KEYSTROKE IDENTIFICATION BASED ON GAUSSIAN MIXTURE MODELS

Danoush Hosseinzadeh, Sridhar Krishnan, April Khademi

Department of Electrical and Computer Engineering
Ryerson University, Toronto, ON - M5B 2K3 Canada
Email: (danoushh@hotmail.com) (krishnan@ee.ryerson.ca) (akhademi@ieee.org)

ABSTRACT

Many computer systems rely on the username and password model to authenticate users. This method is widely used, yet it can be highly insecure if a user's login information has been compromised. To increase security, some authors have proposed keystroke patterns as a biometric tool for user authentication; they can be used to recognize users based on how they type. This paper introduces a novel method that applies GMMs to keystroke identification. The major benefit of this method is the ability to update the user's model each time he or she is authenticated. Therefore, as time goes on, each user model accurately reflects the changes in that user's keystroke pattern. Using this method, a FAR and a FRR rate of approximately 2% was achieved. However, it should be noted that 50% of the test subjects were the traditional "two finger" typists and therefore, this had a disproportionately negative impact on the results.

1. INTRODUCTION

Undeniably, computers have become an essential part of daily life for many people around the world. One of the main reasons for this trend is that computers allow us to access information from any part of the globe. Additionally, they allow us to perform many functions that would otherwise require a physical presence else where, such as banking, shopping and some personal tasks such as online chatting and so on.

Despite their importance, computer systems are generally protected with primitive techniques, such as usernames and passwords. Since passwords can be stolen, accidentally revealed or even cracked by dictionary programs, there has been a great number of electronic crimes in recent years. In fact, reports indicate that in 2002, online retailers lost an estimated US$1.64 billion dollars in fraudulent sales and an additional US$1.82 billion in legitimate sales that looked suspicious [1].

To prevent crime and increase security, access should only be given to the correct users. To achieve this goal, some authors have suggested the use of keystroke identification as a method of preventing unauthorized users from accessing a computer system [2][3][4][5]. Keystroke identification is a biometric tool based on the principle that every person has a unique typing pattern, similar to a hand written signature [2][5]. Particularly, for regularly typed strings, this pattern can be very consistent and therefore, it can be effective for user identification. Furthermore, we argue that a person's keystroke pattern would be harder to duplicate than a signature because an intruder does not have an unlimited number of tries to practice it. In a commercial system, a user who cannot successfully log in after a predetermined number of attempts, could be locked out from the system, therefore, limiting the intruder's practice time. Studies have also shown that even among professional typists there is a great deal of variability in the keystroke patterns [6]. This makes user forgery very difficult.

By exploiting these keystroke patterns, we can add an additional layer of security to the username/password model. Even if authorized persons reveal their passwords, no unauthorized user can gain access to the system. This idea has many internet-based applications, especially for online banking, email and user account protection, just to name a few. In fact, we can completely change the username/password security model to a model which only relies on keystroke patterns. Aside from increased security, this model would benefit users because they will not have to remember many different usernames/passwords pairs for different accounts. Also, the possibility of a user forgetting their password or a user having a password that is easy to decipher would be reduced.

In this paper, a brief review will be presented on what features could be extracted from keystroke patterns and under what conditions good features can be acquired. Also, a new method for modeling these features based on Gaussian Mixture Models (GMMs) is proposed. For completeness, a brief review of GMMs is presented before describing the novel algorithm used. Lastly, the results and conclusions are presented.

2. KEYSTROKE FEATURES

2.1. Features From Keystrokes

It has been shown that for a given user at least two unique features can be extracted from keystroke patterns [6]. Keystroke patterns, which are produced by the user during typing, exhibit unique timing characteristics. One such characteristic is the keystroke latencies (KL), which is the time between strik-

ing two consecutive keys. Another characteristic (feature) is the key down time (KD), which is the time a particular key is held down. These features have been used in previous research to produce good results in user identification.

For a string of length N, there are $N-1$ KL data points and N KD data points. These data points can be used to create two feature vectors. Fig. 1 shows the KL and KD plot for a particular user (one of the authors) that has typed his name repeatedly. Fig. 1 is included to illustrate the stability and strong correlation that exists between each of the feature vectors, KD and KL.

