signMine algorithm for conditioning and analysis of human handwriting

Alexei V. Nikitin[*]
Avatekh LLC,
2131 Quail Creek Dr.,
Lawrence, KS 66047
nikitin@avatekhllc.com

Denis V. Popel[†]
Computer Science Department
Baker University,
Baldwin City, KS 66006
denis.popel@bakeru.edu

Abstract

We describe the signMine *algorithm, which relates to methods for conditioning, representation, modeling, and analysis of variables. In particular,* signMine *is specially adapted for analysis of parametric line objects such as human handwritten signatures. The paper also gives an outline of the* signMine *software package designed for performing signature identification and verification.*

1 Introduction

Consider a curve given in a parametric form $\xi(o) = \xi_x(o) + \mathrm{i}\xi_y(o)$, where o is some continuous *order parameter*. It is convenient to call a representation of a curve 'kinematic' when the order parameter is a physical time t, $\xi = \xi(t)$, and thus the curve can be interpreted as the trajectory of a moving particle. This trajectory can also be presented in a *natural* (or *intrinsic*) form, for example in terms of its arc length s and tangential angle $\varphi(s)$ (Whewell equation), or in terms of its arc length s and curvature $\kappa(s)$ (Cesàro equation). Such an intrinsic equation specifies the shape of a curve, independent of any choice of coordinates or parameterization [9], as a simple scalar function of one argument. If a curve were indeed representing a movement of a particle, the kinematics of this motion can be specified, for example, by providing the speed of the particle's motion along the curve, $v(t) = \dot{s}(t) = |\dot{\xi}(t)|$.

The curvature and the arc length can be expressed as

$$\kappa(t) = \frac{\Im[\dot{\xi}^*(t)\,\ddot{\xi}(t)]}{|\dot{\xi}(t)|^3}\,, \quad \text{and} \quad s(t) = \int_0^t \mathrm{d}t'\,|\dot{\xi}(t')|\,, \tag{1}$$

where z^* denotes the complex conjugate of z, and $\Im[z]$ is the imaginary part of z. The curve itself then can be expressed as

$$\xi(s) = \xi_0 + \int_0^s \mathrm{d}s'\,\mathrm{e}^{\mathrm{i}\varphi(s')}\,, \tag{2}$$

where the tangential angle φ is

$$\varphi(s) = \varphi_0 + \int_0^s \mathrm{d}s'\,\kappa(s')\,. \tag{3}$$

Note that equation (1) is valid only for differentiable and regular curves as it requires finite and nonvanishing speed $|\dot{\xi}(t)|$. This restriction makes equation (1) unsuitable for description such irregular and discontinuous curves as those representing human handwriting, and renders this equation virtually useless when those curves are given as discrete (digital) records.

In this paper, we describe an algorithm which enables accurate representation, in terms a natural equation of the underlying continuous curve, of a *modulated* curve given by a discrete sets of ordered data. Further, we demonstrate how such a representation leads to a set of tools for for conditioning, analysis, comparison, and identification of human handwritten signatures, and provide an outline of the signMine software package.

2 Interpolation in order index

Consider a (raw) digital record which consists of the sets of the Cartesian coordinates $\{\mathbf{r}_i\} = \{x_i, y_i\}$,

[*] Corresponding author. Also with Dept. of Physics & Astronomy, U. of Kansas, Lawrence, KS 66045, USA.

[†] Also with Neotropy LLC, Lawrence, KS.

the time values $\{t_i\}$, and the (optional) modulation $\{\mathbf{f}_i\}$, where $i = 0, 1, 2, \ldots, N$ is an *order index.* It is convenient to use a *normalized order index* o, $0 \le o = i\,N^{-1} \le 1$, instead of an integer i. The modulation vector $\mathbf{f}$ can be, for example, the force (pressure) applied by the writing utensil, the curve's color, etc. The main purpose of (smoothing) interpolation is to (re-)create a continuous representation of a curve from its digital record. This continuous representation must adequately correspond to the raw digital record, and should be suitable for expression in an intrinsic form. When such a continuous (high resolution) record is available, all parameter values along the interpolating curve (the values of the Cartesian coordinates, arc length, tangential angle, curvature, time, speed, modulation, etc.) can be obtained with arbitrary precision. In addition, interpolation allows the reduction of noise and sensitivity to the size of sampling interval(s).

The simplest interpolation is a linear (broken-line) interpolation, which amounts to connecting the sequential points $\mathbf{r}_i$ and $\mathbf{r}_{i+1}$ by straight-line segments and corresponding definition of the values of the other parameters (e.g., the speed and the tangential angle) along those segments. Even though a broken-line curve is not differentiable (and thus, for example, the curvature is zero anywhere between vertices and is infinite at a vertex joining a pair of non-parallel segments), a proper handling of singularities allow its intrinsic-form description, as illustrated in §3.

In a case of noisy finely-sampled data, representation of a (piecewise) smooth curve through a broken-line interpolation is misleading and virtually useless. The main usage of the linear interpolation is as follows: (i) obtain the vertices (their coordinates as well as other parameters at those points) by sampling the piecewise-smooth tangential or smoothing interpolating curve, then (ii) use the linear broken-line representation to obtain the necessary descriptive parameters of the curve suitable for numerical calculations.

2.1 'Tangential' interpolation by a finite-size continuous kernel

Given the values of a function $y(x)$ at a set of points $\{y_i = y(x_i)\}$, $i = 0, 1, 2, \ldots, N$, the values of $y(x)$ and its various derivatives at arbitrary x can be determined through the following interpolation scheme:

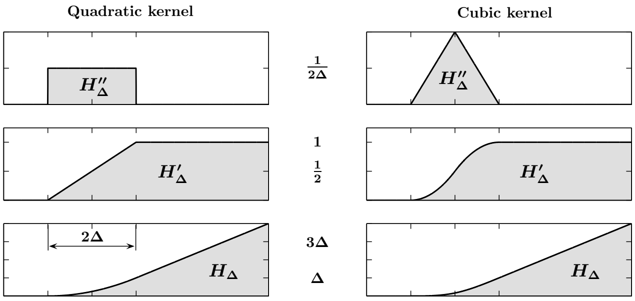


Figure 1. Quadratic and cubic interpolating kernels

$$\frac{\mathrm{d}^n}{\mathrm{d}x^n}\,[y(x) - y_0] = \begin{cases} \sum_{i=0}^{N-1} \Delta y_i \, \frac{\mathrm{d}^{n+1}}{\mathrm{d}x^{n+1}}\, H_\Delta(x - x_i) \\ \quad \text{if} \quad \Delta x_i = 0 \\ \sum_{i=0}^{N-1} \frac{\Delta y_i}{\Delta x_i} \frac{\mathrm{d}^n}{\mathrm{d}x^n} \left[H_\Delta(x - x_i) - H_\Delta(x - x_{i+1})\right] \\ \quad \text{otherwise} \end{cases}, \tag{4}$$

where $\Delta x_i = x_{i+1} - x_i$, $\Delta y_i = y_{i+1} - y_i$, and $H_\Delta(x)$ is a continuous (differentiable) kernel such that

$$\lim_{\Delta \to 0} H_\Delta(x) = x\,\theta(x)\,. \tag{5}$$

Also note that, as follows from equation (5), $\lim_{\Delta\to 0} H'_\Delta(x) = \theta(x)$, and $\lim_{\Delta\to 0} H''_\Delta(x) = \delta(x)$, etc., and in the limit $\Delta \to 0$, for $\Delta x_i > 0$, equation (4) represents a simple linear interpolation. Figure 1 shows the quadratic and cubic interpolating kernels.

