On-line Signature Verification Algorithm Incorporating Pen Position, Pen Pressure and Pen Inclination Trajectories

D.Sakamoto,H.Morita,T.Ohishi,Y.Komiya,T.Matsumoto
Department of Electrical, Electronics and Computer Engineering Waseda University
3-4-1 Ohkubo, Shinjuku-ku, Tokyo, Japan, 169-8555

ABSTRACT

A new algorithm is proposed for pen-input on-line signature verification incorporating pen-position, pen-pressure and pen-inclinations trajectories. Preliminary experimental result looks encouraging.

1. INTRODUCTION

1.1 Propose

Personal identity verification has a great variety of applications including access to computer terminals, buildings, credit card verification, to name a few[1]. Algorithms for personal identity verification can be roughly classified into four categories depending on static/dynamic and biometric/physical or knowledge-based as shown in Fig1.1. (This figure has been partly inspired by a brochure from Cadix Corp, Tokyo.) Fingerprints, iris, retina, DNA, face, blood vessels, for instance, are static and biometric. Algorithms which are biometric and dynamic include lip movements, body movements and on-line signature. Schemes which use passwords are static and knowledge-based, whereas methods using magnetic cards and IC cards are physical.

This paper proposes a new algorithm PPI (pen-position/pen-pressure/pen-inclination) for *on-line pen input signature verification*. The algorithm considers writer's signature as a trajectory of pen-position, pen-pressure and pen-inclination which evolves over time, so that it is dynamic and biometric. Since the algorithm uses pen-trajectory information, it naturally needs to incorporate stroke number (number of pen-ups/pen-downs) variations as well as shape variations. The proposed scheme first generates templates from several authentic signatures of individuals. In the verification phase, the scheme computes a distance between the template and input trajectory. Care needs to be taken in computing the distance function because; (i) length of a pen input trajectory may be different from that of template even if the signature is genuine; (ii) number of strokes of a pen input trajectory may be different from that of template, i.e., the number of pen-ups/pen-downs obtained may differ from that of template even for an authentic signature.

If the computed distance dose not exceed a threshold value, the input signature is predicted to be genuine, otherwise it is predicted to be forgery.

A preliminary experiment is performed on a database consisting of 293 genuine writings and 540 forgery writings, from 8 individuals. Average correct verification rate was 97.6 % whereas average forgery rejection rate was 98.7 %. Since no fine tuning was done, this preliminary result looks very promising.

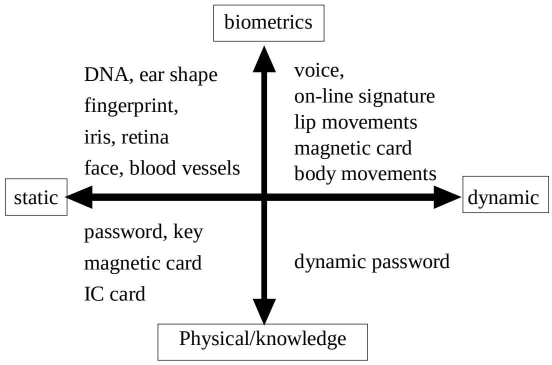

Fig1.1 Authentication Methods

1.2 Related Works

Kato et. al. [2] use pen position and pen pressure for on-line signature verification while Taguchi et. al. [3] use pen inclination. The algorithm proposed in [4] computes distances between input and templates for each stroke so that there are difficulties when stroke number varies.Yoshimura et. al.[5] use the direction of pen movement for on-line signature verification. One of the main distinctions between the previous works and our algorithm PPI given below lies in the fact that the latter uses the trajectory of pen-position, pen-pressure and pen-inclinations in a combined manner.

2. The Algorithm

2.1 Overall algorithm

Figure 2.1 describes an overall algorithm of PPI.

