal point (NP), if the corresponding pixel has only two other adjacent pixels; c) branch points (BP) if the corresponding pixel has more than two other adjacent pixels. A trait of skeleton between two BPs or between a EP and a BP is named branch. A branch is also characterized for its length L and the thickness T of the ink zone it originates from. The length L is equal to the number of points in the branch while the thickness T is the radius of the maximal digital disk centered on each skeletal pixel.

Zones characterized by irregular ink trace thickness, due to either uneven ink deposition or binarization noise, can generate spurious skeleton branches. As spurious branches are associated to small protrusions appearing on the trace main body, and considering that we are dealing with ribbons, i.e. object

whose local thickness is much smaller than the length, the length and the local thickness at the tip (of the spurious branches) are expected to be smaller than the local thickness (T) of the ink trace in correspondence of the protrusion. If we consider the ratio ρ between the difference $\Delta = T_{BP} - T_{EP}$ and the length L of the branch, $\rho = 0$ for a branch whose local thickness at the tip of the protrusion is the same as the local thickness of the trace and therefore cannot be considered as representing a protrusion. On the other hand, a protrusion made by just one pixel, would generate a branch for whom $\Delta = T_{BP} - 1$ and $L = T_{BP} + 1$, so that $\rho \geq 0.5$, and becomes larger and larger as T_{BP} becomes bigger. Consequently, we consider that the skeleton branches starting from a BP and ending in an EP, for whom $\rho \geq 0.5$ are spurious branches and remove them from the skeleton.

As widely stated in the literature, and regardless of the skeletonization algorithm, the skeleton may present geometrical distortions in correspondence of the regions wherein the trace intersects itself. Such distortions, in turn, may induce errors in deciding how to cross the intersection and, eventually, in recovering the trajectory. To deal with such distortions, once the spurious branches are removed, a polygonal approximation of each skeletal branch is performed through a split-and-merge algorithm, wherein the vertices of the polyline are constrained to be the extremes of the sequences of collinear pixels (i.e. pixels aligned along the same rectilinear line) and must include the EPs and the BPs of the skeleton.

In order to illustrate the criteria we have adopted to eliminate the distortions introduced by the skeltonization process, we distinguish three types of distortions, called V-shape, T-shape and X-shape, shown in figure 3.

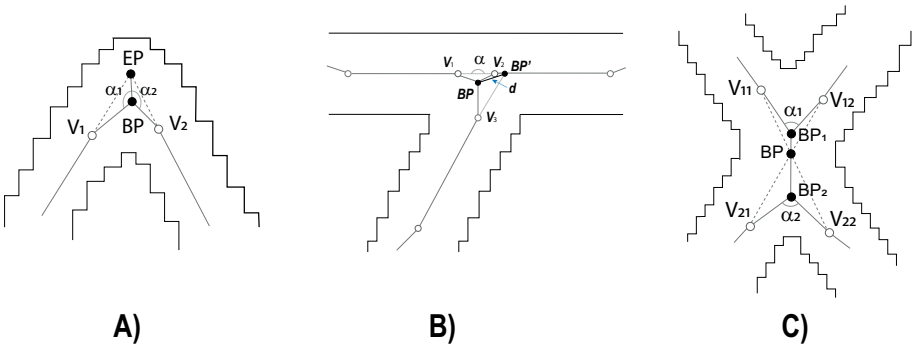

Fig. 3 Distortions of the ink trace skeleton and their correction. External contours represent the original ink trace.

A V-shape distortion occurs when the ink trace presents a sharp curve. In such a case, the polyline of the trace is that of figure 3A, because of the triangular shape of ink trace in correspondence of the sharp curve. Such distortions, thus, can be identified by analyzing the polyline segments joining in a BP that connects three segments. If the following conditions hold:

1. one of the segments is delimited by an EP;
2. the thickness at the BP is bigger than that of the EP;
3. the angles α_1 and α_2 between the segment delimited by the EP and the two other segments joining in the BP have similar amplitude, that is $|\alpha_1 - \alpha_2| \leq \theta_1$;

the spurious segment between the BP and EP is removed and the two segments are connected so that they join in the EP, as represented by the dotted lines in figure 3A.

A T-shape distortion occurs when two parts of ink trace touch each other producing a T-shaped pattern. In these cases, the polyline of the touching region is shown in figure 3B: it presents three segments joining in a BP whose corresponding ink thickness is bigger than those at the other vertices of the

segments connected to the BP, and the segments of the part of ink trace that can be interpreted as the top bar of the T bend toward the vertical bar instead of being parallel to the contour of the top bar. In figure 3B the distorted segments are represented by the continuous lines starting from BP and ending in V_1, V_2 and V_3. The point BP' is the intersection point between the line connecting V_1 and V_2 and the line obtained by extending the segments ending in V_3, α is the angle measured at the intersection obtained by extending the segments passing from V_1 and V_2, and d is the distance between BP and BP'. These distortions are characterized by the following conditions:

1. the ink thickness at the BP is bigger than that of both the vertices V_1 and V_2;
2. the segments connecting V_1 and V_2 with BP form a smooth curve, that is $\alpha \geq \theta_2$;
3. the branch point BP' is centrally locate within the part of the trace corresponding to the top bar, that is $d < T_{BP}$;

When the previous conditions hold, the branch point BP is replaced by BP', and the segments starting from V_1, V_2 and V_3 are connected at BP', as represented by dotted lines in figure 3B.

An X-shape distortion occurs when two part of the ink trace cross each other. In this case, the polyline of the crossing region presents five segments that connect in two BPs, instead of four segments joining in a single BP, as expected (figure 3C). These distortions are detected by considering each pair of BPs connected by a segment of the polyline, from each of which three rectilinear segments start. If the following conditions hold:

1. the ink thickness at the BP_i is bigger than those the vertices V_{ij};
2. the segments connecting V_{ij} with BP_i do not form a smooth curve, that is $\alpha_i \leq \theta_3$;

the five segments are replaced by four segments, connecting the vertices V_{ij} with the new branch points BP' that is located at the intersection between the segments connecting the vertices V_{11} with V_{22} and V_{12} with V_{21}, as represented by dotted lines in figure 3C.

3.2 The Unfolding Module

The unfolding module recovers the ordered sequence of points followed by the pen-tip to produce the ink trace. This goal is achieved by reformulating the problem in terms of a graph search problem (Cordella et al (2010)). The graph is constructed on the basis of the topological properties of the polyline associated to the skeleton ink, considering the EPs and BPs as nodes and the polylines of the skeleton branches connected to them as arcs. Two features characterize each node of the graph: the type (EP or BP) and the degree, i.e. the number of arcs connected to the node. A node whose degree is an odd number is named odd node, otherwise it is named even node.

The unfolding module performs three steps:

1. estimation of Starting and Ending points of the trajectory according to the topological features of the nodes of the graph;
2. transformation of the original graph into a semi-Eulerian one (i.e. a graph in which all the nodes have an even degree), by adding connections between the odd nodes, with the exception of the nodes corresponding to the Starting and Ending points of the trajectory;
3. estimation of the actual trajectory followed by the writer, by applying the Fleury's algorithm, modified on the basis of handwriting generation criteria, for finding the path that originates in the node corresponding to the Starting point (hereinafter referred as source node) and reaches the Ending point of

the trajectory (referred as destination node) crossing all the nodes, while minimizing the number of nodes crossed more than once.

Estimation of the Starting and Ending points of the trajectory

Each node of the graph is characterized by two features: the type (EP or BP) and the degree, i.e. the number of arcs connected to the node.