Notice that the interpolation scheme given by equation (4) can handle discontinuous data (i.e., $\Delta x_i = 0$), and does not require $\{x_i\}$ to be monotonic (i.e., Δx_i can be negative). Thus it is suitable for interpolating discontinuous and noisy data, as illustrated in figure 2.

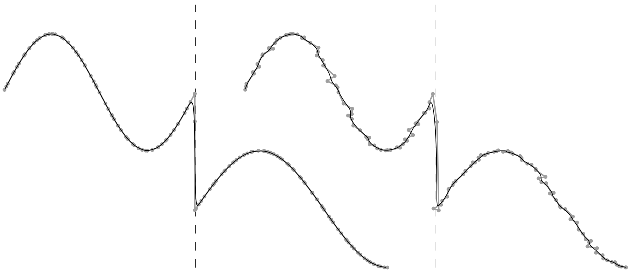


Figure 2. Interpolation of discontinuous and noisy data

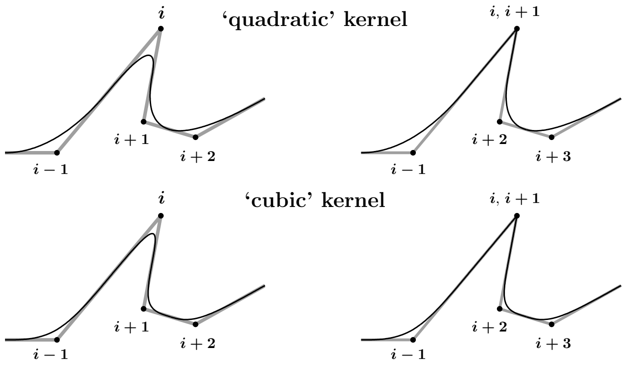


Figure 3. Tangential interpolating curves constructed using quadratic (upper panels) and cubic (lower panels) kernels

If the width of a kernel does not exceed half of the increment in the original order index (i.e., $\Delta o \leq (2N)^{-1}$), interpolation leads to a smooth curve with the following properties:

- the interpolating curve passes through a middle point of each straight-line segment connecting a pair of adjacent vertices while being tangential to the respective segment at this point, and
- the tangential angle to the interpolating curve changes monotonically between the middles of any two adjacent segments of the broken line.

This is illustrated in figure 3 for interpolations using quadratic (upper panels) and cubic kernel (lower panels). Notice that in the righthand panels the vertices i and $i+1$ coincide forming a single vertex, and that the interpolating curve passes through this vertex.

A typical use of a tangential interpolation would be in a case when accuracy of data acquisition is achieved at the expense of the increase in the sampling interval(s), which leads to a too 'rugged' shape of a curve when a linear interpolation is used.

2.2 Smoothing interpolation

In a smoothing interpolation, the width of a kernel exceeds half of the increment in the original order index (i.e., $\Delta o > (2N)^{-1}$), and thus, as described in §2.1, the values of the interpolating curve result from a contribution of more than a single original data point. A typical use of a smoothing interpolation is the reduction of noise when the increase in sampling frequency leads to the loss of accuracy in data acquisition.

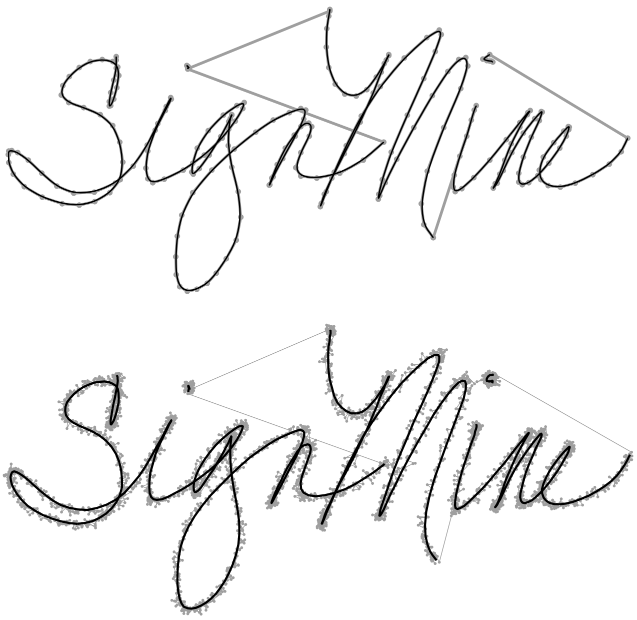


Figure 4. Tangential (upper panel) and smoothing (lower panel) interpolations with a quadratic kernel

Figure 4 illustrates both tangential (upper panel) and smoothing (lower panel) interpolations with a quadratic kernel. In both panels, the raw data is shown in grey (in a form of linear broken-line interpolations), and the interpolating curves are shown by black lines.

3 Description of a broken-line curve

3.1 Description of a piecewise-continuous (segmented) curve

A curve $z = x + \mathrm{i}y$ resulting from human handwriting (e.g., a signature) can consist of only one contiguous component, or a plurality of components. In the latter case, the order and relative positions of the components might be relevant to verification and/or identification of the curve. When the components are arranged in 'chronological' order (e.g., using an order parameter o, $0 \leq o \leq 1$), we can preserve the information about their order and relative positions by connecting the ends of the 'earlier' components with the respective origins of the 'later' components by straight-line segments. In our description of a curve, we want the ability to easily switch between the two representations of the curve, including or excluding the connecting segments, while preserving a unified formalism. We shall use the term ‘connected segmented curve’ when the straight-line segments are included, and the term ‘disconnected curve’ otherwise.

Differential displacement along a connected segmented curve can be formally defined as

$$\mathrm{d}l = \left|\frac{\mathrm{d}}{\mathrm{d}o}\, z(o)\right| \mathrm{d}o\,, \tag{6}$$

where it is assumed that the derivatives at discontinuities of $z(o)$ can be expressed using the Dirac δ-function [see 2, for example].

Differential displacement along a disconnected curve is defined as

$$\mathrm{d}s = \left|\overline{\frac{\mathrm{d}}{\mathrm{d}o}}\, z(o)\right| \mathrm{d}o\,, \tag{7}$$

where

$$\overline{\frac{\mathrm{d}}{\mathrm{d}o}}\, z(o) = \frac{1}{2}\left(\frac{\mathrm{d}}{\mathrm{d}o+} + \frac{\mathrm{d}}{\mathrm{d}o-}\right) z(o)\,, \tag{8}$$

and $\frac{\mathrm{d}z}{\mathrm{d}o+}$ and $\frac{\mathrm{d}z}{\mathrm{d}o-}$ are the right-hand and left-hand, respectively, derivatives of z:

$$\frac{\mathrm{d}}{\mathrm{d}o\pm}\, z(o) = \lim_{\varepsilon\to 0} \frac{z(o\pm\varepsilon) - z(o)}{\pm\varepsilon}\,. \tag{9}$$

It should be easy to see from equations (6) and (7) that $\mathrm{d}l$ and $\mathrm{d}s$ are related as

$$\mathrm{d}l = \mathrm{d}s + \delta l(o) = \mathrm{d}s + \delta l(s)\,, \tag{10}$$

where

$$\delta l(x) = \lim_{\varepsilon\to 0} \left|z(x+\varepsilon) - z(x-\varepsilon)\right|\,, \tag{11}$$

Note that $\mathrm{d}l \equiv \mathrm{d}s$ anywhere within a continuous component of the curve.

The total lengths of a disconnected and a connected segmented curves, respectively, can be expressed as

$$S = \int_0^1 \mathrm{d}o\, \frac{\mathrm{d}s}{\mathrm{d}o}\,, \quad L = \int_0^1 \mathrm{d}o\, \frac{\mathrm{d}l}{\mathrm{d}o} = S + \sum_i \delta l(s_i)\,, \tag{12}$$

where the summation goes over all points s_i where the curve is discontinuous.

