International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 4, Issue 4, April 2015

HMM Based Online Signature Verification

Toai Q.Ton[1], Tai Nhu Do[2]

***Abstract*—In this paper, we present a system using a set of time sequences and Hidden Markov Models (HMMs) to verify the online signatures. The signature is added to computer by the digital pen includes the following information: position trajectories, pressures and tilts of the pen. With the signature obtained, our system will extract 20 features in each points of the signature, andthe system will standardize these features before they enter into the system for training or testing. This standardization aims to enhance the accuracy of the system. When conducting authentication, a user's signature is considered to be correct when the similarity measure calculated by the system on which the signature is greater than a threshold. We have compared our system with other systems: the systems of competition authenticate Signatures First International (SVC2004) [6], the Lei system [5]. In the process of building the system, we have conducted several experiments to find out the specific effect of the signature and the optimal parameters for hidden Markov models.**

***Index Terms*— Biometrics, Online signature verification, Recognition, Verification, Hidden Markov Model, Dynamic Time Wrapping.**

I. INTRODUCTION

For a long time, handwritten signature has been accepted by law for recognition in the financial transaction, contract document, as well as being used to authenticate the materials, document.

When we use computer to collect signature by a digitizing device (electronic pen), we obtain information about the shape of the signature and we also obtained dynamic information of the signature. This dynamic information generates "online"signature.This concept shows a string of sample points shipping information during the process of signing up.In other words, each dynamic information is a function according to time of signing process. Thus, the signing process generates a set of the data function over time. Theonlinesignature helps facilitate forthe authentic signature because the dynamic information is more difficult to forge than the image of the signature. So, if anyone wantsto forge signatures, they need more work. However, this problem is still challenging problem in biometrics becauseof the large intra-class variation and when considering forgeries, small inter-class variation[1].

There are many different approaches in data classification of signature. The current methods can be divided into two classes:

1) Feature based approaches [2]. In this approach, a signature is represented by a vector consisting of a set of global features which are extracted from the trajectory of the signature.

2) Function based approaches [3]. In this approach, a signature is represented by the functions of the time, including the local properties of the signature (ex:position trajectories (x, y), velocities, accelerations, pressures, and more).s

In this paper, we study and apply of Hidden Markov Model (HMM) to model the signatures (the function based approach). The proposed system architecture is depicted in Fig. 1. The experimental results show that the system give a quite good result compared with other systems (will be described in IV section).

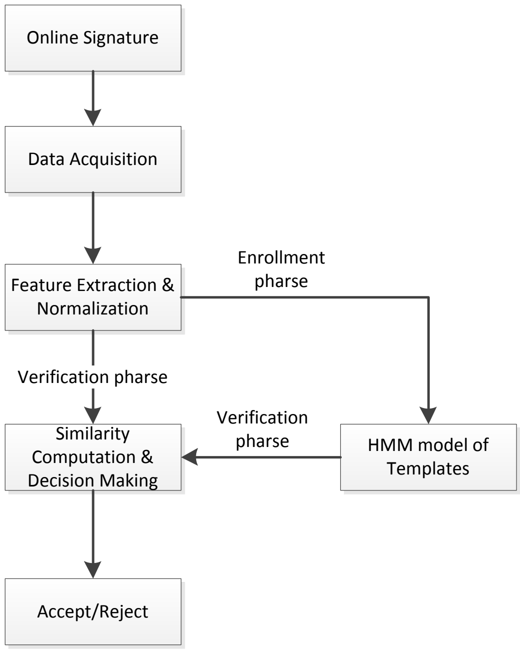

Fig. 1.System architecture of signature data classification

II. FEATURE EXTRACTION

A. Data collection

Signaturesdata are given to the program with the digital pen. Each time a user signs a signature, the program will collect a data stream. They are a set of five components (x, y, p, altitude and azimuth).These parameters can be considered as the function of time t (t is the time index of sampling):

- x (t): x coordinate at time t
- y (t): y coordinate at time t
- p (t): p pressure at time t
- altitude (t): the angle between the pen and the projection of the pen when it draws onto the plane of the drawing equipment (0-90)
- azimuth (t): Clockwise angle to the pen projection onto the plane of the drawing equipment (0-359)

Fig. 2 illustrates the visually information about altitude, azimuth of the digital pen when the user signs a signature.

