IETE Journal of Research
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tijr20

User Authentication Based on Keystroke Dynamics

Rajat Kumar Das[a], Sudipta Mukhopadhyay[a] & Puranjoy Bhattacharya[b]
[a] Department of E & ECE Engineering, IIT Kharagpur, West Bengal, India
[b] Intel Technologies India Ltd, Bangalore, India
Published online: 24 Jul 2014.

To cite this article: Rajat Kumar Das, Sudipta Mukhopadhyay & Puranjoy Bhattacharya (2014) User Authentication Based on Keystroke Dynamics, IETE Journal of Research, 60:3, 229-239, DOI: 10.1080/03772063.2014.914686

To link to this article: http://dx.doi.org/10.1080/03772063.2014.914686

User Authentication Based on Keystroke Dynamics

Rajat Kumar Das[1], Sudipta Mukhopadhyay[1] and Puranjoy Bhattacharya[2]

[1]Department of E & ECE Engineering, IIT Kharagpur, West Bengal, India, [2]Intel Technologies India Ltd, Bangalore, India

ABSTRACT

This paper presents a technique to verify user identity using keystroke dynamics from short text, namely the computer login string. The keystroke behavioural pattern is obtained when a person types with a QWERTY keyboard. Two features hold time of an individual key and the latency of the consecutive keystrokes is used for authentication. Using a small training sample, accuracies of 90% and 99% are achieved for the data-set of 220 login strings per user (40 strings from legal user + 180 strings from nine intruders) using Gaussian mixture model and two-layer feed-forward neural network, respectively, as classifier. The paper then proceeds to a comprehensive study to explain how the accuracy varies with the length of the input string and with negative data in the training set.

Keywords:
Behavioural biometric, Continuous authentication, Free text, Gaussian mixture model (GMM), Hold time, Latency time, Keystroke dynamics, Neural network (NN).

1. INTRODUCTION

1.1 Background

The ever-increasing dependency on the computer system has made it an integral part of our daily life. The computer acts as a gateway to the world to preserve and access information. The computer with internet allows us to do different important online activities like banking, shopping, buying, and selling of stocks [1].

Despite their importance, computer systems today are protected with a primitive security technique of text matching, namely, username and password [2]. If the typed string matches with the stored access control string, users are able to login into the system. Since these passwords are vulnerable to various types of attacks, there is a chance of data being stolen or lost in different ways [5].

In order to avoid the shortcomings of the existing password-based system and to enhance the security, biometric-based authentication can be used. It is a special authentication technique which identifies a person based upon his/her physiological (like face, fingerprint, iris, etc.) or behavioural characteristics (like voice, signature, keystroke, mouse dynamics, etc.) [3]. Most of these biometrics require an extra hardware to acquire data thereby increasing cost and the form factor of the existing system. Keystroke dynamics based biometric system which does not require an extra hardware makes it cost effective and convenient. The said biometric is non-intrusive and non-obstructive making it a perfect choice for authentication [4].

1.2 Related Work

One of the pioneering papers in the area of keystroke recognition was presented by Gaines et al. [5]. The experiment involved six professional secretaries at the Rand Corporation.

They were asked to type three different fixed text passages consisting of 900−1200 words. The passages are same for all the users. The author here shows that the keystroke latencies can be used for authentication. For similarity check, *t*-test statistical tool was used. Although this study was able to obtain a good result (false acceptance rate (FAR) = 0% and false reject rate (FRR) = 4%), it is impractical for login application due to large size of text required for authentication.

Joyce and Gupta [6] conducted another early study where the keystroke latencies from four strings (first name, last name, username, and password) were used to build the reference profile, and predefined threshold was used to check similarity match. The authors were able to achieve a result of 0.17% and 13.3% as FAR and FRR, respectively. It should be noted that the test data are collected immediately after the training session in this experiment. Here the variability in the keystroke pattern over time is not captured. Similar experiment is conducted by Monrose et al. [7], where the four strings were captured in different sessions for training and testing purpose. Four classifiers named as Euclidean distance measure (83%), non-weighted probability (86%), weighted probability measure (87%), and Bayesian classifier (92%) are used in this experiment. The experiment needed a large number of training samples, where

each of the four strings is typed eight times to build the reference profile.

John et al. [8] analysed the digraphs and trigraphs of 1400 characters to build the reference profile. They have used a filter to remove the out layers caused by long pauses or other abnormalities during the typing. The FRR of 5.5% and FAR of 5% were achieved in the experiment.

Obaidat et al. [9] and Brown et al. [10] have used neural network (NN) for authentication using keystroke dynamics. The training samples required are approximately 900 and 70 strings. Yu et al. [11] applied support vector machine (SVM) for identification using 150−400 training samples (string length of 6−10) per user. Chang et al. [12] used hidden Markov model (HMM) for recognition, but the weakness is that the training and testing samples are collected in the same session. Mandujano et al. [13] used fuzzy c-means classifier for authentication using keystroke dynamics.

Hosseinzadeh et al. [14] introduced Gaussian mixture model (GMM) for recognition using keystroke dynamics. The members were enrolled into the system by typing their full name 30 times. A longer text is expected to have a lower classification error. Hence the full name was chosen as entry string. The experiment produced an FRR of 4.8% and an FAR of 4.3%. Here the intruders are not aware of the actual login string information which helped in achieving low value of FAR.

1.3 Motivation and Contribution

Until today, almost all the computer login systems are based on username−password based model, where an intruder could be able to login into the system if he/she knows the particular text. From the literature it is clear that the research efforts are directed toward keystroke dynamics for the development of more robust security system. However, most of them suffer from practical problems due to unrealistic assumptions. In this paper, a static keystroke dynamics based recognition system is developed to enhance the computer login security using the same login string information. The main contribution of this paper includes the following:

(1) A detailed analysis of the authentication accuracy versus the input string length.
(2) The impact of negative (intruder) data-set in training to improve the authentication accuracy.
(3) Use of multiple trials to enhance the accuracy.

1.4 Organization

In Section 2, the detailed description of the building blocks of the system is described. The results and discussions are presented in Section 3 followed by conclusion in Section 4.

2. KEYSTROKE DYNAMICS BASED AUTHENTICATION SYSTEM

The keyboard dynamics based authentication is a practical and natural option as the sensor is available by default along with computer system [7]. So the keystroke dynamics based authentication system can be easily integrated with the existing hardware and login method as shown in Figure 1.

2.1 Enrolment

Since there is no public keystroke dynamics database available, a database is created for this experiment [15]. The timing of keystrokes is recorded by Matlab-provided functions under the following assumptions:

(1) The system will acquire keystroke information from 10 users (fellow researchers in the lab).
(2) Each user has agreed to share his/her login information (username and password), as well as to impersonate other users to generate the intruder data-set.
(3) To capture the variability of the keystroke dynamics behaviour over time, each user's data are collected over a period of eight weeks.
(4) Attempts with typographical error are discarded from the data-set.