3.2 Robust (coincidence) segmentation of a digitally-sampled curve

The formalism of §3.1 allows us do develop a simple robust procedure for segmentation of a digital record. Notice that, as follows from equation (11), the differential δl is zero everywhere except at the ‘breaks’ between the continuous components. Let us define the double differential $\delta^2 l$ as

$$\delta^2 l(o) = \lim_{\varepsilon\to 0} \delta l(o+\varepsilon) - \delta l(o)\,, \tag{13}$$

and point out that $\delta^2 l$ also vanishes at continuous components while taking finite absolute values at discontinuities.

Consider now a curve sampled at discrete values of o, and the finite-difference equivalents of the differentials δl and $\delta^2 l$:

$$\Delta l_i = \left|z(o_{i+1}) - z(o_i)\right|\,, \tag{14}$$

and

$$\left|\Delta^2 l_i\right| = \frac{1}{2}\left[\left|\Delta l_{i+1} - \Delta l_i\right| + \left|\Delta l_i - \Delta l_{i-1}\right|\right]\,. \tag{15}$$

Notice that both Δl_i and $\left|\Delta^2 l_i\right|$ will have pronounced maxima whenever a discontinuity lies between o_i and o_{i+1}. On the other hand, the extrema of Δl_i will correspond to the zeros of $\left|\Delta^2 l_i\right|$ at continuous portions of the curve.

Thus a robust (coincidence) segmentation of a digitally-sampled curve can be performed using the following algorithm: **Discontinuities can be found as *coincident maxima* of Δl_i and $\left|\Delta^2 l_i\right|$ lying above a certain threshold (or respective thresholds).** Since the number of discontinuities is generally much smaller than the total number of the data points in any meaningful digital record, a simple choice for a threshold would be a high percentile of the values of Δl_i and/or $\left|\Delta^2 l_i\right|$.

Figure 5 illustrates the performance of the algorithm on two curves with different sampling (see right-hand panels). The panels on the left show the first differential Δl_i by the solid black line, the second differential $\left|\Delta^2 l_i\right|$ by the solid gray line, and the respective thresholds (90 th percentiles) by the dashed lines. The discontinuous points are indicated by the asterisks. In the right-hand panels, the data points (dots) belonging to continuous portions of the curves are connected by the black lines.

3.3 Intrinsic equation for a piecewise-continuous curve

When the tangential angle is expressed as

$$\phi(s) = \lim_{\varepsilon\to 0} \arg\left[z(s+\varepsilon) - z(s-\varepsilon)\right]\,, \tag{16}$$

where $\arg(z)$ is the (complex) argument of a complex number z (see §3.3.1 below), an intrinsic (Whewell)

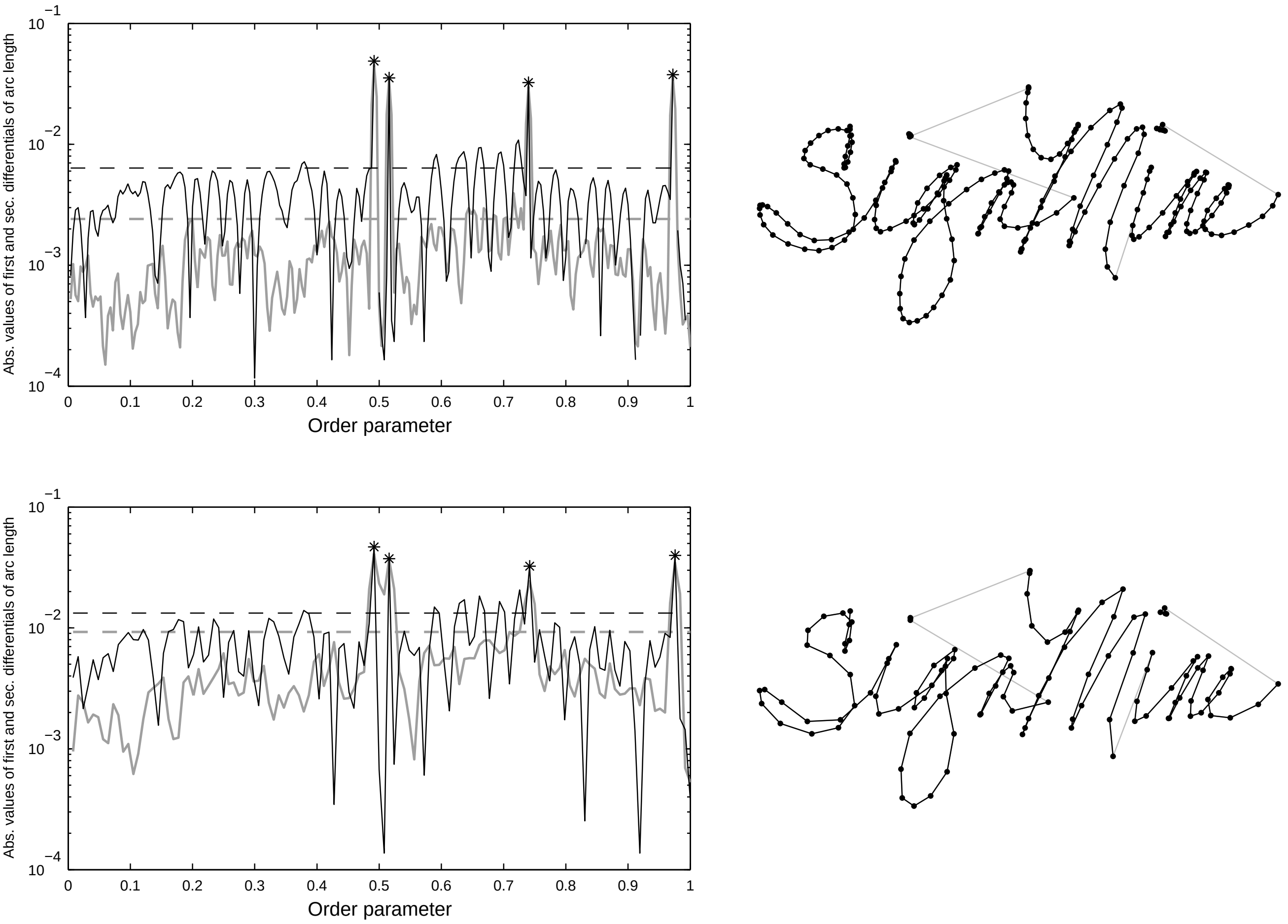


Figure 5. Robust (coincidence) segmentation of a digitally-sampled curve

equation of a piecewise-continuous curve can be written as

$$z(s) = \int_0^s \mathrm{d}s' \, \mathrm{e}^{\mathrm{i}\phi(s')} + \sum_i \delta l(s_i) \, \mathrm{e}^{\mathrm{i}\phi(s_i)} \, \theta(s - s_i) \,, \quad (17)$$

where $\theta(x)$ is the Heaviside unit step function, and the summation goes over all points s_i where the curve is discontinuous.

The kinematic description is obtained by expressing the arc length and the tangential angle as functions of time,

$$z(t) = \int_0^t \mathrm{d}t' \, \dot{s}(t') \, \mathrm{e}^{\mathrm{i}\phi(t')} + \sum_i \delta l(t_i) \, \mathrm{e}^{\mathrm{i}\phi(t_i)} \, \theta(t - t_i) \,, \quad (18)$$

where the dot over s denotes a time derivative.

