Keystroke Dynamics Recognition based on Personal Data: A Comparative Experimental Evaluation Implementing Reproducible Research

Aythami Morales[1], Mario Falanga[1,2], Julian Fierrez[1], Carlo Sansone[2], Javier Ortega-Garcia[1]

[1]ATVS, EPS, Universidad Autonoma de Madrid, C\ Francisco Tomás y Valiente, 11, 28049 Madrid, Spain
[2]DIETI, University of Naples Federico II, Via Claudio 21, 80125 Naples, Italy

{aythami.morales, julian.fierrez, javier.ortega}@uam.es, mario.falanga91@gmail.com carlo.sansone@unina.it

Abstract

This work proposes a new benchmark for keystroke dynamics recognition on the basis of fully reproducible research. Instead of traditional authentication approaches based on complex passwords, we propose a novel keystroke recognition based on typing patterns from personal data. We present a new database made up with the keystroke patterns of 63 users and 7560 samples. The proposed approach eliminates the necessity to memorize complex passwords (something that we know) by replacing them by personal data (something that we are). The results encourage to further explore this new application scenario and the availability of data and source code represent a new valuable resource for the research community.

1. Introduction

Cyber security is a critical concern for governments, companies and end-users. The secure access has become a must in a global society connected by internet and Cyber Security is defined as the body of technologies, services and practices designed to protect computers, networks and data from damage, attack or unauthorized access [1]. Passwords-based authentication is one of the most popular secure access approaches. The security of these systems depends of the password strength and the security policies adopted. A typical security policy includes recommendations as: i) to avoid simple passwords; ii) to change passwords regularly; iii) to use different passwords for different accounts, systems and applications, and iv) to store passwords securely (try to memorize your passwords). The number of users who comply with these security policies is low as they decrease the usability of the systems. As an example, "=/z@l1N]" is a password generated by the automatic password generator of the BTAS2015 Conference Management Toolkit. The requests to reset passwords because they have been forgotten are common.

In this context, keystroke dynamics authentication systems have attracted the interest of both researchers and industry [2][3]. Keystroke dynamics are proposed to improve the security of traditional authentication services based on passwords or PIN numbers. Biometric recognition is commonly related to "something that users are" instead of "something that users have" such as passwords. In the case of keystroke dynamics, the typical approaches based on fixed password authentication combine complex passwords and our keystroke dynamics biometrics. The password acts as a primary security level and the user access is not allowed until the correct password is inserted. The role of the biometric system is a secondary security level which try to detect intruders who are spoofing the identity of the legitimate user.

Why not using the keystroke dynamics authentication as the primary and only security level? Is it possible a reliable recognition by replacing the way the people type complex passwords (something that they know) by the way they type their names (something that they are)?

This work explores keystroke dynamics authentication based on personal data as opposed to complex passwords. The idea underlying this work is that while we can forget a complex password, we will never forget our family or given name. This authentication approach try to improve the security of the access as well as its usability. The main advantages of keystroke recognition based on personal data are: i) the learning curve [4] is minimized because personal data is information that users are accustomed to type; ii) the usability is improved by eliminating the necessity of complex passwords which comply with the security policies but are difficult to memorize. On the other hand, the main disadvantages of this approach are: i) small amount of information available to recognize the users (short words) and ii) the learning curve of the impostors is also minimized when the information is based on common names or simple words instead of complex passwords.

The main contributions of this work can be summarized as follow: i) a new dataset with keystroke dynamics of 63 users acquired on the basis of a real application scenario; ii) new insights in keystroke authentication based on personal data and iii) a fully reproducible and public available benchmark including the database and the source code necessary to reproduce all the results reported in this work[1].

[1] http://atvs.ii.uam.es/keystroke_db.html

Table 1: Survey of publicly available databases for keystroke dynamics recognition.

