An HMM On-line Signature Verifier Incorporating Signature Trajectories

Daigo Muramatsu Takashi Matsumoto
Department of Electrical Engineering and Bioscience
Waseda University, Tokyo, Japan
daigo@matsumoto.elec.waseda.ac.jp
takashi@mse.waseda.ac.jp

Abstract

Authentication of individuals is rapidly becoming an important issue. On-line signature verification is one of the methods that use biometric features. This paper proposes a new HMM algorithm is for on-line signature verification. After preprocessing, input signature is discretized in a polar coordinate system. This particular discretization leads to a simple procedure for assigning initial state and state transition probabilities.

This paper utilizes only pen position trajectories, no other information is used which makes the algorithm simple and fast. A preliminary experiment shows that the proposed algorithm appears to be promising.

1. Introduction

Personal identity verification has a great variety of application including Electrical Commerce, access to computer terminals, buildings, credit card verification, and so on. Algorithm for personal identification can be roughly classified into four categories depending on static/dynamic and biometric/physical-knowledge based as shown in Fig.1.

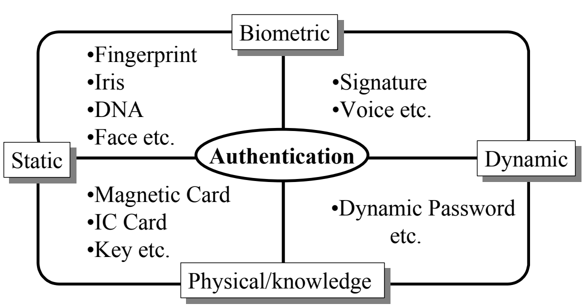


Fig.1 Authentication methods

Fingerprint, iris, DNA, face, for example, are in the group of static and biometric. Dynamic biometric methods include voice and online-signature. Schemes which use passwords are static and knowledge-based, whereas methods using magnetic cards and IC cards are static and physical.

There are at least two reasons that make online pen input signature verification be one of the very promising schemes for personal authentication. First, signature has a long history and is being already built in among many civilizations. Second, with the advent of growing number of pen-input devices including PDA's, tablet PC's among others, pen input environment is rapidly becoming a popular platform.

There are three types of forgery:

(A) Random forgery: Forger has no access to authentic signature.
(B) Simple forgery: Forger knows the name of the person whose signature is to be authenticated.
(C) Skilled forgery: Forger can view and train authentic signature.

In terms of training data sets, there are two cases to be distinguished:

(a) Forgery data as well as authentic signature data are available.
(b) No forgery data are available.

This paper proposes a new algorithm for pen input on-line signature verification using a discrete HMM and incorporating signature trajectories where no forgery data are available for training. Preliminary experiment is performed on a database consisting of 1848 genuine signatures and 3170 skilled forgery signatures, from fourteen individuals. The intersection of Type I Error and Type II Error curves gives 2.6%, which looks promising since only the pen position trajectories are used instead of pen pressure/pen inclination trajectories [1]-[7].

Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003)
0-7695-1960-1/03 $17.00 © 2003 **IEEE**

2. The Algorithm

The overall algorithm is shown in Fig.2. It consists of three sub-algorithms.

(i) Preprocessing, (ii) HMM learning, (iii) Verification

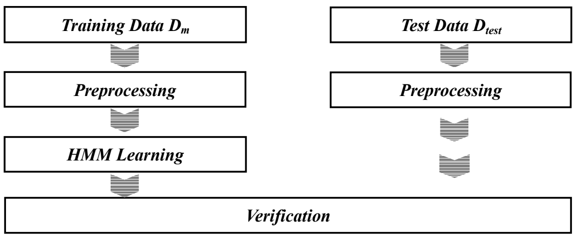


Fig.2 Overall algorithm

2.1 Preprosessing

Typical raw data taken from digitizer (shown in Fig.3) is

$$(x(t), y(t)) \in R^2, t = 1,2,......,T \qquad (1)$$

where $(x(t), y(t))$ is the pen position. The sampling rate is 100 points/sec.

