IJBSCHS (2009-14-01-06)
[Original article]

Biomedical Soft Computing and Human Sciences, Vol.14, No.1, pp.47-53 (2009)
Copyright©1995 Biomedical Fuzzy Systems Association
(Accepted on 2008.10.03)

# Authenticating the Identity of Computer Users with Typing Biometrics and the Fuzzy Min-Max Neural Network

Anas QUTEISHAT*, Chee Peng LIM*+, Chen Change LOY**, and Weng Kin LAI***

*School of Electrical & Electronic Engineering, University of Science Malaysia, Malaysia
**Queen Mary Vision Laboratory, Department of Computer Science, Queen Mary, University of London, England
***Centre for Multimodal Signal Processing, MIMOS Berhad, Malaysia

(The paper was received on Dec. 30, 2007.)

**Abstract:** *In this paper, typing biometrics is applied as an additional security measure to the password-based or Personal Identification Number (PIN)-based systems to authenticate the identity of computer users. In particular, keystroke pressure and latency signals are analyzed using the Fuzzy Min-Max (FMM) neural network for authentication purposes. A special pressure-sensitive keyboard is designed to collect keystroke pressure signals, in addition to the latency signals, from computer users when they type their passwords. Based on the keystroke pressure and latency signals, the FMM network is employed to classify the computer users into two categories, i.e., genuine users or impostors. To assess the effectiveness of the proposed approach, two sets of experiments are conducted, and the results are compared with those from statistical methods and neural network models. The experimental outcomes positively demonstrate the potentials of using typing biometrics and the FMM network to provide an additional security layer for the current password-based or PIN-based methods in authenticating the identity of computer users.*

**Keywords** *Typing biometrics, the Fuzzy Min-Max neural network, keystroke pressure, keystroke latency, computer systems security*

## 1. Introduction

Every human has unique physiological and behavioural characteristics. These characteristics are referred to as biometrics. In general, human biometrics can be divided into two types, i.e., static and dynamic characteristics [1, 2]. As the names imply, static characteristics remain the same throughout the lifespan while dynamic characteristics may change gradually with time. Examples of static biometrics include: fingerprints, iris patterns, face proportions, DNA short tandem repeats, and pupil dilatation and contraction. On the other hand dynamic biometrics includes voice, lips movement, handwriting, and typing biometrics (or keystroke dynamics).

Perhaps one of the most important application areas of biometrics is in security [3]. Since security is always a major concern in our daily activities, many investigations have been conducted to ensure the immunity and invulnerability of a security system from intruders. As technologies develop, it has become clear now that computerized systems need a highly robust and accurate security measure to authenticate the identity of users. This is particular crucial for safety-critical tasks, e.g. entering secure buildings/areas, accessing classified documents, performing internet banking, withdrawing money from ATMs, etc. As such, the use of biometrics in computerized security systems has become increasingly important and widespread [4].

In this work, the biometrics technique under consideration is typing biometrics or keystroke dynamics. Typing biometrics often comes as an additional protection layer for password-based or PIN (Personal Identification Number)-based systems. It is a type of dynamic biometrics that can be exploited to authenticate the identity of an individual based on his/her keystroke patterns. The keystroke patterns come mainly from three sources, *viz.* the duration while pressing a key, the timing delays between successive keypairs (latency), and the pressure exerted on the individual keys on the keyboard or keypad. The underlying principle is that every computer user has a unique typing cadence and force when he/she types words that he/she is familiar with on the keyboard or keypad, for example username and password.

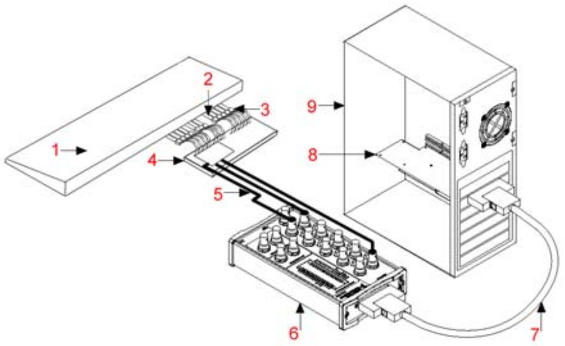
Using typing biometrics for identity authentication has its advantages and disadvantages. One ad-

+Contact: Prof. CP Lim, School of Electrical & Electronic Eng., University of Science Malaysia, Engineering Campus, 14300, Nibong Tebal, Penang, Malaysia. Email: cplim@eng.usm.my.

vantage is concerned with privacy invasion. Since all individuals are used to entering their authentication information during the login section, this technology is considered less invasive to privacy than some other biometrics techniques (e.g. fingerprint or face recognition). On the other hand, individuals who do not type in a consistent manner may have difficulty enrolling and verifying their identities based on keystroke dynamics.