2.2 Keystroke Features

While typing a string, two features, namely, keystroke latency (time interval between two consecutive key strokes) and keystroke hold time (the time for which particular key is pressed) are captured. For a string length of N, there will be $N-1$ latency and N hold

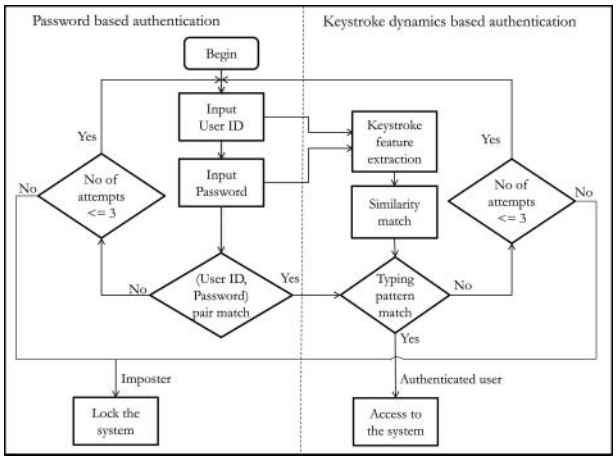

Figure 1: Flow chart of the proposed keystroke dynamics based authentication system.

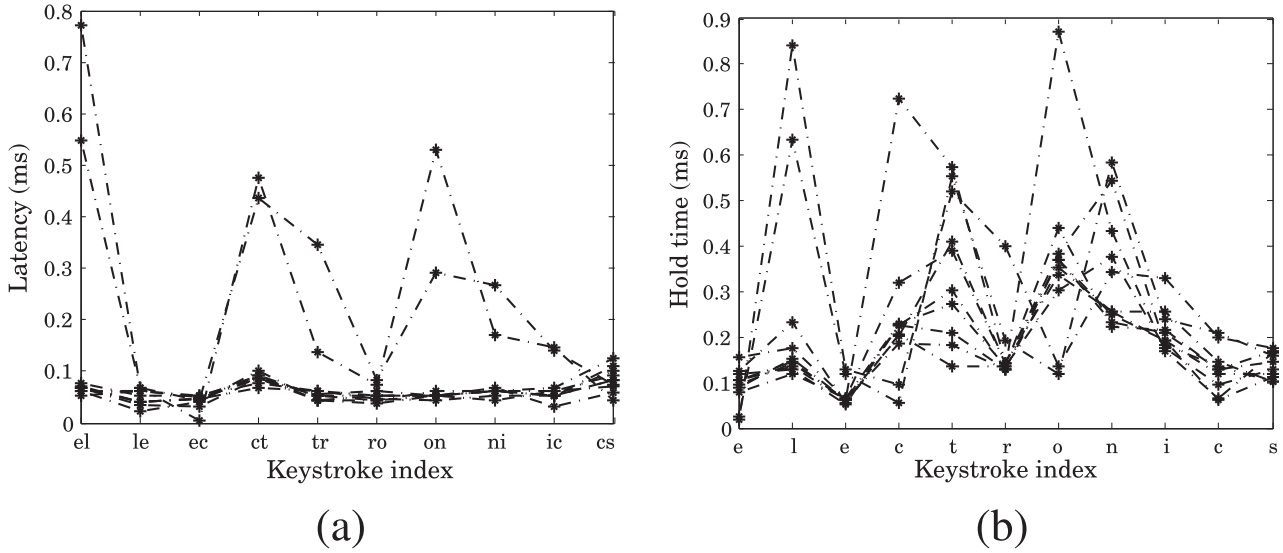

Figure 2: The plots of (a) latency and (b) hold time features derived from keystroke of user 1 for the keyword 'electronics'.

times [9]. In Figure 2, latency and hold time of a particular user are shown.

2.3 Training and Authentication

After extracting the two features from the raw keystroke data, the next task is to train the classifier for authentication. For training, the features can be arranged in two ways, namely, sequential mode and batch mode. In sequential mode, the feature vector length is 2. It is generated from every letter typed during the enrolment phase. In batch mode, the feature vector length is $2N-1$, where N is the number of keystrokes. N keystrokes along with $N-1$ number of latency generate feature vector of length $2N-1$. This increase may be helpful for better discrimination as it captures the chronology of the event. However, increase in feature vector length could pose challenge in training convergence. Two classification techniques, Gaussian mixture model (GMM) and NN, are used in the experiment.

2.3.1 Training of GMM Classifier

The GMM is a parametric probability density function represented as a weighted sum of Gaussian densities [16]. It is commonly used for the data-set with multiple dominant clusters. It can be represented as

$$p(\mathbf{x}|\lambda) = \sum_{k=1}^{M} \pi_k G\left(\mathbf{x}|\mu_k, \sum_k\right) \quad (1)$$

where π_k, $k = 1, 2, \ldots M$, are the mixing coefficients, $G(\mathbf{x}|\mu_k, \sum_k)$ is a multivariate Gaussian distribution with mean μ_k and covariance $\sum_k$. The mixing coefficients π_k should satisfy $\sum_{k=1}^{K} \pi_k = 1$. The GMM can be completely represented by these three parameters, mixing coefficients, mean, and co-variance, and represented by the notation $\lambda = \{\pi_k, \mu, \Sigma\}$. The parameters can be estimated using expectation−maximization algorithm [16]. To do the similarity check during authentication, the user-based threshold needs to be decided during the training period. A leave-one-out cross-validation technique is used to select the threshold due to small training sample size [14].

From total N training samples, $N-1$ samples are used to train the GMM, and the left-out single sample is used to get the likelihood values, i.e. to check the belonging with respect to the GMM model. This procedure is repeated N times to get the likelihood values. From these N log-likelihood values, the final threshold value is calculated experimentally as stated in Section 3.

2.3.2 Training of Neural Network Classifier

An artificial neural network (ANN) is a machine learning approach inspired by the structure of biological NNs [17]. In this method, a simple two-layer feed-forward network is used. The size of the input nodes is the same as the dimensionality of the input feature vector (d), and the number of hidden layer neurons ($2d + 1$) is decided based on the Kolmogrov's theorem [18]. The sigmoid activation function is used in the hidden layer and the output has a single neuron for binary output. A scaled conjugate gradient for fast supervised learning is used as the optimization technique [19].

2.3.3 Authentication

During authentication, either sequential mode technique or batch mode technique is used as shown in Figure 3. In sequential mode, the similarity score (log-likelihood score) values are generated by the classifier for every keystroke and stored until the full-text typing is completed. After completion of the typing, the stored score values are averaged to get the final score value. In batch mode, the similarity value is generated by the classifier after the full text is being typed. The authentication is performed based on the comparison of the scores with respect to the user-dependent thresholds (Th_1, Th_2).