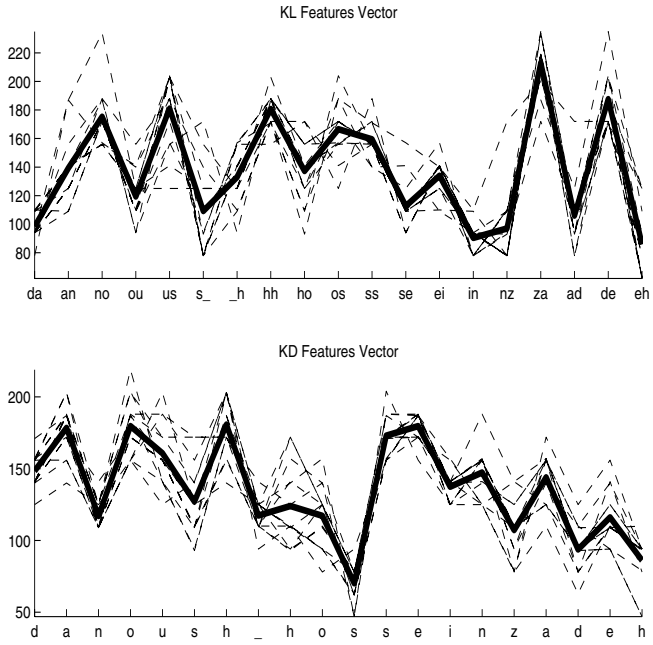

Fig. 1. Several plots of the keystroke latency (KL) and key down time (KD) feature vectors for one user. The bold line is the average of the vectors. The space character is represented by "_".

2.2. Designing Good Features

For keystroke identification, a robust feature pattern is one that is stable over repeated trials. To produce a stable feature pattern, the typist should be able to type the given text without any hesitation. Strings that require the typist to stop and think about the next letter or cause them to pause and search for a certain key, will result in an unstable pattern. As mentioned before, research has shown that the best results are obtained when users type familiar text such as, their first and last names. Such features are intuitively easy to type because they have been used for many years. Therefore, a distinct pattern can be seen when users type their name.

Another important consideration when selecting appropriate text, is the number of characters. Shorter text tends to increase classification error because it can be more easily reproduced by others [5]. This is true because fewer number of characters have a less complex patterns and can be imitated more easily by imposters. The same problem exists with hand written signatures, where short and simple signatures are often easy to copy.

In previous work, it has been suggested that no less than ten characters should be used for keystroke identification [5]. In this work, the user is required to enter at least ten characters, which can be easily accomplished with the first and last name of the individual. At the same time it will be said that no additional effort was made to increase the minimum character length, because it might be difficult or annoying for some users to meet the requirement. This would also pose a strict requirement if the user's full name does not meet the minimum character requirement, or if the user chooses a different string. These factors could have a negative impact on false acceptance rates (FAR) and false rejection rates (FRR).

2.3. Data Acquisition Model

To collect timing information, a data acquisition application named 'KbApp' was designed for the Windows operating system. With this application, keystroke timing error was minimized to less than 0.5 milli-seconds, with the option of reducing it to 100 nano-seconds. However, this error will not have a significant impact on the results because the average feature has a time value that is to the order of 100 milli-seconds.

2.4. Review of GMMs

GMMs are a well known method for modeling the probability distribution of random events. By several weighted L dimensional Gaussian functions, it is possible to closely approximate any distribution, provided that enough training data is available. The complete GMM can be expressed by the mean vector $\vec{\mu_i}$, covariance matrix Σ_i and mixture weights w_i as given below:

$$\lambda = \{w_i, \vec{\mu_i}, \Sigma_i\}, \quad i = 1, ..., K \tag{1}$$

Using the model λ, we can obtain the the likelihood that $\vec{x}$ belongs to the model λ by

$$p(\vec{x}|\lambda) = \sum_{i=1}^{K} w_i b_i(\vec{x}), \tag{2}$$

where b_i is given by an L-dimensional Gaussian PDF as shown below:

$$b_i(\vec{x}) = \frac{1}{(2\pi)^{L/2}} |\Sigma_i|^{-1/2} \, exp\left\{-\frac{1}{2}(\vec{x}-\vec{\mu})^t \, \Sigma_i^{-1}(\vec{x}-\vec{\mu})\right\} \tag{3}$$

GMMs can be very effective in modeling the type of distributions found in keystroke patterns, which are shown in Fig. 1.