Previous research work [5-9] has shown that it is possible to authenticate the identity of a compute user with a high degree of accuracy based on the keystroke latency patterns. In this paper, in addition to keystroke latency, how keystroke pressure can be used to devise a robust typing biometrics system is investigated. To accomplish this task, a pressure-sensitive keyboard that can capture the amount of force that a user exerts on a particular key of the keyboard is designed and developed. Using the keystroke pressure and latency signals collected from an experiment with a group of computer users, the Fuzzy Min-Max [10] (FMM) neural network is applied to classify the users into two categories, i.e. genuine users or impostors.

The organization of this paper is as follows. Section 2 explains the hardware development of the pressure-sensitive keyboard and data collection for experimentation. A description on FMM is given in section 3. The experimental results and a comparison between the performance of FMM and those from other methods are included in section 4. Section 5 presents the conclusions and suggestions for further work.



| | |
|---|---|
| 1 | 104-key Windows keyboard |
| 2 | FlexiForce® standard sensor A201-1, 1 lb, 8” |
| 3 | Berg connector |
| 4 | Signal conditioning circuit (LM741 op-amp, capacitor and resistors) |
| 5 | Probe |
| 6 | BNC2110 (desktop and DIN rail-mountable BNC adapter) |
| 7 | SH68-68-EP cable, 1 meter |
| 8 | NI PCI 6040E (multifunction DAQ device) |
| 9 | CPU |

Fig. 1. The hardware setup of a pressure-sensitive keyboard

## 2. Hardware Setup

To capture the keystroke pressure signals of computer users, a special keyboard with pressure sensors adhered underneath the keys is developed. Figure 1 shows the hardware setup of a pressure-sensitive keyboard. The pressure sensor acts as a variable resistor where its resistance changes in accordance with the amount of force that a user applies when he/she types. A force-to-voltage circuit is used to convert the keystroke pressure to discrete voltage-time signals. The signals are sent to the computer through a data acquisition board.

### 2.1. Data Collection

A set of data, primarily comprising keystroke pressure and latency, from a group of computer users was collected for experimentation. A total of 10 computer-literate users participated in the experiment. Note that the participants were not informed of the data collection and analysis strategy. Initially, each participant had to register his/her username and password during a login session. All participants were requested to enter the same password since the objective was to determine whether the proposed approach could identify and differentiate a particular user from the rest. The password used was “*try4-mbs*”. It was chosen because it fulfilled the common requirements, i.e., at least 8 characters in length combining symbols, numbers, and letters. The participants were asked to practise typing the password beforehand. During the data collection phase, a user typed the password for 10 times. Thus, a database of 10 user profiles, where each profile contained 10 samples of keystroke latency and pressure signals, was established. Keystroke latency was measured in milliseconds, whereas keystroke pressure was measured in volts ranging from 0 to 10 volts in the form of discrete-time signal.

The discrete-time signals of keystroke pressure were transformed into the frequency domain by using the Fast Fourier Transform (FFT). A set of features to represent the pressure signals was extracted for the magnitude information yielded from the FFT process, as shown in Table 1.

Tab. 1. Features extracted from the frequency domain to represent the keystroke pressure signals

| No. | Name of Feature | No. | Name of Feature |
|---|---|---|---|
| 1 | Arithmetic mean | 6 | Fundamental frequency |
| 2 | Root mean square | 7 | Energy |
| 3 | Peak | 8 | Kurtosis |
| 4 | Signal in noise & distortion | 9 | Skewness |
| 5 | Total harmonic distortion | | |

There were various sources of noise associated with the electronic circuit used to acquire the signals. One of the main sources of noise was the customised pressure-sensitive keyboard used in the experiment. The voltage values captured varied within a range. In order to reduce the effect of noise, the data sample of each user profile was pre-processed, as those in [5, 7]. For the keystroke pressure and latency signals, the mean and standard deviations of each feature in the profile were computed. Each feature value was then compared with its respective mean, and any measurements that differed by more than $T$ standard deviations from the mean would be removed. The discarded feature values were replaced by the respective mean value instead. The threshold was set at the mean plus 1.5 standard deviations. All the features were scaled within the range of 0 and 1 before sending them to the FMM network for classification.

