update of a small constant number of integer variables and thus allows a very simple implementation of the digital linearity criterion.

All three algorithms take $O(N)$ steps to decompose a digital curve into digital straight segments. The proposed algorithm uses only a constant $O(1)$ storage and is the only one that has on-line characteristics and requires $O(1)$ steps for each additional point. In these respects, it is advantageous compared with the O'Rourke and Smeulders and Dorst algorithms, which use $O(N)$ and $O(\log N)$ storage, respectively, and may require up to $O(N)$ and $O(\log N)$ steps for some points and cannot be used as on-line algorithms. The Smeulders and Dorst algorithm requires only additions and state changes to compute the representation and for checking the linearity condition, and thus, it is possible that it will turn out to be faster on some machines than the proposed algorithm, which sometimes requires integer multiplications.

As shown previously, the problem of finding whether there is a line $y = mx + b$ that is digitized into a given digital line is identical to the problem of finding whether there are two numbers $m$ and $b$ satisfying $\lfloor mi + b \rfloor = y_i$, where $\{y_i\}_{i=1}^{N}$ is the given $y$ coordinate of the digital curve. Substituting $m$ for $\alpha$, $b$ for $\beta$, and $y_i$ for $a_i$, it is clear that this problem is identical to the problem of finding whether a given sequence $\{a_i\}_{i=1}^{n}$ is a nonhomogeneous spectrum. Hence, we also have a new algorithm for determining nonhomogeneous spectra, which is simple, dynamic, efficient, and requires very little space. A simple modification (of limiting the parameters pairs to the $b = 0$ axis) changes the proposed algorithm into an algorithm for determining homogeneous spectra.

## References


[1] A. Rosenfeld, "Digital straight line segments," *IEEE Trans. Comput.*, vol. C-23, pp. 1264–1269, 1974.

[2] C. E. Kim and A. Rosenfeld, "Digital straight lines and convexity of digital regions," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-4, pp. 149–153, 1982.

[3] L. D. Wu, "On the chain code of a line," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-4, pp. 347–353, 1982.

[4] L. Dorst and A. W. M. Smeulders, "Discrete representation of straight lines," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-6, pp. 450–463, 1984.

[5] S. H. Y. Hung, "On the straightness of digital arcs," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-7, pp. 203–215, 1985.

[6] L. Dorst and A. W. M. Smeulders, "Best linear unbiased estimators for properties of digitized straight lines," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-8, pp. 276–282, 1986.

[7] J. Koplowitz and A. P. Sundar Raj, "A robust filtering algorithm for subpixel reconstruction of chain coded line drawings," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-9, pp. 451–457, 1987.

[8] C. A. Berenstein, L. N. Kanal, D. Lavine, and E. Olson, "A geometric approach to subpixel registration accuracy," *Comput. Vision Graphics Image Processing*, vol. 40, no. 3, pp. 334–360, 1987.

[9] M. Lindenbaum and J. Koplowitz, "Compression of chain codes using digital straight line sequences," *Patt. Recogn. Lett.*, vol. 7, pp. 167–171, 1988.

[10] M. Werman, A. Y. Wu, and R.A. Melter, "Recognition and characterization of digitized curves," *Patt. Recogn. Lett.*, vol. 5, pp. 207–213, 1987.

[11] M. D. McIlroy, "A note on discrete representation of lines," *AT&T Tech. J.*, vol. 64, no. 2, pp. 481–490, 1984.

[12] M. Lindenbaum and J. Koplowitz, "A new parametrization of digital straight lines," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 13, pp. 847–852, 1991.

[13] A. Bruckstein, "The selfsimilarity of digital straight lines," in *Vision Geometry* (Melter, Rosenfeld, and Bhattacharya, Eds.). Contemporary Mathematics, AMS, 1991, pp. 1–20, vol. 119.

[14] J. Koplowitz, M. Lindenbaum, and A. Bruckstein, "On the number of digital straight lines on an $N \times N$ grid," *IEEE Trans. Inform. Theory*, vol. IT-36, pp. 192–197, 1990.

