Extraction of strokes in handwritten characters

Eric L’Homer*

CMLA-DIAM, ENS de Cachan, 94235 Cachan Cedex, France

Received 26 August 1998; received in revised form 6 April 1999; accepted 6 April 1999

Abstract

Among the many handwritten character recognition algorithms that have been proposed in the past few years, few of them use models which are able to simulate handwriting. This can be explained by the fact that simulation models require the estimation of strokes starting from statistic images of letters, while crossing and overlapping strokes make this estimation difficult. The approach we suggest is to efficiently deal with crossing areas and overlaps using parametric representations of lines and thickness of stroke: a probabilistic model of strokes is described to extract non-overlapping strokes of the image. A bayesian approach using a statistical study and a model of stroke crossing is described that optimizes the reconstruction of crossings and permits to characterize image of letters by robust graphs of curves. © 2000 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Handwritten character recognition; Thinning algorithms; Graphs of strokes; Stroke crossing; Stroke path detection

1. Introduction

An algorithm for reconstructing lines in cursive handwriting using a parametric model is presented. Thinning algorithms are used by handwritten characters recognition techniques to represent letters as strokes, loops or crossings. Many thinning algorithms have been proposed in the past few years, but these algorithms are not yet really suited to images of handwritten letters.

Two techniques are usually used by thinning algorithms. Parallel thinning algorithms [1–3] are fast and easy to code, but skeletons of closed loops and single strokes give the same results—a single line—and overlapping strokes often generate short segments which distort the representation of a letter. Fig. 1 shows the skeleton of an image of the word "fissure", which summarizes some of these difficulties. Some algorithms have been proposed to reduce the number of short segments which appear on the crossing [4–6], or to estimate closed loops [7], but these algorithms do not estimate overlapping stroke paths.

Vector thinning algorithms link boundary points of letter in order to find the underlying strokes [8,9]. These algorithms give plane lines as sequences of points, but cannot deal with crossings. Bodies of letters are then split into two parts: one part for regular strokes and one part for components whose boundaries are not linked. This dual description of letters into lines and “crossing areas” is better than the result given by parallel thinning algorithms, but is not satisfying enough to represent and recognize letters.

Doermann and Rosenfeld [10,11] address this problem of the interpretation of inferring strokes to reconstruct crossing areas. Their process first detects stroke-like and non-stroke-like regions of letter. Local configuration of strokes (local measure of the confidence that a given pair of segments are portions of the same stroke), and local compatibility of the reconstruction with the image is used to interpret non-stroke region as high-curvature point, corner, crossing, etc. A cubic-spline approximation is used to reconstruct strokes. Although this work seems to be the most useful, the approach proposed does not use a global model of crossing areas, and then needs to be extended to all simple cases.

The approach we suggest is to efficiently deal with crossing areas and overlaps using parametric representations of lines and thickness of strokes, global model of
crossing of strokes and statistical studies to estimate the model parameters.

Several remarks have convinced us to search such results, in the perspective of using strokes to recognize handwritten words. First, crossings are less stable than strokes in letters and particularly in words, where overlaps between strokes of different letters create crossings which do not appear in single letters. Moreover, strokes that are cut by crossing are less stable than whole strokes. Finally, the process we have tested to decide if the mark which do not appear in single letters. Moreover, strokes that are cut by crossing are less stable than whole strokes. Finally, the process we have tested to decide if the mark of a stroke is, in fact, the path of a closed loop seems to show that the only way to decide this is to try to link the stroke with two other single strokes. This forces us to treat crossing areas, if we want to extract stable structural information on handwritten letters and words.

We first define a model of strokes, and present an algorithm to extract non-overlapping strokes. This allows us to split up letters into two parts. A second algorithm that optimizes the reconstruction of crossings is then proposed. Finally, results of our algorithms on a NIST database is presented.

2. Extraction of simple strokes

We present here a simple model for strokes which uses relations between boundaries and strokes to estimate lines. We consider in this section images of strokes without noise.

2.1. Definitions and properties

Let a stroke be a family of disc \( \{ D_t, t \in [0, 1]\} \), which is defined by the stroke path \( l: [0, 1] \rightarrow \mathbb{R}^2 \) and the radius of the discs \( r: [0, 1] \rightarrow \mathbb{R}^+ \). We assume that the functions \( l \) and \( r \) verify the two following properties: \( l \) and \( r \) are \( C^\infty \) and \( l(t) < r^2(t) \) for \( t \in [0, 1] \), with \( v = dl/dt \). The trace of a stroke on \( \mathbb{R}^2 \) is the set \( G = \{ M \in \mathbb{R}^2, \exists t \in [0, 1], M \in D_t \} \), and the stroke boundaries is \( B = G \setminus \bar{G} \).

With these conditions on \( l \) and \( r \), the two envelopes \( (E_1, E_2) \) of a stroke are always defined. The envelopes of a family of discs \( D \) are points of each circumference of a stroke on \( \mathbb{R}^2 \), and the stroke boundaries is \( B = G \setminus \bar{G} \).

2. Extraction of simple strokes

We present here a simple model for strokes which uses relations between boundaries and strokes to estimate lines. We consider in this section images of strokes without noise.

2.1. Definitions and properties

Let a stroke be a family of disc \( \{ D_t, t \in [0, 1]\} \), which is defined by the stroke path \( l: [0, 1] \rightarrow \mathbb{R}^2 \) and the radius of the discs \( r: [0, 1] \rightarrow \mathbb{R}^+ \). We assume that the functions \( l \) and \( r \) verify the two following properties: \( l \) and \( r \) are \( C^\infty \) and \( l(t) < r^2(t) \) for \( t \in [0, 1] \), with \( v = dl/dt \). The trace of a stroke on \( \mathbb{R}^2 \) is the set \( G = \{ M \in \mathbb{R}^2, \exists t \in [0, 1], M \in D_t \} \), and the stroke boundaries is \( B = G \setminus \bar{G} \).