## 3. The Fuzzy Min-Max Neural Network

In a previous study [11], a number of different methods to classify the keystroke pressure and latency patterns were investigated. The methods used included a Logistic Regression (LR) model as well as the Multi-Layer Perceptron and Fuzzy ARTMAP (FAM) neural network models. In this work, the FMM network is selected as the pattern classifier, and the good results obtained from the experimental study (as presented in Section 4) justify this choice. Indeed, the FMM network was first proposed to meet a number of desirable properties that an ANN classifier should possess in order to function as a reliable and credible tool in solving practical classification problems. These properties include [10]

- *On-line learning*: an ANN classifier should be able to learn input patterns autonomously and adapt itself to the new patterns without forgetting previously learned patterns;
- *Nonlinear Separability*: an ANN classifier should be able to build decision regions that separate classes of different shapes and sizes;
- *Decision boundaries*: an ANN classifier should be able to create class boundaries to reduce its misclassification rates;
- *Learning time*: an ANN classifier should be able to learn in a short time;
- *Soft and hard decisions*: an ANN classifier should be able to provide both hard (crisp, i.e. 0 or 1) and soft decisions;
- *Verification and validation*: an ANN classifier should be able to provide a mechanism to verify and validate its performance, as performance evaluation is an important aspect of an ANN model;
- *Tuning parameters*: an ANN classifier should have a number of tuning parameters to adjust the network to the input patterns, as each classification problem has its own unique properties;
- *Nonparametric Classification*: an ANN classifier should not depend on *a priori* knowledge of the input patterns while performing classification, as this information is not available most of the time

The dynamics of FMM are based on aggregates of fuzzy hyperboxes. A hyperbox defines a region of the $n$-dimensional pattern space that has patterns with full class membership. The hyperbox is completely defined by its minimum and maximum points. The membership function is defined with respect to these hyperbox min-max points, and describes the degree to which a pattern fits in the hyperbox. For an input pattern of $n$-dimensions a unit cube $I^n$ is defined. In this case, the membership value ranges between 0 and 1. A pattern which is contained in the hyperbox has the membership value of one. The definition of each hyperbox fuzzy set $B_j$ is:

$$B_j = \{X, V_j, W_j, f(X, V_j, W_j)\} \quad \forall\ X \in I^n \qquad (1)$$

where $V_j$ and $W_j$ are the min and max points, respectively. Applying the definition of a hyperbox fuzzy set, the combined fuzzy set that classifies the $K^{\text{th}}$ pattern class, $C_k$, is defined as:

$$C_k = \bigcup_{j \in K} B_j \qquad (2)$$

where $K$ is the index set of those hyperboxes associated with class $k$. One important property of this approach is that the majority of the processing is concerned with finding and fine-tuning the boundaries of the classes, as shown in Figure 2.

As shown in Figure 3, FMM is a three layer network. The first layer $F_A$ is the input layer which contains input nodes equal in number to the number of dimensions of the input pattern. Layer $F_C$ is the output layer. It contains nodes equal in number to the number of classes. The hidden layer is called the hyperbox layer $F_B$. Each $F_B$ node represents a hyperbox fuzzy set, where the $F_A$ to $F_B$ connections contain the minimum-maximum points.

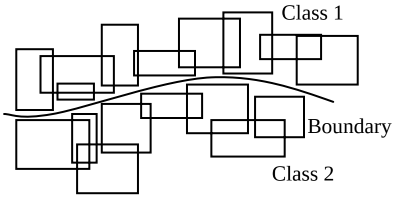


Fig. 2. An example of FMM hyperboxes placed along the boundary of a two-class problem