3.3.1 Quadrant-specific inverse tangent

The (complex) argument of a complex number z can be computed as a quadrant-specific arctangent and defined as follows:

$$\arg(z) = \arg(x + \mathrm{i}y) =$$

$$\begin{cases} \arcsin(y/|z|) & \text{if} \quad x \geq 0 \\ -\arcsin(y/|z|) + \pi & \text{if} \quad x < 0, \ y \geq 0 \\ -\arcsin(y/|z|) - \pi & \text{if} \quad x < 0, \ y < 0 \\ 0 & \text{if} \quad |z| = 0 \end{cases} . \quad (19)$$

3.4 Scaling and alignment along the preferred direction

There are many alternative definitions of such factors as the size (total arc length), orientation, and position of a curve in relation to the coordinates' origin [see 7, for example]. For example, the definitions of the center of a curve and its mean (or preferred) direction can be defined in kinematic and/or geometric sense, and will depend on whether the connecting segments are included into consideration. It may be argued that such factors by themselves are not relevant

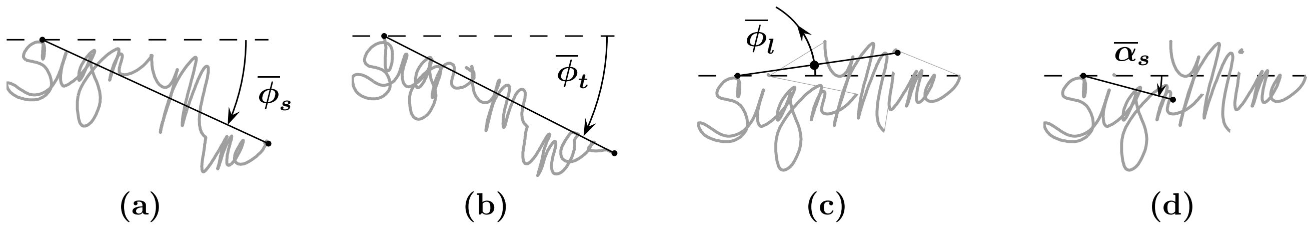


Figure 6. Defining the mean (or preferred) direction

to the curve's verification and/or identification, even though the differences in these factors due to different definitions may serve as descriptive statistics.

The mean (or preferred) direction, $\overline{\phi}$, can be defined in a variety of ways. For example, for a disconnected curve it can be computed in geometric sense as

$$\overline{\phi} = \overline{\phi}_s = \arg\left(\int_0^S \mathrm{d}s\, \mathrm{e}^{\mathrm{i}\phi(s)}\right)\,, \tag{20}$$

and its geometric meaning, as illustrated in figure 6 (a), is the direction of a segment connecting the origin and the end of a curve composed of concatenated continuous components of the curve. The respective kinematic definition is

$$\overline{\phi} = \overline{\phi}_t = \arg\left(\int_0^T \mathrm{d}t\, \mathrm{e}^{\mathrm{i}\phi(t)}\right)\,, \tag{21}$$

and its geometric meaning is illustrated in figure 6 (b).

For a connected segmented curve, the preferred direction can be expressed as

$$\overline{\phi} = \overline{\phi}_l = \arg\left(\int_0^S \mathrm{d}s\, \mathrm{e}^{\mathrm{i}\phi(s)} + \sum_i \delta l(s_i)\, \mathrm{e}^{\mathrm{i}\phi(s_i)}\right)\,, \tag{22}$$

and its geometric meaning, as shown in figure 6 (c), is the direction of a segment connecting the origin and the end of the curve.

As a sensible alternative, the preferred direction can be defined as the direction of a vector connecting the origin of a curve with its center, for example:

$$\begin{gathered}\overline{\phi} = \overline{\alpha}_s = \arg(\overline{z}_s)\,, \qquad \overline{z}_s = \\ \textstyle\int_0^S \mathrm{d}s\, \left(1 - \frac{s}{S}\right)\, \mathrm{e}^{\mathrm{i}\phi(s)} + \sum_i \delta l(s_i)\, \left(1 - \frac{s_i}{S}\right)\, \mathrm{e}^{\mathrm{i}\phi(s_i)}\,,\end{gathered} \tag{23}$$

as shown in figure 6 (d).

A normalized aligned curve can be expressed in an intrinsic form as

$$\begin{gathered}\xi(s) = \\ \textstyle\frac{1}{S}\, \left[\int_0^s \mathrm{d}s'\, \mathrm{e}^{\mathrm{i}\varphi(s')} + \sum_i \delta l(s_i)\, \mathrm{e}^{\mathrm{i}\varphi(s_i)}\, \theta(s - s_i)\right]\,,\end{gathered} \tag{24}$$

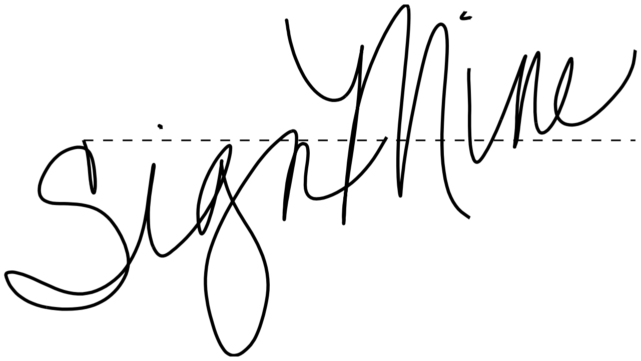


Figure 7. Example of a curve aligned along the preferred direction defined by equation (26)

where $\varphi(s) = \phi(s) - \overline{\phi}$. In polar coordinates, $\xi(s)$ can be written as

$$\begin{gathered}\xi(s) = r(s)\, \mathrm{e}^{\mathrm{i}\alpha(s)}\,, \quad r(s) = \tfrac{1}{S}\, |z(s)|\,, \\ \alpha(s) = \arg\left[z(s)\right] - \overline{\alpha}\,,\end{gathered} \tag{25}$$

where $z(s)$ is given by equation (17) and the preferred direction $\overline{\alpha}$ is defined as

$$\overline{\alpha} = \arg\left(\int_0^S \mathrm{d}s\, |z(s)|^2\, \mathrm{e}^{\mathrm{i}\phi(s)}\right)\,. \tag{26}$$

An example of a curve aligned along the preferred direction defined by equation (26) is shown in figure 7.

4 Circular and linear distributions and the respective densities

4.1 Circular (angular) distributions and the respective densities

The amplitude distribution of an angular (or cyclic with the modulus 2π) variable $\varphi = \varphi(s)$ can be computed as

$$\Psi_s(\beta) = \frac{1}{S} \int_0^S \mathrm{d}s\, \theta\left[\beta - \varphi(s)\right]\,, \tag{27}$$

where we can take, without loss of generality, the range of $\varphi(s)$ to be from $-\pi$ to π. The distribution function $\Psi_s(\beta)$ can be given the following probabilistic interpretation: if s is a uniform deviate in a range 0 to S, then $\Psi_s(\beta)$ is the probability that $\varphi(s)$ does not exceed β.

In practice, the amplitude distribution $\Psi_s(\beta)$ can be computed as [see 5; 6, for example]

$$\Psi_s(\beta) = \frac{1}{S}\int_0^S \mathrm{d}s\, \mathcal{F}_{\Delta D}\left[\beta - \varphi(s)\right] , \tag{28}$$

where $\mathcal{F}_{\Delta D}(x)$ is a *continuous* function which changes monotonically from 0 to 1 so that most of this change occurs over some characteristic range of threshold values Δ, and

$$\lim_{\Delta\to 0} \mathcal{F}_{\Delta D}(x) = \theta(x) . \tag{29}$$

The respective density is a periodic function

$$\psi_s(\beta) = \frac{\mathrm{d}}{\mathrm{d}\beta}\Psi_s^*(\beta) = \psi_s(\beta + 2\pi k) , \tag{30}$$

where $\Psi_s^*(\beta)$ is defined as

$$\begin{gathered} \Psi_s^*(\beta) = \Psi_s(\beta + 2\pi k) - k , \\ -\pi(2k+1) < \beta \le -\pi(2k-1) , \end{gathered} \tag{31}$$

and k is an integer.