With these conditions on \( l \) and \( r \), the two envelopes \( (E_1, E_2) \) of a stroke are always defined. The envelopes of a family of discs \( D \) are points of each circumference of a stroke on \( \mathbb{R}^2 \), and the stroke boundaries is \( B = G \setminus \bar{G} \).

\[ \{ C_{1, r}, t \in [0, 1], \varphi \in [\pi, \pi] \} \] of the family of discs which verify: \( \partial C_{1, r}/\partial t \) is parallel to \( \partial C_{1, r}/\partial \varphi \), that is, with \( l_t = (x_t, y_t)^T \):

\[ E_{1t} = |t - r_t (x_t, y_t) - \sqrt{1 - r_t^2 v_t^2}, t_r \]  

\[ E_{2t} = |t - r_t (x_t, y_t) - \sqrt{1 - r_t^2 v_t^2}, t_r \]  

(1)

Fig. 2 shows an example of a stroke with its envelopes. Let \( E \) be the set of points of the envelopes, then \( B \) is strictly included in \( E \cup C_0 \cup C_1 \), where \( C_0 \) and \( C_1 \) are the circumferences of the first and the last discs of the stroke. So, to study relations between \( B \) and the stroke path, we first describe relations between \( E \) and the stroke path, and then we study cases where \( E \) is different from \( B \).

Given two points \( E_{1t_1} \) and \( E_{2t_2} \) on each stroke envelope, two relations may be used to compute the point \( l_t \) of the path which fit \( E_{1t_1} \) and \( E_{2t_2} \) (Fig. 3):

Relation 1. (a) If the tangents \( t_1 \) and \( t_2 \) are defined on \( E_{1t_1} \) and \( E_{2t_2} \), then the two straight lines which are perpendicular to \( t_1 \) and contain \( E_{1t_1} \), and \( E_{2t_2} \), cross on \( F_n \), and \( |E_{1t_1} F_n| = |E_{2t_2} F_n| \); (b) \( l_t = F_r \).

Relation 2. Let \( H_t \) be the center of \( [E_{1t_1} E_{2t_2}] \), then \( [H_t l_t] \) is perpendicular to \( [E_{1t_1} E_{2t_2}] \), and \( H_t l_t \) is parallel to \( E_{1t_1} E_{2t_2} \). Estimating the path link given the envelopes is the same as matching the first envelope with the second, and one point \( E_{1t_1} \) is fit with the point of \( E_2 \) at the same parameter value \( t \) using Relation 1(a).

However \( B \) and \( E \) do not always fit, especially, when the curvature of the stroke path is high: in this case, one of the envelopes may be strictly included in \( G \). In such cases we cannot fit the two boundaries of the stroke to estimate the path link. We introduce the following definition to describe this case:
Definition 1. The envelopes of a stroke are locally strictly included in \( G \) (l.s.i.) at the parameter value \( t \) if:

\[
\exists \lambda > 0 \text{ such that } \forall \epsilon, |\epsilon| < \lambda, \delta_i(t, \epsilon) < 0, \quad i = 1 \text{ or } 2
\]

with

\[
\delta_i(t, \epsilon) = [E_i(t) - l_{i+\epsilon}]^2 - r_{i+\epsilon}^2.
\]

Even if the envelopes are nowhere l.s.i, this is not sufficient to force their inclusion in the boundaries of \( G \), since envelopes may be hidden by another part of the stroke mark, or by another stroke. This definition, however, allows to obtain a simple and local way to decide whether the two envelopes of a segment of stroke are “visible”, i.e. is \( B \) included, and then to decide if it is possible to fit both boundaries. Let \( \rho \) be the curvature of the stroke path. It is easy to show, with a Taylor’s expansion, that the envelopes are l.s.i if

\[
v^2 (1 - \rho \sqrt{1 - (\vec{r}^2/v^2)} + 2 \frac{\dot{r} \vec{r}}{v} - \dot{r}^2 - \dot{r}^2 > 0. \quad (2)
\]

In practice, we use a pathlink, so \( \alpha(t) = 1 \), and we assume \( \vec{r} = 0 \). The relation becomes

\[
1 - \rho \sqrt{1 - r^2 - \dot{r}^2} < 0. \quad (3)
\]

2.2. First algorithm

Our main hypothesis is to assume that handwritten characters images are a union of noisy stroke traces. The goal of this first algorithm is to divide the trace \( L \) of a letter into two parts \( A \) and \( B \). The first set \( A \) may be described by segments of stroke in which both envelopes are included in theirs boundaries, and \( B = L \setminus A \).

Given a black and white letter image, our first step is to smoothen the local boundaries strokes using classical masks, and to link the edge points (\( B \)) of the image, which corresponds to discrete and noised sequences of points of the stroke boundaries.

Let \( A1 \) and \( B1 \) be two points of the same boundary, for which the distance between \( A1 \) and \( B1 \) is equal to the half average \( h \) of the thickness of the letter’s strokes. The algorithm tests the following hypothesis, \( H_1 \) : \( (A1, B1) \) is the end of a segment with visible envelopes. If the test is positive, the algorithm estimates the corresponding edge points \( A2 \) and \( B2 \) of the second stroke boundary.