***Toai Q. Ton**, Information Technology Department, Ho Chi Minh City University of Foreign Languages and Information Technology, Ho Chi Minh City, Viet Nam, +84947774847.*

***Tai Nhu Do**, Information Technology Department, Ho Chi Minh City University of Foreign Languages and Information Technology, Ho Chi Minh City, Viet Nam, +84909969618.*

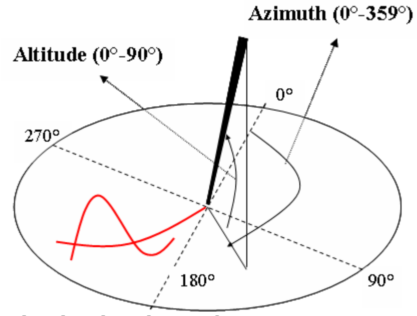

Fig. 2. Two angles:altitude and azimuth

B. *Feature extraction*

Because the features of the signature is depend on the signer very much, so we extracted some dynamic characteristics f(t) at t time.

TABLE 1: 20 CHARACTERISTICS RELATED TO THE MOVEMENT OF THE SIGNER.

No.	Feature name
1-2	The normalized coordinates ($x(t) - x_g$, $y(t) - y_g$) compared to the center of the signature (x_g, y_g)
3	The pressure p(t)
4-5	Two angle: altitude (t) and azimuth (t)
6-7	Speed in x and y directions: $v_x(t) = x'(t)$ $v_y(t) = y'(t)$
8	The magnitude of the velocity line: $v(t)=\sqrt{x'(t)^2 + y'(t)^2}$
9-10	Acceleration in directions of the x and y: $a_x(t) = v_x'(t)$, $a_y(t) = v_y'(t)$
11	Absolute acceleration: $a(t)=\sqrt{a_x^2(t) + a_y^2(t)}$
12	Tangential acceleration: $a_{tt}(t) = v'(t)$
13	The press derivation: $\Delta p(t) = p'(t)$
14	The α angle between the absolute velocity vector and the x axis: $\alpha(t) = \arcsin \frac{v_y(t)}{v(t)}$
15-16	Sine, cosine of the α angle: $\sin \alpha(t) = \frac{v_y(t)}{v(t)}$, $\cos \alpha(t) = \frac{v_x(t)}{v(t)}$
17	Derivation of α angle: $\Delta\alpha(t) = \alpha'(t)$
18-19	$\sin \Delta\alpha(t)$ and $\cos \Delta\alpha(t)$
20	β(t) flexure between two adjacent line segments at each coordinate

C. *Feature normalization*

The features will have different range of values. Without normalization, the feature with large range of value will have more weight than the feature with small range of value. Therefore, we need normalize so that the feature values achieve zero average and unit standard deviation.

$$o_t = \frac{v_t - \mu}{\sqrt{\Sigma}} \quad (1)$$

Whereμ and Σ are the average sample and cross-covariance matrix of the vectors v_t (t = 1, 2 ... T).

So, looking for another aspect, the signature is considered as a feature matrix O=[o_1, o_2… o_T] (see Fig. 3).

III. MODELING SIGNATURE BY HMM

A. *Overview about Hidden Markov Model*

Hidden Markov Modelis used in the identity field as a modeling method of the changes of the signal in discrete time [4]. In essence, HMM is a double statistical process which is bound by Markov chain with a finite set of states and a set of probability functions combined with the output of an observer status. At the specific time t, the process will be in one of the states and generates an observation symbol according to probability function corresponding to the current state.

