User Authentication with Adaptive Keystroke Dynamics

Shimaa I. Hassan[1], Mazen M. Selim[2], and Hala H. zayed[3]

[1] Department of computer systems, Faculty of engineering, Benha university, Shoubra, Egypt

[2] Department of computer Science, Faculty of computers and informatics, Benha university, Benha, Egypt

[3] Department of computer Science, Faculty of computers and informatics, Benha university, Benha, Egypt

Abstract

Recently, keystroke dynamics increasingly being a field of interest for researchers, where users can access different systems through their typing styles, which increases the level of security. This paper tends to implement a robust keystroke dynamics system; it tends to solve the problem of samples variations by using an adaptive threshold. The proposed system is evaluated using CMU dataset, and a new dataset created for this work. Results obtained are compared with others reported in literature and proved to have good performance.

Keywords: *Keystroke dynamics, Timing features, Distance based measures, Leave-One-Out-Method.*

1. Introduction

Biometric systems make use of the physiological and behavioral traits of individuals, for authentication purposes. Physiological traits as: fingerprints, voice, hand-geometry, face, iris, ratina, palm-print …etc., and behavioral traits as: gait, signature, keystroke dynamics, and voice [1].

Keystroke dynamics is the process of authenticating individuals based on their typing style. It is not what you type, but how you type [2, 3]. Recently, keystroke dynamics biometric systems have become the alternative of username/password scheme, which has many drawbacks: passwords may be forgotten, attacked, or shared, so the system will be in danger. A user's typing pattern may be unique because of similar neuro-physiological factors that make written signatures unique.

Unlike other biometric systems that usually require additional hardware and thus are expensive to be implemented, biometrics based on keystroke dynamics are almost free i.e. the only hardware required is the keyboard [4, 5].

The problem of keystroke dynamics is that, it is a behavioral biometric; so there are large intra-class variations in person's typing patterns due to changes in emotional state, position of the user with respect to the keyboard, and type of keyboard used. The collected samples of persons need to be updated periodically [5].

The organization of this paper is as follows. Section 2 presents the types of features that can be extracted and also a brief of the previous work in keystroke dynamics. Section 3 presents the proposed system. Section 4 presents the experimental results, and Section 5 concludes the work.

2. Previous Work

The features extracted from keystroke dynamics pattern in most of researches are timing features. Fig. (1) shows the extracted timing features: [6]

1. Key Hold (KD): time between key pressed and key released.
2. Down-Down Key Latency (DDKL): time between two successive presses.
3. Up-Up Key Latency (UUKL): between two successive releases.
4. Up-Down Key Latency (UDKL): time between the current key release and the next key press.
5. Down-Up Key Latency (DUKL): time between the current key press and the next key release.

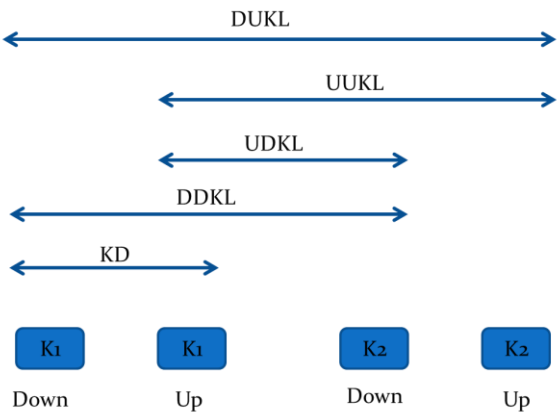

Fig. 1 Extracted features from keystroke timing patterns.

Gaines et al. (1980) were the first researchers showed that keystroke dynamics can be used for authentication [3, 2, 6]; they used long text (900-1200 words) and made their experiments using samples from seven users only.

Hosseinzadeh and Krishnan [6] used three keystroke features to authenticate users: KD, DDKL, and UUKL. UUKL is novel feature proposed during this work. These features were analyzed and modeled using GMM (Gaussian Mixture Modeling). The combination of the KD and UUKL features provided the best performance that led to an equal error rate (EER) of 4.4% based on a database of 41 users, each types 30 times.