First, we estimate the two tangents \( t_{A1} \) and \( t_{B1} \) on both points \( A1 \) and \( B1 \), using their neighboring edge points. Let \( FA \) (res. \( FB \)) be the first edge point that crosses the half-straight line which contains \( A1 \) (res. \( B1 \)), and is perpendicular to \( t_{A1} \) (res. \( t_{B1} \)) (see Fig. 4(I)). These two points are used to obtain, from a neighborhood \( S \) of \( FA \) and \( FB \), a set of positions for \( A2 \) and \( B2 \).

For each couple of points \((A’2, B’2)\) of \( S \), which defines a segment of stroke \( T = [A1, B1; A2, B2] \), the algorithm tests if \( T \) is l.s.i, which is made by testing if relation (3) is verified. We estimate \( \rho, r \), and \( \dot{r} \) using the centers \((C1, C2)\) of both segments of edge points. (Fig. 4(II)).

We arbitrate among the possible couple of points which verify condition (3) by choosing the ones which best verify relation 1(a). A cost function is defined, which depends on the angles between the tangents on the four edge points. A Gaussian noise is introduced to deal with the error on the estimation of these angles. Then the cost function is defined as the quadratic distance between the expectation of the four angles and their estimations.

This process is used to link, step by step, the two boundaries of each stroke with visible envelopes: given the two ends \((B1, B2)\) of the little segment of stroke, let \( D1 \) be an edge point of the right neighborhood of \( B2 \), at a distance \( h \) from \( B2 \), and \( SD \) be the set of left neighboring
edge points of $B_1$, at a distance smaller than $h$ from $B_2$ (Fig. 4(IV)). The algorithm follows the process described below to fit in $SD$ the best corresponding end $D_2$.

2.2.1. End of stroke

Strokes end in three cases:

1. When $r$ is too high on a little segment and thus is not compatible with a definition of a stroke. This happens when the stroke ends at a crossing, like a $T$-junction for example.
2. When the test is negative for all the points of $S$. This corresponds to a segment of stroke whose curvature is high.
3. When the boundaries meet each other, the stroke stops naturally at its end.

Finally we obtain, for each letter, one part described by sequences of linked edge points, which form simple strokes, that is strokes with visible envelopes, and another part which corresponds to traces of stroke crossings and small segments of stroke with high curvature. Fig. 6(I) shows results of this first algorithms for images of the training set of the NIST form-based handprint recognition system.

3. Models for stroke crossings

3.1. Motivation and Hypothesis

Results obtained by this first algorithm are similar to those of vectorial thinning algorithms: letters are described as graphs of strokes, and part of letters where strokes cross are not described, apart from connecting strokes.

The main disadvantages of such results are the description of closed loops as single strokes, and the variation of the graph of curves for letters of the same class; therefore the study and the parameter estimation of the deformation of such graphs to simulate handwritten letters are difficult and inefficient.

Let us take the following examples to clarify both problems: Fig. 5 shows four examples of letters a and the corresponding thinning results obtained with a classical parallel thinning algorithm (the AFP3 algorithm described by Guo and Hall [2]. As usual, lines obtained are often bad estimations of overlapping stroke paths.

Fig. 6(I) shows the corresponding results obtained by our first algorithm, that can be described as graphs of strokes: the graph nodes are strokes and “crossing areas”, and a graph structure connects nodes that join each other (Fig. 6(II)).

For these four examples of the same class of letters, we obtain four different graph structures, whereas a characterization of these examples using stable graphs of curves is possible, if we are able to estimate stroke paths that cross each other, and if we segment paths to obtain smooth lines (Fig. 6(III)). In such descriptions, called graphs of curves, that are close to those of on-line handwriting characters, crossings are “implicit” and do not increase the variation of graph structures by segmenting smooth strokes: graph structures may be reduced to the number of curves.

Our goal is then to deal with crossings and overlapping strokes, to decrease the variation of graph structures. Our approach is first to use a parametric model of strokes in order to merge strokes that are segmented by crossing, and then to determine between the different configurations of crossing the one which best matches the observations.

---

Fig. 5. Examples of results with classical thinning algorithms (the AFP3 algorithm described by Guo and HallGuo [2].
Three main hypotheses are made to make an efficient algorithm of crossing reconstruction:

h1. Curvature of strokes has a low value when crossing.
h2. A branch of a stroke may merge with at least two other branches
h3. The algorithm deals with crossings with less than five branches.

A branch of a crossing \( C_r \) is a stroke which ends in \( C_r \).

The first hypothesis is used to merge two branches \( A, B \) of a crossing into a stroke \( S_{A,B} \) only if \( S_{A,B} \) can be described as one smooth stroke. \( S_{A,B} \) is the link between \( A \) and \( B \). Fig. 7 shows a crossing with four branches on a letter “k”, and three different ways to reconstruct stroke paths into these crossing, which correspond to three different descriptions of \( (A, B) \): as two strokes (Fig. 7(a)), as one stroke whose curvature is high in its center (Fig. 7(b)), and as a loop (Fig. 7(c)). Observations on this image are not sufficient to choose between these three solutions but, if we describe these three solutions using graphs of curves we obtain three similar graph with three curves (Fig. 7(II)). Thus, generally, links that form smooth strokes are the only ones that are useful and easy to estimate.

The second hypothesis is made to deal with closed loops, and the last one is done because in the NIST database we use, the number of crossing with more than four branches is very low and, conversely, the number of configuration of such crossing is too high to test each of them.

3.2. Parametric model of strokes

A crossing \( C \) is a set of overlapping strokes. The trace of a crossing is described by the branches of the crossing, parts of strokes that do not overlap, that are connected with the “crossing area”, the center of the crossing that is not described as a stroke.