HMM is defined as a set of five λ=(S, V, π, A, B), where:

- S = {s_1, s_2 ... s_N}: A finite set of N states. State time t is called $q_t \in$ S
- V = {v_1, v_2 ... v_M}: A set of M symbols in the vocabulary that can be observed on the states.
- $A = \{a_{ij}\}$: The displacement matrix with the displacement probability a_{ij} (i=1..N, j=1..N) indicating the probability of moving from state s_i to state s_j

$$a_{ij} = P(q_{t+1} = s_j \mid q_t = s_i) \ (2)$$

- $B = \{ bj(k) \}$: Aset of probability which generates v_k symbol in state s_j.

$$b_j(k) = P(o_t = v_k \mid q_t = s_j) \ (3)$$

- $\pi = \{\pi_i | i = 1, \ldots, N\}$: The probability of the system at state $s_i$at the beginning.

$$\pi_i = P(q_1 = s_i) \ (4)$$

B. *Signature observations*

Each signature is represented by a matrix O = [o_1, o_2... o_T], in which o_t (t=1, 2... T) is the feature vector. Feature vector at each sample point includes 20 local features, called one signature observer.

Fig. 3 illustrates the continuous signature observation includingextraction features based on above techniques.

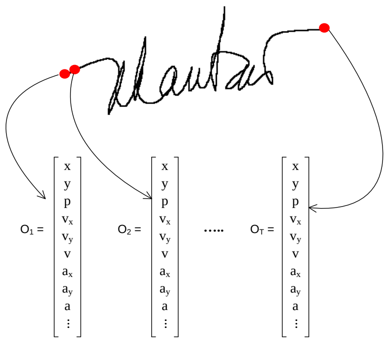

Fig. 3. The continuous signature observation

C. *Structure of HMM*

HMM used in the system is left-to-right Hidden Markov Model or Bakis Hidden Markov Model. With this model, the state only allowed to shift to itself or to the next state. This model reflects the data for the duration while data is processed.

Fig. 4 shows a picture of the left-right HMM model with five states

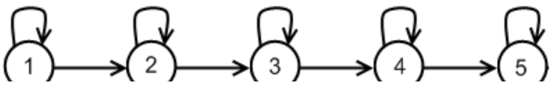

Fig. 4. HMM structure

The matrix A below describes left-to-right hidden Markov model with five left-right states:

$$A = \begin{pmatrix} a_{11} & a_{12} & 0 & 0 & 0 \\ 0 & a_{22} & a_{23} & 0 & 0 \\ 0 & 0 & a_{33} & a_{34} & 0 \\ 0 & 0 & 0 & a_{44} & a_{45} \\ 0 & 0 & 0 & 0 & a_{55} \end{pmatrix} \quad (5)$$

To describe the probability of $b_j(o)$ observations in each state, we use a Gauss mixture distribution function with M continuous multivariable. The number of variables in the Gauss function is L (L = 20, the number of feature of each signature point):

$$b_j(o) = \sum_{m=1}^{M} c_{jm}\, p\left(o|\mu_{jm}, \textstyle\sum_{jm}\right), j = 1, \ldots, N \quad (6)$$

Where $p\left(o|\mu_{jm}, \sum_{jm}\right)$ is the Gauss distribution function with multivariable, and the average vector μ_{jm} and the diagonal covariance matrix $\sum_{jm}$. Thus, the density function of the observations are described by theparameters $B = \left\{c_{jm}\, \mu_{jm}, \sum_{jm}\right\}$ $with\ 1 \le j \le N, 1 \le m \le M$.

D. *Initialize the parameter*

$\lambda = (S, V, \pi, A, B)$is used to model K training signatures called $\{O^{(1)}, O^{(2)}, \ldots, O^{(K)}\}$ where $O^{(k)} = [o_1^{(k)}, o_2^{(k)}, \ldots, o_{T_k}^{(k)}]$.