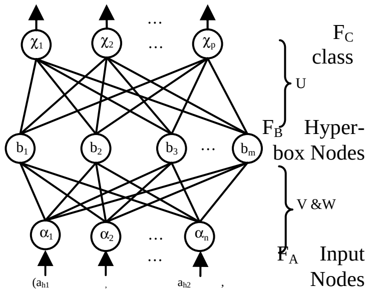


Fig. 3. A three layer FMM network

The $F_B$ transfer function is the hyperbox membership function defined by

$$b_j(A_h) = \frac{1}{2n}\sum_{i=1}^{n}\Big[\max\big(0, 1 - \max\big(0, \gamma \min\big(1, a_{hi} - w_{ji}\big)\big)\big) + \max\big(0, 1 - \max\big(0, \gamma \min\big(1, v_{ji} - a_{hi}\big)\big)\big)\Big] \qquad (3)$$

where, $A_h = (a_{h1}, a_{h2}, \ldots, a_{hn}) \in I^n$ is the $h^{\text{th}}$ input pattern, $V_j = (v_{j1}, v_{j2}, \ldots, v_{jn})$ is the minimum point for $B_j$, $W_j = (w_{j1}, w_{j2}, \ldots, w_{jn})$ is the maximum point for $B_j$, and $\gamma$

is the sensitivity parameter that regulates how fast the membership values decrease as the distance between $A_h$ and $B_j$ increases. The connections between the $F_B$ and $F_C$ nodes are binary valued and are stored in matrix *U*. The equation for assigning the values from $F_B$ to $F_C$ connections is

$$u_{jk} = \begin{cases} 1 & \text{if } b_j \text{ is a hyperbox for class } C_k \\ 0 & \text{otherwise} \end{cases} \qquad (4)$$

where $b_j$ is the $j^{\text{th}}$ node and $C_k$ is the $k^{\text{th}}$ node. Each $F_C$ node represents a class. The output of the $F_C$ node represents the degree to which input pattern $A_h$ fits within class *k*. The transfer function for each of the $F_C$ nodes performs the fuzzy union of the appropriate hyperbox fuzzy set values. This operation is defined as

$$c_k = \max_{j=1}^{m} b_j u_{jk} \qquad (5)$$

In FMM, the fuzzy min-max learning methodology is an expansion/contraction process. The training set $D$ consists of a set of *M* ordered pairs $\{X_h, d_h\}$, where, $X_h = (x_{h1}, x_{h2}, ..., x_{hn}) \in I^n$ is the input pattern and $d_h \in \{1, 2, ..., m\}$ is the index of one of the *m* classes. The learning process begins by selecting an ordered pair from *D* and finding a hyperbox of the same class that can expand, if necessary, to include the input. The expansion criteria have a constraint that must be met, and is defined as:

$$n\theta \geq \sum_{i=1}^{n} \left( \max(w_{ji}, x_{hi}) - \min(v_{ji}, x_{hi}) \right) \qquad (6)$$

where $0 \leq \theta \leq 1$ is the hyperbox size. If a hyperbox that meets the expansion criteria cannot be found, a new hyperbox is formed in the network. This growth process allows classes to be formed that are nonlinearly separable. It also allows new classes to be added without retraining. When hyperboxes expand there is a possibility of overlap among these boxes. So, an overlap test is introduced to check if the overlap is among the same or different classes. If overlapping occurs among the same classes nothing is done. But if overlapping occurs among different classes, the contraction process is activated and the overlap is eliminated. Basically, the fuzzy min-max learning process comprises a series of activities that create and expand hyperboxes, and fine-tune these hyperboxes by the overlap test and the contraction process. Details of the FMM dynamics can be obtained in [10, 12].

## 4. Experimental Studies

The 10-fold cross validation method [13] was employed in the experimental study. Initially, all keystroke data samples (pressure and latency) were divided into 10 profiles, where each profile represented a user. In each cross-validation run, a different profile was chosen to be a genuine user, and was labelled as “Class 0”. The rest (9 profiles) were assumed to be impostors, and were labelled as “Class 1”. As each user repeated his/her typing patterns 10 times during the experiment, a total of 100 samples were generated (in each run). These data samples were randomly partitioned into 90%/10% for training/test, respectively. The performance evaluation was based on the following indicators:

- *Accuracy*–the ratio of the number of correctly predicted users to the total number of users;
- *Sensitivity*–the ratio of the number of correctly predicted genuine users to the total number of genuine users;
- *Specificity*–the ratio of the number of correctly predicted impostors to the total number of impostors

The *False Acceptance Rate* (FAR) (the rate that an impostor is falsely identified as a genuine user) and *False Rejection Rate* (FRR) (the rate that a genuine user is incorrectly identified as an impostor) are: FAR=100-*Specificty* and FRR=100-*Sensitivity*, respectively. Note that the above performance indicators are measured in terms of percentages.