The estimation of the source/destination nodes derives from the following observations on the handwriting kinematics:

1. EPs represent either the positions where the pen-tip touches or leaves the writing surface or the position when the handwriting movement stops and move in the opposite direction along the same trajectory, as in case of a retracing. Therefore they can be the source or the destination node of a path;
2. Starting/Ending points may be hidden in the trace when the different parts of the trace interact, and therefore BPs can be source/destination nodes of a path as well;
3. in case of language based on Latin alphabet:
   - handwriting proceeds from left to right;
   - handwriting movements whose traces extend between the top/central zone and the central/bottom zone are mostly performed moving downwards;
   - loops are drawn clockwise or counterclockwise depending on whether they are located in the bottom zone or above, respectively.

Consequently, the criteria for selecting the source/destination node take into account the relative position of the nodes and their degree. According to the characteristics of the ink trace, three conditions can be observed, as shown in figure 4:

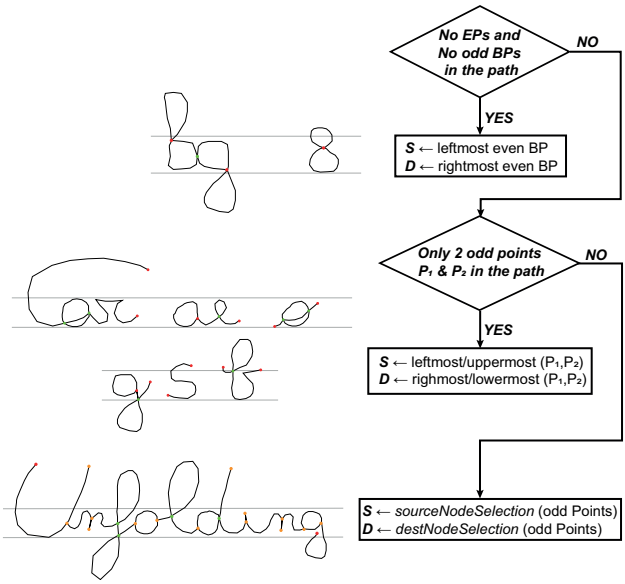

Fig. 4 Source and Destination node selection. Orange points represent odd degree nodes, green points represent even degree nodes and red points represent the estimated source/destination nodes. Grey lines represent the center zone of the handwritten sample.

1. the graph contains only even degree BPs. The leftmost BP is selected as source node and the rightmost BP is selected as destination node. An example of this condition is represented by the shape of the digit '8'.
2. the graph contains only two odd degree nodes and any number of even degree nodes. According to the orientation of the connected component represented by the graph (horizontally or vertically extended) and the distance (horizontal or vertical, respectively) between the nodes, the leftmost or the uppermost of the two points is selected as source, and the rightmost or the lowermost of the two points is selected as destination, as depicted in the flowchart of figure 5.
3. the graph contains more than two odd nodes and any number of even nodes. Source and destination nodes are selected according to the flowchart reported in figure 6. The source/destination node is selected among the two leftmost/rightmost odd nodes of the graph depending on the type of both

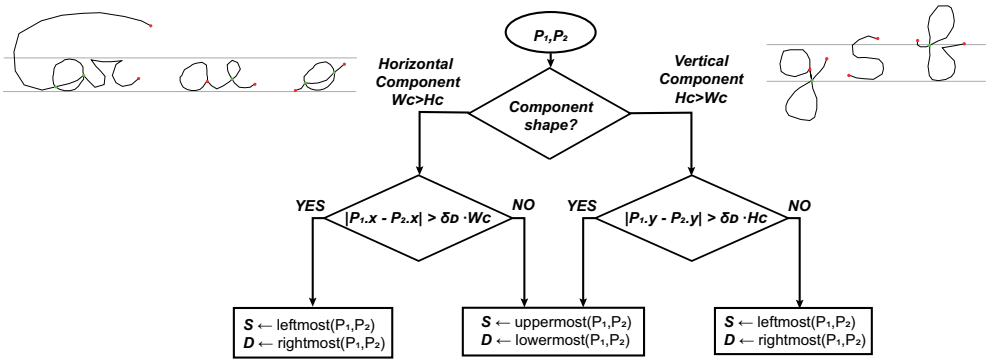

Fig. 5 Source and Destination nodes selection: criteria for graphs containing only two odd nodes and any number of even degree nodes. Green points represent even degree nodes and red points represent the source/destination nodes. Grey lines represent the center zone of the handwritten sample

the candidate node (EP or BP) and the nodes connected to it. In particular, the function *selectSrc(NodeSet)* selects as source destination node the leftmost or the uppermost among those belonging to the NodeSet depending on whether the horizontal component of the distance between the two nodes is bigger than the vertical one or vice versa. Similarly, the function *selectDst(NodeSet)* selects as destination node the rightmost or the lowermost among those belonging to the NodeSet depending on whether the horizontal component of the distance between the two nodes is bigger than the vertical one or vice versa.

A particular case occurs when the ink trace contains only normal points, as it happens with a skeleton of a trace representing a closed loop (i.e. an isolated letter o). In this case the topleft polygonal approximation point is selected as both source and destination node and the corresponding graph degenerates into a single node.

Construction of the semi-Eulerian graph

In line with the hypothesis of minimum energy consumption, the path that corresponds to the trajectory followed by the pen-tip should be the one that

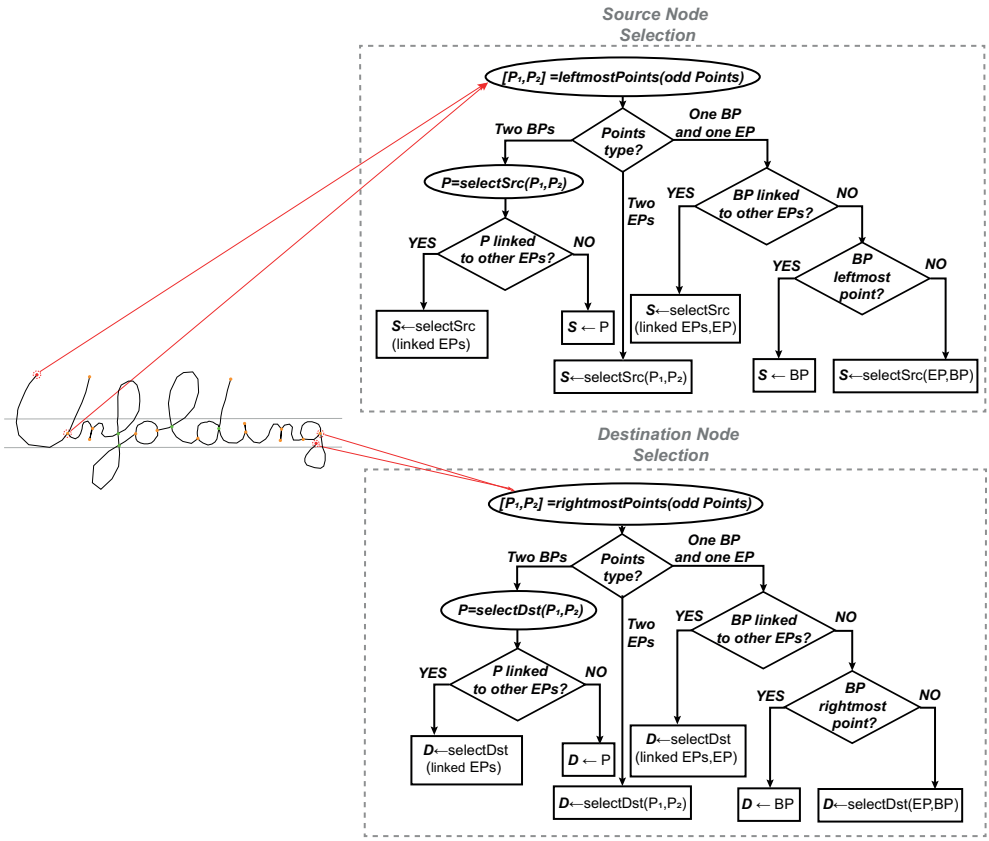

Fig. 6 Source and Destination nodes selection: Criteria for graphs containing more than two odd nodes and any number of even degree nodes. Orange points represent odd degree nodes, green points represent even degree nodes and red points represent the nodes selected as starting and ending nodes according to the criteria. Grey lines represent the center zone of the handwritten sample.

crosses all the nodes and, at the same time, minimizes the number of nodes crossed more than once. In order to ensure the existence of such a path, the original graph has to be transformed into a semi-Eulerian one. This is achieved by adding further connections between odd degree nodes, referred in the literature as Chinese arcs, so that the resulting graph contains only even degrees nodes.