[15] C. Ronse, "A bibliography on digital and computational convexity (1961–1988)," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 11, pp. 181–190, 1989.

[16] B. A. Venkov, *Elementary Number Theory*, (translated and edited by H. Anderson). Groningen: Wolters-Noordhoff, 1970.

[17] K. Stolarsky, "Beatty sequences, continued fractions, and certain shift operators," *Canadian Math. Bull.* , vol. 19, no. 4, pp. 473–482, 1976.

[18] A. S. Fraenkel, M. Mushkin, and U. Tassa, "Determination of $[n\theta]$ from its sequence of differences," *Canadian Math. Bull.* , vol. 21, no. 4, pp. 441–446, 1978.

[19] R. L. Graham, S. Lin, and C. S. Lin, "Spectra of numbers," *Math. Mag.* , vol. 51, pp. 174–176, 1978.

[20] M. Boshernitzan and A.S. Fraenkel, "A linear algorithm for nonhomogeneous spectra of numbers," *J. Algorithms* , vol. 5, pp. 187–198, 1984.

[21] A. W. M. Smeulders and L. Dorst, "Decomposition of discrete curves into piecewise straight segments in linear time," in *Vision Geometry* (Melter, Rosenfeld and Bhattacharya, Eds.). Contemporary Mathematics, AMS, 1991, pp. 169–195, vol. 119.

[22] G. H. Hardy and E. M. Wright , *An Introduction to the Theory of Numbers*. Oxford: Oxford University Press, 1979.

[23] F. P. Preparata and M. I. Shamos, *Computational Geometry*, . Berlin: Springer Verlag, 1985.

[24] D. E. Knuth, *The Art of Computer Programming, Vol. 2, Seminumerical Algorithms*. Reading, MA: Addison-Wesley, 1969.

[25] L. Dorst and A. W. M. Smeulders, "Length estimators for digital contours," *Comput. Vision Graphics Image Processing*, vol. 40, pp. 311–333, 1987.

[26] J. O'Rourke, "An on-line algorithm for fitting straight lines between data ranges," *Commun. ACM* , vol. 24, no. 9, pp. 574–578, 1981.


# Segmenting Handwritten Signatures at Their Perceptually Important Points

Jean-Jules Brault and Réjean Plamondon

***Abstract*— This correspondence describes a new algorithm for segmenting continuous handwritten signatures sampled by a digitizer. The segmentation points are found by means of a two-step procedure. The principal step is to construct a function that weights the perceptual importance of every signature point according to its specific neighboring points. The second step points out the various local maxima of this function that correspond where the signature should be segmented. The method is well illustrated and tested on a number of signatures that require different kinds of segmentation decisions.**

***Index Terms*—Corner detection, handwritten curve partitioning, handwritten signature, perceptual importance of an angle, segmentation.**

## I. Introduction

The goal of an automatic signature verification (ASV) system is to confirm or invalidate the presumed identity of a signer from information obtained during execution of its signature. The techniques

Manuscript received December 20, 1989; revised February 8, 1993. This work was supported by grant CRSNG-A0915 and FCAR-AS-2655 and by the Ecole Polytechnique d Montréal. Recommended for acceptance by Editor-in-Chief A. K. Jain.

The authors are with Laboratoire Scribens, Département de Génie Electrique et Génie Informatique, Ecole Polytechnique de Montréal, Montréal, Canada H3C 3A7.

IEEE Log Number 9209988.

that have been proposed in the literature for designing ASV systems may be divided into two major groups: those based on functions derived from recorded signatures during their execution and those based on certain specific global parameters. The first approach has yielded better results to date (as measured by rates of false rejections and acceptances), but further improvements must be made in order to lower the error rates to acceptable levels for general public applications [9]. We are concerned here by systems that record dynamic information (mainly the execution sequence) from a digitizer.