The bootstrap method [14] was applied to calculate the FMM results, i.e., mean and standard deviation. In the experiment, the mean and standard deviation values were obtained by re-sampling 5000 times. As it is known, the performance of FMM (as well as FAM) network is sensitive to the presentation order of the training samples owing to its incremental learning properties. To overcome this problem, a pool of networks is trained, each with different presentation order of the training samples, and the results are then combined using some decision combination methods, e.g. by voting [10]. In this study, the majority voting strategy was used. The training data was randomized three times, and three FMM networks were created. During the experiment, each FMM network yielded a prediction for each test sample. The final prediction for the given test sample was the one with the majority number of votes. In other words, the final decision was the prediction made by more than half of the classifiers. The results obtained were known as Voting FMM results. The results of the three FMM networks were also averaged to produce the Average FMM results.

Two studies were conducted. The first aimed to examine the performance of Average FMM using only the keystroke pressure data set. The second examined Average FMM and Voting FMM using keystroke pressure signals, keystroke latency signals, as well as pressure and latency signals. In all the experiments, the hyperbox size of FMM ($\theta$) was set to 0.01.

### 4.1. Experiment I

The keystroke pressure signals were first evaluated. Figure 4 shows the normalized distribution of features extracted from the pressure data set as well as the average values of all 10 users. Notice that the average values ranged between 0.4 and 0.5. However, each user exhibited a different distribution pattern. This indicated that the selected features were useful for the classification task.

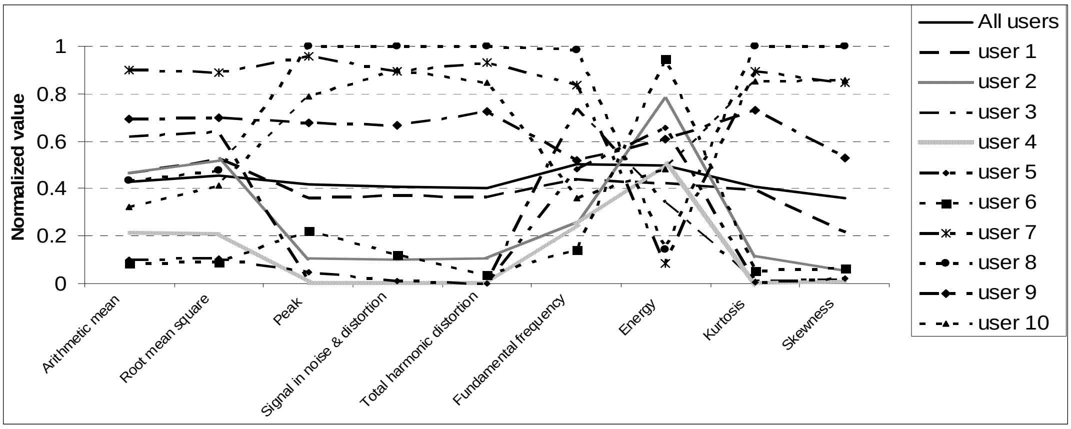


Fig. 4. Distribution of the pressure data set

Table 2 shows the experimental results using Average FMM. The results were compared with those from Logistic Regression, Multi-Layer Perceptron, and Average Fuzzy ARTMAP [15] as published in [11]. As shown in Table 2, Average FMM achieved the best performance in terms of Accuracy, Sensitivity, and Specificity. This demonstrated the usefulness of FMM as the underlying classifier for authenticating genuine users from impostors in this study.

Tab. 2. Performance comparison of different methods using the keystroke pressure data set (the shaded results are those published in [11])

| | Accuracy | Sensitivity | Specificity |
|---|---|---|---|
| Logistic Regression [11] | 94.22 ± 3.79 % | 77.50 ± 17.21 % | 96.08 ± 2.56 % |
| Multi-Layer Perceptron [11] | 89.70 ± 5.86 % | 53.60 ± 27.43 % | 93.71 ± 3.89 % |
| Average Fuzzy ARTMAP [11] | 93.41 ± 3.68 % | 76.20 ± 18.16 % | 95.32 ± 2.64 % |
| Average FMM | 95.00 ± 5.85 % | 83.89 ± 27.26 % | 97.90 ± 3.91 % |

The performance of Average FMM was further evaluated against different number of users. Table 3 shows the average test accuracy rates of 10 runs from 3-10 users. One can notice that as the number of users increased, the number of training patterns increased.