Rybnik et al. [7] proposed a new approach to authenticate users using short fixed text. The extracted features were KD and UDKL. Classification of samples is based on k-nearest neighbor algorithm; the best accuracy obtained is 90.38% for 21 users.

Killourhy and Maxion [8] use a 14 keystroke dynamics detectors to authenticate users, 11 detectors was proposed by previous researchers, and 3 classic pattern recognition detectors (Euclidean, Manhattan, and Mahalanobis distance measures). Their data were collected from 51 individuals, each typed the same password 400 times along 8 sessions (50 times/session), 200 samples are used for training and the other 200 samples are used for test, the features extracted from each sample are: DDKL, UDKL, and KD. Scaled Manhattan provided the best results, and reduces EER to 9.6%.

Romain Giot et al. [9, 10] proposed a new method based on SVM (Support Vector Machine) learning. Data were collected by allowing each user to type a fixed password “greyclaboratory”. The features extracted from each sample were: UUKL, DDKL, UDKL, DUKL, and the total typing time. They use a population of 100 persons; each produced only 5 captures for the enrollment step. SVM with intelligent adaptation mechanism and the individual threshold produced the best results, which reduces EER to 6.95%.

Pin Shen Teh et al. [11] proposed a new system that uses two measures to calculate the similarity score between the two given samples: Direction Similarity Measure (DSM) [12] and Gaussian probability Density function (GPD). They evaluated their system over 100 persons; each typed 10 times their usernames, passwords, and a fixed phrase "the brown fox". The results were obtained by applying a two layer fusion approach on both GPD scores and DSN scores. The extracted features were KD, DDKL, UUKL, and UDKL. KD and UDKL yields the best results with all used fusion rules, using them with "And Voting Fusion Rule" produced EER near 1.4%.

Deian Stefan et al. [13] used keystroke dynamics for authentication and detecting imposters, they showed its robustness against forgery attacks. They presented a framework called TUBA for monitoring a user’s typing patterns. They used the total typing time, KD, DDKL, UUKL, UDKL, and DUKL features to authenticate users. Support vector machine is used for classification. They evaluated their system using 20 users' keystrokes. The best result they obtained is 4.2% for average false positive rate.

Yu zhong et al. [14] evaluated a keystroke biometrics algorithms based on a new distance metric on the keystroke dynamics dataset created in (CMU Dataset) [8]. The new distance metric combined both Mahalanobis distance and Manhattan distance. Using the Nearest Neighbor classifier with the new distance metric achieved an average EER of 8.7%.

In previous work there are some constraints. In [7, 13], users authenticated through long phrases which is not real in case of passwords. Where in [6, 8, 14], users are asked to type several times in different sessions which could be refused by some users as it is time consuming and needs additional efforts. Also in [8, 9, 11], all users typed the same word, but in the real login system each user types his own password.

The main idea of our work is to allow users to access different systems by typing their own usernames and passwords as usual. Then, the users' typing styles features are extracted from their passwords, so there is no additional text required for authentication.

3. Proposed System

The block diagram of the proposed system is given in Fig. (2).

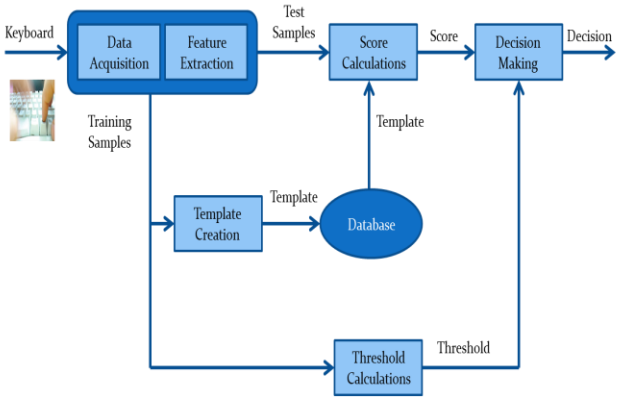

Fig. 2: Block diagram of the proposed keystroke dynamics system.