To verify the likelihood that a given feature vector $\vec{x}$ belongs to the a model λ, the natural logarithm of the associated probability is used. This value, which we call the Log-Likelihood (LL) is given below:

$$LL = log\left\{p(\vec{x}|\lambda)\right\} = log\left\{\sum_{i=1}^{K} w_i b_i(\vec{x})\right\} \tag{4}$$

3. A NOVEL KEYSTROKE IDENTIFICATION METHOD

The novel method proposed in this paper uses GMMs to model keystroke timing information and uses the log-likelihood measure to the authenticate the user based on a threshold.

3.1. GMM Training and Verification

To produce a GMM, the user is first required to enroll into the system by typing their full name ten times. These ten samples produce twenty feature vectors; ten KL vectors and ten KD vectors. From these two sets of ten sample vectors, two GMMs can be trained, one for the KD feature and one for the KL feature. The expectation maximization (EM) algorithm was used to train the GMMs.

Upon verification, the user is required to re-enter their full name. From this test vector, the KL and KD feature vectors are extracted and compared with the user's model, which is obtained from the enrolment session. Equation 4, is used to calculate the log-likelihood that the test vector ($\vec{x}$) belong the the given model. This result is then compared with the user's threshold before access is granted or denied.

The results show the statistics for the system when access is based on using the KD feature, the KL feature and a combination of KL and KD features. In the later case, the test vector is compared with both user models (KL model and KD model) before access is granted. Also, each time the user is authenticated successfully, both GMM models and thresholds are updated with the new information.

3.2. Calculating Model Thresholds

To obtain the user's threshold, the Leave-One-Out-Method (LOOM) is used. The LOOM is as follows: for N feature vectors, $N-1$ vectors are used to train the model and the last vector is used to test the likelihood that it belongs to that model, using Equation 4. This test can be performed N times, where each time a different vector is used to test the model. The final results of the LOOM produces N likelihood measures and can be expressed by

$$LL_j = log\left\{p(\vec{x_j}|\lambda)\right\}, \quad j = 1, 2, .., N \tag{5}$$

where λ is a GMM that has been trained with $N-1$ vectors not including the j^{th} vector and $\vec{x_j}$ is the test vector.

These N log-likelihood results are further processed before selecting the models threshold. From these likelihood values, the minimum value that falls within the range of three standard deviations away from the mean is set as the model threshold, as given below:

$$Threshold = \min_{\forall j}\left\{LL_j \,|\, (\, LL_j - \overline{LL}\,) < 3\sigma\right\} \tag{6}$$

where $\overline{LL}$ is the mean and σ is the standard deviation of the LL values obtained from the leave-one-out-method.

The model generation and threshold calculation procedures are repeated every time the user has been verified so that the model and threshold are adaptive and can change with the user over time.

3.3. Authenticating The User

User authentication is the main goal of this work. To achieve this, the keystroke model should be robust enough to produce a low false rejection rate (FRR) and a low false acceptance rate (FAR). FAR is the rate at which intruders can gain access to a valid user's account, and FRR is the rate at which valid user's are denied access to their own account. Obviously, both these measures should be as low as possible.

In this approach, authentication is performed in two stages. In the first stage, if the user is denied access, they are given a second chance to entre their name. By using this method a significant improvement was be seen in the FRR and is discussed in the results section.

4. EXPERIMENTAL RESULTS

Before presenting the results, the reader is reminded that the number of initial training vectors used to calculate the model thresholds was ten. Because it is desired to have an accurate threshold based on the training vectors, the LOOM was used, as described in Section 3.2. It has been shown that the LOOM provides the least unbiased estimate for small databases [7]. Therefore, the model thresholds used to authenticate the users are optimal given the size of the database.

The results for FRRs and FARs for three different cases are presented in Table 1. It should be noted by the reader that the algorithm should also function well in terms of FRR and FAR, over time. The main reason for this behavior is that the proposed method adaptively selects the threshold that best suits the individual user, based on the LOOM. Also, the algorithm has shown that using a two stage verification process (ie. the user is given a two chances for authentication), decreases the FRR significantly.