It is generally accepted that highly unstable and easily imitated signatures are among the main causes of deterioration in ASV system performances since the threshold to settle genuine and authentic signatures is then very difficult to adjust. One proposed way to help solve this problem is to model a typical imitator by specifying the various steps that must be done to copy, both visually and dynamically, any signature [2], [4]. Indeed, with this model, it is possible to adapt the threshold according to the intrapersonal variations of a signature and according to how difficult it is to forge by a potential imitator [3].

One of the first steps a forger has to do is look into a signature to extract its perceptually important points. Consequently, a segmenting algorithm must be able to find accurately the *apex* of each peak present along the handwritten curve. However, according to Fischler and Bolles [5], different goals for the segmenting of a *continuous* plane curve could result in different segmentation points, even if the curve is very simple. Our goal here is to segment a handwritten signature to allow a person to rebuild it with a minimal number of points and to produce segments that are as close as possible to the psychomotor reality of their execution.

Several interesting techniques that segment continuous lines in various ways have already been proposed in the literature in fields other than ASV. Three typical approaches are briefly presented in the following paragraphs for illustrative and comparative purposes.

The "split and merge" algorithm proposed by Pavlidis and Horowitz [8] is based on an iterative approximation of a curve by straight segments that drive an error norm under a specified threshold. This method is suitable for the approximation of a curve by a polygonal line but has a tendency to result in too many segmentation points; moreover, they are not always well centered on the apex of the peaks.

The technique proposed by Kruse and Rao [7] is based on calculating a "sliding correlation" between a mathematical "model of a corner (or vertex)" and portions of the curve joining $s$ points. The apex of the "corner" must correspond to the local maxima of the correlation function. The main shortcomings of this method are, in our opinion, the too-restrictive definition of the corner model and the arbitrary fixed domain ($s$ points) of every possible vertex. Consequently, it would make it impossible to adequately quantify the importance of a vertex made up of much more than $s$ points.

Freeman and Davis [6] proposed another type of segmentation technique that also involves a "sliding" analysis of portions of the curve joining $s$ points. In this case, however, the $s$ points are used to locate the discontinuities along the curve. Three indices are calculated for each successive (and overlapping) portion of $s$ points; one index $C(i)$ is related to the severity of the curvature combining these $s$ points, and the other two are related to the length of the discontinuity-free region (backwards ($l_b(i)$) and forwards ($l_f(i)$)) from the point $i$. The importance of a given vertex $i$ is calculated with the formula

$$Imp(i) = C(i)\sqrt{l_b(i) * l_f(i)}. \quad (1)$$

The segmentation points are the ones located, after a proper filtering, at the maxima of $Imp(i)$. This interesting method, however, suffers the same kind of shortcoming as the Kruse and Rao technique [7]; it fixes arbitrarily the domain where a potential vertex could be found and is not able to detect "long and smooth" corner.

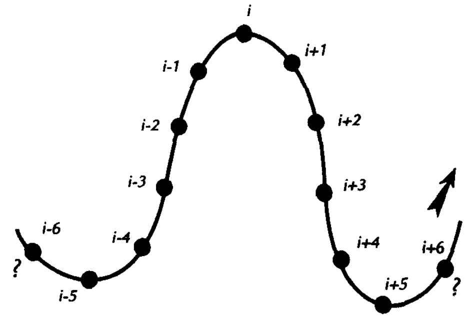


Fig. 1. Curve showing the points belonging to the domain of vertex $i$.

## II. A New Segmentation Algorithm

The segmentation algorithm that we propose in this paper (an early version was published in [1]) is roughly similar in some respects to the two proposed by Kruse and Rao [7] and by Freeman and Davis [6]. However, we consider that a corner could be made of any number of points, and the algorithm itself must determine the length and the specific domain of every potential vertex.

Moreover, in a recent model, Plamondon [11] has proposed a general segmentation framework for the analysis of handwriting based on a theory of rapid movements [10]. According to that theory, handwriting is made of curvilinear and angular strokes that are partially superimposed due to some anticipation effects occurring in the generation of fast complex movements. Since the beginning and end of these strokes are partially hidden in the signal, it is shown that a consistent handwritten segmentation theory should take into account large units of handwritten signals because one of the most efficient ways to extract these underling strokes is to perform an analysis-by-synthesis experiment over a whole *component* (i.e., pentip traces produced during a continuous pen-down movement).