In our case, additional arcs may represent pen-up movements (i.e. in-air trajectories of the pen-tip) if they connect two nodes that are not already

connected, or retracing movements (i.e. a trajectory drawn two times in opposite directions) if they connect two nodes already connected. Pen-up movements involve greater energy consumption than retracing movements, since they require to lift the pen-tip from the paper, perform an in-air trajectory and reset the motor parameters for starting a new movement when the pen-tip touches the paper. Consequently, adopted criteria tends to introduce additional connections mainly between already connected nodes than between unconnected nodes. Chinese arcs between odd nodes are included in the graph according to the nodes type (EP or BP), the topology of branches connecting them, and the euclidean distance between them, as shown in the flowchart of figure 7.

Starting from the set of odd degree nodes, the leftmost node n_i is chosen (in case two nodes are located at the same horizontal coordinate, the uppermost is chosen):

- If n_i is an EP, a Chinese arc to the leftmost node n_{i+1}, chosen among the remaining odd nodes, is added. This criterion arises from the considerations that handwriting is performed from the left to the right and that an EP usually represents either a point from which a retracing movement is started or a point where the pen-tip is lifted from the paper.
- If n_i is a BP, among the odd nodes connected to it, those connected through a "loop" or a "bridge" are not considered for being further connected. This criterion follows from the particular topology of loops and bridges. Here we defined as "loop" a path that starts and ends in the same BP (i.e. the upper part of the trajectory of the letter e and l), and as "bridge" an arc whose removal disconnects the graph (i.e. the ligature between two characters). Being very unlikely that a loop or a bridge be drawn twice when writing

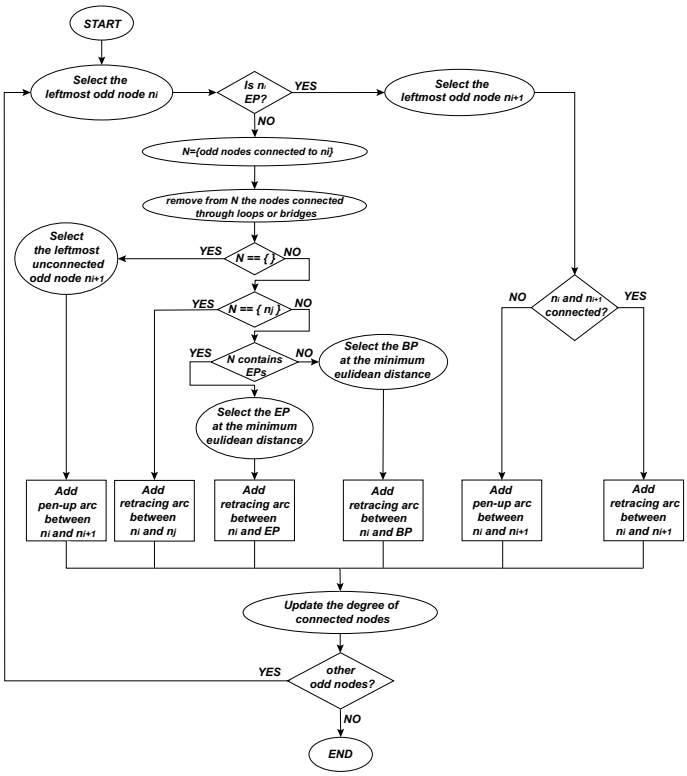

Fig. 7 Criteria for adding further arcs for trasforming the original graph into a semi-Eulerian one.

the word, we exclude the presence of a retracing involving these two types of branches. Once the odd nodes connected through loops or bridge were excluded from being further connected, three cases can be observed:

- No already connected nodes are available for being connected. In this case a pen-up is added between the node n_i and the leftmost unconnected odd node n_{i+1}.
- Only one connected node is available, and is connected to n_i with a retracing arc.
- Many odd nodes already connected are available for being connected. Whether this set contains EPs, the closest EP is connected to n_i, otherwise a retracing is added between the closest BP and n_i.

This procedure is performed until all the odd nodes are connected and transformed into even ones and the graph becomes semi-Eulerian.

Each arc (or a group of arcs) of the semi-Eulerian graph is then labeled according to the topological characteristics of the trajectory associated to it. In particular, we define:

- *loop*: arc starting and ending in the same BP node;
- *two-way ring*: a pair of arcs starting and ending in the same nodes;
- *three-way ring*: three arcs starting and ending in the same nodes;
- *bridge*: arc whose removal disconnects the graph.
- *simple arc*: arc that does not belongs to the previous classes;

Figure 8 shows the ink trace of the word *Unfolding*, containing an example for each arch type. Dotted orange lines represent the added retracing arcs, whereas the dotted blue line represents an added pen-up arc. For the sake of clarity, the bridges, are left unlabeled.

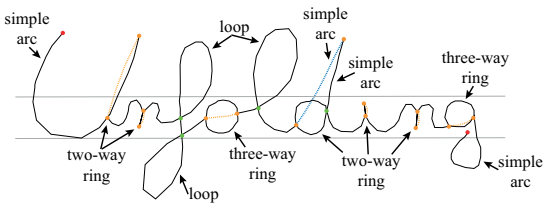

Fig. 8 Ink trace of the word *Unfolding*, to which estimated retracing and pen-up movements were added. Dotted orange lines represent the retracing arcs, whereas the dotted blue line represents a pen-up arc. Unlabeled arcs represent the bridges. Orange points represent odd degree nodes that are transformed into even ones by adding further connections, green points represent even degree nodes, red points represent the source and destination nodes. Grey lines represent the center zone of the handwritten sample.

Trajectory Estimation

The trajectory followed by the writer for drawing the ink traces is estimated by applying a modified version of the Fleury's algorithm to the semi-Eulerian graph.

Starting from the source node the algorithm selects, among all the possible ones, the next arc to traverse and the way to traverse it (in case of loops or rings), according to the following order: 1) loops; 2) two-way rings; 3) three-way rings; 4) simple arcs; 5) bridges, as reported in figure 9. Once the selected arc is traversed and the ending point of the arc is reached, the criteria are applied on the arrival node and the procedure is repeated until the destination node is reached and no more arcs have to be traversed.

Loops ad rings are traversed according to their position within the ink trace. If their center of mass is located below the center zone of the word (e.g. the lower part of the letters *g* and *f*), they are traversed clockwise. Conversely, if their center of mass is located above or within the center zone of the word (e.g. the upper part of the letters *b* and *f*, or the body of the letters *a*, *e*, *o*), they are traversed counterclockwise. In particular, the external ring of the three-way rings is traversed first (clockwise or counterclockwise, according to the position of the center of mass) and then the internal arc is traversed.