Database	#users	#samples	#sessions	Properties
CMU [5]	51	400	8	Same password for all users: ".tie5Roanl"
MIMOS [6]	100	10	-*	Same password for all users: "try4-mbs"
Clarkson [7]	39	20	1	Same password for all users: 3 different passwords "yesnomaybe", "bahaNe312!" and "ballzonecart"
GREYC [8]	133			Same password for all users: "greyc laboratory"
BeiHang [9]	117	4-16	1	Different password per user
Present work	**63**	**120**	**2**	**Different password per user**

*Different depending the user

1.1. Publicly Available Databases

The availability of public databases and benchmarks is crucial for the development of biometric recognition technologies. Keystroke recognition is strongly user and application dependent. As a behavioral biometric, the intraclass variability (samples from the same user) and interclass variability (samples from different users) are affected by several factors including those inherent to the user (e.g. we never type two times in the exact same way) and others depending of the application scenario (e.g. length of the password, fixed or non-fixed password). Table 1 shows a list of the most popular keystroke dynamics public databases. Most of these databases are based on a single hypothetical password or passphrase for all subjects. Some examples are: ".tie5Roanl" for the CMU database, "greyc laboratory" for the GREYC database or "try4-mbs" for the MIMOS database.

While real password authentication services are based on personal passwords (chosen by the user or automatically generated by a platform), the number of databases and public benchmarks based on this real application scenario is scarce. The BeiHang keystroke database is an example of database acquired under real application assumptions. The database includes keystroke samples from 117 subjects in two different environments. The acquisition of the samples was made without any supervision and the number of samples per subject ranges from 2 to 16 which make difficult to propose a balanced experimental protocol.

The database made publicly available together with the present work complements the existing databases by: i) each user has his own password (in the form of his real personal data) and ii) each user provides 5 different keystroke sequences (given name, family name,…) with different properties (e.g. length). Although the keystroke dynamics community offers public databases, the availability of source code to fully reproduce the experimental framework is rare. The database presented here together with the source code is a new valuable research resource for the biometric community. In this regard, i) we propose a novel recognition approach based on personal information of the users and ii) the source code to reproduce all the experiments is publicly available

2. Acquisition setup and database

The design of the acquisition platform is inspired by the use of web-based application forms (e.g. the USA Electronic System for Travel Authorization). The idea was to provide a familiar environment which allows natural user behavior. The acquisition platform includes 5 forms to provide the following personal data: given name, family name, email, nationality and national ID number, see Fig. 1.

The database comprises 63 users with 12 genuine access and 12 impostor access for each user for a total number of samples equal to 7560 (63 users × 24 accesses × 5 fields). There are people from two different nationalities with 60% of males and 40% females. The acquisition was divided into two sessions according a semi-supervised protocol:

- **First session:** the users were asked to introduce their personal data in the platform. This process was repeated 6 times.
- **Second session:** after at least 24 hours, the users were asked to introduce once again their personal data in the platform. The process was repeated 6 times. In addition, in this second session, each user acted as an imposter trying to spoof the system with the personal data of other users. The personal data of three users was showed to each of the imposters and they introduced them four times for a total number of impostor accesses of twelve per user.

The information provided by the users includes sensitive data and therefore it has been post-processed to remove all the personal information (the characters pressed) maintaining in this way the privacy of the users enrolled in the database. The keystroke dynamic patterns were recorded using a key-logger (programmed in Java). The key-logger detects two different types of events: press and release from the QWERTY keyboard used. The timestamps for each of the detected events were recorded in milliseconds.

3. Methodology

The benchmark proposed in this work includes the database, the features obtained from the raw data (timestamps), an experimental protocol, and baseline

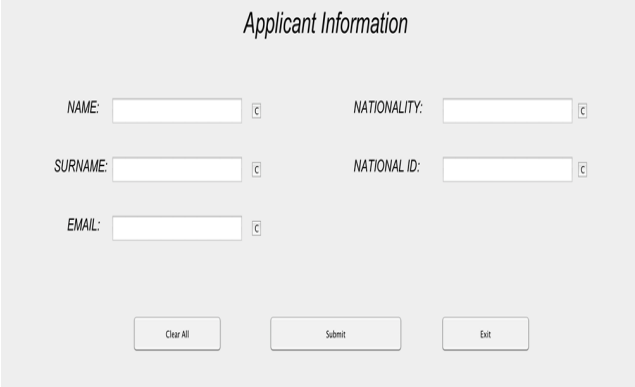

Figure 1: Web-form for the acquisition of the keystroke patterns.