The Average FMM network was able to fine-tune the hyperboxes formed and to formulate a more accurate classifier, hence improvement in the performance. The results stabilized at around 95% when all 10 users were tested.

Tab. 3. Classification accuracy rates subject to different number of users

| Number of Users | Test Accuracy (%) |
|---|---|
| 3 | 92.71 |
| 4 | 93.21 |
| 5 | 94.60 |
| 6 | 94.45 |
| 7 | 95.98 |
| 8 | 96.68 |
| 9 | 95.90 |
| 10 | 95.01 |

### 4.2 Experiment II

The aim of this experiment was to compare the performance of Average FMM and Voting FMM using three different data sets, namely the keystroke pressure data set alone, the keystroke latency data set alone, and the combined keystroke latency and pressure data set. The results are summarized in Table 4. The Average Fuzzy ARTMAP results from [11] are also listed.

As can be observed, the latency data set yielded better results as compared with those from the pressure data set. However, combining the latency and pressure data samples in to one could lead to improved performances from the three classifiers, as listed in Table 4. Comparing the results of Average FMM and Average FAM, one can see that Average FMM outperformed Average FAM when the pressure or latency data set was used alone. However, for the combined pressure and latency data set, Average FAM performed marginally better than Average FMM in terms of accuracy and sensitivity, but not specificity. Nevertheless, Voting FMM yielded the best results for Accuracy, Sensitivity, and Specificity by using the combined pressure and latency data set.

Tab. 4. Performance comparison of different methods using different keystroke data sets (the shaded results are those published in [11])

| Classifier | Features | Accuracy | Sensitivity | Specificity |
|---|---|---|---|---|
| Average Fuzzy ARTMAP | Pressure | 93.41 ± 3.68 % | 76.20 ± 18.16 % | 95.32 ± 2.64 % |
| | Latency | 96.17 ± 3.81 % | 85.40 ± 15.56 % | 97.37 ± 2.83 % |
| | Pressure + Latency | 98.78 ± 1.42 % | 95.60 ± 5.42 % | 99.13 ± 1.18 % |
| Average FMM | Pressure | 95.00 ± 5.85 % | 83.89 ± 27.26 % | 97.90 ± 3.91 % |
| | Latency | 97.05 ± 5.07 % | 90.82 ± 19.62 % | 98.63 ± 3.39 % |
| | Pressure + Latency | 98.32 ± 3.13 % | 94.79 ± 11.79 % | 99.40 ±1.90 % |
| Voting FMM | Pressure | 96.18 ± 6.28 % | 85.81 ± 30.31 % | 97.1 ± 6.27 % |
| | Latency | 97.60 ± 5.63 % | 95.32 ± 13.55 % | 98.47 ± 5.1 % |
| | Pressure + Latency | 99.30 ± 2.51 % | 98.68 ± 7.21 % | 99.49 ± 2.42 % |

Another observation is that Average FMM and Voting FMM produced larger values of standard deviations when compared with those of Average FAM. In other words, the results of FMM varied from one run to another. This implied that the presentation order of training samples had a large effect on the performance of FMM as compared with that of FAM.

## 5. Summary

In this paper, use of typing biometrics and the FMM neural network, which acts as an additional security layer for conventional password-based or PIN-based protection systems for computer users, has been investigated. A pressure-sensitive keyboard has been constructed to collected keystroke pressure signals from computer users. In addition to keystroke pressures, keystroke latency signals of the users have been captured. The keystroke pressure and latency signals were presented as the input patterns to the FMM network for it to differentiate between genuine users and impostors. A series of experiments was conducted using FMM with two different operating strategies, i.e., Average FMM and Voting FMM. In general, voting FMM achieved the best results, with 99.30% Accuracy, 98.68% Sensitivity (or 1.32% FRR), and 99.49% Specificity (or 0.51% FAR) when both the keystroke pressure and latency signals were combined for classification. Besides, the FMM results compared favourably with those from other classification methods.

Although the performance of FMM is good, the standard deviations associated with the results are large. This implies the instability of the FMM performance from one run to another. Thus, further investigation on how to improve the stability of the FMM performance is needed. On the other hand, more data samples from computer users of varying ages, computer literacy, and typing abilities need to be collected for analysis. Besides, more experiments by using typing patterns at different phases, where each phase extends from a few days to a few weeks apart, can be conducted to examine the robustness of the proposed approach in authenticating the identity of computer users.