To perform imposter tests, two typists were chosen to observe and imitate the other users' typing pattern. The results indicate an average FAR and FRR of about 2% using both features. These figures are comparable to other techniques however, a direct comparison with other methods cannot be justified because in each experiment a different database has been used. In our database, four out of the eight typists were the traditional "two finger" typers. We believe this led to poor performance in both the FAR and the FRR because these types of users do not produce a very stable keystroke pattern and at the same time can be copied easily. Therefore, because their finger patterns can be easily seen and imitated by the imposter users, the FAR results presented here are skewed. In terms of FRR, these users also do not perform well because they have a lot of variation in their typing pattern. In fact,

Table 1. Experimental Results for FRR and FAR

	KL Feature		**KD Features**		**KL&Kd Features**	
User	$FRR(\%)$	$FAR(\%)$	$FRR(\%)$	$FAR(\%)$	$FRR(\%)$	$FAR(\%)$
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	5.3	14.3	0	14.3	5.3	7.1
4	0	9.5	0	0	0	0
5	0	0	0	0	0	0
6	5.6	0	5.6	0	8.3	0
7	0	50	0	10	0	10
8	0	0	5.9	20	5.9	0
Average(%)	1.4	9.2	1.4	5.5	**2.4**	**2.1**

more users should be enrolled before the performance can be fully evaluated.

This experiment obtained two features from the keystroke data and performed three similarity tests. The combination of the KL and KD features should produce a lower FAR and higher FRR compared to using either of the features individually. This is due to the fact that a user must correctly produce both features simultaneously. These trends were observed in the results, as can be seen from Table 1. A major benefit of this method over existing techniques is the ability to update the user model each time as he/she is successfully authenticated. Therefore, as time goes on, each user's model accurately reflects the changes in that person's keystroke pattern.

5. CONCLUSIONS

A novel method for authenticating computer users based on keystroke identification was presented. Upon verification, the keystroke latencies and key hold-down times for the user's keyboard inputs were recorded and compared with a predefined individualistic model. Access was granted if the user's input reached a certain threshold. A new method for calculating model threshold was also introduced using the LOOM and log-likelihood of the feature vectors.

Ideally the FAR and the FRR should be very small with more emphasis give to former because a security breach is more critical than a valid user being forced to re-authenticate. Based on this logic, the best results were obtained using both the KL and KD features simultaneously; which produced a FRR of 2.4% and a FAR of 2.1%.

Despite the fact that these results are based on a small database, it has been shown by this work that GMMs can be used effectively to identify users based on their keystroke patterns. Furthermore, despite the fact that 100% classification accuracy was not achieved, more users should be enrolled using this approach before a definitive answer can be given about the capability of the system. As mentioned earlier, the results presented are skewed because of the type of users enrolled (50% of users were two finger typers). This technique could be further improved to incorporate the varied nature of the different typists.

GMMs may be used with other metrics to improve both the FAR and FRR, or the threshold procedure can be modified to produce more accurate resutls. In future works, we intend to investigate these areas with a larger database.

6. REFERENCES

[1] Alen Peacock, Xian Ke, and Matthew Wilkerson, "Typing patterns: A key to user identification," *IEEE Security & Privacy Magazie*, vol. 2, no. 5, pp. 40–47, Oct. 2004.

[2] Rick Joyce and Gopal Gupta, "Identity authentication based on keystroke latencies," *Communication of the ACM*, vol. 33, no. 2, pp. 168–176, February 1990.

[3] Oscar Coltell, Jose M. Dabia, and Guillermo Torres, "Biometric identification system based on keyboard filtering," in *Proc. IEEE 33rd Int. Carnahan Conf. on Secutrity Technology*, Madrid, Oct. 1999, pp. 203–209.

[4] Saleh Bleha, Charles Slivinsky, and Bassam Hussien, "Computer-access security systems using keystroke dynamics," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 12, no. 12, pp. 1217–1222, December 1990.

[5] Livia C. F. Araujo, Luiz H. R. Sucupira Jr., Miguel G. Lizarraga, Lee L. Ling, and Joao B. T. Yabu-Uti, "User authentication through typing biometrics features," *Signal Processing, IEEE Transactions on*, vol. 53, no. 2, pp. 851–855, Feb. 2005.

[6] R. Gaines, W. Lisowski, S. Press, and N. Shapiro, "Authentication by keystroke timing: Some preliminary results," Tech. Rep. R-256-NSF, Rand Corporation, Santa Monica, CA, USA, May 1980.

[7] Keinosuke Fukunaga, *Introduction to Statistical Pattern Recognition (2nd ed.)*, Academic Press Professional Inc., San Diego, CA, USA, 1990.