For the GMM-based authentication, the average (L_{mean}) and the standard deviations (L_{sd}) of the N scores are calculated. The decision rule derived by the multiple trials can be specified as follows:

$$\begin{aligned}
&\text{If Score} = Th_1(= L_{\text{mean}} + k_1 \times L_{\text{sd}})\\
&\text{then user} \rightarrow \text{legal user;}\\
&/^*\text{The user is granted access to the system}^*/\\
&\text{else if } Th_1 = \text{Score} = Th_2,\\
&\text{then user} \rightarrow \text{trial user;}\\
&/^*\text{The user is allowed to re-attem pt}^*/\\
&\text{else Score} = Th_2(= L_{\text{mean}} - k_2 \times L_{\text{sd}}),\\
&\text{then user} \rightarrow \text{intruder;}\\
&/^*\text{The user is denied access to the system}^*/
\end{aligned} \tag{2}$$

where k_1 and k_2 are positive constants. The constants k_1 and k_2 are empirically selected.

For the NN classifier, the output is bounded ($0 \leq oi \leq 1$). The NN output score can provide the authentication as follows:

$$\begin{aligned}
&\text{If Score} = Th_1(= 0.5 + k_3)\\
&\text{then user} \rightarrow \text{legal user;}\\
&/^*\text{The user is granted access to the system}^*/\\
&\text{else if } Th_1 = \text{Score} = Th_2,\\
&\text{then user} \rightarrow \text{trial user;}\\
&/^*\text{The user is allowed to re-attempt}^*/\\
&\text{else Score} = Th_2(= 0.4 - k_4),\\
&\text{then user} \rightarrow \text{intruder;}\\
&/^*\text{The user is denied access to the system}^*/
\end{aligned} \tag{3}$$

where k_3 and k_4 are positive constants ($0 \leq ki \leq 0.5$, $i = 3, 4$). The constants k_3 and k_4 are empirically selected. For both the classifiers, after three chances, if the user score still remains in the trial zone, then the user is treated as an intruder.

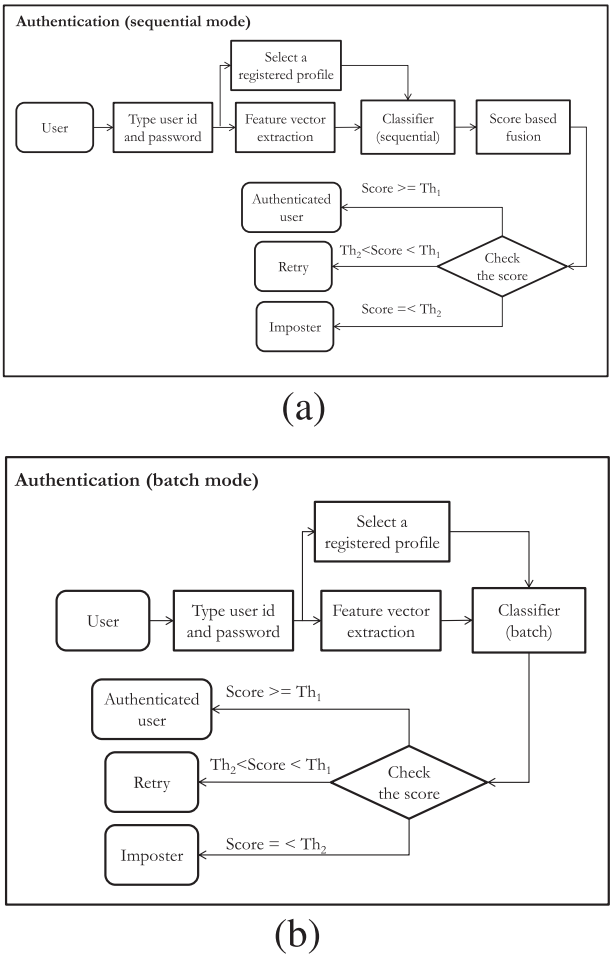

Figure 3: Block diagram of (a) sequential mode and (b) batch mode authentications based on keystroke dynamics.

3. RESULTS AND DISCUSSION

3.1 Performance Metrics

The performance of an authentication system can be quantified by different metrics. To define them, consider a two-class problem, where the data are labelled either as positive or negative and the outcome of the binary classifier will fall in one of the following four categories: true positive (TP), true negative (TN), false positive (FP), and false negative (FN) [20]. For this experiment, the FRR, FAR, accuracy, and area under the receiver operating characteristics (ROC) curve (A_z) matrices are used [3,21].

3.2 Experimental Setting

A summary of the keystroke database is shown in Table 1.

For authentication, a classifier is trained for each user. For GMM, only the legal user data are used for

Table 1: Summary of the keystroke database and experimental set-up

Number of users	10
Legal user sample size	200 usernames and 200 passwords (20 samples from each user)
Intruder sample size	900 usernames and 900 passwords (10 samples from each user for each login-id)
Username	First name of the user (size: 7−11 characters)
Password	Fixed-text 'electronics' (size: 11 characters)
Age	24−30
Gender	9 males and 1 female
Typing error	Not allowed
Duration of session	Eight weeks for each user
User profession	Research (typing with two fingers of each hand)
Hardware	Capacitive QWERTY keyboard, OEM-Compaq

training. For NN model of each user, the data from legal user and nine intruders are used for training. For GMM, the legal user data-set of the respective user is divided into two halves: training set and testing set. Using 10 samples, GMM model parameters are estimated and stored for every individual along with the threshold values (Th_1, Th_2) as described in Section 2.3.3. For NN model, the entire data-set (one legal user + nine intruders) of the respective user is divided into three parts: training (35%), testing (15%), and validation sets (50%).

3.3 Results Based on GMM and NN

To estimate the GMM for an individual, 10 positive training samples are used and the decision is produced by the rule as described in Eq. (2). Similarly, for NN classification for an individual, 10 positive samples and half of the negative samples (450) are used, and the final decision is derived following the rule as described in Eq. (3). In this section, all the values of the tabulated results are performance metric at the end of the three attempts. Two-fold cross validation is performed always, unless otherwise stated.

By examining different combinations of input string (fixed, variable), classifier and arrangement of feature vector (sequential or batch mode), the most suitable combination is found out in this experiment. In sequential mode, the size of the input feature vector is 2 irrespective of the length of the input string, whereas in batch mode for a string length of N, the size of the input feature vector is $2N-1$.

In Figure 4, graphs depict the relationship between classifier accuracy and the length of the input string using the GMM and NN classifiers. From Figure 4, it is evident that the accuracy increases with the length of the input string. It is also noticed that for first few letters (seven letters for GMM and eight letters for NN), the accuracy of the sequential mode is better than that of the batch mode, and later on, this trend is reversed. Authentication using the features in sequential mode enjoys the benefit of score level fusion; on the other hand, authentication using the features in batch mode gets the benefit of chronology of data. As the length of the input string increases, the benefit of the chronology of data supersedes that of the score level fusion resulting in superior performance of authentication using features in batch mode for long strings.