### A. Background

Knowing that signatures are considered to be plane curves made of sequentially equidistant points, the main idea of the algorithm is the following. For each point $i$ of this curve, the algorithm tries to iteratively construct a vertex centered on that point with the help of neighboring points to either side of it until some conditions were met.

As an example, it is seen in Fig. 1 that there must exist some geometric conditions such that some pairs of points cannot be considered to be part of the *domain* of a vertex centred on $i$ (such as the pair of neighbors $(i \pm 6)$). Furthermore, we note that the pairs of points of the $i$ domain do not contribute with equal weight to making the vertex $i$ look important.

### B. Parametric Formulation

More precisely, the geometric parameters shown in Fig. 2 are calculated for each pair of neighbors $i \pm n($ *for* $n = 1, 2\ldots)$. Intuitively, the more the two angles $\theta_f(i,n)$ or $\theta_b(i,n)$ approach $\pi/2$, and the fewer the pair of straight segments associated with these angles contributes to making the point $i$ singular (relatively to its neighbors). As a result, by a suitable analysis of the angles $\theta_f(i,n)$ and $\theta_b(i,n)$, one can determine whether or not the pair of points $i \pm n$ are parts of the domain of $i$ and, in addition, estimates the importance of the contribution of these points.

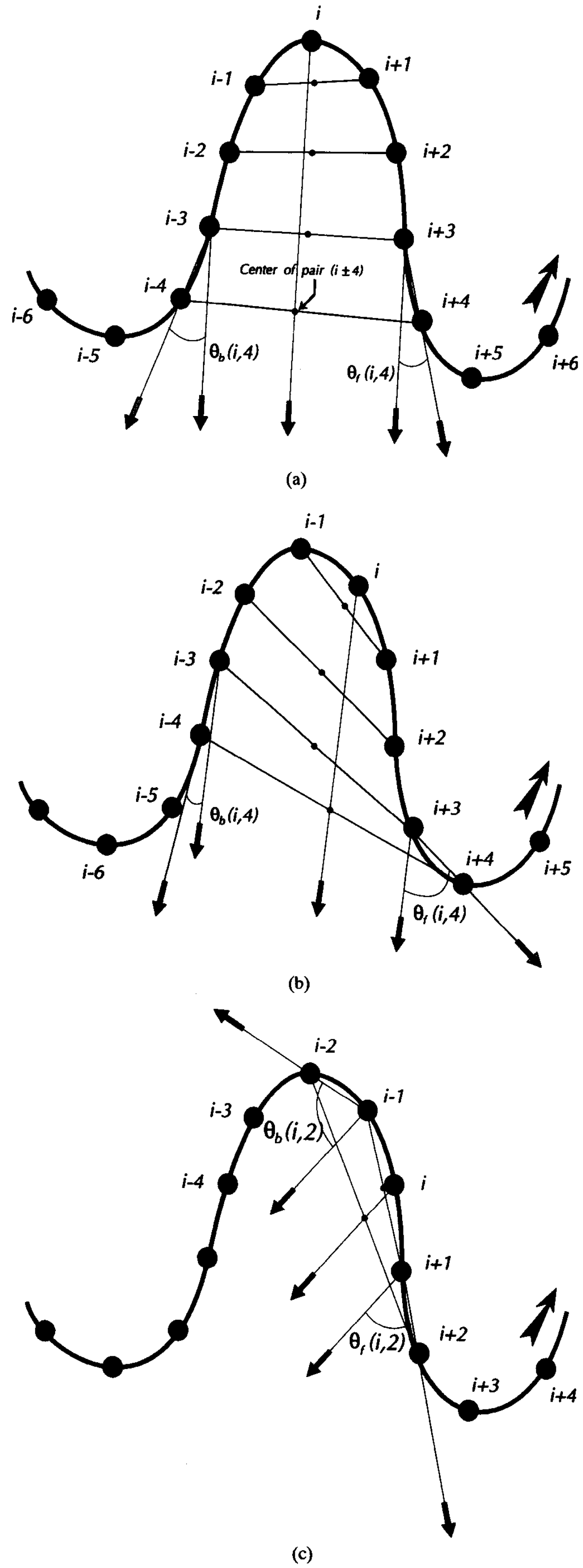


Fig. 2. Curve showing the various geometric parameters used to determine the domain and the perceptual importance of point $i$.