4.1.1 Examples of angular distributions

Several examples of angular distributions can be given as follows:

$$\Psi_s(\beta) = \frac{1}{S}\int_0^S \mathrm{d}s\, \theta\left[\beta - \varphi(s)\right] , \tag{32}$$

$$\Psi_l(\beta) = \frac{1}{L}\int_0^L \mathrm{d}l\, \theta\left[\beta - \varphi(l)\right] , \tag{33}$$

$$\Psi_t(\beta) = \frac{1}{T}\int_0^T \mathrm{d}t\, \theta\left[\beta - \varphi(t)\right] , \tag{34}$$

where φ is the tangential angle, and

$$\Xi_s(\beta) = \frac{1}{S}\int_0^S \mathrm{d}s\, \theta\left[\beta - \alpha(s)\right] , \tag{35}$$

$$\Xi_l(\beta) = \frac{1}{L}\int_0^L \mathrm{d}l\, \theta\left[\beta - \alpha(l)\right] , \tag{36}$$

$$\Xi_s(\beta) = \frac{1}{T}\int_0^T \mathrm{d}t\, \theta\left[\beta - \alpha(t)\right] , \tag{37}$$

where α is the polar angle of equation (25). Note that equations (32), (33), (35), and (36) relate to the *geometric* description of a curve, while equations (34) and (37) relate to its *kinematic* description. Figure 8 shows the distributions, along with their respective densities, given by equations (32) through (37) in the left-half panels. Ψ_s, ψ_s, Ξ_s, and ξ_s are shown by the solid black lines, Ψ_l, ψ_l, Ξ_l, and ξ_l are shown by the gray lines, and Ψ_t, ψ_t, Ξ_t, and ξ_t are plotted by the dashed black lines.

4.2 Linear distributions and the respective densities

Various linear distributions and the respective densities of a variable $x = x(s)$ can be viewed as different appearances of general *modulated* distributions

$$\Phi(D) = \frac{\int_0^S \mathrm{d}s\, K(s)\, \mathcal{F}_{\Delta D}\left[D - x(s)\right]}{\int_0^S \mathrm{d}s\, K(s)} \tag{38}$$

and densities

$$\phi(D) = \frac{\mathrm{d}\Phi(D)}{\mathrm{d}D} = \frac{\int_0^S \mathrm{d}s\, K(s)\, f_{\Delta D}\left[D - x(s)\right]}{\int_0^S \mathrm{d}s\, K(s)} , \tag{39}$$

where $K(s)$ is a unipolar *modulating signal* [see 6, for example]. Various choices of the modulating signal allow us to introduce different types of threshold densities and impose different conditions on these densities.

4.2.1 Examples of linear distributions

Several examples of linear distributions can be given as follows:

$$\begin{gathered} F_s\left(\tfrac{t}{T}\right) = \tfrac{s(t)}{S} , \qquad F_l\left(\tfrac{t}{T}\right) = \tfrac{l(t)}{L} , \\ G_s\left(\chi\right) = \tfrac{1}{S}\int_0^S \mathrm{d}s\, \theta\left[\chi - \tfrac{r(s)}{r_{\max}}\right] , \quad \text{and} \\ G_t\left(\chi\right) = \tfrac{1}{T}\int_0^T \mathrm{d}t\, \theta\left[\chi - \tfrac{r(t)}{r_{\max}}\right] . \end{gathered} \tag{40}$$

Figure 8 shows the distributions, along with their respective densities, given by equation (40). F_s, f_s, G_s, and g_s are shown by the solid black lines, F_l and f_l are shown by the gray lines, and G_t and g_t are shown by the dashed black lines.

Note that the interpolation scheme described in §2 allows easy numerical computation of the densities from known distributions.

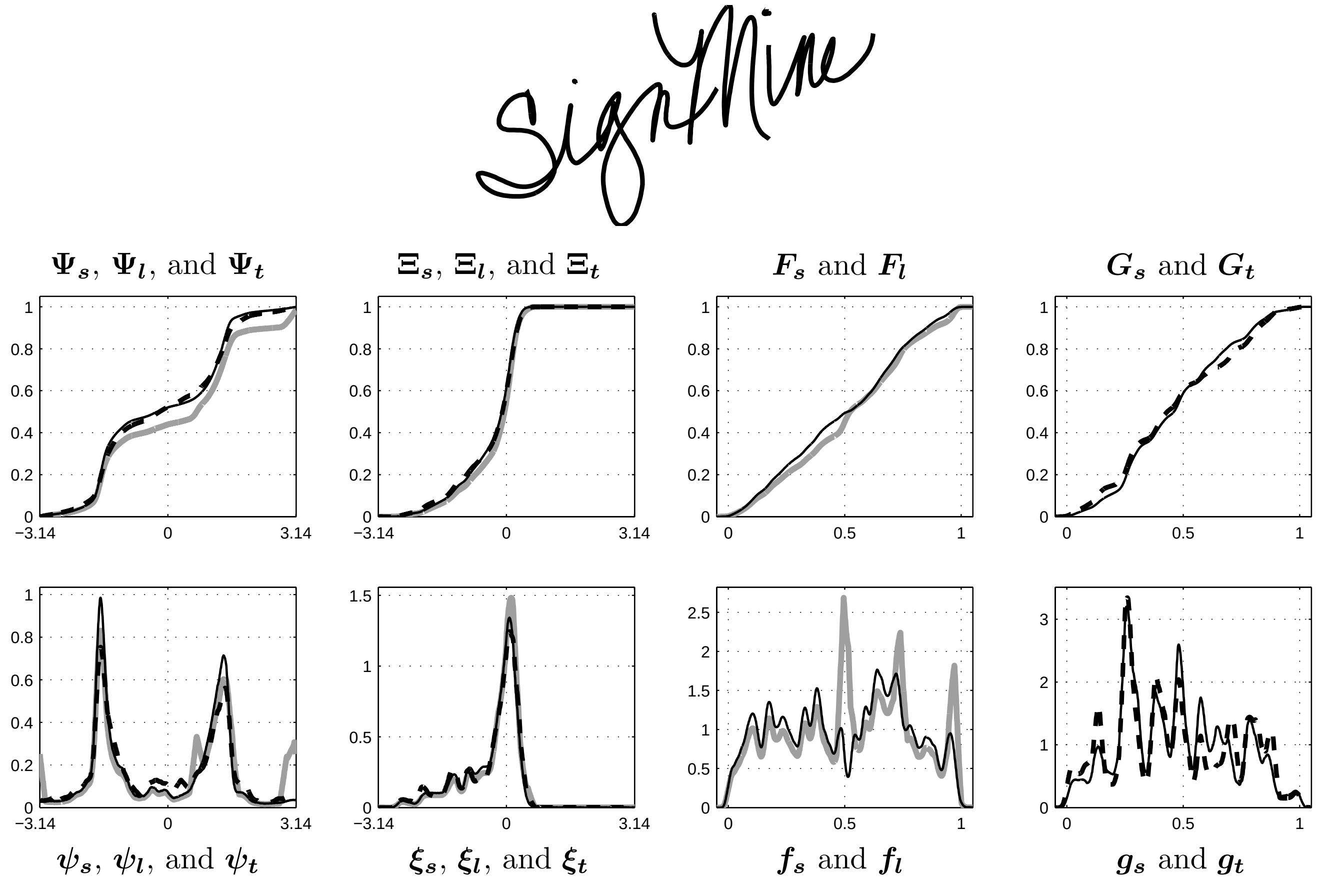


Figure 8. Examples of angular and linear distributions and their respective densities

5 Descriptive statistics

We now have a variety of (equivalent) representations of a curve, and the ability to focus on its different features. For example, we can separate geometric properties of the curve from its kinematic properties, consider or disregard the order and connectivity of contiguous components of the curve, etc.