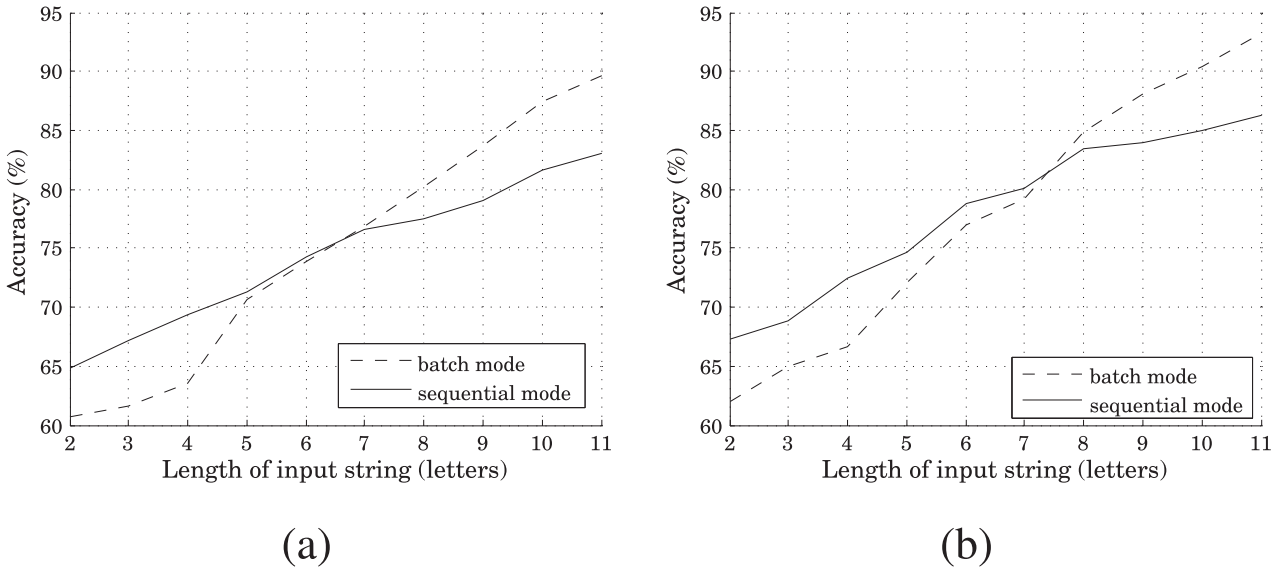

Figure 4: Results of average authentication accuracy versus the input string length (password) using (a) GMM in sequential and batch modes, and (b) NN in sequential and batch modes.

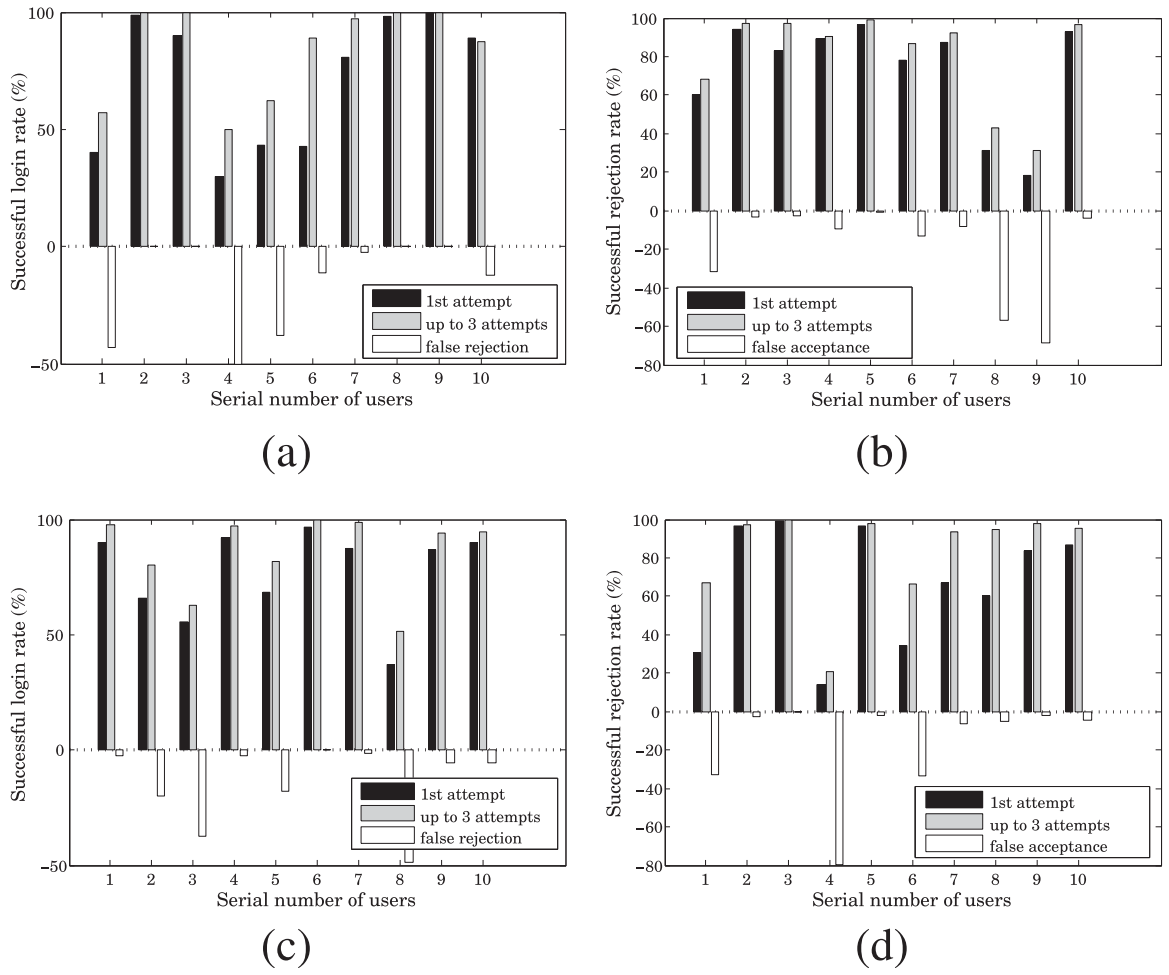

Figure 5: Results of GMM in sequential mode: (a) percentage of successful login in single and multiple attempts for legal users with password as input string; (b) percentage of successful rejection in single and multiple attempts for intruders with password as input string; (c) percentage of successful login in single and multiple attempts for legal users with both username and password as input strings; (d) percentage of successful rejection in single and multiple attempts for intruders with both username and password as input strings.

After exploring the effect of string length on the authentication performance, the GMM with feature in sequential mode is investigated for different possible input texts like fixed (password), variable (username), and combination of both fixed and variable (username and password).