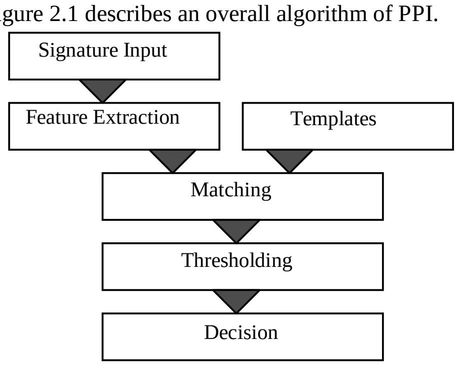

Fig 2.1 Overall algorithm

2.2 Feature Extraction

The raw data available from our tablet (WACOM Art Pad 2 pro Serial) consists of five dimensional time series data:

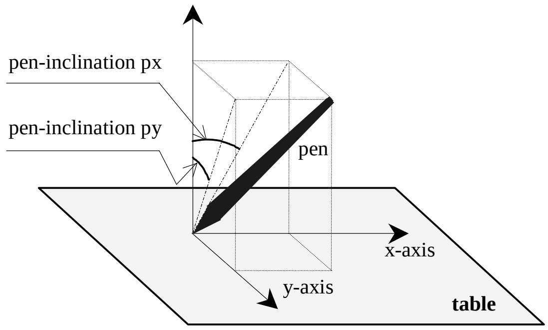

Fig2.2 Raw data from tablet

$$(x(t_i), y(t_i), p(t_i), px(t_i), py(t_i)) \in R^2 \times \{0,1,...,255\} \times R^2 \quad i = 1,2,..., I \qquad (2.1)$$

where $(x(t_i), y(t_i)) \in R^2$ is the pen position at time t_i, $p(t_i) \in \{0,1,...,255\}$ represents the pen pressure, $px(t_i)$ and $py(t_i)$ are pen inclinations with respect to the x - and y -axis as shown in Fig 2.2. Usually, $t_i - t_{i-1} \approx 5ms$ so that there are too many points which is not appropriate. Uniform resampling often

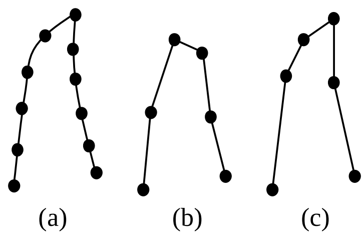

Fig2.3 Our resampling algorithm preserves sharp corners

results in a loss of important features. Consider, for instance, the raw data given in Fig 2.3(a). If one resamples the data uniformly then the sharp corner may be lost as is shown in Fig 2.3(b). Our resampling procedure checks if

$$\theta_i := \tan^{-1} \frac{y(t_i) - y(t_{i-1})}{x(t_i) - x(t_{i-1})} \leq \theta^* \qquad (2.2)$$

where θ^* is a threshold value. If (2.2) holds, then $(x(t_i), y(t_i))$ is eliminated, otherwise it is kept. This typically gives Fig 2.3(c) which retains a sharp corner while portions of pen trajectory without sharp corners retain information with smaller number of points. This is a preprocessing done in our pen-input on-line character recognizers which worked very well [6][7]. Details are omitted.

Let

$$\Delta f_i := \sqrt{(x(t_i) - (x(t_{i-1}))^2 + (y(t_i) - (y(t_{i-1}))^2} \qquad (2.3)$$

then our feature consists of the following five dimensional data

$$(\theta_j, \Delta f_j, p_j, px(t_i), py(t_i)) \in R^2 \times \{0,1,..., N\} \times R^2 \quad i = 1,2,..., I \quad j = 1,2,..., J \qquad (2.4)$$

Our verification algorithm described below computes a weighted sum of three different distance measures between an input data and stored templates.