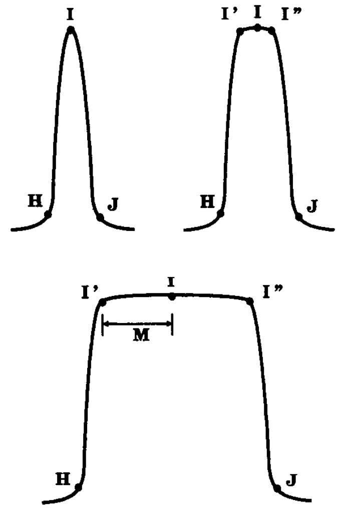


Fig. 3. Effect on perception of flattening the vertex of a curve.

Determining the conditions for which a pair $i \pm n$ belongs to the domain of vertex $i$ is straightforward. The angles $\theta_f(i,n)$ and $\theta_b(i,n)$ must satisfy the following inequality:

$$|\theta_f(i,n)| \text{ and } |\theta_b(i,n)| \prec \theta_{\max} \tag{2}$$

where $\theta_{\max}$ is a threshold value between 0 and $\pi/2$. The importance of their contribution is calculated by this empirical formula

$$IMP(i,n) = \cos(\theta_b(i,n)) * \cos(\theta_f(i,n)). \tag{3}$$

The trigonometric function cos was used for its adequate behavior in the range of angles concerned, whereas the multiplication operation takes into account the required simultaneous effect of the pair of points to make the vertex $i$ important.

The total contribution of the $N_d(i)$ points belonging to the domain of $i$ (the ones that satisfy the inequality (2)) is calculated by

$$FI(i) = \sum_{n=1}^{N_d(i)} IMP(i,n). \tag{4}$$

### C. Necessary Refinement

Conceptually, a vertex usually appears between two sides. However, a vertex is not always limited to one single point, and mutually, sides are not always made of straight lines. Therefore, a vertex could be interpreted as a side and vice versa. Fig. 3 illustrates the problem very well. In Fig. 3(a), the point I is, without question, the vertex of the curve but, as the vertex flattens (see Fig. 4(b) and (c)), two secondary vertices ($I'$ and $I''$) appear and became important at the expense of vertex $I$. One way of resolving this difficulty is to replace (4) with the following equation:

$$FI(i) = \sum_{n=N_0(i)}^{N_d(i)} IMP(i,n). \tag{5}$$

The original algebraic sum of (4) has been amputated by the $N_0(i)$ first contributions, where the value $N_0(i)$ is determined in the following way. Suppose that the *first* $M(i)$ *pairs* of neighbors of $i$ have their angles $\theta_f$ and $\theta_b$ greater than $\theta_{\max}$ (for example $\frac{3\pi}{8}$) since each point of these first $M(i)$ pairs go in nearly opposite directions from $i$; they weaken the perception of $i$ as a vertex. Consequently, it would be reasonable to inhibit the sum by an additional 2 $M(i)$ neighbors. In other words, the points $I', II', J$, and $H$ of Fig. 3(c) could be seen to be the vertex of a square, and thus, the point $I$