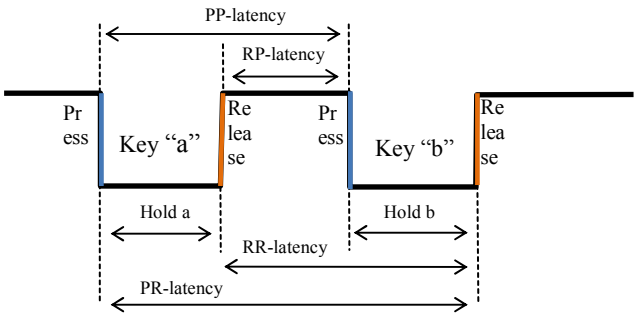

Figure 2: Keystroke dynamics features from a digraphs sequence.

authentication performance results obtained with four popular keystroke dynamics classifiers.

3.1. Features

The keystroke dynamics data extracted from a sequence of N keys consist of a vector $\mathbf{t}$ which contains the time stamp of every key-press (t^p) and key-release (t^r) event. These time stamps can be used to model the way a subject types but it is necessary to process the data to normalize the features with respect to a reference. This normalization on time can be achieved considering intervals between consecutive key events instead of absolute time stamps, see Fig. 2. The most popular features to characterize the keystroke patterns are:

- **Hold Time:** it is the difference between the time of pressure and release of the *i*th key.
- **Release-Press latency (RP-latency):** is the difference between the time of pressure of the (*i*+*1*)th key and the release of the *i*th key.
- **Press-Press latency (PP-latency):** is the difference between the time of pressure of (*i*+1)th key and the pressure of the *i*th key.
- **Release-Release latency (RR-latency):** is the difference between the time of release of (*i*+1)th key and the release of the *i*th key.
- **Press-Release latency (PR-latency):** is the difference between the time of release of the (*i*+1)th key and the pressure of the *i*th key.

Both the raw data and the feature vectors are publicly available (see *Section 5*).

3.2. Classifiers

The benchmark proposed in this work includes baseline authentication performance results obtained with four keystroke dynamics classification algorithms. The algorithms could be used as baseline to further research with this dataset.

Assume $\mathbf{f} = [f_1, f_2, \dots, f_M]$ as the feature vector (with M features) of a given test sample and $\mathbf{g}^k = [g_1^k, g_2^k, \dots, g_M^k]$ $k \in 1, \dots T$ as an enrollment set with T samples. The four keystroke dynamics classifiers included in the benchmark are:

- **Scaled Manhattan Distance** [5]: based on the one proposed by Araujo et al. [10]. The distance between a feature vector $\mathbf{f}$ of the test sample and the enrollment set $\{\mathbf{g}^k\}_{k=1}^T$ is calculated as:

$$d_1 = \sum_{i=1}^{M} |f_i - \bar{g}_i| / a_i \qquad (1)$$

where $\bar{\mathbf{g}}$ is the average of the enrollment set $\bar{\mathbf{g}} = \frac{1}{T}\sum_{k=1}^{T}\mathbf{g}^k$ and $\mathbf{a} = [a_1, a_2, \dots, a_M]$ is the average absolute deviation of the enrollment features, $a_i = \frac{1}{T}\sum_{k=1}^{T}|g_i^k - \bar{g}_i|$ $\forall\, i \in 1, \dots M$.