The first algorithm does not use a parametric model to define strokes, that are defined as sequences of edge points. To reconstruct smooth strokes whose edges are hidden by a crossing, we introduce the following parametric models on paths and thickness of strokes \( D(l, r) \):

\[
\begin{align*}
    l(t) &= \begin{cases} 
        x(t) = f(t) = \sum_{i=0}^{4} a_i t^i, \\
        y(t) = g(t) = \sum_{i=0}^{4} b_i t^i
    \end{cases} \\
    r(t) &= c_0 + c_1 t + c_2 \exp\left(-t \frac{L}{p}\right) + c_3 \exp\left((t - 1) \frac{L}{p}\right) \\
    &= \sum_{q=0, 0.3} c_q h_q(t),
\end{align*}
\]

with \( 0 \leq t \leq 1 \). We note \( S(\theta) \) such a stroke, with \( \theta = (a, b, c) \).

Observations on a trace of a stroke is defined by two sequences of edge points. Let \( (X_l)_{1,n} \) and \( (X_r)_{1,n} \) be the sequences of these points. Noise on the edge points
3.3. Topology of crossings

Let \( T_m \) be the set of \( m \times m \) binary (0 or 1) symmetric matrix \( c \) that verify: \( \forall j \sum c(i, j) \leq 2 \) and \( c(i, i) = 0 \) \( \forall i \). If we assume that \( c(i, j) = 1 \) corresponds to a link between branches \( i \) and \( j \) of a crossing with \( m \) branches, and \( c(i, j) = 0 \) corresponds to an absence of such link between both branches, then \( T_m \) is the set of crossing configurations: according to the main hypothesis, \( T_m \) describes the different reconstructions of crossings with \( m \) branches. For example, the matrix \( c_{3,2} \) (8) corresponds to a crossing with three branches \( A, B, C \), parameterized by \( \theta_{B,C} \), the parameters of the link \( S(\theta_{B,C}) \), and by \( \theta_A \) the parameters of the stroke \( S(\theta_A) \) whose trace is \( A \), and the matrix \( c_{3,3} \) corresponds to a three-branch crossing parametrized by \( (\theta_{A,B}, \theta_{A,C}) \), the parameters of the two links \( S(\theta_{A,B}) \) and \( S(\theta_{A,C}) \) between the first branch and the two others: this configuration corresponds to a closed loop.

\[
\begin{pmatrix}
0 & 1 & 0 \\
0 & 0 & 1 \\
1 & 0 & 0
\end{pmatrix}
\]

Fig. 9 shows the set \( T_3 \).

Let \( C_m \) be the group of cyclic permutations on sets with \( m \) elements. \( T_m / C_m \) is the set of types of crossings, i.e. the set of configurations modulo cyclic permutations on branches. We assume that configurations with the same type have the same probability of appearance, as we depend on different factors like pen or pencil type, paper and ink quality, scanning accuracy. We use a Gaussian additive and independent noise on each edge point to model such variations. Let \( \{k_1, \ldots, k_n\} \) be the abscissa of the sequences of edge points. We have supposed that the envelope is included in the edge stroke. Then, let \( t \) and \( n \) be the unit tangent and normal vectors of the stroke path, and \( (v_r, v_t, \xi_r, \xi_t) \) four \( n \)-dimensional Gaussian vectors, with \( k = k_1, \ldots, k_n \).

\[
X_1(k) = E_x(k) + v_t \xi_r(k) + \xi_t(k),
\]

\[
X_2(k) = E_y(k) + v_t \xi_r(k) + \xi_t(k),
\]

with \((E_x, E_y)\) the two sequences of points of the envelopes (Fig. 8):

\[
E_x(k) = \dot{y}(k) [1 - \frac{\dot{r}^2}{v^2(k)}],
\]

\[
E_y(k) = \dot{y}(k) [1 - \frac{\dot{r}^2}{v^2(k)}],
\]

3.3. Topology of crossings

Let \( T_m \) be the set of \( m \times m \) binary (0 or 1) symmetric matrix \( c \) that verify: \( \forall j, \sum c(i, j) \leq 2 \) and \( c(i, i) = 0 \) \( \forall i \). If we assume that \( c(i, j) = 1 \) corresponds to a link between branches \( i \) and \( j \) of a crossing with \( m \) branches, and \( c(i, j) = 0 \) corresponds to an absence of such link between both branches, then \( T_m \) is the set of crossing configurations: according to the main hypothesis, \( T_m \) describes the different reconstructions of crossings with \( m \) branches. For example, the matrix \( c_{3,2} \) (8) corresponds to a crossing with three branches \( A, B, C \), parameterized by \( \theta_{B,C} \), the parameters of the link \( S(\theta_{B,C}) \), and by \( \theta_A \) the parameters of the stroke \( S(\theta_A) \) whose trace is \( A \), and the matrix \( c_{3,3} \) corresponds to a three-branch crossing parametrized by \( (\theta_{A,B}, \theta_{A,C}) \), the parameters of the two links \( S(\theta_{A,B}) \) and \( S(\theta_{A,C}) \) between the first branch and the two others: this configuration corresponds to a closed loop.

\[
\begin{pmatrix}
0 & 0 & 0 \\
0 & 1 & 1 \\
1 & 0 & 0
\end{pmatrix}
\]

Fig. 9 shows the set \( T_3 \).

Let \( C_m \) be the group of cyclic permutations on sets with \( m \) elements. \( T_m / C_m \) is the set of types of crossings, i.e. the set of configurations modulo cyclic permutations on branches. We assume that configurations with the same type have the same probability of appearance, as we
choose not to take account of the orientation of strokes in our a priori knowledge of configuration distribution. Fig. 10 shows the set of types of $\mathbb{T}_4$.