Whenever neither loops nor rings start from the analyzed node and there are many arcs (either simple arcs or bridges) starting from it, the selection of the next arc to traverse is restricted to the simple arcs. As a bridge represents an arc whose removal disconnects the graph, traversing it when other simple arcs are connected to the analyzed node would prevent traversing those arcs. The simple arc to traverse is chosen according to a) the angle it forms with the arc previously traversed, and b) the length of the arc, i.e. the Euclidean distance between the node delimiting the arc. Selection is performed through the function *selectArc(arcs)* according to the following criteria: 1) if one or more arcs form an angle smaller than θ_4, the arc that forms the smallest angle is traversed; 2) if no arcs form an angle smaller than θ_4, the shortest arc is traversed; in case there are two or more arcs of the same length, the arc that

forms the smallest angle is traversed.

Figure 10 shows the application of the described algorithm to the word "Unfolding".

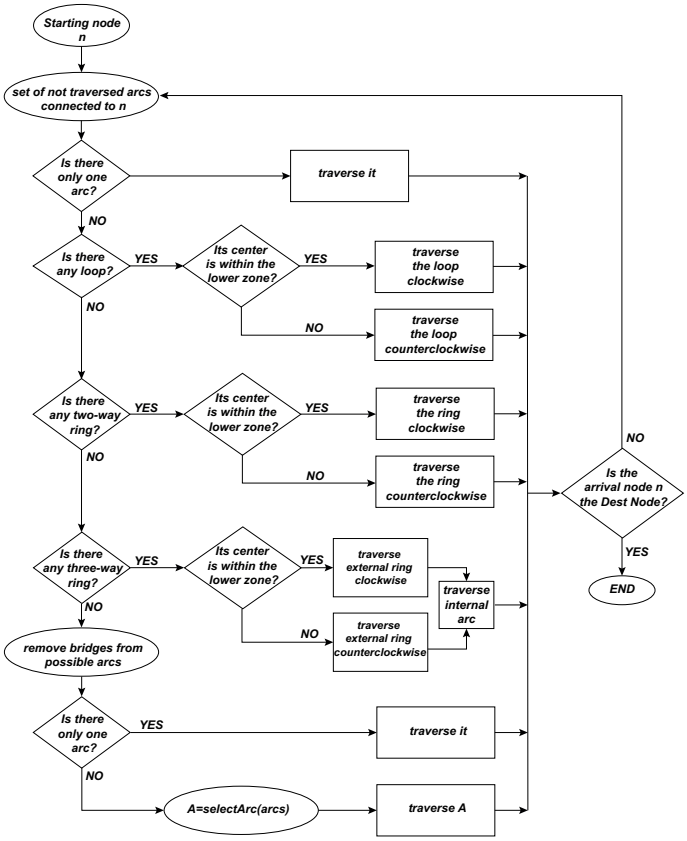

Fig. 9 Trajectory estimation. Application of Fleury's algorithm, modified according to handwriting generation criteria.

3.3 The Segmentation module

Segmentation is performed through the method presented in De Stefano et al (2004). From the ordered sequence of points composing the ink trace, as provided by the unfolding algorithm, the segmentation module identifies the elementary movements performed by the writers, i.e. the strokes. Each stroke

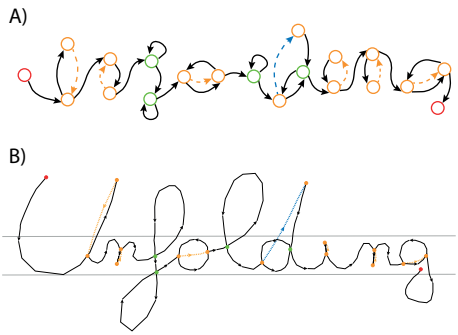

Fig. 10 Pen-tip trajectory estimation of the word "Unfolding". A) Semi-Eulerian graph and B) ink trace associated to the word. Red nodes(points) represent the starting and ending nodes(points) of the graph path (pen-tip trajectory). Orange nodes(points) represent odd degree nodes(points) and green nodes(points) represent even degree nodes(points). The orange arcs of the graph (branches of the ink trace) represent retracing and the blue one represent pen-up.

is delimited by segmentation points, that are located by applying a decomposition method based on the concept of saliency, used to model attentive vision in primate visual system. In brief, the method builds a saliency map according to the multiscale representation of the unfolded ink trace. The obtained map is then used to highlight the regions of the image representing salient information (the "foci of attention") that, in this case, are the points of the ink trace in which significant curvature variations are recorded at different scales. These points represent the segmentation points and can be considered a good approximation of the target points selected by the writer for producing the ink trace (for more details see (De Stefano et al (2004))).

3.4 The Energy Evaluation module

Over the years, various psychological theories have embraced notions of *economy*, *efficiency*, or *least effort* to explain how complex movement sequences are organized and modified. Sparrow and Newell reviewed these theories and derived a conceptual framework that explains how movement economy is regulated. They conclude that "In all human motor skills, there is, therefore, an advantage to be gained by those who are most economical in their movements"

(Sparrow and Newell (1998)). More recently, by using a neurocomputational model of the spinal cord, which includes connections with the motor cortex, the cerebellum and the proprioceptive feedback provided by tendons and Golgi organs, we have showed that movements aimed at reaching a spatial target are learned by a trial-and-error process, driven by two goals: reaching the desired position and minimizing the metabolic energy consumption to perform the movement (Parziale et al (2020)). We have also shown that when subjects are required to produce an handwritten sample by following a trajectory that is different from the one they normally draw, the trajectory is made of a larger number of strokes than the usual one (Senatore and Marcelli (2019)). As each stroke is generated by a motor command (either by the motor cortex or by the spinal cord) and each motor command is executed by activating the necessary muscles, the more commands are generated the higher is the energy needed to perform them. Thus, to recover the trajectory followed by a skilled writer, and considering that the metabolic energy consumption is not available nor can be computed from the available data, we have addressed the problem by searching within the sequence of strokes whether or not there are subsequences whose execution is metabolic energy demanding. If this is the case, we searched for alternative ways of unfolding the ink trace that lead to trajectories made of sequence of strokes that are less voracious in terms of metabolic energy.

To estimate if and to which extent the estimated trajectory requires more metabolic energy with respect to the actual one (i.e. the one actually followed by the pen tip), we associated to the trajectory a score, called *energy score*, computed as the sum of the value of a parameter we called *additional energy* and the values of the elements of an *energy vector*. Additional energy takes into account the length and the type of the added connections (i.e. retracing and pen-up arcs), whereas the energy vector, which has as many elements as

the total number of EPs and BPs of the unfolded skeleton, takes into account the proximity between segmentation points and EPs/BPs. The former is used to estimate the extra energy requested to produce the trajectory, while the latter is used to decide whether an alternative trajectory, i.e. a new sequence of strokes, should be generated through the unfolding and segmentation modules and its energy score evaluated, in search for the cheapest one. In the following paragraphs, we describe how the energy consumption is estimated and how such an estimate is used to verify whether or not a path satisfies the conditions to be considered as a likely trajectory or, in the negative, how alternative sequences of strokes can be generated for evaluation.

Energy Expenditure Estimation

Additional energy value is computed by summing up the energy consumption for retracing and pen-ups, as they have been added for transforming the original graph into a semi-Eulerian one, and therefore may not be part of the actual trajectory. It follows that, in case of a trajectory like the one of the character "8", i.e. a trajectory whose original graph contains only even degree nodes, its additional energy value is zero. In the case of a retracing arc, its energy consumption is given by the number of the pixels of the skeletal branches associated to it, while in case of a pen-up arc, its energy consumption is given by twice the Euclidean distance between the starting and the ending points of the movement. These choices follow from the observation that in the handwriting a retracing is performed by executing a single command, and therefore the energy to produce the movement is the energy for drawing the corresponding stroke: the longer the stroke, the bigger the metabolic energy consumption to draw it. On the contrary, a pen-up movement is much more expensive because it requires to move the pen tip while contrasting the gravity. As both actions are in place during the movement, the energy consumption is twice the length

of the stroke. Moreover, because pen-up movements aim at reaching the target in the most efficient (i.e. less energy consuming) way, it is expected (and confirmed by experiments with subjects performing reaching experiment) that such movements exhibit a trajectory which is a good approximation of the segment connecting the starting and ending points of the movement(Plamondon and Djioua (2006)). Consequently, their length is represented by the Euclidean distance between the starting and the ending points.