A new keystroke dynamics system was proposed. The following steps were used to implement the proposed system as follows:

1. The individual types their username and previously trained eight character password several times through separate sessions.
2. Features are extracted when individuals press and release keys.
3. User's Template and threshold was calculated through the extracted features.
4. Calculate the distance between template and the test samples to get the user's score
5. Finally, the user's score is compared against its threshold to make the decision.

3.1 Data acquisition and Feature Extraction

New dataset is created to evaluate the proposed system. A software application is implemented to acquire samples from individuals and extract their features, simply, user types his username and password, and individuals can easily run this program on their own PCs or Laptops. Users are allowed to enter their own eight characters passwords containing only uppercase characters, lowercase characters, and numbers, where special characters are not allowed. Time stamps of each key press (Down) and release (Up) are stored in a logout file and used to calculate KD, DDKL, UUKL, UDKL, and DUKL.

3.2 Keystroke Dynamics Algorithms

Four distance based algorithms were used to evaluate the system: Manhattan, Manhattan with standard deviation, Euclidean, and Mahanabolis.

3.2.1 Manhattan Distance

The score is calculated as in Eq. (1) which represents Manhattan distance [15]:

$$M = \sum_{i}^{n} (x_i - y_i) \qquad (1)$$

Where $x = (x_1, x_2,...,x_n)$ represents test vector and $y = (y_1,y_2,...,y_n)$ represents the mean vector of the training samples

3.2.2 Manhattan with Standard Deviation Distance (std)

The standard deviation of each feature is calculated as well [8]. Eq. (2) will be in the form:

$$Ms = \sum_{i}^{n} (x_i - y_i)/\alpha_i \qquad (2)$$

3.2.3 Euclidean Distance

The score is calculated as the squared Euclidean distance between the test vector and the mean vector as in the following Eq. (3):

$$E = \sqrt[2]{\sum_{i}^{n} (x_i - y_i)^2} \qquad (3)$$

3.2.4 Mahanabolis Distance

The standard deviation of each feature is calculated, where the Mahanabolis distance is presented by Eq. (4) [15]:

$$Mh = \sqrt[2]{\sum_{i}^{n} ((x_i - y_i)/\alpha_i)^2} \quad (4)$$

3.3 Thresholds calculation

User's threshold is more recommended than global threshold in user authentication, it is proved to produce better performance. Leave-One-Out-Method (LOOM) [6] is used to calculate thresholds for individuals through the following steps:

1. Dividing the training space of (n) samples to one sample used as test sample, and (n-1) samples used to create the training sample.
2. Applying a distance measure (Euclidean for example) to calculate the distance between the selected test sample and the mean vector of the (n-1) training samples.
3. This process is repeated (n) times and produce (n) different thresholds for each feature vector.
4. The average of these (n) thresholds is calculated to produce the individual threshold.
5. These steps are repeated to calculate the individual thresholds for the other three distance measures.

3.4 Scores calculation

A match score is known as a genuine score if it results from matching of two samples of the same user, otherwise it is known as an impostor score if it involves comparing biometric samples of two different users [1].

In the proposed system two sets of genuine samples and one set of imposters are used to authenticate users, the scores are calculated through the following steps:

1. The individual types its username and eight character password.
2. Features are extracted from the typed sample to produce KD, DDKL, UUKL, UDKL, and DUKL feature vectors.
3. The new feature vectors are compared with those of the individual stored template using one the four distance measures.
4. The obtained scores for each feature are reported.
5. These steps are repeated to calculate the individual scores within the other three distance measures

3.5 Decision making

The proposed system is evaluated using: False Rejection Rate (FRR), which is the refused fraction of genuine individuals, and False Acceptance Rate (FAR), which is the accepted fraction of impostor individuals. Eq. (5) and (6) shows FRR and FAR respectively.

$$FRR = \frac{Number\ of\ refused\ genuines}{Total\ number\ of\ genuines} \quad (5)$$

$$FAR = \frac{Number\ of\ accepted\ imposters}{Total\ number\ of\ imposters} \quad (6)$$

The biometric system performance could be measured using Equal Error Rate (EER) which refers to the point on the ROC (Receiver Operating Characteristic) curve where the FAR and the FRR are equal [1, 3].