Initial probability matrix π:

$$\pi = \{\pi_1, \pi_2, \ldots, \pi_N\} = \{1, 0, 0, \ldots, 0\} \quad (7)$$

Initial shift-state matrix A:

$$A = \begin{pmatrix} 0.5 & 0.5 & 0 & 0 & 0 & \ldots & 0 \\ 0 & 0.5 & 0.5 & 0 & 0 & \ldots & 0 \\ 0 & 0 & 0.5 & 0.5 & 0 & \ldots & 0 \\ 0 & 0 & 0 & 0 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & 0 & 0 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad (8)$$

Initialize values of B: With N predetermined state in K training signatures, each signature divided into N sub segments:

$$S_1^{(k)}, S_2^{(k)}, \ldots, S_N^{(k)}$$

Where the size of (N-1) sub segments:

$$size = \frac{T_k}{N} \quad (9)$$

$$S_i^{(k)} = [o_{(i-1)size+1}^{(k)}, o_{(i-1)size+2}^{(k)}, \ldots, o_{(i)size}^{(k)}] \quad (10)$$

The last sub segments contain the remaining observations:

$$S_N^{(k)} = [o_{(N-1)size+1}^{(k)}, o_{(N-1)size+2}^{(k)}, \ldots, o_{T_k}^{(k)}] \quad (11)$$

Thus, the i^{th} state (i = 1, 2... N) has K sub observed segments $S_i^{(1)}, S_i^{(2)}, \ldots, S_i^{(K)}$ used to initialize the parameters of B by clustering into M groups according to K-Means algorithm. Each group has some Gaussians, which we calculate the center and covariance matrix, which is also the initial value of the Gaussian parameters.

The number of Gaussian was determined by experiments.We choose the Gaussian number with M = 5. The number of HMM states depends on the length of the signature and the number of parameters needed training. Through experimental average, we need 30 points to establish the average matrix and the Gaussian covariance matrix. To train a HMM, we need a set of 30*M sample points.

E. *Training*

The paper uses the Baum-Welch algorithm to train HMM. New model λ_{new} from old model λ according to Baum-Welch algorithm will give the cumulative probabilities of K signatures on λ_{new} model. They are greater or equal to the cumulative probability of K signatureson λ model:

$$\prod_{k=1}^{K} P(O^{(k)} \mid \lambda_{new}) \ge \prod_{k=1}^{K} P(O^{(k)} \mid \lambda) \quad (12)$$

Where P is calculated by the Forward algorithm.

Stop condition is defined:

$$\prod_{k=1}^{K} P(O^{(k)} \mid \lambda_{new}) - \prod_{k=1}^{K} P(O^{(k)} \mid \lambda) \le threshold \quad (13)$$

Threshold value are selected by heuristic (threshold = 0.001) and maximum loop is 20, but experimentally values will converge after 5-10 loops.

F. *Verification*

After HMM trained, we calculate the average of log likelihood of K signatures used to train the model. This is the average that the training signatures achievedon the model. We denote as S_{train}. With the verify signature O = [o_1, o_2 …ot] and an id to identify the model, we calculate the log likelihood of O by Viterbi algorithm, called S_{test}.

$$S_{test} = \frac{1}{T} \log P(O \mid \lambda) \quad (14)$$

We use the difference between S_{test} and S_{train} to determine the verify signature which is real or fake.

$$d = | S_{test}\,\text{test} - S_{train} | \quad (15)$$

d is the value of difference between testing signature and patterns. The more value d is close to 0, the more testing signature is close to real signature and vice versa. Thisdistance d is transformed into similarity measure s with range values between 0 and 1 by using Gaussian function:

$$s = \exp\left(\frac{-d}{L}\right) \quad (16)$$

Where L is the feature number of each point.

IV. EXPERIMENTS

To test the research results, we carried out the experiments on a SVC2004 database [6] including:

- 40 users
- 1 user: 20 real signatures + 20 professional forged signatures

Testing:

- Select randomly 5 real signatures for training
- Test 1: 10 real signatures + 20 professional forged signatures
- Test 2: 10 real signatures + 20 pseudo-random

signatures(obtained from other users)

The above process is repeated 10 times to ensure reliability with each time we can calculate EER deviation. After 10 trials, we can calculate the EER average.