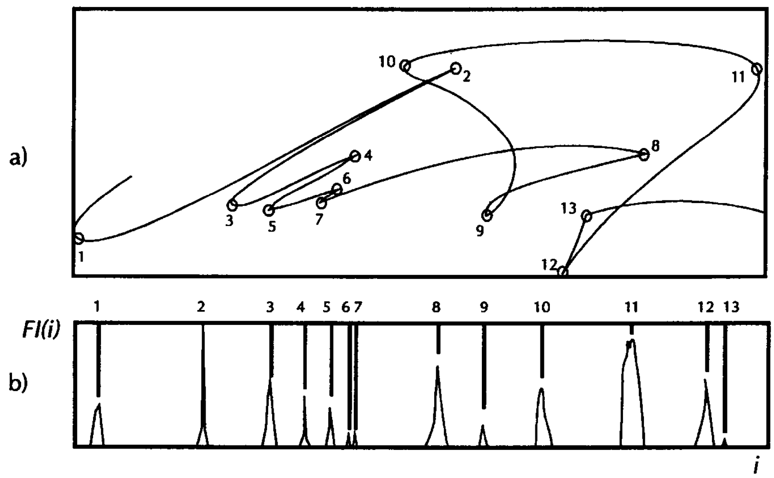


Fig. 4. (a) Signature and its corresponding segmentation points as found with function $FI(i)$; (b) function $FI(i)$ and the localization of local maxima $\theta_{\max} = 3\pi/8$, $K = 3$).

could no longer be considered to be a vertex. The value $N_0(i)$ would then be equal to $KM(i)$ with $K$ preferably being greater than or equal to 3.

### D. Locating the Segmentation Points

The identification of the segmentation points from the function $FI(i)$ is very simple because this function usually comes in the form of groups of nonzero values spaced by group of zero values. Each of the nonzero groups represents a vertex, and the maximum value of each group is said to be its most perceptually important point. To be efficient, the sampled signature data must be preprocessed with a moving average filter (with a gaussian kernel) to remove the noise that comes with any acquisition made with a digitizer.

There are, therefore, only two parameters to be predetermined ($\theta_{\max}$ and $K$) and, as will be shown in Section III, their exact values are not very crucial.

## III. Experimental Results

This new algorithm has been applied in an experiment [2] to segment 24 signatures requiring around 700 different segmenting decisions. Note that this signature database was built with the help of right-handed, North American, European, and Asian signers.

Figs. 4 to 7 show typical results obtained through the application of the method described in the preceding section. Fig. 4(a) depicts a signature to be segmented, and Fig. 4(b) shows the function $FI(i)$ obtained with parameters $\theta_{\max} = 3\pi/8$ and $K = 3$. The segmentation points numbered on the signature correspond to those shown on function $FI(i)$. It is interesting to note that the *amplitude* of $FI(i)$ indicates the necessity of choosing $i$ as segmentation points while its *width* indicates the uncertainty involved in choosing a single segmentation point for a given vertex. For example, the amplitude of $FI(i)$ at point #2 is greater than that at point #3, which is itself greater than the one at point #4. On the other hand, the large width of vertex #B expresses the fact that adjacent points could have also been chosen to the right, or to the left, of the one chosen automatically by the algorithm. In the case of vertex #2, the narrow width of $FI(i)$ expresses no ambiguity at all. These results seem to agree with our relative perception of each of these vertices.

Fig. 5 shows how function $FI(i)$ changes with variation of the parameter $\theta_{\max}$ (where parameter $K$ is fixed at 2 instead of 3 to better see the effect of $\theta_{\max}$). It may be seen that function $FI(i)$ becomes more selective as $\theta_{\max}$ diminishes. When $\theta_{\max}$ is fixed at $\pi/2$ (its maximum value), the central point between vertices #A and #B in Fig. 4(a) (indicated by an arrow in Fig. 5(a)) is considered to be a vertex, as previously discussed about point I of Fig. 3(c). When $\theta_{\max}$ diminishes, the importance of this point tends rapidly toward zero. If, however, $\theta_{\max}$ becomes too small, the value of $FI(i)$ at vertex #B, for example, will also become too small.