We can introduce many 'direct' comparison measures, such as the 'distance' estimates, etc. However, most of those measures would have a computational complexity in $O(N^2)$. This is appropriate for comparison and/or verification, but is not suitable for identification and search.

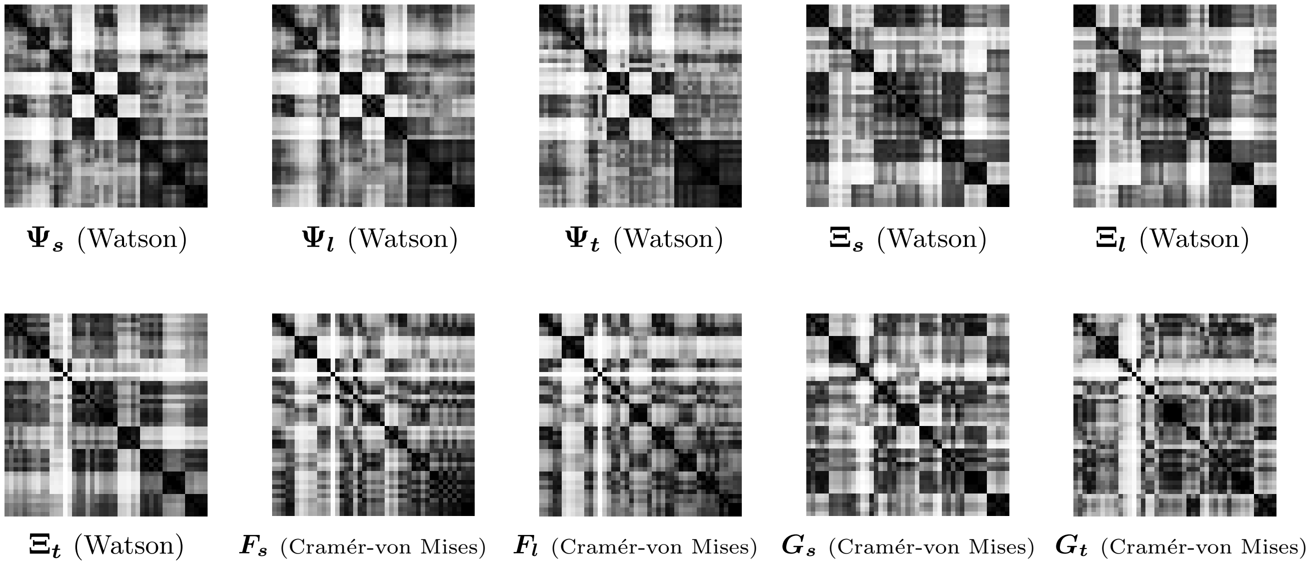
We can also construct a variety of distributions of the variables expressing a curve, and introduce a large number of statistics for those distributions. We can then characterize the curve in terms of those statistics and/or distributions. This allows us to reduce both the size of the inputs (by an order of magnitude or more) and the computational complexity (to $O(N)$ or even $O(\log N)$). It also enables a 'hierarchical' organization of search and retrieval.

Even though different forms of expressing a curve may be equivalent, various distributions constructed for different variables may be different in terms of their 'descriptive' ability, and have different robustness and selectivity with respect to different variations in the curve (e.g., due to noise, discontinuities, singular and/or improper points, etc.).

The main challenge is that the variations due to an 'overall human factor' are not known *a priori*. This is why a database with self-learning capabilities is required.

6 Goodness-of-fit tests

Note that even though the properties of the threshold distributions and densities defined above are usually associated with those of the probability distributions and densities, the above definitions are given for deterministic signals and do not rely on the



$\boldsymbol{\Psi_s}$ (Watson) $\boldsymbol{\Psi_l}$ (Watson) $\boldsymbol{\Psi_t}$ (Watson) $\boldsymbol{\Xi_s}$ (Watson) $\boldsymbol{\Xi_l}$ (Watson)

$\boldsymbol{\Xi_t}$ (Watson) $\boldsymbol{F_s}$ (Cramér-von Mises) $\boldsymbol{F_l}$ (Cramér-von Mises) $\boldsymbol{G_s}$ (Cramér-von Mises) $\boldsymbol{G_t}$ (Cramér-von Mises)

Figure 9. Examples of comparison through two-sample statistics

usual axioms of probability and statistics. The formal similarity of the latter with the probability functions, however, allows us to explore probabilistic analogies and interpretations. Such interpretations enable the construction of a variety of 'statistical' estimators to evaluate the similarity between a pair of variables in a flexible way, permitting a meaningful adaptation to particular problems [see 5; 6, for example].

6.1 Goodness-of-fit tests for linear distributions

As a measure of discrepancy between two distributions, one can use such statistics as Kolmogorov-Smirnov and Cramér-von Mises [see 1; 3, for example].

6.1.1 Two-sample Cramér-von Mises statistic

For two linear distributions F and G, the following statistic of Cramér-von Mises type [see 1; 3, for example] can be used:

$$\gamma^2(F,G) = \frac{3}{2}\int_{-\infty}^{\infty} \mathrm{d}\left[F(x)+G(x)\right]\, W\left[F(x)+G(x)\right]\, \left[F(x)-G(x)\right]^2 \,, \tag{41}$$

where W is a (normalized) weight function and, if both F and G are continuous, the integration may be carried out with respect to either $2F$ or $2G$ instead of $F+G$, since

$$\int_{-\infty}^{\infty} \mathrm{d}\left[F(x)-G(x)\right]\, \left[F(x)-G(x)\right]^2 = 0\,. \tag{42}$$

6.2 Goodness-of-fit tests for circular distributions

For circular distributions, one can use the circular-invariant modifications of the Kolmogorov-Smirnov and Cramér-von Mises tests [see 1, for example], such as the Kuiper [4] and Watson [8] statistics.

6.2.1 Two-sample Watson statistic

Two-sample Watson statistic w^2, $0 \le w^2 \le 1$, can be defined as

$$w^2(\Psi_1,\Psi_2) = 6\int_{-\pi}^{\pi} \mathrm{d}\beta\, \psi_{12}(\beta)\, W\left[\Psi_1(\beta)+\Psi_2(\beta)\right]\, \left[\Psi_1(\beta)-\Psi_2(\beta)\right]^2 - \overline{\Delta\Psi}_{12}^2 \,, \tag{43}$$

where W is a (normalized) weight function, $\psi_{12} = \psi_1 + \psi_2$, and

$$\overline{\Delta\Psi}_{12} = \sqrt{3}\int_{-\pi}^{\pi} \mathrm{d}\beta\, \psi_{12}(\beta)\, W\left[\Psi_1(\beta)+\Psi_2(\beta)\right]\, \left[\Psi_1(\beta)-\Psi_2(\beta)\right] \,. \tag{44}$$

6.3 Percentile comparison for identification and/or comparison

If q_{ij} is the statistic resulting from a similarity (goodness-of-fit) test between i th and j th distributions, then the similarity score assigned

