21st International Conference on Pattern Recognition (ICPR 2012)
November 11-15, 2012. Tsukuba, Japan

Tablet Owner Authentication Based on Behavioral Characteristics of Multi-Touch Actions

Kumi Nakamura
Osaka University
nakamura@nanase.comm.eng.osaka-u.ac.jp

Kazuhiro Kono
Kansai University
k-kono@kansai-u.ac.jp

Yoshimichi Ito
Osaka University
ito@comm.eng.osaka-u.ac.jp

Noboru Babaguchi
Osaka University
babaguchi@comm.eng.osaka-u.ac.jp

Abstract

This paper proposes a method for tablet owner authentication based on behavioral characteristics of multiple fingers' actions called multi-touch actions. The method is based on dynamic time warping, which has been commonly used for authentication using pen-tablet or single finger's actions, but another problem arises due to the use of multi-touch actions (e.g., identifying fingers). We also provide methods for these problems. Using proposed method, we evaluate the authentication accuracies for several types of multi-touch actions through experiments.

1. Introduction

Tablet-type devices that have multi-touch screen as an input interface are commonly used in recent years. This enables us to perform complex actions by two or more fingers because these devices recognize touches of multiple points at the same time. The actions using multiple fingers, such as pinch-in and pinch-out, are called multi-touch actions.

As for the owner authentication of tablet-type devices, password authentication is generally used. However, it would be inappropriate because tablet-type devices are smaller than desktop/laptop computers, and they are hard to enter passwords. Authentication method using physical biometrics like fingerprint authentication is also common for desktop/laptop computers, but it would also be unsuitable for tablet-type devices. This is because it requires a sensor for detecting physical biometric features of owners, but it is hard to implement such devices in tablet-type devices.

From the practical point of view, it would be more appropriate to use multi-touch screen for owner authentication of tablet-type devices, and thus, the authentication using multi-touch actions is quite natural.

Concerning owner authentication using multi-touch actions, several methods have been proposed in recent years [1, 4, 6, 8]. However, these methods are not fully utilize the advantage of multi-touch screen, that is, it can handle with complex actions by multiple fingers. For example, in [1], only simple physical characteristics of the shape of hands are used. Methods proposed in [4, 6, 8] impose a burden on users because users have to remember some kind of information for authentication (e.g., pre-determined images or finger movements).

From the viewpoints of making use of multi-touch screen and reducing user burden, a biometric authentication using *behavioral characteristics of multi-touch actions* would be more favorable. For example, when two users perform a certain multi-touch action, they move their fingers in different ways according to their behavioral characteristics. By discriminating their behavioral characteristics, we can establish an owner authentication method that is suitable for tablet-type devices. The purpose of this paper is to establish a fundamental tool to realize such authentication, and to evaluate authentication accuracies for several types of multi-touch actions.

The novelty of this paper lies in the fact that our authentication method uses multi-touch actions as behavioral characteristics, which is in sharp contrast with the fact that existing authentication methods use single-touch actions. The use of multi-touch actions would attain higher authentication accuracy than the use of single touch action.

2. Owner authentication using behavioral characteristics of multi-touch action

Our method consists of the following steps: 1) taking motion data of multi-touch actions of the tablet owner, such as pinch-in, drawing a circle by two fingers, and so on; 2) producing several time-series data of features from the motion data; 3) decision of master data and thresholds for authentication by calculating distances between the time series-data using dynamic time warping [7] like existing matching methods; 4) owner/non-owner judgement by comparing the threshold with the distance between the master data and his data.

2.1. Taking motion data of multi-touch actions

We first obtain motion data of multi-touch actions. Let $\boldsymbol{I} = \{I_0, I_1, ..., I_T\}$ be a time-series data obtained from a tablet. I_t $(t = 0, \ldots, T)$ is given by

$$I_t = \left\{(x_k^t, y_k^t, m_k^t) | k = 1, ..., n\right\} \tag{1}$$

where n is the number of fingers used in multi-touch actions, k is the ID of fingers, x_k^t and y_k^t are x and y coordinate values of finger k at t, respectively. m_k^t is the state of finger k at time t, which takes values 0, 1, and 2, when finger k does not touch the screen, the finger just touch the screen, and the finger has been touching the screen, respectively. The coordinate system is defined as follows: the origin is located at the upper-left corner of the screen, and the directions of x-axis and y-axis are rightward and downward from the origin, respectively.