2.3 Angle-Arc Length Distance Measure

Let

$$(\eta_l, \Delta g_l, q_l, qx(t_k), qy(t_k)) \in R^2 \times \{0,1,..., N'\} \times R^2 \quad k = 1,2,..., K \quad l = 1,2,..., L \qquad (2.5)$$

be the resampled feature trajectory of a template and consider

$$\left|\theta_j - \eta_l\right| d(p_j, q_l) \rho(\Delta f_j, \Delta g_l) \qquad (2.6)$$

where

$$d(p_i, q_j) := \left|p_i - q_j\right| + 1$$

incorporates pen-pressure information. The last term "1" is to avoid zero value of a $d(p_i, q_j)$. Function ρ is defined by

$$\rho(\Delta f_j, \Delta g_l) := \sqrt{\Delta f_j^2 + \Delta g_l^2} \qquad (2.7)$$

which is to take into account local arc length of the trajectories. Generally $K \neq I, L \neq J$ even when signatures are written by the same person so that time-warping is necessary to compute (2.6) over the whole trajectories. The following is our angle arc length distance measure

$$\boxed{D1 := \min_{\substack{j_s \leq j_{s+1} \leq j_s + 1 \\ l_s \leq l_{s+1} \leq l_s + 1}} \sum_{s=1}^{S} \left|\theta_{j_s} - \eta_{l_s}\right| d(p_{j_s}, q_{l_s}) \rho(\Delta f_{j_s}, \Delta g_{l_s}) \qquad (2.8)}$$

where

$$j_1 = l_1 = 1, \; j_s = J, l_s = L$$

are fixed.

Because of the sequential nature of the distance function, Dynamic Programming [8] is a feasible means of the computation:

$$D1(0,0) = 0$$

$$D1(j_s, l_s) = \min \begin{cases} D1(j_s - 1, l_s - 1) + \left|\theta_{jS} - \eta_{lS}\right| \\ \quad \times d(p_{jS}, q_{lS})\rho(\Delta f_{jS}, \Delta g_{lS}) \\ D1(j_s - 1, l_s) + \left|\theta_{jS} - \eta_{lS}\right| \\ \quad \times d(p_{jS}, q_{lS})\rho(\Delta f_{jS}, 0) \\ D1(j_s, l_s - 1) + \left|\theta_{jS} - \eta_{lS}\right| \\ \quad \times d(p_{jS}, q_{lS})\rho(0, \Delta g_{lS}) \end{cases}$$

$$d(p,q) = |p - q| + 1$$

$$\rho(\Delta f, \Delta g) = \sqrt{\Delta f^2 + \Delta g^2}$$

2.4 Pen Inclination Distances

Define pen-inclination distances

$$\boxed{D2 := \min_{\substack{i_{s'} \le i_{s'+1} \le i_{s'}+1 \\ k_{s'} \le k_{s'+1} \le k_{s'}+1}} \sum_{s'=1}^{S'} \left| px_{i_{s'}} - qx_{k_{s'}} \right| \qquad (2.9)}$$

$$\boxed{D3 := \min_{\substack{i_{s''} \le i_{s''+1} \le i_{s''}+1 \\ k_{s''} \le k_{s''+1} \le k_{s''}+1}} \sum_{s''=1}^{S''} \left| py_{i_{s''}} - qy_{k_{s''}} \right| \qquad (2.10)}$$

which are computable via DP also.

2.5 Distance Measure Plots

Figure 2.4(a) is a scatter plot of $(D1, D2, D3)$ consisting of 150 authentic signatures (triangle) and 351 forgery signatures (square) taken from eight individuals. Figure 2.4(b),(c), and (d) shows the projections onto the $(D1, D2)$ -plane, $(D2, D3)$ -plane and $(D1, D3)$ -plane respectively

These plots naturally suggest that there should be a two dimensional surface which could separate authentic signatures from forgeries reasonably well even though perfect separation may not be achieved.