## References

[1] Brömme, A. (2003): "A classification of biometric signatures", *Proc. Int. Conf. on Multimedia and Expo ICME '03,* pp. III-17-20.

[2] Brömme, A., (2002): "A Classification of Biometric Applications wanted by Politics: Passports, Person Tracking and Fight against Terror", *Proc. of the IFIP 17th World Computer Congress*, pp. 207 - 219.

[3] Yanushkevich, S.N. (2006): "Synthetic Biometrics: A Survey", in *Int. Joint Conf. on Neural Networks IJCNN '06*, pp. 676-683.

[4] Jain, A.K. (2004): "Biometric recognition: how do I know who you are?", *Proc. of the IEEE 12th Signal Processing and Communications Applications Conf.*, pp. 3-5.

[5] Joyce R. and Gupta, G. (1990): "Identity authentication based on keystroke latencies", *Comm. ACM,* vol. 33, pp. 168–176.

[6] Mandujano, S. and Soto, R. (2004): "Deterring password sharing: user authentication via fuzzy c-means clustering applied to keystroke biometric data", *Proc. of the Fifth Mexican Int. Conf. in Computer Science ENC 2004,* pp. 181-187.

[7] Monrose, F., and Rubin, A. (1997): "Authentication via keystroke dynamics", *Proc. of the Fourth*

*ACM Conf. on Computer and Communications Security*, pp. 48-56.

[8] Ru, W.D. and Eloff, J. (1997): "Enhanced password authentication through fuzzy logic", *IEEE Expert,* vol. 12, pp. 38–45.

[9] Obaidat, M.S. and Sadoun, B. (1997): "Verification of computer users using keystroke dynamics", *IEEE Trans. Syst. Man, and Cybernet,* vol. 27, pp. 261–269.

[10] Simpson, P.K. (1992): "Fuzzy Min-Max neural networks-Part 1: Classification", *IEEE Trans. on Neural Networks,* vol. 3, pp. 776-786.

[11] Loy, C.C., Lim, C.P., and Lai, W.K. (2005): "Pressure-Based Typing Biometrics User Authentication Using the Fuzzy ARTMAP Neural Network", *Proc. of the 12th Int. Conf. on Neural Information Processing*, pp. 647-652.

[12] Gabrys, B. and Bargiela, A. (2000): "General Fuzzy Min-Max Neural Network for Clustering and Classification", *IEEE Trans on Neural Networks,* vol. 11, pp. 769-783.

[13] Stone, M. (1974): "Cross-validatory choice and assessment of statistical prediction", *Journal of the Royal Statistical Society,* vol. 36, pp. 111-147.

[14] Efron, B. (1979): "Bootstrap methods: another look at the Jackknife", *The Annals of Statistics,* vol. 7, pp. 1-26.

[15] Carpenter, G.A., Grossberg, S., Markuzon, N., Reynolds, J., and Rosen, D. (1992): "Fuzzy ARTMAP: A neural network architecture for incremental learning of analog multidimensional maps", *IEEE Trans. on Neural Networks,* vol. 3, pp. 698-713.

**Anas QUTEISHAT**

He received his BEng (Electronics) degree from Princess Sumaya University of Technology, Jordan, and both the MSc (Electronic Systems Design) and PhD degrees from University of Science Malaysia, Malaysia in 2003, 2005, and 2008, respectively. His research interests include neural networks, pattern recognition, rule extraction, and multi-agent systems.

**Chee Peng LIM**

He received his PhD degree from the University of Sheffield in 1996. Currently, he is a professor at School of Electrical & Electronic Engineering, University of Science Malaysia. His research interests include soft computing, pattern recognition, medical decision support systems, condition monitoring, intelligent manufacturing and control.

**Chen Change LOY**

He received his BEng (Electronics) (1$^{st}$ class honours) degree from University of Science Malaysia in 2005. Currently, he is working towards his PhD at Queen Mary, University of London. His research interests include soft computing, biometrics, visual surveillance, novelty detection, and semi-supervised learning.

**Weng Kin LAI**

He received his PhD degree from the University of Auckland in 1995. Currently, he is Head of the Advanced Informatics Centre, MIMOS Berhad, Malaysia. His research interests include machine learning, pattern recognition, image processing, multi-constraint optimisation, and complex systems.