In order to describe multi-touch actions, it is necessary to figure out which finger is touched on a tablet. However, tablet-type devices do not have such function. In the following, we present a method for identifying fingers.

Finger identification

When the maximum value $n_{\max}$ of fingers is equal to two, we regard the multi-touch action is performed using thumb and index finger, like pinch-in or pinch-out. In this case, the finger with smaller y_k^t is identified as index finger and the other is identified as thumb.

When the maximum value $n_{\max}$ is more than or equal to 3, the gravity point $G_t = (x_g, y_g)$ of I_t is computed where $x_g = \sum_{k=1}^{n_{\max}} x_k^t / n_{\max}$ and $y_g = \sum_{k=1}^{n_{\max}} y_k^t / n_{\max}$. Then, θ_k $(k = 1, ..., n_{\max})$ is computed where θ_k is an angle between the vector $(x_k^t - x_g, y_k^t - y_g)$ and the vector $(\frac{1}{2}, \frac{\sqrt{3}}{2})$, where we use $(\frac{1}{2}, \frac{\sqrt{3}}{2})$ instead of $(0, 1)$ because the initial finger position of the multi-touch action "turning a circle with five fingers" is taken into account.

Here, suppose that $n_{\max} = 5$. The finger with the smallest θ_k is identified as thumb because thumb is located at the lower left area on the tablet, as shown in Figure 1. In a similar manner, the finger with the second smallest θ_k is identified as index finger. The remaining fingers are identified in a similar way. This approach is performed for each t.

Finger trajectory interpolation

When several fingers are apart from the screen and

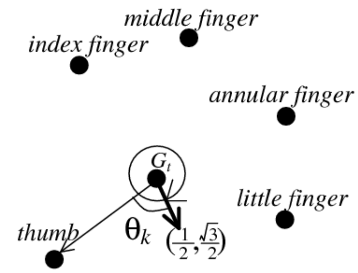

Figure 1. Identifying fingers.

they touch the screen again at the same time, we need to interpolate trajectories. In this case, we linearly interpolate the points that the corresponding fingers that are identified by the above algorithm are identical.

2.2. Producing time-series data of features

We use four features $\boldsymbol{P}$, $\overline{\boldsymbol{P}}$, $\boldsymbol{V}$, $\overline{\boldsymbol{V}}$, where $\boldsymbol{P}$ and $\boldsymbol{V}$ represent the time-series data of fingers' position and velocity, respectively, and $\overline{\boldsymbol{P}}$ and $\overline{\boldsymbol{V}}$ represent the normalized data of $\boldsymbol{P}$ and $\boldsymbol{V}$, respectively. From $\boldsymbol{I}$, $\boldsymbol{P} = \{P_0, P_1, \ldots, P_T\}$ is produced by

$$P_t = \{(x_k^t - X_G, y_k^t - Y_G, m_k^t) | k=1, \ldots, n\}, \tag{2}$$

where (X_G, Y_G) is the gravity point of all fingers given by $X_G = \sum_{k=1}^{n_{\max}} \sum_{t=0}^{T} x_k^t / (T+1) n_{\max}$, $Y_G = \sum_{k=1}^{n_{\max}} \sum_{t=0}^{T} y_k^t / (T+1) n_{\max}$. From $\boldsymbol{P}$, $\overline{\boldsymbol{P}} = \{\overline{P_0}, \overline{P_1}, \ldots, \overline{P_T}\}$, the normalized data of $\boldsymbol{P}$, is calculated by

$$\begin{cases} \overline{P_t} = \{(\overline{x_k^t}, \overline{y_k^t}, m_k^t) | k=1, \ldots, n\}, \\ \left(\overline{x_k^t}, \overline{y_k^t}\right) = \left(\dfrac{100\,(x_k^t - X_G)}{X_{\max} - X_{\min}}, \dfrac{100\,(y_k^t - Y_G)}{Y_{\max} - Y_{\min}}\right), \end{cases} \tag{3}$$