Finally, a crossing $Cr$ with $m$ branches, whose configuration is $c \in \mathbb{T}_m$, is parameterized with $\theta_c = \{\theta_{k, l} \mid c(k, l) \neq 0\} \cup \{\theta_{k, l} \mid c(k, l) = 0\}$, the parameters of links $S(\theta_{k, l})$ and branches $S(\theta_{k})$ with no links on $Cr$.

Then the likelihood of the observations $(X)_1^r$, is defined as follows: define for each point $X_i$ the point $E_i$ in the trajectories of $S(\theta)$ which minimizes the difference between the thickness $t_r$ on this point and $|E_i - X_i|$. Let $R_i = |E_i - X_i| - r$. We assume that $(R)$ is a sufficient statistic for $(X)$, and that $R \sim \mathcal{N}(0, \sigma^2 \cdot Id)$, where $\sigma^2$, the variance of the noise on the edges of strokes, is estimated on the image of letter.

4. Estimation of crossing configurations

In this section, we detail the principles of our second algorithm, an algorithm of crossing estimation. Let $Cr$ be a stroke crossing with $m$ branches. We assume, under the hypothesis of the existence of a link $S(\theta_{A,B})$ between two branches $A$ and $B$, that $\theta_{A,B}$ is independent of the other links and of the crossing branches. Thus in a first step, the second algorithm makes a least square estimation of all the possible links between two branches on $Cr$. In a second step, the algorithm computes for each configuration
an estimation of the maximum a posteriori loglikelihood of the observation, and finally chooses the ones which maximize it.

4.1. Two branch merging

Let \((A, B)\) be two branches of a crossing, and \(H_{A-B}\) the hypothesis that \((A, B)\) merge into one stroke \(S(\theta_{A-B})\) whose path curvature remains at a low value. Given the sequences of edge points \((Ar, Al)_{i=1}, \ldots, n\) and \((Br, Bl)_{i=1}, \ldots, n\) which describe the two branches, let \(k = \{k_1, \ldots, k_n, k_{n+1}, \ldots, k_{n+m}\}\) be the abscissa of these sequences under \(H_{A-B}\). \((Ar, Al)\) and \((Br, Bl)\) are sequences of noisy observations of points of the two envelopes of \(S_{A-B}\). Let \((C_r, C_l)\) be the sequences \((A_r, A_l)\) and \((B_r, B_l)\). The estimator \(\hat{\theta}_{A-B}\) is computed with the parametric model of strokes, using estimation of the abscissa and tangents, and simplification of the formula 7.

First, we use the main hypothesis \(h_1\) to make an affine interpolation \(\hat{k}\) of the abscissa (Fig. 8(II)). Unit tangents and normal vectors are then locally estimated using this affine interpolation of the path.

We note these estimators, \(t_i\) and \(n_i\), at the abscissa \(i\). Furthermore, we assume that \(r/k < 1\), so as to neglect the term \(t_i(t_i)/r(k_i)\), and we simplify \(r(k_i)n_i\sqrt{1 - r^2(k_i)/r^2(k_i)}\) into \(r(k_i)n_i\). Finally, we obtain

\[
C_r(k_i) = l(k_i) - r(k_i)n_i + v_i,t_i + \xi_i,t_i,
\]

\[
C_l(k_i) = l(k_i) + r(k_i)n_i + v_i,t_i + \xi_i,t_i. \tag{9}
\]

Then a least-squares linear estimation \(\hat{\theta}_{A-B}\) of the parameters of \(S_{A-B}\) is obtained from the linear system (9). Fig. 12 shows some examples of stroke estimated using this process.

**Remark.** The efficiency of the affine estimation of the abscissa depends on the main hypothesis \(h_1\): without it, such linear estimations would not be acceptable.

If one uses this algorithm to merge a branch \(A\) with another \(B\) whose thickness is higher, for instance when \(B\) is the trace of a closed loop, then the result is often wrong because one of the edges of \(S_{A-B}\) is hidden in the trace \(B\) (Fig. 11). So in such a case, we introduce two sub-hypotheses \(H_{A-B}^i\) and \(H_{A-B}^r\):

1. \(H_{A-B}^i\): Only the right edge of \(S_{A-B}\) is visible in the trace of the branch \(B\).
2. \(H_{A-B}^r\): Only the left edge of \(S_{A-B}\) is visible in the trace of the branch \(B\).

---

**Fig. 11.** Examples of the merging of a simple stroke with a closed loop (a–c), and the final estimation of the crossing (d). (a) and (b) correspond to false hypotheses, and (c) is the true one.
Under these two hypotheses, we assume that the thickness of $S_{A,B}$ is constant, to estimate the hidden edge, and to obtain a least-square estimation of $\hat{\theta}_{A-Br}$ and $\hat{\theta}_{A-Bl}$ (Fig. 11).
We obtain for each hypothesis of branch merging a quick and linear estimation of the corresponding strokes. Using the same process to estimate the parameters $h_{kA}$ of each branch of the crossing, an estimation $h_{Kc}$ of the parameters of each configuration $c$ is done (Fig. 12).

The last step of the algorithm is to identify, within the set $\mathbb{T}_m$, which configuration best fits the observations.

4.2. Crossing identification

Let $(X)_i=1,...,n$ be the observations on a crossing $Cr$ with $m$ branches, and let $n=2r$ be the dimension of the observation on the crossing. Let $\Theta$ be the sample space of $\hat{\theta}(X) = \{\hat{\theta}_{k,l}, k,l \in \{1,\ldots,q\}\}$, where $q$ is the cardinal of $\mathbb{T}_m$. We use a bayesian approach to identify crossing configuration. We define a decision function $d: \chi \times \Theta \rightarrow \mathbb{T}$, which selects estimate $\hat{\hat{c}} = d(X, \hat{\theta}(X))$ of the configuration of $Cr$.