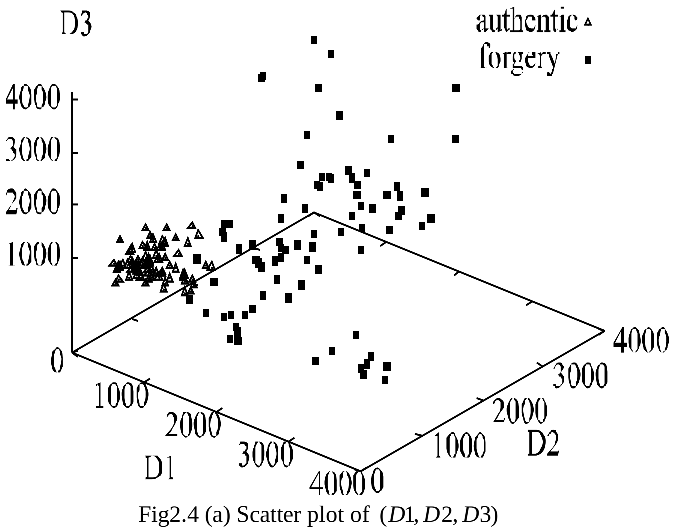

Fig2.4 (a) Scatter plot of $(D1, D2, D3)$

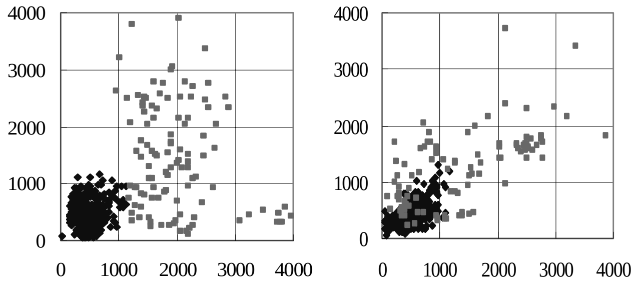

Fig2.4 (b) ,(c) Projection onto the $(D1, D2)$ -plane and Projection onto the $(D2, D3)$ -plane

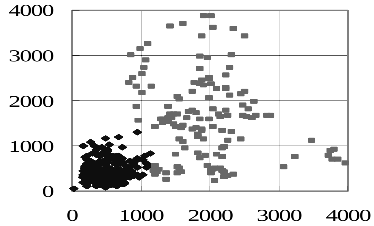

Fig2.4 (d) Projection onto the $(D1, D3)$ -plane

2.6 Template Generation

In order to explain our template generation procedure, recall two types of errors in signature verification:

a) Type I Error (False Rejection Error)

b) Type II Error (False Acceptance Error)

Given m_0 authentic signature trajectories, divide them into two group S_1 and S_2 consisting of m_1 and m_2 trajectories, respectively, where the former is to generate templates while the latter is for verification test. We compute the total squared distance $D^2 = (D1)^2 + (D2)^2 + (D3)^2$ between each of the signatures in S_1 and sort them according to their distances between each other. Choose three signatures with the smallest D^2. These three will be used as templates.

2.7 Threshold Value

In order to select the threshold value for distance between input and template, compute the $3 \times (m_1 - 3)$ distances between the chosen three and the remaining $m_1 - 3$ signatures and let the threshold value Th be the average of five largest distances.

2.8 Signature Verification

Note that three template signatures are generated for each individual. Given an input signature, compute the squared distance measure between it and the three templates and let $(D_{\min})^2$ be the smallest. We introduce a parameter $c \in [0.5, 2.0]$ to be selected and the input is predicted to be authentic if

$$(D_{\min})^2 \le c \cdot Th$$

while the input is predicted as a forgery if

$$(D_{\min})^2 > c \cdot Th\,.$$

3. Experiment

This section reports our preliminary experiment using the algorithm described above. Eight individuals participated the experiment. The data were taken for the period of three months. There are 861 authentic signatures, 1921 forgery signatures and 205 signatures for template generation. Table3.1 shows the details. Figure 3.1 shows average verification error as a function of parameter c described above, where the intersection between Type I Error and Type II Error curves gives 3.0%. Figure 3.2 shows the error curves of individual "B" where zero error is achieved at c =1.1

Figure 3.3(a) is an unsuccessful attempt of a forgery rejected by our algorithm while Fig. 3.3(b) is an authentic signature accepted by the PPI.