The energy vector is computed by performing a local analysis on the BPs and EPs of the segmented ink trace, evaluating the presence of segmentation points in the proximity of them. The reason behind this choice arises from the characteristics of the ink traces in correspondence of stroke junctions and stroke crossings. A junction appears in the ink trace because the starting/ending points of one stroke is within the body of the other (for example, the intersection between the vertical and horizontal strokes of the letter "T"). Consequently, one segmentation point can appear in the proximity of the BP created at the junction, since a stroke starts or ends in that point. On the contrary, a crossing appears in the ink trace when two strokes intersect each other and none of them starts or ends within the intersecting region (for example, the intersection of the two strokes forming the letter "x"). Consequently, segmentation points cannot appear in the proximity of the BP, since none of the intersecting strokes starts or ends in that point. The energy vector values are therefore computed by analyzing the skeletal pixels located at a distance smaller then the ink thickness at the EP or BP under consideration. Whenever it happens that one of the skeletal pixels is a segmentation point, the energy consumption associated to the EP/BP is the number of pixels of the longest between the two strokes joining in it, since the longer the strokes the bigger the effects of the segmentation error.

Changes in the estimated trajectory according to the estimated energy

The energy vector is exploited for deciding if and how to change the choices related to the selected source and destination nodes, the additional connecting arcs and the way the graph (i.e. the ink trace) is traversed. Estimated trajectory is changed according to the following steps, as illustrate in figure 11:

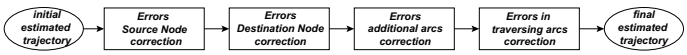

Fig. 11 Changes of the estimated trajectory according to the values of the additional energy value and the energy vector.

1. the energy vector is analyzed in search for a non-zero value in correspondence of the EP or BP that is the starting point of the trajectory. If such an element is found and there are alternatives nodes to be considered as source node, the current source node is removed from the possible candidates, a new source node is selected, as described in the paragraph 2.2, a new trajectory is generated by the unfolding module, segmented in strokes and its energy consumption estimated by computing the energy score. Eventually, between the two trajectories, the one with the lowest energy score is selected for the next step, as illustrated in figure 12;
2. the energy vector of the selected trajectory is analyzed in search for a non-zero values in correspondence of the EP or BP that is the ending point of the trajectory. If such an element is found and there are alternatives nodes to be considered as destination node, the current destination node is removed from the possible candidates, a new destination node is selected as described in the paragraph 2.2, a new trajectory generated, and its energy

consumption estimated, as illustrated in figure 13. As in the previous case, the trajectory with the lowest energy score is selected for the next step;

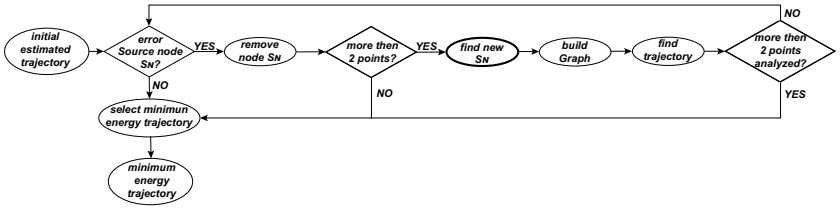

Fig. 12 Selection of the alternative starting point of the trajectory according to the info carried by the energy vector and the value of the energy score.

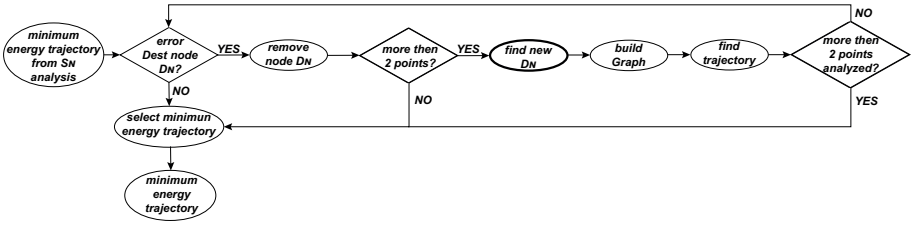

Fig. 13 Selection of the alternative ending point of the trajectory according to the info carried by the energy vector and the value of the energy score.

3. if the selected trajectory contains additional arcs and non-zero values are found in correspondence of the EP/BP representing the starting/ending of the additional arc, the additional arc is removed, other possible combinations for connecting odd nodes are evaluated following the procedure for building the semi-Eulerian graph described in the paragraph 2.2, a new trajectory is generated and its energy consumption estimated, as illustrated in figure 14. This step is iterated until a trajectory is generated whose energy vector elements in correspondence of the starting/ending points of the additional arcs are equal to zero, or all alternatives additional arcs have been evaluated. Therefore, in the worst case, i.e. when all the possible alternatives must be evaluated, the step is repeated n_{IT} times, where n_{IT} depends

on the number of odd nodes n to be connected:

$$n_{IT} = \prod_{i=0}^{\frac{n}{2}-1} (n - 2i - 1)$$

Among the set of estimated trajectories, the one with the lowest energy score is selected for the last step;

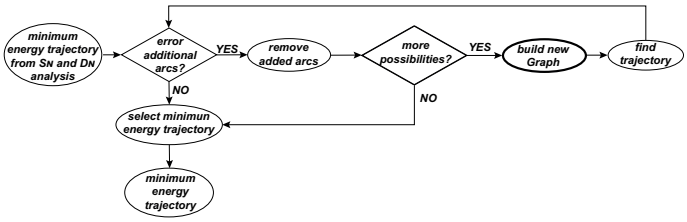

Fig. 14 Changes of the additional arcs for constructing the semi-Eulerian graph according to the info carried by the energy vector.

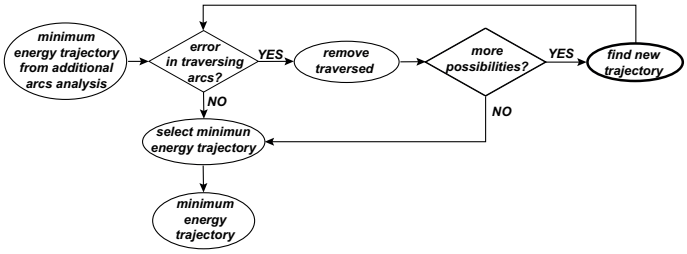

Fig. 15 Changes of the traverse path according to the info carried by the energy vector.

4. the energy vector is analyzed in search for the non-zero value in correspondence of BPs that are not the starting/ending point of the trajectory, and if such elements are found, for each BP another possible choices of coupling the arcs connected to it is made, a new trajectory is generated and its energy consumption evaluated, as illustrated in figure 15. This step is iterated until a trajectory is generated, whose energy vector elements corresponding to the BPs that are non starting/ending points are equal to zero,

which therefore represents the desired one, or all alternative couplings of the arcs connected to the BP have been evaluated. As at each step an alternative is evaluate for each BP, in the worst case, i.e. when all the possible alternatives must be evaluated, the step is repeated $n-1$ times, where n is the largest degree of the BPs under investigation. Among the set of estimated trajectory, the one with the lowest energy score is selected as the final one.

4 Results

The method has been evaluated on datasets extracted from the IRONOFF database, which contains on-line traces (x and y coordinates) and the corresponding off-line images of isolated characters, digits and cursive words written by about 700 different writers (Viard-Gaudin et al (1999)). On-line data were acquired with a spatial resolution of 300 dpi and a sampling rate of 100 points/sec through a Wacom UltraPadA4. Off-line data consists of gray scale TIFF images scanned at 300 dpi.