4. Experimental Results

Two datasets are used to evaluate the system; the first is CMU created by Kevin Killourhy [8], the second is created through this work. CMU dataset contains 51 individuals; each one typed the same password 400 times along 8 sessions (50 times/session). In this work, eight samples are used for each individual (one sample/session), six samples are used to create the training space, two samples are used to evaluate the system based on FRR, and two imposter samples are used to evaluate the system based on FAR. The extracted features for this dataset are: KD, DDKL, and UDKL and the previously stated matching algorithms were used to calculate scores and thresholds.

The most common approach for decision level fusion is majority voting (MV) [1]. If there are n features, the input sample is assigned an identity when at least k of the features agree on that identity, where k = (n/2) + 1 if n is even and k = (n+1) / 2 if n is odd.

Firstly, users are authenticated based on each feature separately; Table (1) shows the results of each feature, UDKL produces the best results is 8.8% for EER using Manhattan with standard deviation.

Table 1: EER of each feature for the CMU dataset

	Euclidean	Manhattan	Mahanabolis	Manhattan with std
KD	20.3	18.0	19.7	18.0
DDKL	17.2	14.7	15.3	12.3
UDKL	16.0	13.7	19.0	8.8

Then, individuals are authenticated based on different features combinations by comparing the average scores of combined features with their average thresholds. Finally, individuals are authenticated based on majority voting (MV), the best results is 7.0 % for EER using Manhattan with standard deviation and MV. Table (2) shows the best results for the features combinations.

Table 2: EER for CMU dataset based on two features and all features combinations

	Euclidean	Manhattan	Mahanabolis	Manhattan with std
KD & DDKL	16.0	14.5	10.3	8.1
KD & UDKL	14.9	13.0	10.3	8.9
DDKL & UDKL	16.7	14.3	16.7	11.3
KD & DDKL & UDKL	15.5	14.8	11.1	8.7
MV	15.5	12.1	12.5	7.0

EER is used to compare the results of the proposed system on CMU dataset with two existing systems: Kevin S. Killourhy (2009) [8] and Yu Zong (2012) [14], see table (3).

Table 3: shows the comparison between the proposed and other two systems based on EER

System	EER
Kevin S. Killourhy (2009) [8]	9.6
Yu Zong (2012) [14]	8.4
The proposed system	7.0

Fig. (3) shows a comparison among four distance measures using different features combinations based on EER.

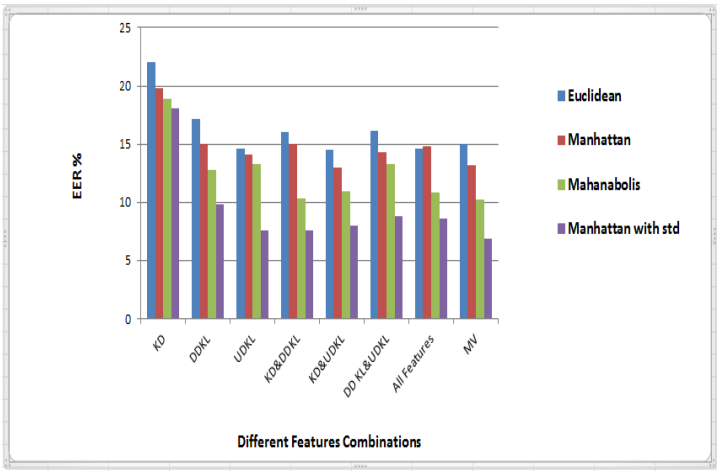

Fig. 3: a comparison among four distance measures using different features combinations based on EER for CMU dataset.

Fig. (4) shows the way to calculate the EER in the case of using Manhattan with standard deviation and MV on CMU dataset.