Table 3.1 Data for Experiment

individuals	authentic		forgery	total
	test	Template generation	test	
A	184	45	585	814
B	40	10	81	131
C	126	30	237	393
D	24	6	68	98
E	173	39	435	473
F	52	12	71	135
G	172	42	288	502
H	91	21	156	268
total	861	205	1921	2987

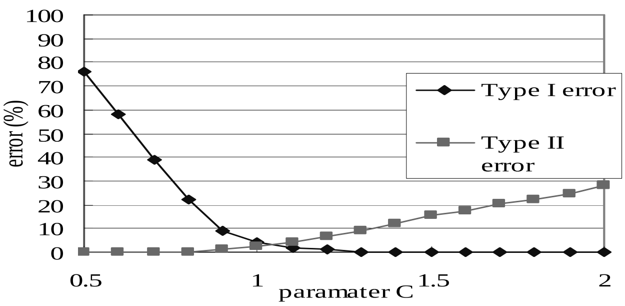

Fig 3.1 Average verification error

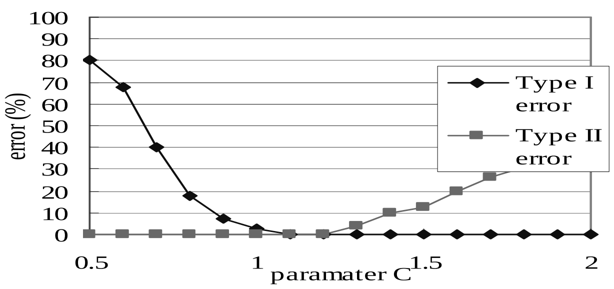

Fig 3.2 The error curves of individual "B"

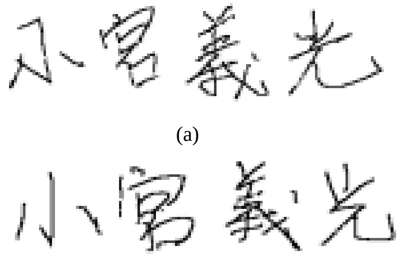

Fig 3.3 (a) Forgery rejected by the PPI algorithm.
(b) Genuine signature accepted by the PPI.

.

4. Discussion

So far signatures written in the Japanese Kanji characters are considered. Signatures written in the Western alphabets are interesting to study. Statistical methods will be of worth considering.

5. REFERENCES

[1] T. Tabuki, "Personal Identification by Signature, Proc. Seminar on Sensing Technology and Applications, Osaka, Japan, 1997.

[2] Masami KATO, Yoshihiro fKAWASHIMA, "Signature Verification Using Online Data such as Pen Pressure and Velocity", IPSJ, pp.2-199, 1993.

[3] Hideo TAGUCHI, Koichi KIRIYAMA, Eiji TANAKA and Katsuhiko FUJII, "On-Line Recognition of Handwritten Signatures by Feature Extraction of the Pen Movement",IEICE, D Vol.J71-D No.5 pp.830-840, 1988.

[4] Chang Ji JIN, Masaharu WATANABE, Toshio KAWASHIMA and Yoshinao AOKI "On-Line Signature Verification by Non-public Parameters", IEICE, D-II Vol.J75-D-II No.1 pp.121-127, 1992.

[5] I Yoshimura, and M Yoshimura, "On-line Signature Verification Incorporating the Direction of Pen Movement", Trans.IEICE Jpn., E74, pp.2803-2092, 1991.

[6] S.Masaki, M.Kobayashi, O.Miyamoto, Y. Nakagawa and T.Matsumoto "An On-Line Hand-writing Character Recognition Algorithm RAV（Re-parameterized Angle Variations）", 4th ICDAR, Proceedings Vol.2, pp.919-925, 1997.

[7] T.Takahashi, H.Yasuda and T.Matsumoto "On-LineHandwriting Character Recognition Using Hidden Markov Models", 4th ICDAR97, Proceed-ings Vol.1, pp.369-375, 1997.

[8] R.Bellman "Dynamic Programming", Princeton University Press, Princeton, New Jersey, 1957.

[9]Y. Komiya and T. Matsumoto. "On-line Pen Input Signature Verification PPI (pen-Position /pen-Pressure /pen-Inclinations)", IEEE SMC'99, Vol.4, pp41-46, 1999.