We perform our experiments on 19,495 handwritten samples (both French and English words), extracted from the entire dataset and representing 168 different words written by 479 different writers. Characteristics of the datasets used for evaluating the method are reported in table 1.

Table 1 Characteristics of the datasets used for evaluating the method

Dataset name	Number of writers	Number of unique words	Mean number of characters per word	Number of samples
Dataset C	157	50	7.4	7686
Dataset E	130	26	8.6	3333
Dataset F	104	26	6.3	2689
Dataset G	88	66	3.4	5787

The quality of the reconstructed trajectories has been evaluated in terms of Root Mean Square Error (RMSE) and Dynamic Time Warping (DTW)-based error between the sequence of points of the actual trajectory and the sequence of points of the recovered trajectory. Table 2 reports the mean of both measures on each data set and the average across them. It is noteworthy

Table 2 Mean RMSE and DTW-based error measures obtained by comparing the estimated and the actual trajectory on the samples of each dataset and over all of them.

Dataset	All Words		
	Number of unique words	**RMSE**	**DTW-based error**
Dataset C	50	538.77	310.74
Dataset E	26	596.75	326.78
Dataset F	26	505.60	268.02
Dataset G	66	243.96	138.89
All		471.27	261.10

that French words contain more accents than English words, which introduces one more degree of freedom in the transcription of the word. Indeed, while some people draw the accent while writing the word, others insert it only after the transcription of the entire word. This, in turn, will introduce a penalty in the measured performance, since the proposed method recovers the dynamics of the ink trace according to the position of the ink components, and therefore tends to reconstruct the trajectory as the accents were drawn while writing the word. In order to support this assumption, we have reported in Table 3 the performance obtained only on words devoid of accents. They show that in this last case the overall performance improves of 11,55% and of 17,43% in terms of average RMSE and DTW-based error, respectively.

Table 4 reports the comparison between the performance of our method and those reported in two recent works on the dataset F. Phan et al (2015) proposed a new skeletonization method based on polygonal contours and applied

Table 3 Mean RMSE and DTW-based error obtained by comparing the estimated and the actual trajectory for words devoid of accents belonging to the datasets C, E and G and over all of them.

Dataset	No Accented Words		
	number of unique words	**RMSE**	**DTW-based error**
Dataset C	30	476.64	261.44
Dataset E	13	503.55	253.00
Dataset G	65	239.95	134.81
All		406.72	216.44

a greedy algorithm for searching the optimal path on the graph obtained from the skeleton. Dinh et al (2016) performed an ambiguous zone analysis, in which some elaborations were made on the skeleton of the ink trace (by separating touching characters and crossing strokes) for reducing its complexity, and applied the greedy and the Dijkstra algorithm for finding the optimal path in all the detected skeleton components.

Table 4 Mean RMSE and DTW-based error measures obtained by comparing the estimated and the actual trajectory on the samples of the dataset F

Method	**RMSE**	**DTW-based error**
Phan et al. (2015)	2513.57	5345.18
Dinh et al. (2016)	669.03	278.27
Proposed method	505.60	268.02

The performance reported in Table 2, 3 and 4 were obtained by setting the thresholds θ_1, θ_2, θ_3 and θ_4 to the values reported in the column titled *selectedvalue* of Table 5. To assess the robustness of the proposed method

Table 5 Selected values and range of values for evaluating the robustness of the proposed method with respect to the values of the threshold.

Threshold	**selected value**	**Analyzed Range**	**Step (Δ)**
θ_1	20°	$10° - 30°$	5°
θ_2	150°	$140° - 160°$	5°
θ_3	120°	$110° - 130°$	5°
θ_4	30°	$20° - 40°$	5°

with respect to the values of the thresholds, we have changed the value of one

threshold at a time within the range reported in Table 5, while keeping all the other fixed to the selected values, performed the whole process of trajectory recovery and evaluated the RMSE between the recovered trajectory and the actual one on the samples of the dataset F, as before. By using this procedure, we elicited the robustness of the heuristic criteria where the threshold is used. The experimental results, reported in figure 16 show the robustness of the criteria we have defined for detecting and correcting the distortions (where the thresholds θ_1, θ_2 and θ_3 are used) and for traversing junctions or crossings (threshold θ_4).

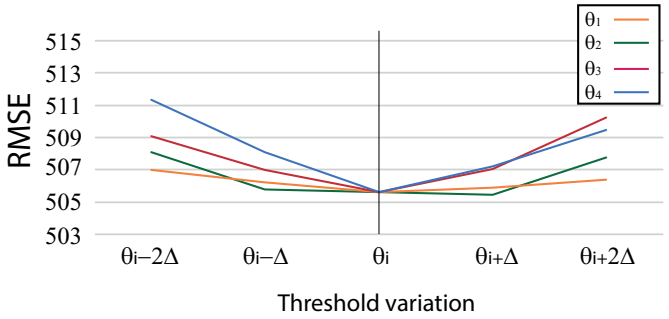

Fig. 16 Mean RMSE values obtained by changing the threshold values within a neighborhood of the selected values. Δ represents the step used for the analysis, as reported in Table 5.

Eventually, we performed a off-line handwriting recognition experiment to assess to which extent our method recovers the actual writing order and how its performance reflects on the overall word recognition rate. For this purpose, we used the on-line recognizer available in the Interactive Ink SDK toolkit provided by MyScript© for on-line recognition, equipped with a dictionary containing as many entries as the number of different words belonging to the each dataset.

We performed the experiment in three different scenarios. In the first scenario (off-line image), the input to the recognizer is the sequence of skeletal points provided by our method, as depicted in figure 1. In order to assess to which extent the loss of performance depends on the trajectory recovery step, we designed a second scenario (ideal skeleton), where we bypassed the preprocessing module and compute the image of the skeleton by connecting the on-line coordinates with a digital line, on which we applied the bwmorph Matlab© function for finding the connected components, which were passed to the unfolding-segmentation loop module. Eventually, in the third scenario (ideal trajectory) we provided the recognizer with the "ideal" trajectory, i.e. the skeleton computed as before with the temporal order obtained from on-line data. Table 6 reports the recognition rates obtained in the three scenarios for each of the dataset, as well as on the whole dataset. As the classifier provided up to 5 possible interpretations for each sample, ranked according to the lexicographic distance between the recognized word and the dictionary entries, the performance is reported by considering the top 1, top 3 and top 5 interpretations provided by the recognizer.

As in the previous case, we have evaluated the robustness of the performance with respect to the values of the thresholds. Figure 17 reports the Top 3 recognition rate values obtained on the dataset F by using the same procedure as the one we have used for the RMSE. Again, the plots confirm the robustness of the recognition rate with respect to the thresholds values.

5 Discussion

We have proposed a novel approach for trajectory reconstruction from offline images of handwriting and have evaluated its effectiveness both quantitatively and extensively on a large and well-known database, consisting on multi-stroke

Table 6 Word recognition rates (%) obtained in the different scenarios.