where $X_{\max} = \max_{k,t} x_k^t$, $Y_{\max} = \max_{k,t} y_k^t$, $X_{\min} = \min_{k,t} x_t^k$, and $Y_{\min} = \min_{k,t} y_t^k$. Taking difference between P_{t-1} and P_t, the time-series data of finger velocity $\boldsymbol{V} = \{V_1, V_2, \ldots, V_T\}$, is calculated as

$$V_t = \{(u_k^t, v_k^t, m_k^t) | k=1, \ldots, n\}, \tag{4}$$

where u_k^t and v_k^t are x-component and y-component of velocity for finger k at a time t, respectively, and are computed as follows: $u_k^t = x_k^t - x_k^{t-1}$; $v_k^t = y_k^t - y_k^{t-1}$. Finally, from $\boldsymbol{V}$, $\overline{\boldsymbol{V}} = \{\overline{V_1}, \overline{V_2}, \ldots, \overline{V_T}\}$, the normalized data of $\boldsymbol{V}$, is calculated by

$$\begin{cases} \overline{V_t} = \{(\overline{u_k^t}, \overline{u_k^t}, m_k^t) | k=1, \ldots, n\}, \\ (\overline{u_k^t}, \overline{v_k^t}) = \left(\dfrac{100 u_k^t}{U_{\max} - U_{\min}}, \dfrac{100 v_k^t}{V_{\max} - V_{\min}}\right), \end{cases} \tag{5}$$

where $U_{\max} = \max_{k,t} |u_k^t|$, $V_{\max} = \max_{k,t} |v_k^t|$, $U_{\min} = \min_{k,t} |u_k^t|$, and $V_{\min} = \min_{k,t} |v_k^t|$.

2.3. Distance Calculation

Let $\boldsymbol{S}$ and $\boldsymbol{R}$ be two different time-series data (they correspond to $\boldsymbol{I}$ in Section 2.1). Also, let $(\boldsymbol{P}_S, \overline{\boldsymbol{P}_S}, \boldsymbol{V}_S, \overline{\boldsymbol{V}_S})$ and $(\boldsymbol{P}_R, \overline{\boldsymbol{P}_R}, \boldsymbol{V}_R, \overline{\boldsymbol{V}_R})$ be the datasets of four time-series data of features calculated from $\boldsymbol{S}$ and $\boldsymbol{R}$, respectively, as shown in Section 2.2.

We obtain an optical match and the smallest distance between two time-series data by using dynamic time warping(DTW) [7], which is often used in authentication [2, 3, 5] because DTW enables us to consider the distance between two data of different length.

2.4. Decision of master data and thresholds

We decide master data M and a threshold C for authentication by calculating distances between training data as shown in Section 2.3. Master data M is data where the sum of square values of distances between M and the other training data is the smallest. A threshold C is given by $C = \beta L$, where L is the largest distance between M and an element in training data, and β is a constant for adjusting the threshold. Thus, we obtain four master data $M_{\boldsymbol{P}}$, $M_{\overline{\boldsymbol{P}}}$, $M_{\boldsymbol{V}}$, and $M_{\overline{\boldsymbol{V}}}$ and four thresholds $C_{\boldsymbol{P}}$, $C_{\overline{\boldsymbol{P}}}$, $C_{\boldsymbol{V}}$, and $C_{\overline{\boldsymbol{V}}}$ for four time-series data of features $\boldsymbol{P}$, $\overline{\boldsymbol{P}}$, $\boldsymbol{V}$, and $\overline{\boldsymbol{V}}$, respectively.

Note that the decision of master data and thresholds are affected significantly by outliers when we use all training data. We regard the data that attains L as an outlier if L is 1.5 times larger than the median of the distances between training data and master data M, and eliminate the outlier. And after that, we find new master data M and compute new L again.

2.5. Owner/non-owner judgment

One is authenticated as the tablet owner if the distance between the input data and the master data M is smaller than the threshold C. Otherwise, we judge that the user is not the owner.