Figure 5(a) displays the result of successful login and false rejection in single and multiple attempts for individual legal users, and Figure 5(b) shows the successful rejection rate and false acceptance rate in single and multiple attempts of the intruders. Here fixed-text password is used as input string. The average FRR, FAR, and accuracy achieved are 15.69%, 19.83%, and 82.24%, respectively. Using variable text (username), the average FRR, FAR, and accuracy achieved are 21.79%, 13.22%, and 82.49%, respectively. The corresponding figures are not shown for brevity. In the next test, both the username and password are together used as test string. Successful login and false rejection for legal users and the successful rejection and false acceptance in the case of intruders in single and multiple attempts are shown in Figure 5(c) and 5(d), respectively. The average FRR, FAR, and accuracy achieved are 14.11%, 16.85%, and 84.52%, respectively.

The trade-off between FAR and FRR prevents comparison of the experiments using these metrics and hence accuracy is used to compare different experiments. For GMM with features in sequential mode, the accuracies achieved are 71.4%, 72.1%, and 73.4% for only password, only username, and both username and password, respectively, after the first attempt. After three attempts, the respective accuracies have increased to 82.24%, 82.49%, and 84.52%. Therefore, for GMM with features in sequential mode, authentication accuracy

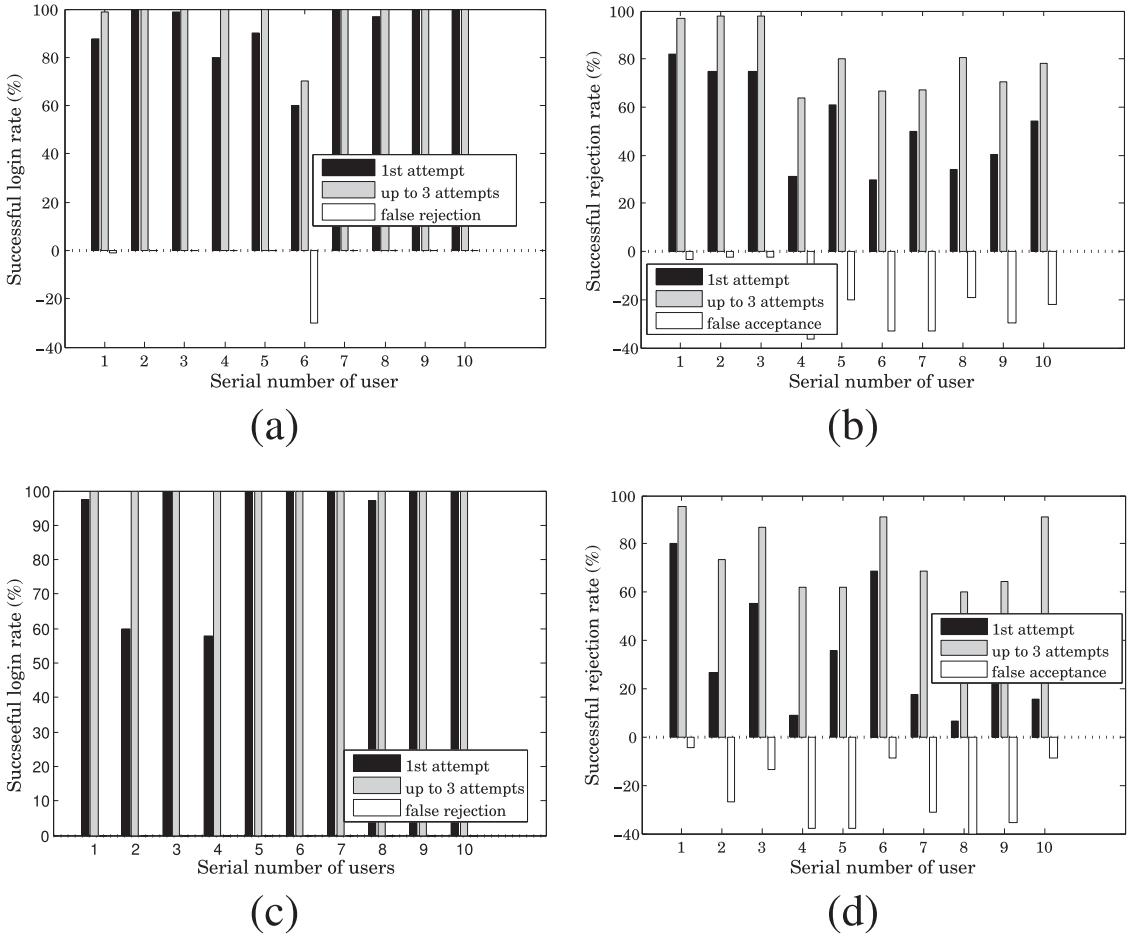

Figure 6: Results of NN in sequential mode: (a) percentage of successful login in single and multiple attempts for legal users with password as input string; (b) percentage of successful rejection in single and multiple attempts for intruders with password as input string; (c) percentage of successful login in single and multiple attempts for legal users with both username and password as input strings; (d) percentage of successful rejection in single and multiple attempts for intruders with both username and password as input strings.

has improved with the length of the input string and multiple attempts (up to three attempts).

Now features arranged in sequential mode using NN are investigated. Figure 6(a) displays the result of successful login and false rejection in single and multiple attempts for individual legal users, and Figure 6(b) shows the successful rejection rate and false acceptance rate in single and multiple attempts of the intruders. Here fixed text password is used as input string. The average FRR, FAR, and accuracy achieved are 3.11%, 20.09%, and 85.4%, respectively. Using variable text (username), the average FRR, FAR, and accuracy achieved are 0.66%, 29.47%, and 84.93%, respectively. The corresponding figures are not shown for brevity. In the next test, both the username and password are together used as test string. Successful login and false rejection for legal users and the successful rejection and false acceptance in case of intruders in single and multiple attempts are shown in Figure 6(c) and 6(d), respectively. The average FRR, FAR, and accuracy achieved are 0%, 24.43%, and 87.78%, respectively.

For NN with features in sequential mode, the accuracies achieved are 72%, 70.74%, and 73.48% for only password, only username and both username and password, respectively, after the first attempt. After three attempts, the respective accuracies have increased to 85.4%, 84.93%, and 87.78%. Therefore, for NN with features in sequential mode, authentication accuracy has improved with the length of the input string and multiple attempts (up to three attempts). Using feature in sequential mode, NN has outperformed GMM in all the cases. NN-based classification using username and password together as input string is found to be the most effective combination.