		Word recognition rate		
		TOP 1	**TOP 3**	**TOP 5**
Dataset C	Off-line image	73.7	84.1	87.1
	Ideal skeleton	84.2	90.8	92.4
	Ideal trajectory	90.4	94.5	95.5
Dataset E	Off-line image	79.1	87.5	90.0
	Ideal skeleton	85.4	91.6	93.7
	Ideal trajectory	88.5	93.2	95.0
Dataset F	Off-line image	81.3	90.9	93.1
	Ideal skeleton	88.9	91.6	93.7
	Ideal trajectory	93.5	96.9	97.8
Dataset G	Off-line image	75.8	85.2	87.3
	Ideal skeleton	84.7	90.4	91.5
	Ideal trajectory	89.9	93.6	94.2
All	Off-line image	76.3	85.9	88.5
	Ideal skeleton	85.2	91.4	92.9
	Ideal trajectory	90.3	94.3	95.3

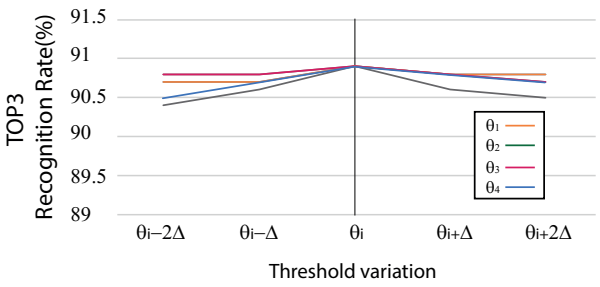

Fig. 17 TOP 3 recognition rate values(%) obtained on the dataset F as the threshold values vary in their respective ranges.Δ represents the step used for the analysis, as reported in table 5.

words. The comparison of the different techniques has been done both in terms of the accuracy of the reconstructed trajectory, as in (Hassaïne et al (2013);

Nguyen and Blumenstein (2010)), and in terms of word recognition rate, as in (Noubigh and Kheralla (2017)). Consequently:

1. we computed the RMSE and the DTW-based error between the actual trajectories of the on-line samples and the trajectories reconstructed from the corresponding off-line samples;
2. we used an on-line classifier for recognizing handwriting samples using the actual (on-line) and reconstructed (from off-line) trajectories and compared the recognition rates.

The comparison of the performance obtained on the dataset F with respect to that obtained on the same dataset through two other approaches (Dinh et al (2016); Phan et al (2015)) shows that our method outperforms the existing ones, allowing to achieve lower RMSE and DTW-based error (Table 4). Indeed, in terms of RMSE, our method achieves a performance improvement of 80% and 25% compared to Phan et al (2015) and Dinh et al (2016), respectively. Likewise, we obtained an improvement of 95% and 4% in terms of DTW-based error.

In order to extensively assess the performance, the method has been applied on the remaining part of the IRONOFF dataset (datasets C, E and G), containing French words. Table 2 shows that the proposed method achieves similar results in terms of performance.

Interestingly, it can be observed that performance depends on words complexity (in terms of number of characters and pen-up). This result was expected, since an increased number of characters composing the word leads to an increased number of EPs and BPs in the skeleton ink. This, in turn, leads to more complex graphs, characterized by a greater number of nodes, possible ways of connections between odd nodes (for obtaining the semi-Eulerian

graph) and possible ways of traversing the graph. Indeed, looking at the average number of characters composing the words contained in each dataset (table 1), the obtained results in terms of RMSE and DTW distance are inversely proportional to the mean number of characters composing the words contained in the dataset. On the contrary, a bigger word complexity can be due to a greater number of pen-ups. An example is the presence of accents in the word. Unlike English words, French words frequently contain accents (acute, grave and circumflex), which introduce another degree of freedom in the transcription of the word, since they can be drawn by the writer either during or at the end of the writing of the word. The proposed method recovers the dynamics according to the position of the ink components, and therefore tends to reconstruct the trajectory as the accents were drawn while writing the word. This, in turn, could negatively affect the performance. Indeed, removing from the evaluation the performance obtained on accented words, (table 3) an increase of performance can be observed (16% and 23% of improvement for datasets C and E, respectively).

In order to show the effectiveness of our method in practical applications, we used an on-line classifier (implemented through the toolkit provided by MyScript©) in three different scenarios. The obtained results show that our method is able to reconstruct the ink trace with reduced loss of performance (table 6). Indeed, by considering the top 5 interpretations when providing the recognizer with the reconstructed ink trace, we observed good recognition performance, reaching 88.5% mean accuracy and a performance loss of 7.1% with respect to the ideal trajectory on average over all the datasets. Looking at the recognition rate by taking into account word complexity (in terms of number of characters composing the word) it can be observed that the performance on datasets composed by long words are better than those on datasets composed

by short words. This result, that seems in contrast to that observed in trajectory recovery performance, is due to the fact that increasing the word length increases the distance between the words of the lexicon, and so it reduces the probability of misclassification.

Eventually, we point out that the reported performance takes into account the entire process performed on the static image of the handwritten word, which includes binarization, component extraction, thinning, skeleton pruning and distortions correction. Consequently, errors in the preprocessing steps may cause a decrease in the performance of the reconstructed ink trace that is not due to the unfolding-segmentation loop. Indeed, we have shown that providing the ideal skeleton (i.e. the static image built from the on-line coordinates) to the unfolding-segmentation loop module, the recognition rate over all the data sets achieved considering the top 5 interpretations reaches 94.3%, with a performance loss of 1.0% with respect to the one achieved with the ideal trajectories.

6 Conclusions

In this work we have addressed the problem of recovering the trajectory from off-line handwritten traces by proposing a method inspired by the processes involved in handwriting generation by humans.

The proposed method exploits criteria obtained from the analysis of physiological process underlying handwriting generation for evaluating the energy consumption associated to a sequence of movements and providing a set of corrective measures on the trajectory associated to it. We have included these modules into a feedback loop, which estimates the trajectory, computes the

corresponding sequence of strokes, evaluates the energy consumption and eventually generates alternatives trajectories, among whom the final one is that produced with the lowest energy consumption.

Experiments performed on the IRONOFF dataset have shown the effectiveness of the proposed approach. In particular, we have proved that our approach outperforms the existing ones in terms of the quality of the reconstructed trajectory and shown its efficacy in practical applications, such as handwriting recognition. We have also shown that a large fraction of the loss of performance is due to the preprocessing steps performed to achieve a line representation of the trajectory, which may introduce (or not remove) artifacts that alter the shape of the line representation with respect to the shape of the actual trajectory.

Future work will be aimed at evaluating the proposed approach in other practical applications, such as writer identification, signature verification, and handwriting analysis. On the other hand, experiments will be done to ascertain if and to which extent, the observations on handwriting kinematics and the heuristic criteria we derived for designing our method for recovery the trajectory of handwriting movements still hold or need to be adapted, and how, in case of languages based on different alphabets.

7 Acknowledgements

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

8 Statements and Declarations

Authors have no financial or non-financial interests that are directly or indirectly related to the work submitted for publication.

The data that support the findings of this study are available on request from C. Viard-Gaudin, P. M. Lallican, S. Knerr and P. Binter, "The IRESTE On/Off (IRONOFF) dual handwriting database,doi:10.1109/ICDAR.1999.791823.

References

Boccignone G, Chianese A, Cordella L, et al (1993) Recovering dynamic information from static handwriting. Pattern Recognit 26(3):409–418. https://doi.org/10.1016/0031-3203(93)90168-V

Bunke H, Ammann R, Kaufmann G, et al (1997) Recovery of temporal information of cursively handwritten words for on-line recognition. In: Proceedings Fourth International Conference on Document Analysis and Recognition, pp 931–935, https://doi.org/10.1109/ICDAR.1997.620647

Cordella LP, De Stefano C, Marcelli A, et al (2010) Writing order recovery from off-line handwriting by graph traversal. In: 2010 20th International Conference on Pattern Recognition, pp 1896–1899

De Stefano C, Guadagno G, Marcelli A (2004) A saliency-based segmentation method for online cursive handwriting. Int J Pattern Recognit and Artif Intell 18(07):1139–1156. https://doi.org/10.1142/S021800140400368X

Diaz M, Crispo G, Parziale A, et al (2021) Writing Order Recovery in Complex and Long Static Handwriting. International Journal of Interactive Multimedia and Artificial Intelligence In Press(In Press). https://doi.org/10.9781/ijimai.2021.04.003