3. Experiments

Using proposed method, we evaluate the authentication accuracies for several types of multi-touch actions through experiments. The tablet used is SONY Tablet, whose OS is Android 3.2. We take time-series data for 14 users. Each user performs 11 actions that are described in Section 3.1 with his/her right hand. Each action is performed 13 times, where first 3 sets are for practice, next 5 sets are for deciding master data and thresholds, and the last 5 sets are for test data for authentication. The sampling interval of data is set to 60 frames per second.

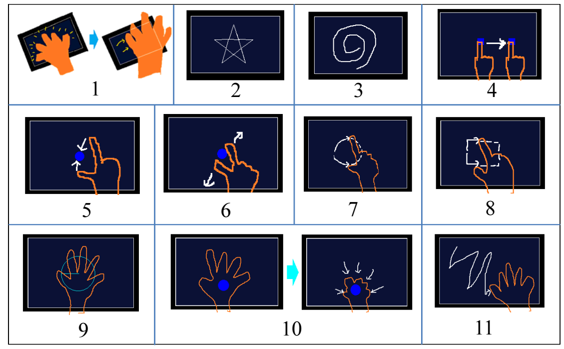

Figure 2. Multi-touch actions for test.

3.1. Multi-touch actions for experiments

As shown in Figure 2, the users perform 11 actions listed below:
1) touch screen with five fingers and let five fingers up,
2) draw pentacle,
3) draw spiral pattern,
4) flick,
5) pinch in,
6) pinch out,
7) draw a circle by two fingers,
8) draw a box by two fingers,
9) turn a circle with five fingers,
10) hold a circle by closing five fingers,
11) draw zig-zag shape by two or more fingers,

Actions 2-4 are single-touch actions and Actions 1 and 5-11 are multi-touch actions. We do not use V_n and $\overline{V_n}$ for Action 1 because fingers are not moved on the screen in this case.

3.2. Results and discussions

For evaluating authentication accuracy, false acceptance rate (FAR) and false rejection rate (FRR) are often used as standard measures. FAR and FRR are defined, respectively, as FAR = FA/(FA+TR) and FRR = FR/(FR+TA), where FA, TR, FR, and TA are numbers of false acceptance, true reject, false reject, and true accept, respectively. Therefore, we can achieve good authentication accuracy when FAR and FRR are small.

It is known that FAR monotonically increases from zero to one whereas FRR monotonically decreases from one to zero, when β increases (and thus, a threshold becomes large). Therefore, by selecting β appropriately, we can set FAR=FRR. In this case, the value FAR=FRR is called equal error rate (EER). This measure is also used quite often for evaluating authentication accuracy [1, 5, 6]. Here, we use EER for evaluating authentication accuracy.

Experimental results of EER for four features $\boldsymbol{P}$, $\overline{\boldsymbol{P}}$, $\boldsymbol{V}$, and $\overline{\boldsymbol{V}}$ are shown in Table 1. We first consider the

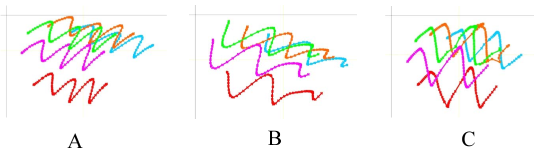

Figure 3. Examples of drawing zig-zag shape by 3 users.

influence of four features on EER. From Table 1, it is shown that EER in $\boldsymbol{P}$ is smaller than EER in $\overline{\boldsymbol{P}}$. This is because a difference between users is lost by normalization. EERs in $\boldsymbol{V}$ and $\overline{\boldsymbol{V}}$ tend to be larger than EER in $\boldsymbol{P}$. This is because the velocity of drawing actions varies every time. Such a tendency becomes significant especially in complex multi-touch action like Actions 9, 10, and 11. In what follows, we discuss EER in $\boldsymbol{P}$.