The results on the skill forged signatures and the random forged signatures corresponding:

TABLE 2: AVERAGE SKILL FORGED SIGNATURE ERROR EER 7.02%

No.	%EER	Global Threshold
1	7.30	0.656
2	6.80	0.669
3	8.25	0.644
4	7.50	0.682
5	7.12	0.635
6	6.50	0.678
7	6.55	0.683
8	6.25	0.657
9	7.20	0.647
10	6.75	0.669

TABLE 3: AVERAGE RANDOM FORGED SIGNATURE ERROR EER 1.93%

No.	%EER	Global Threshold
1	2.10	0.559
2	2.14	0.501
3	1.75	0.55
4	1.25	0.543
5	1.50	0.555
6	2.75	0.57
7	2.35	0.535
8	1.62	0.59
9	2.26	0.57
10	1.60	0.574

TABLE 4: RESULTS ON DTW+ER2 [5]

	Skill forged signature	Random forged signature
Global Threshold	7.2%	0.9%
Local Threshold	4.9%	0.2%

TABLE 5: RESULT OF GROUPS ARE TESTED IN SVC2004 COMPETITIONS [6]

Group Code	EER1	EER2
219b	6.90%	3.02%
219c	6.91%	3.02%
206	6.96%	3.47%
229	7.64%	4.45%
219a	8.90%	3.08%
214	11.29%	4.41%
218	15.36%	6.39%
217	19.00%	4.29%
203	20.01%	5.07%
204	21.89%	8.75%

V. CONCLUSION

Compared with the systems of SVC2004 competition, our system resultsin the top 4 on skill forged signature and have better results than other systems on random forged signature.

Compared with the DTW + ER^2 method of Lei offer:

- When compared on a global threshold: Results of system is studied better of 0.18% on the professional forged signature.
- However, Lei system is better of nearly 1% on pseudo-random signature (The system is not implemented on the local threshold, so it cannot compare the results on the local threshold per user).

REFERENCES

[1] Impedovo, D., Pirlo, G.,"Automatic signature verification: the state of the art", *IEEE Trans. Syst. Man Cybern. C*, Appl. Rev., 38, (5), pp. 609–635, 2008

[2] Lee, L.L., Berger, T., Aviczer, E., "Reliable on-line human signature verification systems", *IEEE Trans. Pattern Anal. Mach. Intell.*,,18, (6), pp. 643–647, 1996

[3] Jain, A. K., Ross, A., Flynn, P. (Eds.), "Handbook of biometrics", pp. 189–209, 2008

[4] Rabiner, L.R.,"A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", *Proceeding of the IEEE*, vol. 77, pp.257-286, 1989.

[5] Lei, H.,"Sequential Pattern Classification Without Explicit Feature Extraction", pp16-38, 2005.

[6] Yeung, D.Y., Chang, H., Xiong, Y., George, S., Kashi, R., Matsumoto, T., Rigoll,G. "SVC2004: First International Signature Verification Competition" In Proc. of ICBA, Springer LNCS-3072, 2005.

[7] E. Argones Rua, E. Maiorana, J.L. Alba Castro, and P. Campisi,"Biometric Template Protection Using Universal Background Models: An Application to Online Signature". Information Forensics and Security, IEEE Transactions on, 7(1):269 –282, Feb. 2012.

[8] K.N. Ballantyne C.L. Bird, B. Found and D. Rogers, "Dynamic features of naturally written, disguised and forged handwritten text", In Proc. of 15th Conf. of the Int. Graphonom. Soc., Cancun, Mexico, June 2011.

[9] E.J. Will, "Inferring relative speed of handwriting from the static trace", J. of Forensic Doc. Examination, 22, 2012.

Toai Q. Ton received the Master degree in computer science from University of Information Technology, Ho Chi Minh City, Vietnam, in 2008.
He has joined the Faculty of Information Technology, Ho Chi Minh City University of Foreign Languages and Information Technology, Ho Chi Minh City, Vietnam, where he is currently a Lecturer. His research interests includepattern recognition, biometric recognition.

Tai Nhu Do received the Bachelor degree in information technology from Ho Chi Minh City University of Foreign Language – Information Technology, Ho Chi Minh City, Vietnam, in 2005.
In 2005, he has joined the Faculty of Information Technology, Ho Chi Minh City University of Foreign Languages and Information Technology, Ho Chi Minh City, Vietnam, where he is currently a Lecturer. His research interests include computer vision, image processing, and parallel programming.