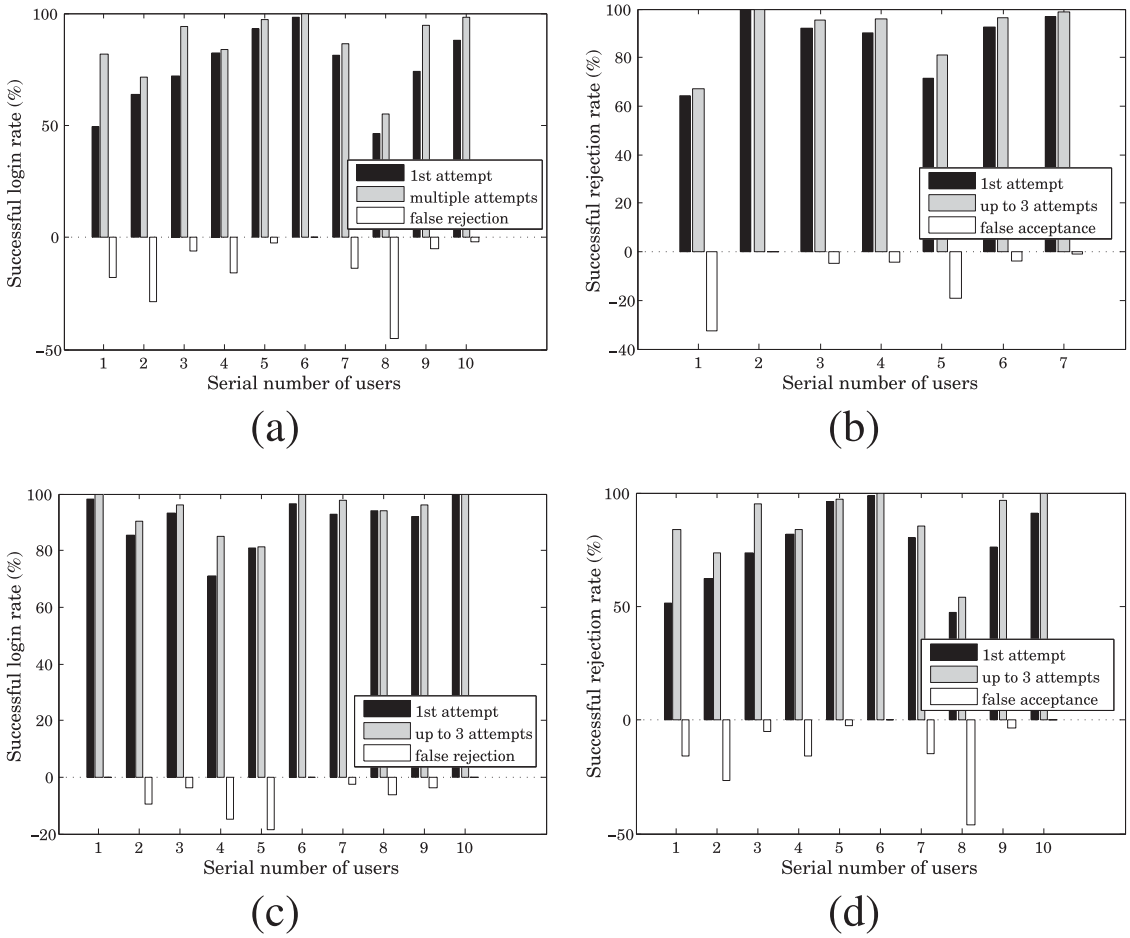

Figure 7: Results of GMM in batch mode: (a) percentage of successful login in single and multiple attempts for legal users with password as input string; (b) percentage of successful rejection in single and multiple attempts for intruders with password as input string; (c) percentage of successful login in single and multiple attempts for legal users with both username and password as input strings; (d) percentage of successful rejection in single and multiple attempts for intruders with both username and password as input strings.

Next, features arranged in batch mode using GMM are investigated. Figure 7(a) displays the result of successful login and false rejection in single and multiple attempts for individual legal users, and Figure 7(b) shows the successful rejection rate and false acceptance rate in single and multiple attempts of the intruders. Here fixed text password is used as input string. The average FRR, FAR, and accuracy achieved are 13.71%, 9.34%, and 88.47%, respectively. Using variable text (username), the average FRR, FAR, and accuracy achieved are 18.32%, 5.98%, and 87.85%, respectively. The corresponding figures are not shown for brevity. In the next test, both the username and password are together used as test string. Successful login and false rejection for legal users and the successful rejection and false acceptance in case of intruders in single and multiple attempts are shown in Figure 7(c) and 7(d), respectively. The average FRR, FAR, and accuracy achieved are 5.92%, 13.04%, and 90.52%, respectively.

For GMM with features in batch mode, the accuracies are 80.37%, 74.2%, and 84.0% for only password, only username, and both username and password, respectively, after the first attempt. After three attempts, the respective accuracies have increased to 88.47%, 87.85%, and 90.52%. Therefore, for GMM with features in batch mode, authentication accuracies have improved with the length of the input string and multiple attempts (up to three attempts). The GMM with features in batch mode proved to be more accurate compared to GMM and NN with features arranged in sequential mode.

Next, features arranged in batch mode using NN are investigated. Figure 8(a) displays the result of successful login and false rejection in single and multiple attempts for individual legal users, and Figure 8(b) shows the successful rejection rate and false acceptance rate in single and multiple attempts of the intruders. Here fixed text