To construct $d$, we introduce a loss function $C: \mathbb{T} \times \mathbb{T} \rightarrow [0,1]$ which measures the cost introduced by the identification of a configuration $c$ by $\hat{\hat{c}}$, and we take the function which minimizes the expected risk $R$:

$$R(d, X, \hat{\theta}(X)) = \sum_{c=1}^{q} P(c|X, \hat{\theta}(X)) \cdot C(c, \hat{\hat{c}})$$

$$= \sum_{c=1}^{q} \frac{p(X|\hat{\theta}(X), c) \cdot P_m(c)}{p(X, \hat{\theta}(X))} \cdot C(c, \hat{\hat{c}})$$

$$R(d, X, \hat{\theta}) = \sum_{c=1}^{q} \frac{p(X|\hat{\theta}_c(X), c) \cdot p(\hat{\theta}(X)|c) \cdot P_m(c)}{p(X, \hat{\theta}(X))} \cdot C(c, \hat{\hat{c}}).$$

The function $C$ we used is such that $C(c, \hat{\hat{c}}) = 0$ if $c = \hat{\hat{c}}$ and $C(c, \hat{\hat{c}}) = 1$ if $c \neq \hat{\hat{c}}$, so the choice of $\hat{\hat{c}}$ becomes the maximum a posteriori likelihood (MAP) estimator:

$$\hat{\hat{c}} = \arg\max_c L(X, \hat{\theta}, c)$$

$$= \arg\max_c \log p(X|\hat{\theta}_c, c) + (\log p(\hat{\theta}|c) + \log P_m(c)),$$

that can be split into two terms, one for the loglikelihood of the observations, and one for an a priori logdensity of the parameter estimators.

4.2.1. A priori distribution of the parameters

In the context described in the beginning of this article, no information about the class of the image of the letter which is treated is available. So we introduce a sufficient statistic of the parameters $\theta$ which does not depend on

Fig. 12. The six different simple links between branches for a crossing with 4 branches.
Fig. 13. A false link between two branches on a crossing with 4 branches.

Fig. 14. Examples of estimations of stroke paths for images of letters of a NIST database.
the orientation of the stroke but depends only on the intrinsic shape of the stroke, that is a function of the curvature. For each stroke $S_{ik}$, which merges one branch $k$ with another $l$, there are two options: this stroke is a real one, or this stroke does not exist, and we take a statistic $\phi_{ik}$, function of the curvature, whose densities are different enough to discriminate the two hypothesis. For example, functions like the mean of the curvature will not be suitable because even if the link does not exist, the mean of the curvature of the line may have a low value (Fig. 13).

Let $L$ be the length of a stroke. The statistics we use is:

$$\Delta \rho(\theta) = \int_0^{rt} |\rho, \rho'(s)| \, ds,$$

Thus $\Delta \rho(\theta)$ allows one to take into account the curvature variations.

As we assume that the density of $\Delta \rho$ depends only on the fact that the stroke exists or does not exist, an empiric estimation of the density of $\Delta \rho$ for both hypotheses has been done on 200 crossings of the database images. A parametric estimation of the density of these two empiric densities was computed using a gamma law $\rho(a, b)$ with $a_1 = 0.68, b_1 = 2.15$ when the connection really exists, and a law $\rho(a_2, b_2)$ with $a_2 = 1.51, b_2 = 0.01$ when the link is missing. Let $g_1(\cdot | 0)$ be such density when the link does not exist, and $g_0(\cdot | 1)$ when the link exists. Eventually, the a priori density of $\Delta \rho(\theta)$ is:

$$p(\Delta \rho(\theta) | c) = \prod_{k,l=1 \ldots q, k \neq l} g(\Delta \rho(\delta_{ik}), c(k, l)).$$

The a priori distribution of the configuration $P_m(c)$ is estimated on a test set of letters of the database, under the hypothesis that the a priori probability of a configuration depends only on the type of the configuration.

### 4.2.2. Loglikelihood estimation

To compute the loglikelihood $\log p(X|c, \hat{\theta})$, we have to estimate the series ($R$) of errors on the thickness on the edge points for each configuration $c$. When $X_i$ stays on a branch of the crossing that is described with only one stroke $S_i$, the least-square estimation of the parameters of $S$ computes straight the error $R_i$ of the thickness on the edge point $X_i$. When two strokes overlap on a branch, or when $X_i$ is an edge point of the area of the crossing, the algorithm chooses for $X_i$ the minimum of errors for all the strokes of $c$. But in practice the series $R$ is not computed for each configuration and sub-configurations. A minimal difference $R_{i,ik}$ is computed for each edge points $X_i$ and for each stroke $S_{\delta_{ik}}$, and for each configuration we take: $R_i = \min(R_{i,ik}) | c(k, l) = 1).

A problem appears when one follows this process: the dimension of the parameters which describe a crossing depends on the type of configuration, i.e. on the number of strokes of the configuration. So the law of $R^2 = \sum_{i=1 \ldots r} R_i^2$, which is a sufficient statistics for the loglikelihood, depends on the configuration: a bias appears that gives an advantage to the configurations whose number of parameters is higher. So we use a loglikelihood that is penalized by a penalty term $pen(dim(X), dim(c))$ to take into account this bias: it is described in the appendix.