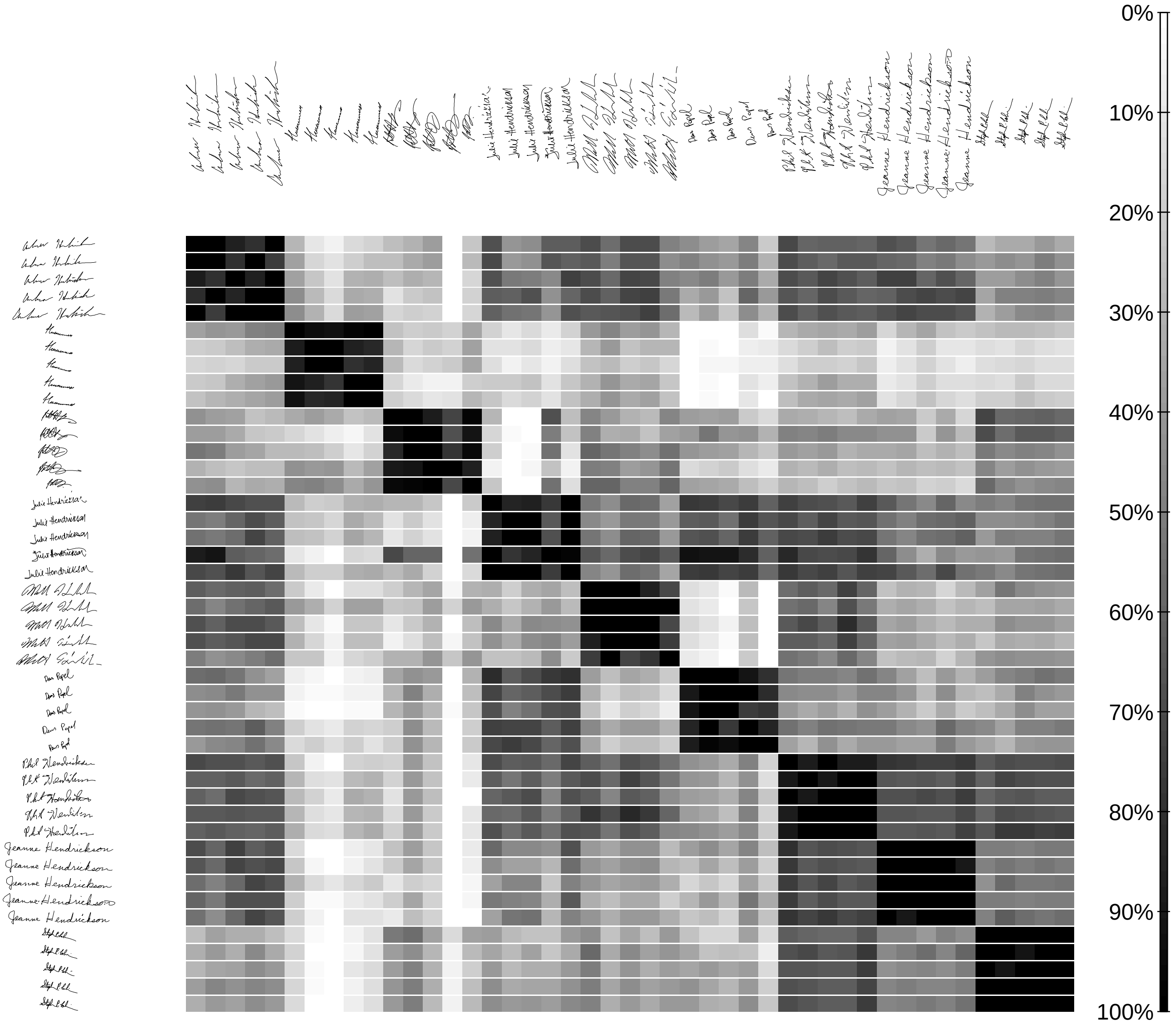


Figure 10. Examples of combined percentile comparison

to this value can be calculated as, for example,

$$P_{ij} = P(q_{ij}) = \frac{1}{N^2} \sum_{k=1}^{N} \sum_{l=1}^{N} \theta \left(q_{kl} - q_{ij} \right) , \qquad (45)$$

where the summation is carried out over all distributions, and can be interpreted as the probability to find a *worse* match between all available pairs of distributions. It is assumed in equation (45) that the statistic q_{ij} is a *non-increasing* measure of similarity.

Figure 9 provides an example of the matrices P_{ij} constructed for various distributions described in § 4. Here, a sample of 45 signatures taken from 9 persons (5 signatures per person) was used. Notice that signatures taken from the same person consistently exhibit high level of similarity (5-by-5 blocks along the diagonals of the matrices) regardless the type of the distribution, while the measures of similarity of the signatures taken from different persons vary in a wide range, depending on the distribution used. Thus the total percentile comparison matrix $\overline{P}_{ij}$ can be constructed as a measure of central tendency of the elements P_{ij} calculated for different types of distributions, and the 'reliability' of this estimate can be calculated as the respective measure of dispersion. Figure 10 provides an example of such a matrix $\overline{P}_{ij}$ calculated for the comparison matrices depicted in figure 9.

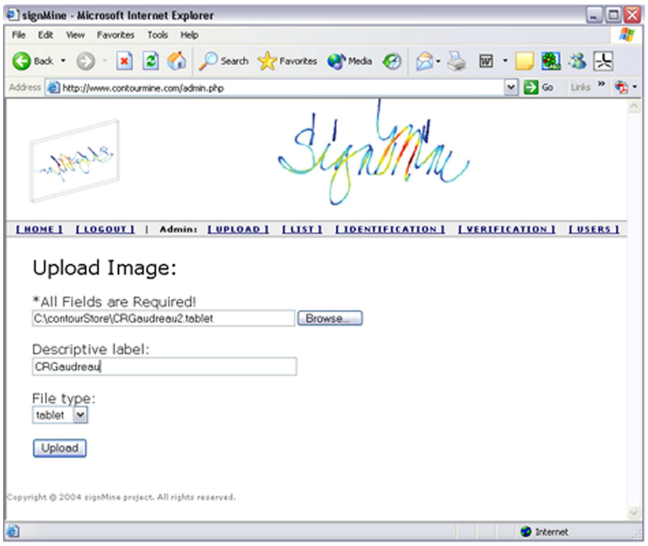
7 Example of online database of handwritten signatures

This section gives an outline of a software package for automated handwritten signature recognition, verification, and mining, signMine, which includes (i) signature acquisition tools, (ii) a searchable signature database (the signMine engine), and (iii) an online interface. The signMine package currently supports pressure sensitive tablets which allow recording both geometric (signature contours, shapes, etc.) and kinematic/dynamic characteristics (pressure, time stamps, etc.).

The signMine engine is a key component of the software package, it includes the tools for generating multiple distributions, the relational database, scoring mechanisms, and decision making tools. Signature databases are currently considered to be a part of multimedia databases, and they differ from traditional information databases based on textual searching. This attributes to the fact that a text-based query is computationally more efficient to perform than the image analysis and comparison. Since a database of signatures based on textual searching alone is inadequate for a qualitative analysis in the areas of biometrics and security, the signMine implementation incorporates distinctions based on the image data. Some of the components of our solution include the server-based database (a relational database), different types of image acquisition tools (pressure sensitive tablets), signature processing and classification algorithms (external modules), and a web-based user interface (dynamically generated web pages). signMine engine is a robust and scalable technology designed to support behavioral authentication mechanisms based on handwritten electronic signatures for identification and verification.

The web-based interface has five basic modules: `login`, `upload`, `list`, `verify`, and `identify`. The database is protected against any unauthorized access by the login module. After the successful login, the user is given administrative rights to the upload and list functions. The upload module allows the user to upload a signature image providing a descriptive keyword (e.g., a person's name), and to choose a file type from the drop down list (see figure 11). After clicking `submit`, the web script updates the database and generates all the necessary distributions for the given image.