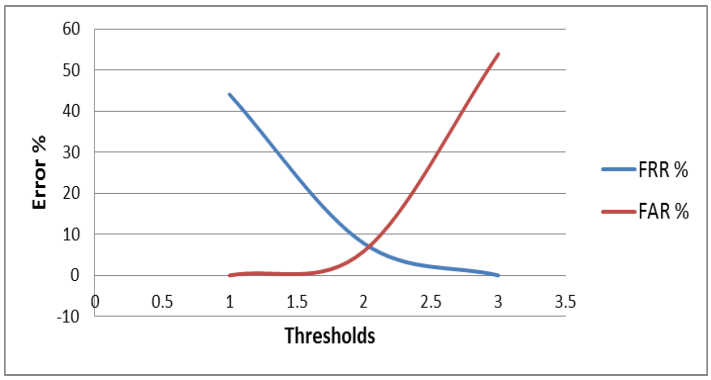

Fig. 4: ROC curve represents FRR and FAR for different thresholds using Manhattan with standard deviation and MV, the point of intersection represent the EER

New dataset of 62 individuals was created; each individual types his own eight character

password. Six samples are used to create the user template which stored in the database, and two genuine samples are used to evaluate the system, on the other hand two other imposter samples are used to evaluate the ability of the system to discover the forgery attacks. The features extracted are: KD, DDKL, UUKL, UDKL, and DUKL. The matching distance based algorithms are used to calculate scores and thresholds are: Euclidean, Manhattan, Manhattan with standard deviation, and Mahanabolis.

Firstly, users are authenticated based on each feature separately; Table (4) shows the results of each feature, UDKL produces the best results is 7.5% for EER using Manhattan with standard deviation.

Table 4: shows EER of each feature for the new dataset

	Euclidean	Manhattan	Mahanabolis	Manhattan with std
KD	16.5	15.0	15.0	14.7
DDKL	11.3	9.4	11.6	9.3
UUKL	12.0	9.5	11.3	8.3
UDKL	13.8	9.1	10.1	7.5
DUKL	10.1	9.0	8.2	7.8

Then, individuals are authenticated based on combinations of each two, three, and four features respectively, by comparing the average scores of combined features with their average thresholds. Table (5) shows the best results for the features combinations.

Finally, individuals are authenticated based on a combination of all features by comparing the average score all features with their average threshold, or by making vote (MV), the best results is 4.9% for EER using Manhattan with standard deviation and MV. Table (6) shows these results.

Table 5: shows EER for two, three, and four features combinations

	Euclidean	Manhattan	Mahanabolis	Manhattan with std
KD&UDKL	9.9	9.1	7.0	6.5
UDKL& DUKL	11.4	9.5	7.3	6.5
KD&DDKL &UDKL	9.3	8.0	6.7	5.7
KD&UUKL &UDKL	9.5	7.1	7.3	5.8
KD&DDKL& UUKL &UDKL	9.9	6.8	7.5	6.2
KD&DDKL &UUKL &DUKL	10.3	8.0	7.9	6.7

Table 6: shows EER of the new dataset based on MV and on all features combination

	Euclidean	Manhattan	Mahanabolis	Manhattan with std
MV	10.6	7.8	7.0	4.9
All Features	10.1	8.1	7.9	6.6

Fig. (5) shows a comparison among four distance measures using different features combinations based on EER for new dataset.

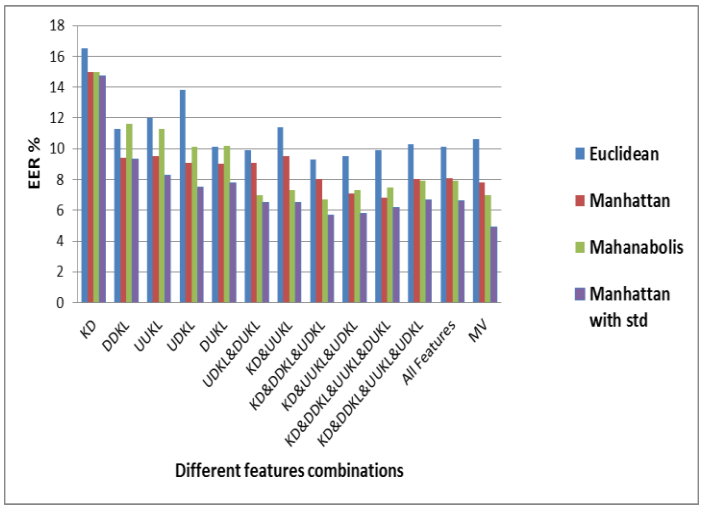

Fig. 5: a comparison among four distance measures using different features combinations based on EER for new dataset.