Next, we consider the influence of multi-touch actions, that is, Actions 5-11. From Table 1, we observe that EER for Action 11 (i.e., drawing zig-zag shape) is the best in multi-touch actions. This is because differences between users are large as shown in Figure 3. On the other hand, other multi-touch actions are quite worse. This is because complex multi-touch actions are difficult to perform and they are poorly reproducible.

Another problem is that touched points are not detected accurately when the positions of several fingers are very close. In this case, our method fails to identify fingers as shown in Figure 4.

Comparing the result of multi-touch actions with single-touch actions (i.e., Actions 2-4), EERs of single-touch actions are smaller than those of multi-touch actions. In particular, Action 2 attains the best authentication accuracy. One of the reasons is that the starting point for drawing pentacle is different for each user.

4. Conclusion

In this paper, we have proposed a method for tablet owner authentication using multi-touch actions. The

Table 1. EER at each time-series data of each designed actions.

	EER			
Action	$\boldsymbol{P}$	$\overline{\boldsymbol{P}}$	$\boldsymbol{V}$	$\overline{\boldsymbol{V}}$
1	53.6%	49.1%		
2	1.9%	5.8%	7.1%	7.6%
3	5.7%	6.8%	6.0%	3.8%
4	21.5%	25.7%	18.4%	35.7%
5	14.4%	24.6%	22.1%	25.7%
6	15.1%	32.8%	28.4%	32.1%
7	15.5%	23.2%	18.2%	20.1%
8	15.6%	19.0%	17.2%	25.7%
9	12.9%	18.6%	35.6%	34.5%
10	13.6%	16.0%	28.8%	28.0%
11	8.3%	8.2%	23.3%	21.8%

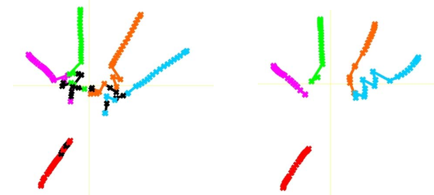

Figure 4. Examples of holding a circle by an user.

main feature of our method is to authenticate owners with multiple fingers using behavioral characteristics only. By using proposed method, we evaluate authentication accuracy for several types of multi-touch actions, and show that drawing zig-zag shape attains the best accuracy in multi-touch actions. However, it is shown that complex multi-touch actions are difficult to use for authentication. This work is supported in part by a Grant-in-Aid for scientific research from the Japan Society of the Promotion of Science.

References

[1] H. Iseri and E. Okamoto. A study of touch panels biometric based on behavioral characteristics. In *Computer Security Symposium 2011*, volume 3, pages 84–88, October 2011. (in Japanese).

[2] S. Ishihara, M. Ohta, E. Namikata, and T. Mizuno. Individual authentication for portable devices using motion of the devices. *Information Processing Society of Japan Journal*, 46(12):2997–3007, December 2005. (in Japanese).

[3] A. Kholmatov and B. Yanikoglu. Identity authentication using improved online signature verification method. *Pattern Recognition Letters*, 26(15):2400–2408, November 2005.

[4] D. Kim, P. Dunphy, P. Briggs, J. Hook, J. W. Nicholson, J. Nicholson, and P. Olivier. Multi-touch authentication on tabletops. In *Proceedings of the 28th international conference on Human factors in computing systems*, CHI '10, pages 1093–1102. ACM, April 2010.

[5] O. Miguel-Hurtado, L. Mengibar-Pozo, M. Lorenz, and J. Liu-Jimenez. On-line signature verification by dynamic time warping and gaussian mixture models. In *41st Annual IEEE International Carnahan Conference on Security Technology*, pages 23 –29, October 2007.

[6] A. Noguchi, M. Takahashi, K. Notomi, and K. Saito. Evaluation of rhythm authentication method without buttons. In *Proceedings of the 74th National Convention IPSJ*, volume 3, pages 640–641, March 2012. (in Japanese).

[7] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 26(1):43 – 49, February 1978.

[8] T. Sawamura, M. Narita, N. Noji, B. B. Bista, and T. Takata. A proposal of an authentication method using multi-touch screen. In *Computer Security Symposium 2010*, volume 9, pages 645–650. Information Processing Society of Japan, October 2010. (in Japanese).