Dinh M, Yang HJ, Lee GS, et al (2016) Recovery of drawing order from multi-stroke english handwritten images based on graph models and ambiguous

zone analysis. Expert Syst Appl 64:352 – 364. https://doi.org/10.1016/j.eswa.2016.08.017

Doermann DS, Rosenfeld A (1995) Recovery of temporal information from static images of handwriting. Int J Comput Vision 15:143 – 164. https://doi.org/10.1007/BF01450853

Elbaati A, Kherallah M, Ennaji A, et al (2009) Temporal order recovery of the scanned handwriting. In: Proceedings International Conference on Document Analysis and Recognition, pp 1116–1120, https://doi.org/10.1109/ICDAR.2009.266

Elbaati A, Hamdi Y, Alimi AM (2019) Handwriting recognition based on temporal order restored by the end-to-end system. In: Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, https://doi.org/10.1109/ICDAR.2019.00199

Grossberg S, Paine RW (2000) A neural model of cortico-cerebellar interactions during attentive imitation and predictive learning of sequential handwriting movements. Neural Networks 13(8-9):999–1046. https://doi.org/10.1016/S0893-6080(00)00065-4

Hassaïne A, Al Maadeed S, Bouridane A (2013) Icdar 2013 competition on handwriting stroke recovery from offline data. In: 2013 12th International Conference on Document Analysis and Recognition, pp 1412–1416

Jager S (1996) Recovering writing traces in off-line handwriting recognition: Using a global optimization technique. In: Proceedings 13th International Conference on Pattern Recognition, pp 150–154

Kato Y, Yasuhara M (1999) Recovery of drawing order from scanned images of multi-stroke handwriting. In: Proceedings of the Fifth International Conference on Document Analysis and Recognition, pp 261–264, https://doi.org/10.1109/ICDAR.1999.791774

Kato Y, Yasuhara M (2000) Recovery of drawing order from single-stroke handwriting images. IEEE Trans Pattern Anal Mach Intell 22(9):938–949. https://doi.org/10.1109/34.877517

Kumarbhunia A, Bhowmick A, Kumar Bhunia A, et al (2018) Handwriting Trajectory Recovery using End-to-End Deep Encoder-Decoder Network. In: Proceedings - International Conference on Pattern Recognition, https://doi.org/10.1109/ICPR.2018.8546093

Marcelli A, Santoro A, De Stefano C, et al (2017) Process of handwriting recognition and related apparatus. US Patent: US9665768, 2017

Nagoya T, Fujioka H (2011) Recovering drawing order from static handwritten images using probabilistic tabu search. In: TENCON 2011 - 2011 IEEE Region 10 Conference, pp 379–384

Nagoya T, Fujioka H (2012) Recovering dynamic stroke information of multi-stroke handwritten characters with complex patterns. In: International Workshop on Frontiers in Handwriting Recognition, pp 722–727, https://doi.org/10.1109/ICFHR.2012.258

Nguyen HT, Nakamura T, Nguyen CT, et al (2020) Online trajectory recovery from offline handwritten Japanese kanji characters of multiple strokes. In: Proceedings - International Conference on Pattern Recognition, https://doi.org/10.1109/ICPR48806.2021.9413294

Nguyen V, Blumenstein M (2010) Techniques for static handwriting trajectory recovery: A survey. In: Proceedings of the 9th IAPR International Workshop on Document Analysis Systems, DAS '10, p 463–470, https://doi.org/10.1145/1815330.1815390

Noubigh Z, Kheralla M (2017) A survey on handwriting recognition based on the trajectory recovery technique. In: 2017 1st International Workshop on Arabic Script Analysis and Recognition (ASAR), pp 69–73

Parziale A, Senatore R, Marcelli A (2020) Exploring speed–accuracy tradeoff in reaching movements: a neurocomputational model. Neural Computing and Applications pp 1–27

Pervouchine V, Leedham G, Melikhov K (2005) Three-stage handwriting stroke extraction method with hidden loop recovery. In: International Conference on Document Analysis and Recognition (ICDAR'05), pp 307–311, https://doi.org/10.1109/ICDAR.2005.241

Phan D, Na IS, Kim SH, et al (2015) Triangulation based skeletonization and trajectory recovery for handwritten character patterns. KSII Trans Internet Inf Syst 9(1):358–377. https://doi.org/10.3837/tiis.2015.01.022

Plamondon R, Djioua M (2006) A multi-level representation paradigm for handwriting stroke generation. Hum Movement Sci 25(4-5):586–607. https://doi.org/10.1016/j.humov.2006.07.004

Plamondon R, Privitera CM (1999) The segmentation of cursive handwriting: an approach based on off-line recovery of the motor-temporal information. IEEE Trans Image Process 8(1):80–91. https://doi.org/10.1109/83.736691

Qiao Y, Nishiara M, Yasuhara M (2006) A framework toward restoration of writing order from single-stroked handwriting image. IEEE Trans Pattern Anal Mach Intell 28(11):1724-1737. https://doi.org/10.1109/TPAMI.2006.216

Rabhi B, Elbaati A, Boubaker H, et al (2021) Multi-lingual character handwriting framework based on an integrated deep learning based sequence-to-sequence attention model. Memetic Computing 13(4). https://doi.org/10.1007/s12293-021-00345-6

Rijntjes M, Dettmers C, Büchel C, et al (1999) A blueprint for movement: Functional and anatomical representations in the human motor system. J Neurosci 19(18):8043–8048. https://doi.org/10.1523/JNEUROSCI.19-18-08043.1999

Rousseau L, Anquetil E, Camillerapp J (2005) Recovery of a drawing order from off-line isolated letters dedicated to on-line recognition. In: Eighth International Conference on Document Analysis and Recognition (ICDAR'05), pp 1121–1125

Senatore R, Marcelli A (2012) A neural scheme for procedural motor learning of handwriting. In: 2012 International Conference on Frontiers in Handwriting Recognition, pp 659–664, https://doi.org/10.1109/ICFHR.2012.160

Senatore R, Marcelli A (2019) A paradigm for emulating the early learning stage of handwriting: Performance comparison between healthy controls and parkinson's disease patients in drawing loop shapes. Hum Movement Sci 65:89–101. https://doi.org/10.1016/j.humov.2018.04.007

Sparrow WA, Newell KM (1998) Metabolic energy expenditure and the regulation of movement economy. Psycho B Rev 5(2):173–196. https://doi.org/10.3758/BF03212943

Sumi T, Iwana BK, Hayashi H, et al (2019) Modality conversion of handwritten patterns by cross variational autoencoders. In: Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, https://doi.org/10.1109/ICDAR.2019.00072

Viard-Gaudin C, Lallican PM, Knerr S, et al (1999) The ireste on/off (ironoff) dual handwriting database. In: Proceedings of the International Conference on Document Analysis and Recognition, pp 455–458, https://doi.org/10.1109/ICDAR.1999.791823

Viard-Gaudin C, Lallican PM, Knerr S (2005) Recognition-directed recovering of temporal information from handwriting images. Pattern Recognit Lett 26(16):2537 – 2548. https://doi.org/10.1016/j.patrec.2005.04.019

Yu Qiao, Yasuhara M (2006) Recover writing trajectory from multiple stroked image using bidirectional dynamic search. In: 18th International Conference on Pattern Recognition (ICPR'06), pp 970–973

Zhang R, Chen J, Yang M (2019) Drawing Order Recovery based on deep learning. In: 11th International Conference on Advanced Computational Intelligence, ICACI 2019, https://doi.org/10.1109/ICACI.2019.8778533

Zhao B, Yang M, Tao J (2018) Pen tip motion prediction for handwriting drawing order recovery using deep neural network. In: 2018 24th International Conference on Pattern Recognition (ICPR), pp 704–709