5. Conclusion

Keystroke Dynamics can be used as a digital signature to authenticate persons; it has no additional efforts from users to login systems, as users already type their passwords to be allowed to access. In some of the previous work users were authenticated by allowing them to type the same word or long phrases, while in the proposed system users are authenticated through their own eight character passwords; then timing features (KD, DDKL, UUKL, UDKL, and DUKL) are extracted for the typed characters. The system is evaluated based on each feature separately, and on different combinations of features.

The proposed system is evaluated using four distance measures: Manhattan, Manhattan with standard deviation, Euclidean, and Mahanabolis, for the matching process, taking the standard deviation into consideration increases the performance. Manhattan with standard deviation produces the best results as it takes into account the standard deviation of the training samples, so using it with the user based threshold calculated using LOOM could improve the system performance as it could solve the problem of large intra-class variations in user's samples. Two data sets were used in this work, the CMU data set and the other one is created via 51 individuals. Results obtained show a better performance while compared with the others.

6. References

[1] A. K. Jain, P. Flynn and A. A. Ross, Handbook of Biometrics, Springer, 2008.

[2] F. Monrose and A. D. rubin, "Keystroke dynamics as a biometric for authenticatio," Science Publishers B. V. Amsterdam, The Netherlands, Feb, Elsevier , 2000 .

[3] M. Karnan and K. M. Akila, "Biometric personal authentication using keystroke dynamics: A review," Applied Soft Computing, Elsevier,2011.

[4] D. Jamil and M. N. A. Khan, "Keystroke Pattern RecognitionPreventing Online Fraud," International Journal of Engineering Science and Technology (IJEST) vol. 3, March, 2011.

[5] E. Lau, X. Liu, C. Xiao and a. X. Yu, "Enhanced User Authentication Through Keystroke Biometrics," Computer and Network Security Final Project Report, Massachusetts Institute of Technology, December 9, 2004.

[6] D. Hosseinzadeh and S. Krishnan, "Gaussian Mixture Modeling of Keystroke Patterns for Biometric Applications," IEEE Systems, Man, and Cybernetics Society , November, 2008, Vol. 38, Toronto.

[7] M. Rybnik, P. Panasiuk and K. Saeed, "User Authentication with Keystroke Dynamics using Fixed Text," International Conference on Biometrics and Kansei Engineering, 2009.

[8] K. S. Killourhy and R. A. Maxion, "Comparing Anomaly-Detection Algorithms for Keystroke Dynamics," IEEE, Dependable Systems and Networks - DSN, 2009.

[9] R. Giot, M. El-Abed and C. Rosenberger, "Keystroke Dynamics With Low Constraints SVM Based Passphrase Enrollment," IEEE International Conference on Biometrics: Theory, Applications and Systems, 2009.

[10] R. Giot, M. El-Abed, B. Hemery and C. Rosenberger, "Unconstrained Keystroke Dynamics Authentication with Shared Secret," Computer & Security, ScienceDirect, 2011.

[11] P. S. Teh, A. B. J. Teoh, T. S. Ong and C. Tee, "Keystroke dynamics in password authentication enhancement," Expert Systems with Applications 37 (2010) 8618–8627, Elsevier,2010.

[12] P. S. Teh, A. B. J. Teoh, T. S. Ong and H. F. Neo, "Statistical Fusion Approach on Keystroke Dynamics," Third International IEEE Conference on Signal-Image Technologies and Internet-Based System , 2008.

[13] D. Stefan and D. (. Yao, "Keystroke-Dynamics Authentication Against Synthetic Forgeries," Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2010 6th International Conference on.

[14] y. Zhong, Y. Deng and A. K. Jain, "Keystroke Dynamics for User Authentication," Computer Vision and Pattern Recognition Workshop, IEEE

computer society conference, June, 2012.

[15] S. Hocquet, J. Ramel, and H. Cardot."User Classification for Keystroke Dynamics," Seoul, Korea, Advances in Biometrics, International Conference, ICB, 2007.