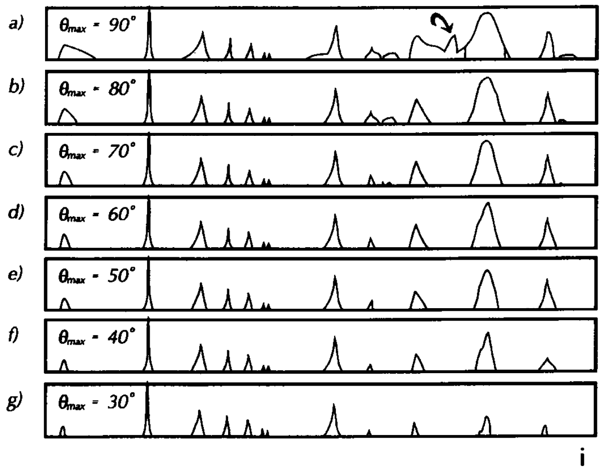


Fig. 5. Evolution of the function $FI(i)$ with variation of parameter $\theta_{\max}$ (with $K = 2$).

The particular values of the thresholds, once fixed, are perhaps debatable since thresholds that are considered to be adequate for one signature may not be adequate for another (pattern recognition is the art of the thresholding ...). Nevertheless, the variations in $FI(i)$ are minor within a relatively wide range of values of its parameters. For example, some signatures segmented with the same parameters as those used for a signature in Fig. 4 are depicted in Figs. 6 and 7 with their segmentation points.

One of the main qualities of the method, besides its simplicity, is probably its scale factor independence. Indeed, it is able to identify the segmentation point of the very large and smooth vertex #1 (or

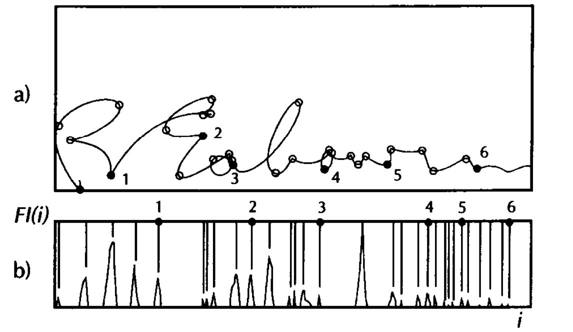


Fig. 6. Signature segmented with function $FI(i)(\theta_{\max} = 3\pi/8, K = 3)$.

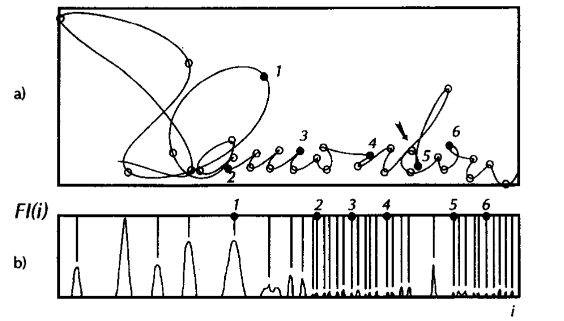


Fig. 7. Signature segmented with function $FI(i)(\theta_{\max} = 3\pi/8, K = 3)$.

the next one after #1) of Fig. 7 as well as the small and acute vertex #4. A drawback of the method (and perhaps of every method?) is its difficulty to segment an almost complete circle, as in the one shown after point #5 in Fig. 7. Indeed, function $FI(i)$ of Fig. 7(b) appears with two local maxima without the function passing through zero. The segmenting point indicated by an arrow on Fig. 7(a) was added manually afterwards for the sake of the discussion. To overcome this problem, it is necessary to diminish the value of $\theta_{\max}$, allowing $FI(i)$ to reach zero between the two peaks.

## IV. Conclusion

We have presented an algorithm that makes it possible to estimate the perceptual importance of each of the points of a signature (or other types of continuous cursive handwriting) as a basis for its segmentation. The main idea of the algorithm is that for each point $i$ of the signature, it tries to iteratively construct a vertex centred on that point with the help of neighboring points to either sides of it until certain geometric conditions are met. The method has been applied successfully to a signature database, and the location and relative importance of the segmentation points are generally in agreement with human perception. Moreover, they are also in accordance with our most recent segmentation theory [11]. An interesting application of the algorithm is to use it to quantify one of the difficulties (at the perception level) that could be experienced by a typical imitator in reproducing a signature [2], [4]. This difficulty index, together with an intrapersonal variation index, could be used to identify problematic signers in a particular signature database and adapt the thresholds of the ASV system to improve its overall performance.