Finally, the algorithm computes for each configuration and possible subconfiguration $c$, an estimation of the maximum of the penalized a posteriori loglikelihood:

$$LC(X, \hat{\theta}, c) = \log(p(X|\hat{\theta}, c) - pen(dim(X), dim(c)) + \log p(\rho(\hat{\theta}) | c) + \log P_m(c)$$

$$= \log(p(\rho(\hat{\theta}) | c) + \log P_m(c)) + \log P_m(c) + \sum_{k,l=1 \ldots m, k \neq l} g(\Delta \rho(\delta_{ik}), c(k, l))$$

The estimator of the configuration becomes:

$$\hat{c} = \arg \max_{c \in \Omega} LC(X, \hat{\theta}, c)$$

and the crossing is described by strokes corresponding to $\hat{c}$.

### 5. Results

The estimation of the a priori distribution of the crossing types has been done on 520 images of letters (20 letters for each class) of the NIST database. The following arrays summarize the distribution of the number of crossings by letters, the number of branches by crossing, and the distribution of crossing types for crossings with 3 and 4 branches.

<table>
<thead>
<tr>
<th>Crossing by letter</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3 +</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of observed letters</td>
<td>162</td>
<td>291</td>
<td>62</td>
<td>5</td>
</tr>
<tr>
<td>%</td>
<td>31%</td>
<td>56%</td>
<td>12%</td>
<td>1%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Number of branches</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of observed crossings</td>
<td>354</td>
<td>75</td>
<td>2</td>
</tr>
<tr>
<td>%</td>
<td>82.1%</td>
<td>17.4%</td>
<td>0.5%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Types</th>
<th>T3.1</th>
<th>T3.2</th>
<th>T3.3</th>
<th>Others</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of strokes</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>—</td>
</tr>
<tr>
<td>Number of configurations</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>—</td>
</tr>
<tr>
<td>Number of observed crossings</td>
<td>6</td>
<td>97</td>
<td>245</td>
<td>6</td>
</tr>
<tr>
<td>%</td>
<td>1.7%</td>
<td>27.4%</td>
<td>69.2%</td>
<td>1.7%</td>
</tr>
</tbody>
</table>

(Continued on next page)
becomes possible, and an adaptation of on-line handwriting recognition methods on-line description of characters. Therefore, unlike the characters. A statistical study of the stroke crossings on handwritten images of word without any a priori knowledge of the result indicates that our method will not be useful for such a letter, but will be acceptable to the true line of the letter, and estimate the corresponding unknown parameters with a 4-branch crossing and a 5-branch crossing that are not reconstructed. The letters (b) and (c) show the main branches are linked and form one stroke (H\textsubscript{0}).

In order to illustrate the importance of a priori knowledge to estimate strokes, the last example (d) shows a crossing identification that does not correspond to the true line of the letter, but will be acceptable without any knowledge of the class of the letter. Such result indicates that our method will not be useful for images of word without any a priori knowledge of the graph of curves to be estimated conditionally to the class of letters.

### 6. Conclusion

We have described an algorithm for reconstructing stroke paths of off-line handwritten characters, which deals with the case of stroke crossings. A simple parametric model of stroke has been introduced. The method is based on the properties of the edges of this model, and on a statistical study of the stroke crossings on handwritten characters.

With this method, handwritten characters are described by simple graphs of curves that are similar to the on-line description of characters. Therefore, unlike the description obtained by classical thinning algorithms, an adaptation of on-line handwriting recognition methods becomes possible, and an efficient model for the law of graphs of curves can be used.

### Appendix A

In this section, we first analyze the bias of the loglikelihood estimator in a simple case, then we generalize the results to the case of stroke crossings.

Let \( X \) be a random vector on \( \mathbb{R}^{2d} \), which verifies under the hypothesis \( H_0 \):

\[
X(i) = \sum_{k=0}^{d-1} a_k \left( \frac{i}{2n} \right)^k + \xi_i, \quad i = 1 \ldots 2n
\]

and, under the hypothesis \( H_1 \):

\[
X(i) = \sum_{k=0}^{d-1} b_k \left( \frac{i}{n} \right)^k 1_{[i \leq n]} + b_k \left( \frac{i}{n} \right)^k 1_{[i > n]} + \xi_i, \quad i = 1 \ldots 2n,
\]

\( \xi \sim \mathcal{N}(0, I_{2n}) \). With an observation \( x \) of this random vector, we like to identify the true hypothesis, and estimate the corresponding unknown parameters \( A = (a_0, \ldots, a_{d-1})^\text{tr} \) or \( B = (b_0, \ldots, b_{2d-1})^\text{tr} \).

We can recognize, with simplifications, the two hypothesis between two branches of a crossing: the two branches are linked and form one stroke (\( H_0 \)), or they are the mark of two different strokes (\( H_1 \)).

With the following notations: \( T_{2n} = ((i/2n)^n, T_n = ((i/n)^n) \) and \( T_{n0} = (T_0^n, T_n^0) \), the two models can be rewritten like this:

\[
H_0: X = AT_{2n} + \xi,
\]

\[
H_1: X = BT_{n0} + \xi.
\]

For each hypothesis, the estimator of the maximum likelihood for \( A \) and \( B \) corresponds to the least-square estimator:

\[
\hat{A} = (T_{2n}^\text{tr} T_{2n})^{-1} T_{2n}^\text{tr} X,
\]

\[
\hat{B} = (T_{n0}^\text{tr} T_{n0})^{-1} T_{n0}^\text{tr} X.
\]

Thus the estimator of the maximum of the loglikelihood for \( H_0 \) and for \( H_1 \) is:

\[
H_0: L_0(X) = -\frac{1}{2} (X - \hat{A} T_{2n})^\text{tr} (X - \hat{A} T_{2n}) + \text{cst}(n),
\]

\[
H_1: L_1(X) = -\frac{1}{2} (X - \hat{B} T_{n0})^\text{tr} (X - \hat{B} T_{n0}) + \text{cst}(n).
\]