The `list` script creates a table, listing all the data from the database. For signature images, the data are listed in the form of thumbnails (see figure 12). A button labelled `regenerate` is also available for administrative users to automatically regenerate distributions for all signatures. This is especially useful when a new classification feature is added to signMine engine. By clicking `regenerate`, all previously stored data are recalculated every signature in the database. Images can be inspected and deleted when necessary.


signMine - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address
Go Links
[HOME] [LOGOUT] | Admin: [UPLOAD] [LIST] [IDENTIFICATION] [VERIFICATION] [USERS]
Browse...
Upload
Internet

Address: http://www.contourmine.com/admin.php

Upload Image:

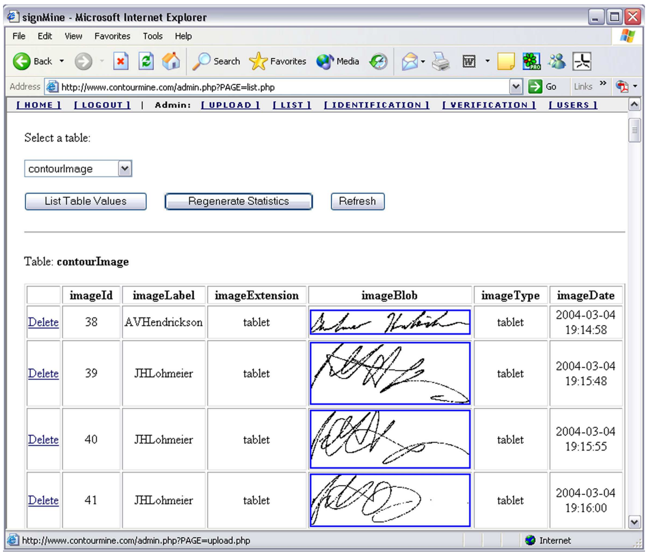
*All Fields are Required!
C:\contourStore\CRGaudreau2.tablet

Descriptive label:
CRGaudreau

File type:
tablet

Copyright © 2004 signMine project. All rights reserved.

Figure 11. Screenshot of the upload module


signMine - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address
Go Links
[HOME] [LOGOUT] | Admin: [UPLOAD] [LIST] [IDENTIFICATION] [VERIFICATION] [USERS]
List Table Values
Regenerate Statistics
Refresh
Internet

Address: http://www.contourmine.com/admin.php?PAGE=list.php

Select a table:
contourImage

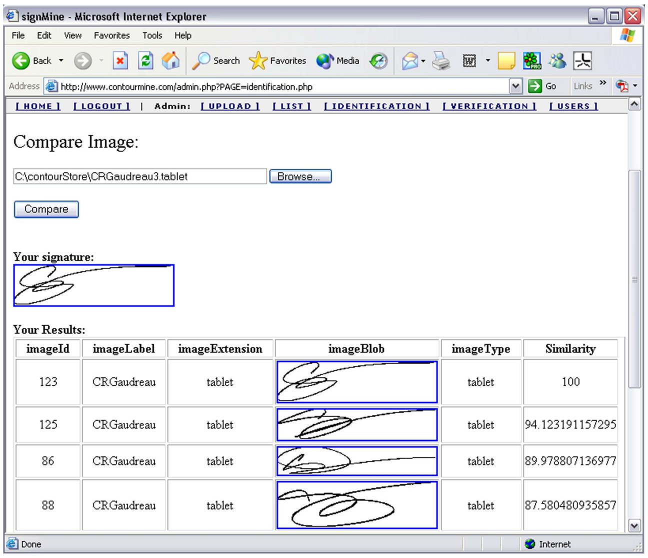
Table: **contourImage**

	imageId	imageLabel	imageExtension	imageBlob	imageType	imageDate
Delete	38	AVHendrickson	tablet		tablet	2004-03-04 19:14:58
Delete	39	JHLohmeier	tablet		tablet	2004-03-04 19:15:48
Delete	40	JHLohmeier	tablet		tablet	2004-03-04 19:15:55
Delete	41	JHLohmeier	tablet		tablet	2004-03-04 19:16:00

http://www.contourmine.com/admin.php?PAGE=upload.php

Figure 12. Screenshot of the list module

The only functions accessible to non-administrative users are `verify` and `identify`, because they do not alter the database. `Identify` is a module that allows the user to upload a signature image, generate distribution data, and compare the generated data


signMine - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://www.contourmine.com/admin.php?PAGE=identification.php
Go Links
[HOME] [LOGOUT] | Admin: [UPLOAD] [LIST] [IDENTIFICATION] [VERIFICATION] [USERS]
Browse...
Compare
Done
Internet

Compare Image:

C:\contourStore\CRGaudreau3.tablet

Your signature:

Your Results:

imageId	imageLabel	imageExtension	imageBlob	imageType	Similarity
123	CRGaudreau	tablet		tablet	100
125	CRGaudreau	tablet		tablet	94.123191157295
86	CRGaudreau	tablet		tablet	89.978807136977
88	CRGaudreau	tablet		tablet	87.580480935857

Figure 13. Screenshot of the identification module

against the data of all images in the database. The verification module collects the keyword label from the user and compares the generated data against a limited set of images. Both modules create a table displaying the testing signature and listing the top ten signatures from the database along with similarity ratings (see figure 13).

8 Summary

In this paper, we provided an outline of the signMine algorithm specially adapted for analysis of parametric line objects such as human handwritten signatures. This algorithm represents signatures given by discrete data in terms of continuous quantities, and enables novel approach to analysis of human handwriting. We also provide a brief description of a complete life-cycle software package for signature identification and verification. The signMine engine stands in the middle, it has image processing tools and internal formats built-in and incorporated with the database. The input data comes from image acquisition devices like scanners or pressure-sensitive tablets, the output is interfaced for other applications (web systems, control systems, etc.). In general, the signMine engine uses drivers to integrate with many off-the-shelf image acquisition devices and standardized software platforms, and connectors to interface with legacy and commonly used authentication systems and applications. The signMine package will have applicability in all areas where signature identification or verification is desirable or required.

References

[1] D. A. Darling. The Kolmogorov-Smirnov, Cramér-von Mises tests. *Ann. Math. Stat.*, 28:823–838, 1957.

[2] P. A. M. Dirac. *The Principles of Quantum Mechanics.* Oxford University Press, London, 4th edition, 1958.

[3] M. Kac, J. Kiefer, and J. Wolfowitz. On tests of normality and other tests of goodness of fit based on distance methods. *Ann. Math. Stat.*, 26:189–211, 1955.

[4] N. H. Kuiper. Tests concerning random points on a circle. *Proceedings of the Koninklijke Nederlandse Akademie van Wetenschappen*, A63:38–47, 1962.

[5] A. V. Nikitin and R. L. Davidchack. Method and apparatus for analysis of variables. Geneva: World Intellectual Property Organization, International Publication Number WO 03/025512, 2003.

[6] A. V. Nikitin and R. L. Davidchack. Signal analysis through analog representation. *Proc. R. Soc. Lond. A*, 459(2033):1171–1192, 2003.

[7] A. V. Nikitin and D. V. Popel. Analog approach to analysis and modeling of biometric information. International Workshop on Modeling and Simulation in Biometric Technology (BT'2004), The University of Calgary, Canada, June 22-23, 2004.

[8] G. S. Watson. Goodness-of-fit tests on the circle. *Biometrika*, 48:109–114, 1961.

[9] R. C. Yates. *Curves and their Properties.* National Council of Teachers of Mathematics, Reston, VA, 1974.

Authors

Alexei V Nikitin, PhD, is the founder and CEO of Avatekh LLC, Lawrence, KS, USA. He has over 17 years of experience working in the areas of applied physics and mathematics, experimental physical chemistry, and engineering. Dr. Nikitin is the principal inventor of the AVATAR technology, which has been the main focus of his research since 1997.

Denis V Popel, PhD, Member IEEE, is an Assistant Professor in the Computer Science Department, Baker University, KS, USA. He has over 9 years of experience working in the area of knowledge discovery including data mining, decision making, data representation, data retrieval, and data manipulation. Dr. Popel participated as the PI and as a senior person on a number of decision making and pattern recognition projects, and published articles and papers in these areas.