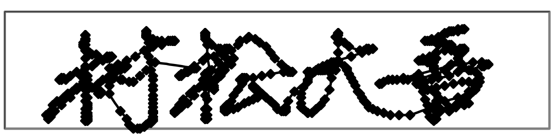


Fig.3 Typical raw data

Let

$$\theta := \tan^{-1} \frac{y(t+1) - y(t)}{x(t+1) - x(t)}, -\frac{\pi}{2} < \theta < \frac{\pi}{2} \qquad (2)$$

and define θ' by

$$\begin{aligned} &i)\ \theta' := \theta && if\ x(t+1) - x(t) \geq 0 \\ &ii)\ \theta' := \theta + \pi && if\ x(t+1) - x(t) < 0 \end{aligned} \qquad (3)$$

In order to formulate the problem in terms of discrete HMM, consider $V(t)$ which represents the quantized angle information defined by

$$i)\ V(t) := 1\ if\ \theta' < -\frac{\pi}{2} + \frac{\pi}{L}, and\ \theta' \geq \frac{3\pi}{2} - \frac{\pi}{L}$$

$$ii) V(t) := n\ if -\frac{\pi}{2} + \frac{(2n-3)\pi}{L} \leq \theta' < -\frac{\pi}{2} + \frac{(2n-1)\pi}{L}$$

$$n = 2,3,..., L \qquad (4)$$

Thus $(x(t), y(t))$ is transformed into L discretized angles shown in Fig. 4.

Thus, the given trajectory (1) is transformed into the

$$O(t) := V(t) \in \{1,......, L\} \qquad (5)$$

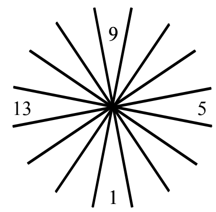


Fig. 4 Quantized directions (L=16)

2.2 HMM Learning

2.2.1 Proposed Structure of HMM. HMM is a general probabilistic structure which is applicable to a broad class of problems where time evolution is important.

Every general discipline must be tailored before being applied to a specific type of problem, which is a basic engineering function, and HMM is no exception. The general framework of HMM must be carefully tuned to the on-line signature verification problem. This section gives the complete details of the proposed HMM structure.

In order to make an HMM precise, let us first recall the output symbols defined in the previous section:

$$O(t) = V(t), t = 1,2,......, T-1 \qquad (6)$$

An HMM of a signature

$$H = H(\{a_{ij}\}, \{b_{ik}\}, \{\pi_j\}, N) \qquad (7)$$

is defined by the joint probability distribution of $\{Q(t), O(t)\}_{t=1}^{T-1}$ given H ;

$$P(\{Q(t), O(t)\}_{t=1}^{T-1} \mid H) = \pi_{Q(1)} \prod_{t=1}^{T-2} a_{Q(t)Q(t+1)} \prod_{t=0}^{T-2} b_{Q(t+1)V(t)} \qquad (8)$$

Where $Q(t)$ stands for hidden state at time t, $\{a_{ij}\}$ the state transition probability, $\{b_{ik}\}$, the output emission probability, and $\{\pi_i\}$ the initial state probability.

Learning in HMM amounts to an estimation of parameters $\{a_{ij}\}, \{b_{ik}\}, \{\pi_i\}$ and N, whereas verification is to compute posterior Probability $P(H \mid \{O(t)\}_{t=1}^{T-1})$ given a test signature $\{O(t)\}$ and attempt to make decisions.

In order to tune HMM to our current type of problem, we will use the **left to right model** so we put the following constraints on $\{a_{ij}\}$ and $\{\pi_j\}$:

i) $a_{ij} = 0$, unless $i = j$ or $i = j-1$, $a_{NN} = 1$

So that $\{a_{ij}\}$ matrix is restricted to the one in the form

$$\{a_{ij}\} = \begin{bmatrix} a_{11} & a_{12} & & & \\ & a_{22} & \ddots & 0 & \\ & & \ddots & a_{N-2,N-1} & \\ & & & a_{N-1,N-1} & a_{N-1,N} \\ & 0 & & & 1 \end{bmatrix} \quad (9)$$

ii) $\{\pi_j\} = (1, 0, \ldots, 0)$ (10)

Note that i) indicates that the last state, q_N , cannot transit any other state and also demands that state q_i cannot jump to $q_{j+k}, k \geq 2$, which corresponds to the causality of a trajectory associated with H , and ii) demands that the initial state $Q(1)$ must be always q_1 , The reasons for these constraints will become clear below.