Under \( H_0 \) we have:

\[
L_0(X)_{H_0} = \text{cst}(n) - \frac{1}{2} (AT_{2n} + \xi - \hat{A} T_{2n})^\text{tr}
\times (AT_{2n} + \xi - \hat{A} T_{2n})
\]

\[
= \text{cst}(n) - \frac{1}{2} \xi^\text{tr} \Lambda_{2n} \xi
\]
The calculus is the same for $H_1$:

\[ L_1(X)_{H_1} = \text{cst}(n) - \frac{1}{2}(T_{2n}A + \xi)^*\Delta_{n0}(T_{2n}A + \xi). \]  

Under $H_1$, both statistics become:

\[ L_0(X)_{H_1} = \text{cst}(n) - \frac{1}{2}(T_{n0}B + \xi)^*\Delta_{2n}(T_{n0}B + \xi) \]  

\[ L_1(X)_{H_1} = \text{cst}(n) - \frac{1}{2}(T_{n0}B + \xi)^*\Delta_{2n}(T_{n0}B + \xi) \]

Thus the difference $\Lambda = L_1(X) - L_0(X)$ which is used to identify the hypothesis, is, under $H_0$:

\[ \Lambda_{H_0} = L_1(X)_{H_0} - L_0(X)_{H_0} \]

\[ = -\frac{1}{2}((\xi^*\Delta_{2n} - \Delta_{n0})\xi - 2(A^*T_{2n}\Delta_{n0}T_{2n}A) \]

\[ + A^*W^{-1}\sigma^2 T_{2n}\Delta_{n0}\xi) \]  

and under $H_1$:

\[ \Lambda_{H_1} = L_1(X)_{H_1} - L_0(X)_{H_1} \]

\[ = -\frac{1}{2}((\xi^*\Delta_{2n} - \Delta_{n0})\xi - 2(B^*T_{n0}\Delta_{2n}T_{n0}B) \]

\[ + B^*T_{n0}\Delta_{2n}\xi) \]  

In both cases, the difference is the sum of two terms; the law of the first term is $-\frac{1}{2}\chi^2(d)$, and the law of the second is a Gaussian $\mathcal{N}(\delta - 2\text{d}(n), 4\text{d}(n))$. Under $H_0$, $\text{d}(n) = A^*T_{2n}\Delta_{n0}T_{2n}A$ and under $H_1$, $\text{d}(n) = B^*T_{n0}\Delta_{2n}T_{n0}B$, but in both cases, $\text{d}(n)$ is equal to the quadratic distance between the series of points of the true curve and the ones estimated on the false hypothesis: under $H_0$, $T_{2n}A$ is the vector of the true points of the curve (without any noise), and $T_{n0}B$ is the vector of the estimation of these points under the false hypothesis. Therfore, $\text{d}(n)_{H_0} = (T_{2n}A - T_{n0}\tilde{B})^*(T_{2n}A - T_{n0}\tilde{B})$

\[ = A^*T_{2n}\Delta_{n0}T_{2n}A \]

The calculus is the same for $H_1$.

Under $H_0$, $\text{d}(n) \sim 10^{-14}$ for the curves observed on the stroke crossings, which is not surprising because the number of parameters of the false models is higher than the number of parameters of the true model. So we can neglect this term and, under this hypothesis, if we compare both loglikelihood, there is a systematic bias on account of $H_1$, whose law is $-\frac{1}{2}\chi^2(d)$.

Under $H_1$ (2 curves), the quadratic distance $\text{d}(n)$ between the true curves and the one estimated with $H_0$ is a measure of the error between both models. Under $H_1$, the bias on account of $H_1$ is the sum of two terms, whose law are $\frac{1}{2}\chi^2(d)$ and $\mathcal{N}(\text{d}(n), \text{d}(n))$.

Thus the choice of a penalty term depends closely on the quadratic distance between both models: if under $H_1$ $\text{d}(n)$ has a low value, which shows that the observations are well described by the model with one curve, we have to choose this hypothesis, to favour the model with less parameters. So we turn the hypothesis $H_1$ into $H_{1,p}$, which is the model $H_1$ with $D(n) = n.D_{\text{min}}$, where $D_{\text{min}}$ is the average distance for which under $H_1$, $H_0$ cannot be chosen. Thus let $L_{C_i}(X)$ be the penalized loglikelihood, $i = 0, 1$:

\[ L_{C_0}(X) = L_0(X) - \text{pen}(d, n, D_{\text{min}}). \]

\[ L_{C_1}(X) = L_1(X) - \text{pen}(2d, n, D_{\text{min}}). \]

where $C$ is such that:

\[ P(L_{C_0}(X) - L_{C_1}(X) < 0|H_0) \]

\[ = P(L_{C_1}(X) - L_{C_0}(X) < 0|H_1). \]

The error rates are the same under both hypothesis. Giving $n$ and $D_{\text{min}}$, such function $\text{pen}(\ )$ can be easily estimated using simulation of the loglikelihood bias.

The same process is used to compensate for the loglikelihood bias of the stroke crossing observations, as this simple case corresponds to the model of one edge with two branches. So we just have to make an empirical choice of a “limit” crossing for which two branches, which are not linked, can be described as one stroke, to determine $D_{\text{min}}$.

References


About the Author—ERIC L’HOMER received a Ph.D. degree in applied mathematics from the University of Orsay, France, in 1998. Since 1993, he has been on the CMLA, Ecole Normale Superieure de Cachan. He is currently a research scientist at Paris 13 University. His research interests include pattern recognition, stochastic neural networks and Gaussian mixture distribution.