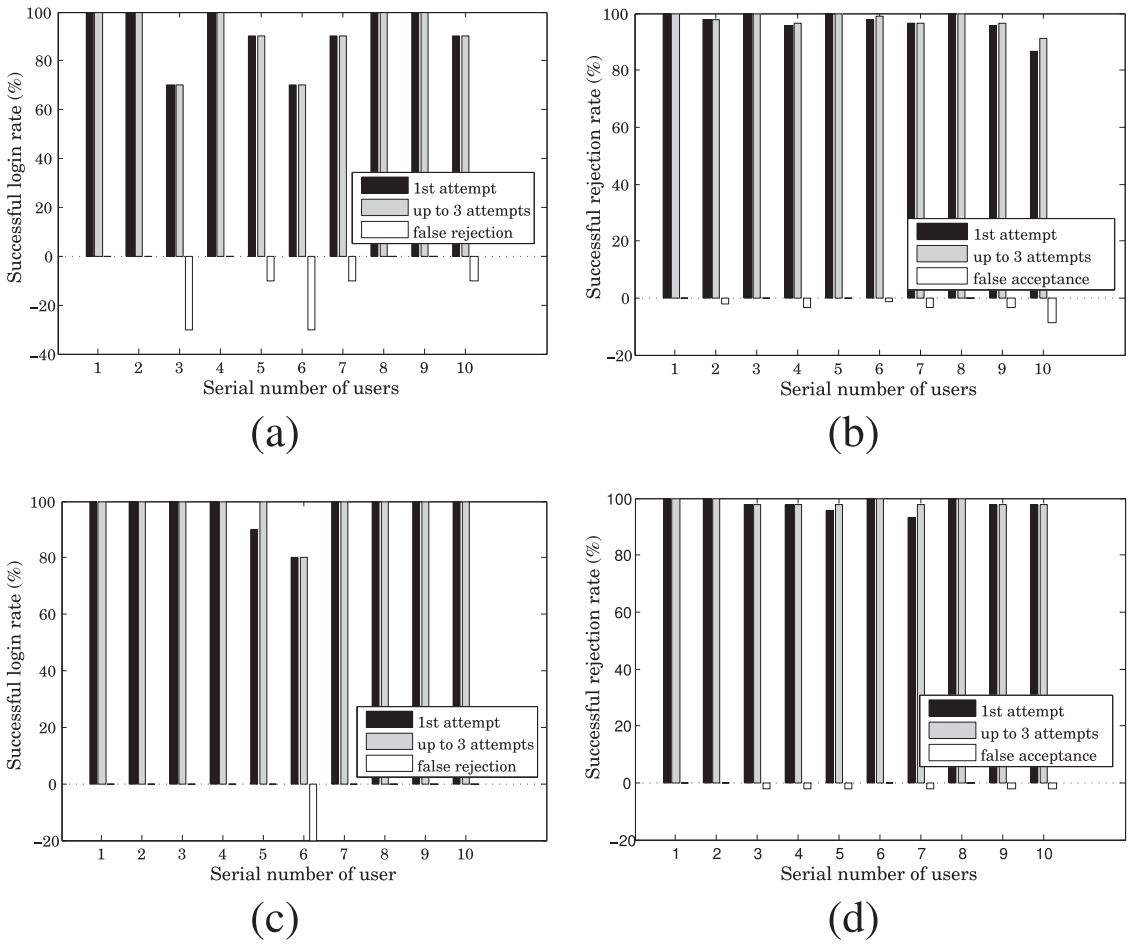

Figure 8: Results of NN in batch mode: (a) percentage of successful login in single and multiple attempts for legal users with password as input string; (b) percentage of successful rejection in single and multiple attempts for intruders with password as input string. (c) Percentage of successful login in single and multiple attempts for legal users with both username and password as input strings; (d) percentage of successful rejection in single and multiple attempts for intruders with both username and password as input strings.

password is used as input string. The average FRR, FAR, and accuracy achieved are 9%, 2.22%, and 94%, respectively. Using variable text (username), the average FRR, FAR, and accuracy achieved are 1%, 3.99%, and 97.5%, respectively. The corresponding figures are not shown for brevity. In the next test, both the username and password are together used as test string. Successful login and false rejection for legal users and the successful rejection and false acceptance in case of intruders in single and multiple attempts are shown in Figure 8(c) and 8(d), respectively. The average FRR, FAR, and accuracy achieved are 2%, 1.33%, and 98.53%, respectively. For NN with features in batch mode, accuracies achieved are 93.99%, 95.94%, and 97.5% for only password, only username, and both username and password, respectively, after the first attempt. After three attempts, the respective accuracies have increased to 94%, 97.5%, and 98.53%. Therefore, after investigating all the possible combinations, it is evident that the performance of NN with features in batch mode using username and password together as input text gives the best accuracy.

For better comparison, all the competing techniques are tuned to a particular FAR value and the corresponding FRR, the accuracies are tabulated in Table 2. To judge the overall performance of the techniques, area under the ROC curve (A_z) values are also displayed.

From Table 2, the following conclusions are derived:

(1) The batch mode classification results are good compared to the sequential-based results.
(2) The NN classification performs better than the GMM. The use of both the positive and negative data-sets in the NN results in better accuracy compared to that of the GMM.
(3) The best results of 90% and 99% accuracies are achieved using GMM and NN classifiers, respectively.

Table 2: Performance figures of keystroke dynamics based authentication using GMM and NN classifiers (after three attempts). Here password and username are denoted by 'P' and 'U', respectively

Classifier	Input text	Feature vector size	FAR (%)	FRR (%)	Accuracy (%)	*Az*
GMM	P	2	2.5	57.5	70	0.816
	P	$2N-1$	2.5	47.5	75	0.911
	U	2	2.5	70	63.75	0.863
	U	2N-1	2.5	52.5	72.5	0.921
	U and P	2	2.5	65	66.25	0.894
	U and P	$2N-1$	2.5	67.5	65	0.901
NN	P	2	2.5	80	58.75	0.930
	P	$2N-1$	2.5	5	96.25	0.989
	U	2	2.5	50	73.75	0.803
	U	$2N-1$	2.5	15	91.25	0.940
	U and P	2	2.5	80	58.75	0.914
	U and P	$2N-1$	2.5	0	98.53	0.999

The corresponding ROC of the two are plotted in Figure 9. The superiority of NN over GMM is reconfirmed. Both the GMM and NN classifiers perform the best when username and password together are used as input string and the derived features are used in batch mode.

The proposed NN classifier with features fed in batch mode technique can be compared with the results reported in the literature. Obiadat and Sasoun [9] have used NN for keystroke classification and the features are arranged in sequential mode. This paper reported FAR = 0% and FRR = 0%, for 7140 number of training samples, with an average string length of 7 (7140 × 7 = 49,980 letters). In the proposed method, only 110 training samples (from both legal user and intruders using username and password) with an average string length of 20.5 (110 × 20.5 = 2255 letters) are used. Hence the proposed method has shown that using a small-size training sample and arranging the features in batch mode, comparable accuracy can be achieved. However, the inclusion of intruder data-set (negative dataset) into the training samples is not a suitable characteristic for the biometric authentication, because true representation of all the intruders is quite impossible to collect. Since the GMM can be trained only with the positive data-set, Hosseinzadeh and Krishnan [14] had used the technique. Their approach resulted in FAR = 4.3% and FRR = 4.8%, based on 30 training samples of average length of 10+ (300 letters). The assumption that the intruders were not aware of the actual input login string indicates minimal dependence on biometric [14]. In the proposed GMM-based batch mode experiment with 20 samples and an average string length of 20.5 (10 × 20.5 = 205 letters), the FAR and FRR are 13.04% and 5.92%, respectively, depending only on keystroke biometric, i.e. the intruders are completely aware about the respective legal users' input login strings.

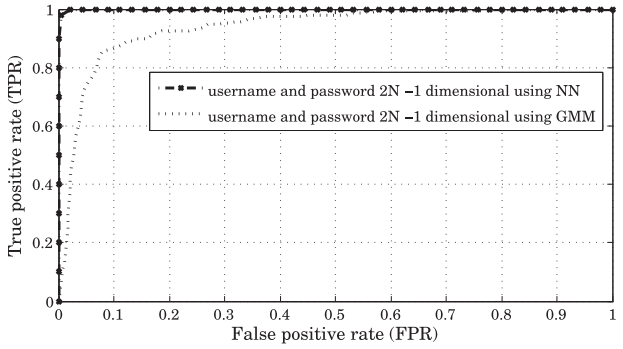

Figure 9: ROC of the GMM and NN classifiers using username and password as input text and features in batch mode.