2.2.2 Learning. The well-known Baum-Welch learning algorithm uses a gradient search to compute

$$\arg \max_{\{a_{ij}\},\{b_{ik}\},\{\pi_i\}} P(\{O(t)\}_{t=1}^{T-1} \mid \{a_{ij}\},\{b_{ik}\},\{\pi_i\},H) \quad (11)$$

with N fixed. The efficiency of this method naturally depends on each problem. In the current on-line signature verification problem, there are major hurdles; (i) In a realistic signature verification problem, only very few data sets are available for training. Namely, individuals will be willing to register only several signatures; ten signatures registrations will be out of question, even five signature registration is felt inconvenient, unless there is a particular reason. (ii) The likelihood function (11) is typically non-convex with respect to the parameters, and the serious problems are created by the local minima. (iii) The dimensions of the parameter space are determined by the sum of the non-zero parameters of $\{a_{ij}\},\{b_{ik}\},\{\pi_i\}$ and, N which amounts to $2N-1+N \times L+N$; the computational effort is significant.

Our learning algorithm given below is fast because it is one-shot, and hence non-iterative, and most importantly, works well.

2.2.3. Model Generation. When the data set is given for learning, our proposed algorithm first attempts to associate a clear meaning to the states. The algorithm, however, is still HMM in that it has nontrivial $\{b_{ik}\}$.

Let the data set $\{O(t)\}_{t=1}^{T-1}$ be given.

Step.1 State Clarification

Given the data set

$$O(t) = V(t), t = 1, \ldots\ldots, T-1 \quad (12)$$

make a division between $O(t)$ and $O(t+1)$, if $V(t) \neq V(t+1)$ i.e., if the angle value changes.

Consider the following simple example:

5 , 5 , 5	**8 , 8**	**9 , 9 , 9 , 9**	**10 , 10 , 10**
q_1	q_2	q_3	q_4

The first data $\{5\}$ means that the trajectory angle is in the direction 5 as shown in Fig. 4. After three such angles, the angle value changes to $\{8\}$ so that a division is made as indicated by a vertical line $\{|\}$, and so forth. The three repeated symbols $\{5,5,5\}$ are associated with state q_1 , the two repeated symbols $\{8,8\}$ are associated with state q_2 , and so forth.

Thus, given a training data set, the number of states is well defined, whereas the number of states in a general HMM learning is one of the most difficult parameters to be estimated.

Step.2 Learning $\{a_{ij}\},\{\pi_j\}$

Let $n(O,q_i)$ be the number of repetitions of the symbol $V(t)$ associated with q_i . Define

$$a_{ij} \equiv 0 \qquad (i \neq j, i \neq j-1) \quad (13)$$

$$a_{ii} \equiv \frac{n(O,q_i)-1}{n(O,q_i)} \qquad (1 \leq i \leq N-1) \quad (14)$$

$$a_{ii+1} \equiv \frac{1}{n(O,q_i)} \qquad (1 \leq i \leq N-1) \quad (15)$$

$$a_{NN} \equiv 1 \quad (16)$$

$$\pi \equiv (1, 0, \ldots\ldots, 0) \quad (17)$$

Care needs to be exercised in learning output emission probabilities in order to circumvent over-fitting.

Step.3 Learning $\{b_{jk}\}$

Let $V(j)$ be the angle value associated with state q_j and define

$$b_{jk} := \frac{\alpha}{Z(\sigma_V)} \int_{x=k-0.5}^{x=k+0.5} \exp\left(-\frac{(V(j)-x)^2}{2\sigma_V^2}\right) dx + \frac{1-\alpha}{L} \quad (18)$$

$Z(\sigma_V)$ is the normalization constant in order that

$$\sum_{k=1}^{L} b_{jk} = 1 \quad (19)$$

is satisfied, and α, σ_v are the hyperperameters.

2.3 Verification

The purpose of signature verification is to infer whether the test signature was written by the registered person or not.