One object of our continuing research effort is to implement the algorithm on a neural network and automatically fix the optimal thresholds of the only two parameters of the method.

## References

[1] J. -J. Brault and R. Plamondon, "Handwritten curve partitioning based on geometrical and sequential information," in *Proc. Third Int. Symp. Handwriting Comput. Applications* (Montréal), July 1987, pp. 56–58.

[2] J. -J. Brault, "Proposition et vérification d'un coefficient de difficulté d'imitation des signatures manuscrites," Ph.D. thesis, Ecole Polytechnique de Montréal, Canada, 384 pages, Dec. 1988.

[3] J. -J. Brault and R. Plamondon, "How to detect problematic signers for automatic signature verification," in *Proc. 1989 Int. Carnahan Conf. Security Technology* (Zurich), Oct. 1989, pp. 127–132.

[4] ——, "A complexity measure of handwritten curves: Modeling of dynamic signature forgery," to be published in *IEEE Trans. Syst. Man Cybern.*, vol. 23, no. 2, Mar./Apr. 1993.

[5] A. Fishler and R. C. Bolles, "Perceptual organisation and curve partitioning," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. PAMI-8, no. 1, pp. 100–105, Jan. 1986.

[6] H. Freeman and L. Davis, "A corner-finding algorithm for chain-coded curves," *IEEE Trans. Comput.*, vol. C-26, pp. 297–303, Mar. 1977.

[7] B. Kruse and C. V. K. Rao, "A matched filtering technique for corner detection," in *Proc. 4th Int. Conf. Patt. Recogn.* , 1978, pp. 642–644.

[8] T. Pavlidis and S. T. Horowitz, "Segmentation of plane curves," *IEEE Trans. Comput.*, vol. C-23, pp. 860–870, Aug. 1974.

[9] R. Plamondon and G. Lorette, "Automatic signature verification and writer authentication: The state of the art," *Patt. Recogn.*, vol 22, no. 2, pp 107–131, 1989.

[10] R. Plamondon, "A theory of rapid movements," in *Tutorials in Motor Behavour II* (G. E. Stelmach and J. Requin, Eds.). Amsterdam: Elsevier, North-Holland, 1992, pp. 55–69.

[11] ——, "A model-based segmentation framework for computer processing of handwriting," in *Proc. 11th Int. Conf. Patt. Recogn.* (The Hague, The Netherlands), Sept. 1992, pp. 303–307.

# Fast Nearest-Neighbor Search in Dissimilarity Spaces

András Faragó, Tamás Linder, and Gábor Lugosi

***Abstract*—A fast nearest-neighbor algorithm is presented. It works in general spaces where the known cell (bucketing) techniques cannot be implemented for various reasons, such as the absence of coordinate structure and/or high dimensionality. The central idea has already appeared several times in the literature with extensive computer simulation results. This paper provides an exact probabilistic analysis of this family of algorithms, proving its $O(1)$ asymptotic average complexity measured in the number of dissimilarity calculations.**

***Index Terms*—Average complexity, dissimilarity spaces, fast nearest-neighbor search, pattern recognition, probabilistic analysis of algorithms.**

## I. Introduction

Finding a nearest neighbor of a point among several others is a task one often encounters in a number of practical situations such as vector quantization of signals, pattern recognition, etc. In a Euclidean space, this is one of the so-called closest-point problems of computational

Manuscript received June 25, 1991; revised January 6, 1992. Recommended for acceptance by Editor-in-Chief A. K. Jain.

The authors are with the Technical University of Budapest, Budapest, Hungary.

IEEE Log Number 9209991.

0162–8828/93$03.00 © 1993 IEEE