4. CONCLUSION

In this paper, it is shown that the features extracted from the keystroke dynamics behaviour are distinct enough to verify computer-literate users properly after a small training. The GMM and NN classifiers based authentication techniques are applied during the login as the user keyed in their username and/or password. The extracted features are arranged both in sequential and batch modes. This paper shows that the accuracy has increased with the length of the input string and by arranging the features in batch mode compared to the sequential mode. However, because of the raised dimensionality of the feature vector, the complexity is comparatively higher in the batch mode. The best authentication accuracy of 90% and 99% are achieved in the batch mode configuration for multiple login attempts using GMM and NN classifiers, respectively, using two-fold cross-validation technique. It should be noted that the use of GMM is more practical, as in real life only positive data-set is available.

The keystroke-based authentication can be extended for continuous authentication during the use of computer due to its non-intrusive and non-obstructive nature [4]. It can be integrated with the operating system to enhance the existing login security system with a minimal requirement of the processor time.

ACKNOWLEDGEMENTS

This work has been supported by Signal Processing and Machine Learning in Pervasive Healthcare project sponsored by Intel Technologies India Pvt. Ltd, Bangalore. We would like to thank Dr Satish Prasad Rath, Sreenivas Subramoney, and Sumeet Verma of Intel Technologies India Pvt. Ltd. for their constant encouragement and support. We are also thankful to co-workers Lavanya Sainik, Subhadeep

Mukhopadhyay, and our laboratory in-charge, Mr Arumoy Mukhopadhyay, for their helpful suggestions and advice at the moments of crisis.

REFERENCES

1. N. L. Clarke, and S. M. Furnell, "Advanced user authentication for mobile devices," *Comput. Sec*, Vol. 26, no. 2, pp. 109–19, 2007.
2. D. Hosseinzadeh, S. Krishnan, and A. Khademi, "Keystroke identification based on Gaussian mixture models," *IEEE Int. Conf. Acoust. Speech Signal Process*., Vol. 3, pp. 1144–7, 2006
3. A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Trans. Circuits Syst. Video Technol. Special Issue Image Video-Based Biometr.*, Vol. 14, no. 1, pp. 1–29, 2004.
4. R. K. Das, S. Mukhopadhyay, and P. Bhattacharya, "Continuous multimodal biometric authentication for PC and handheld devices," *IETE J. Educ.*, Vol. 52, pp. 59–69, 2011.
5. R. Gaines, W. Lisowski, S. Press, and N. Shapiro, "Authentication by keystroke timing: Some preliminary results (Report style)," Rand Corporation, Santa Monica, CA, Rand Rep. R-2560-NSF, pp. 1–52, 1980.
6. R. Joyce, and G. Gupta, "Identity authentication based on keystroke latencies," *Commun. ACM*, Vol. 33, no. 2, pp. 168–76, 1990.
7. F. Monrose, and A. Rubin, "Keystroke dynamics as a biometric for authentication," *Future Gener. Comput. Syst.*, Vol. 16, no. 4, pp. 351–9, 2000.
8. J. J. Leggett, and G. Williams, "Verifying identity via keystroke characteristics," *Int. J. Man Mach. Stud.*, Vol. 28, no. 1, pp. 67–76, 1988.
9. R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. 2nd ed. Hoboken, NJ: Wiley, 2007.
10. M. Obaidat, and B. Sadoun, "Verification of computer users using keystroke dynamics," *IEEE Trans. Syst. Man Cybern. Part B*, Vol. 27, no. 2, pp. 261–9, 1997.
11. M. Brown, and S. J. Rogers, "User identification via keystroke characteristics of typed names using neural networks," *Int. J. Man Mach. Stud.*, Vol. 39, no. 6, pp. 999–1014, 1993.
12. E. Yu, and S. Cho, "GA-SVM wrapper approach for feature subset selection in keystroke dynamics identity verification," in *Proceedings of the IEEE International Joint Conference on Neural Networks*, Portland, Oregon, Jul. 2003, pp. 2253–7.
13. W. Chen, and W. Chang, "Applying hidden Markov models to keystroke pattern analysis for password verification," in *Proceedings of the IEEE International Conference on Information Reuse and Integration (IRI)*, Las Vegas, Nov. 2004, pp. 467–74.
14. S. Mandujano, and R. Soto, "Deterring password sharing: User authentication via fuzzy c-means clustering applied to keystroke biometric data," in *Proceedings of the Fifth Mexican International Conference on Computer Science (ECN04)*, Colima, Mexico, pp. 181–7.
15. D. Hosseinzadeh, and S. Krishnan, "Gaussian mixture modeling of keystroke patterns for biometric applications," *IEEE Tran. Syst. Man Cybern. Part C*, Vol. 38, no. 6, pp. 816–26, 2008.
16. P. S. Teh, B. Andrew, and S. O. Thian, "Keystroke dynamics in password authentication enhancement," *Expert Syst. Appl.*, Vol. 37, pp. 8618–27, 2010.
17. C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY: Springer, 2009.
18. T. M. Mitchell, *Machine Learning*. New York, NY: McGraw-Hill Science, 1997.
19. A. N. Kolmogorov, "On the representations of continuous functions of many variables by super positions of continuous functions of one variable and addition," *Dokl. Akad. Nauk. USSR*, Vol. 114, no. 5, pp. 953–6, 1957.
20. M. F. Moller, "A scaled conjugate gradient algorithm for fast supervised learning," *Neural Netw.*, Vol. 6, pp. 525–33, 1993.
21. L. Hong, and A. K. Jain, " Integrating faces and fingerprints for personal identification," *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 20, no. 12, 1998.

Authors

Rajat Kumar Das received his B.E. degree in electronics and communication engineering from Visvesvaraya Technological University, Karnataka, India, in 2005. He worked as an engineer in Cranes software international limited, Bangalore, India, for 4 years. At present he is an MS student in IIT-Kharagpur, under Prof. Sudipta Mukhopadhyay in the Electronics & Electrical Communication Engineering, where he is working on biometric recognition.

E-mail: rajatdas2005@gmail.com

Sudipta Mukhopadhyay received his B.E. (Electrical) from Jadavpur University, India, in 1988, M.Tech. and Ph.D. in Electrical Engineering from the Indian Institute of Technology, Kanpur in 1991 and 1996, respectively. After 10 years in various industries – Tata Consultancy Services Limited, Satyam Computers, Silicon Automation Systems (SASKEN Communications), General Electric India Technology Center, and Philips – he joined the Indian Institute of Technology, Kharagpur in 2005, as an assistant professor. At present he is serving IIT Kharagpur as an associate professor.

E-mail: smukho@gmail.com

Puranjoy Bhattacharya received his B.Tech., M.Tech., and Ph.D. in Electrical Engineering from the Indian Institute of Technology, Kanpur, India, in 1990, 1993 and 1999, respectively. Since 1998, he has been working in industrial R&D in the areas of signal processing, pattern recognition and machine intelligence, particularly as applied to biomedical, audio and speech signals. He currently works at Intel Labs at Bengaluru, India.

E-mail: puranjoy.bhattacharya@intel.com

DOI: 10.1080/03772063.2014.914686; Copyright © 2014 by the IETE