Given training data set

$$D := \{D_1, \ldots, D_m, \ldots, D_M\},\ D_m = \{O(t)\}_{t=1}^{T_m - 1} \qquad (20)$$

consisting of M signature trajectories from a registered person, we create associated HMM's by the above algorithm. Note that the above HMM generation is one shot, i.e., each training data D_m generates one HMM.

In the verification phase, we are given a signature trajectory $D_{test} = \{O(t)\}_{t=1}^{T_{test}-1}$, from which inference should be made. We are interested in the model likelihood:

$$P(D_{test} \mid H_m) = \sum_{\substack{\text{All possible paths} \\ \{Q(t)\}_{t=1}^{T_{test}-1}}} P(\{Q(t), O(t)\}_{t=1}^{T_{test}-1} \mid H_m) \qquad (21)$$

This paper proposes the following performance index derived from (21):

$$\Theta_{test\,m} = \frac{\ln P(D_{test} \mid H_m)}{T_{test} - 1} \qquad (22)$$

Here T_{test} is the number of the raw data before preprocessing.

We infer that

$$D_{test} \text{ is authentic } \quad if \sum_m f(\Theta_{test\,m}, \lambda_m) \geq G$$

$$D_{test} \text{ is forgery } \quad if \sum_m f(\Theta_{test\,m}, \lambda_m) < G$$

where G is an empirical value indicating the number of times that the performance index exceeds threshold, and

$$f(\Theta_{test\,m}, \lambda_m) = \begin{cases} 1 & (\Theta_{test\,m} \geq \lambda_m) \\ 0 & (\Theta_{test\,m} < \lambda_m) \end{cases} \qquad (23)$$

λ_m is a threshold value, which is computed from the training data set D_m:

$$\lambda_m := \frac{1}{M}\sum_{n=1}^{M}\Theta_{n\,m} + c \times \sqrt{\frac{1}{M}\sum_{n=1}^{M}\left(\Theta_{n\,m} - \frac{1}{M}\sum_{l=1}^{M}\Theta_{l\,m}\right)^2} \qquad (24)$$

$$m = 1,2,\ldots,M,\ n = 1,2,\ldots,M,\ l = 1,2,\ldots,M$$

The error rates in the experiment in the next section will be reported as a function of c.

3. Experiment

This section reports our preliminary experiment using the algorithm described above. Fourteen individuals participated the experiment. The data was taken for the period of three months. There were 1848 authentic signatures and 3170 *skilled* forgery signatures.

Table1 shows the details of the signature data and Fig.5 shows some of the test data. Figure 6 shows the verification error of two individuals (Fig.6(a)(b)) and Total average verification error of 14 individuals (Fig.6(c)) as a function of parameter c described above, where the intersection of Type I and Type II Error curves gives 2.6%.

Table 1 The data used for experiment

Individuals	Authentic		Forgery	Total
	Test	Template generation	Test	
A	204	25	585	814
B	45	5	81	131
C	141	15	237	393
D	25	5	68	98
E	187	25	435	647
F	153	15	357	525
G	56	5	71	132
H	54	5	73	132
I	205	10	288	503
J	210	20	396	626
K	73	5	69	147
L	102	10	156	268
M	94	5	81	180
N	134	15	273	422
Total	1683	165	3170	5018

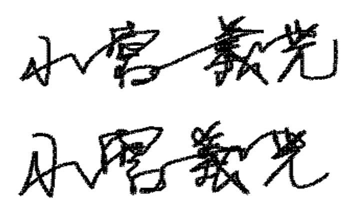


Fig. 5(a) Authentic Signatures

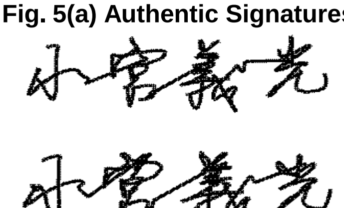
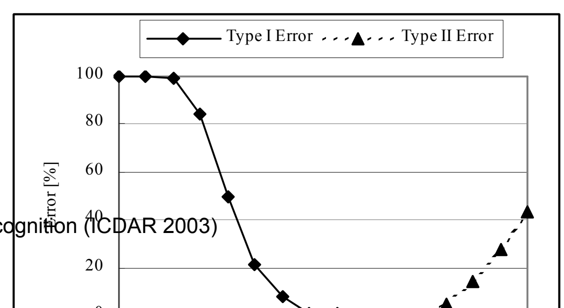


Fig. 5(b) Forgery Signatures

Fig.5 Test data for experiment



Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003)
0-7695-1960-1/03 $17.00 © 2003 **IEEE**

Fig.6(a) Verification error of Individual A

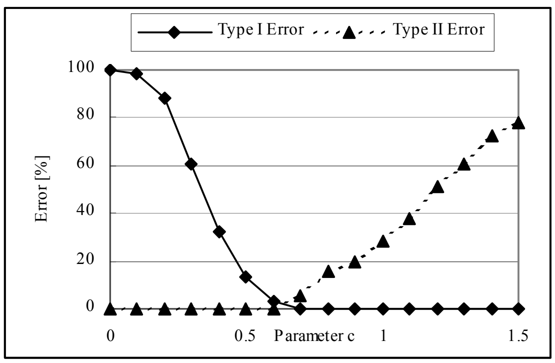


Fig.6(b) Verification error of Individual L

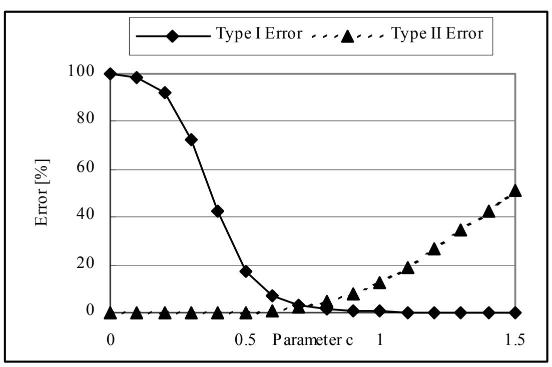


Fig.6(c) Total average verification error

4. Conclusions and Future Work

Since the proposed algorithm utilizes only pen position trajectories, there are at least three directions to pursue:

(i) Attempt to install the algorithm on a PDA where computational power is extremely limited;

(ii) Attempt to improve the verification ability by using additional information such as pen pressure, pen inclinations and so forth;

(iii) Try different HMM topologies.

5. Reference

[1] Y. Komiya and T. Matsumoto, "On-line Pen Input Signature Verification PPI (pen-Position /pen-Pressure /pen-Inclinations)", Proc. IEEE SMC'99, Vol.4, pp. 41-46, 1999.

[2] Y. Komiya, H. Morita, T. Matsumoto, "Pen-input On-line Signature Verification with Position, Pressure, Inclination Trajectories", Pros. IEEE IPDPS 2001, pp. 170, April 2001.

[3] Y. Komiya, T. Ohishi, T. Matsumoto: "A Pen Input On-Line Signature Verifier Integrating Position, Pressure and Inclination Trajectories", IEICE Trans. Information Systems, vol. E84 - D, No.7, July, 2001.

[4] T. Ohishi, Y. Komiya and T. Matsumoto, "An On-Line Pen Input Signature Verification Algorithm", Proc. IEEE ISPACS 2000, Vol.2, pp. 589-592, 2000.

[5] T. Ohishi, Y. Komiya and T. Matsumoto, "On-line Signature Verification using Pen Position, Pen Pressure and Pen Inclination Trajectories", Proc. IEEE ICPR 2000, Vol. 4, pp. 547-550, 2000.

[6] T. Ohishi, Y. Komiya, H. Morita, T. Matsumoto, "Pen-input On-line Signature Verification with Position, Pressure, Inclination Trajectories", Pros. IEEE IPDPS 2001, pp. 170, April 2001.

[7] D. Sakamoto, T. Ohishi, Y. Komiya, H. Morita and T. Matsumoto, "On-line Signature Verification Algorithm Incorporating Pen Position, Pen Pressure and Pen Inclination Trajectories", Proc. IEEE ICASSP 2001, Vol. 2, pp. 993-996, 2001.

[8] H. Yasuda, T. Takahashi and T. Matsumoto, "A Discrete HMM For Online Handwriting Recognition", International Journal of Pattern Recognition and Artificial Intelligence, Vol. 14, No.5, pp.675-688, 2000.

Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003)
0-7695-1960-1/03 $17.00 © 2003 **IEEE**