- **Combined Manhattan-Mahalanobis distance** [11]: the test samples $\mathbf{f}$ and the enrollment set $\{\mathbf{g}^k\}_{k=1}^T$ are first normalized as $\hat{\mathbf{f}} = \mathbf{S}^{-1/2}\mathbf{f}^{\mathrm{T}}$ and $\hat{\mathbf{g}}^k = \mathbf{S}^{-1/2}(\mathbf{g}^k)^{\mathrm{T}}$, where $\mathbf{S}$ is the covariance matrix of the enrollment set and $(\cdot)^{\mathrm{T}}$ is the transpose. The distance d_2 is then calculated as the Manhattan distance between the normalized test sample and the normalized enrollment set:

$$d_2^k = \sum_{i=1}^{M} |\hat{f}_i - \hat{g}_i^k| \quad \forall\, k \in 1, \dots T \qquad (2)$$

The final distance d_2 is obtained as the minimum in k.

- **Mahalanobis + Nearest Neighbor** [12]: the distance between a test sample $\mathbf{f}$ and each of the enrollment samples $\mathbf{g}^k$ is calculated as:

$$d_3^k = (\mathbf{f} - \mathbf{g}^k)\mathbf{S}^{-1}(\mathbf{f} - \mathbf{g}^k)^{\mathrm{T}} \;\forall\, k \in 1, \dots T \qquad (3)$$

where the covariance matrix of the gallery set, $\mathbf{S}$, is introduced to increase the impact of those features with a smaller variance. The final distance d_3 is obtained as the minimum in k.

Table 2: Performance of the four classifiers using the CMU benchmark dataset. The table also shows the performance of other keystroke dynamics algorithms [5]

Classifier	Average EER (std dev)
d_1	9.62 (0.0694)
d_2	9.96 (0.0642)
d_3	8.40 (0.0560)
d_4	8.84 (0.0627)
z-score [5]	10.22 (0.0767)
SVM [5]	10.25 (0.0650)
Mahalanobis [5]	11.01 (0.0645)
Mahalanobis norm. [5]	11.01 (0.0645)

- **Modified Scaled Manhattan distance** [10]**:** the distance between a feature vector of the test sample **f** and the enrollment set $\{\mathbf{g}^k\}_{k=1}^T$ is calculated as:

$$d_4 = \sum_{i=1}^{M} |f_i - \bar{g}_i| / \sigma_i' \qquad (4)$$

where $\boldsymbol{\sigma}' = [\sigma_1', \sigma_2', \ldots, \sigma_M']$ is a modification of the standard deviation:

$$\sigma_i' = \begin{cases} \frac{0.2}{M}\sum_{j=1}^{M}\sigma_j & if\ \ \sigma_i < \frac{0.2}{M}\sum_{j=1}^{M}\sigma_j \\ \sigma_i & rest \end{cases} \qquad (5)$$

and σ_i is the standard deviation of features computed over $\{\mathbf{g}^k\}_{k=1}^T$. This simple modification tries to mitigate the effects of samples with very low variance during the normalization (low variance means high weight).

The matching score, s_i, between the test sample and the enrollment set is defined as the inverse of each of these distances: $s_i = 1/d_i$.

These algorithms were selected among some of the most competitive algorithms tested on the CMU benchmark dataset [5], see Table 2. As we can see, the performance achieved by the four distances is similar with average EER of all subjects ranked between 8.40% and 9.96%.

3.3. Experimental Protocol

The main aim of the experimental protocol is to establish a baseline to further experiment with the proposed database. The experiments included in this section are divided into different scenarios or case studies. The protocol proposed to evaluate the database is as follows:

- The six genuine samples of the first session are used as enrollment set and the remaining six genuine samples of the second session are used as genuine test set (to obtain the False Rejection Rate).
- The twelve impostor samples available for each user comprise the impostor test set (to obtain the False Acceptance Rate).
- The performance is provided as the mean Equal Error Rate and its standard deviation for all subjects in the database.

4. Experiments

The experiments are aimed to analyze the discriminative ability of the keystroke patterns obtained from the personal data of the users. The experiments include an analysis at multiple levels: feature, data, classification algorithms and number of enrollment samples.

4.1. Feature Comparison

The first study was carried out to ascertain the performance of the different features and the four classifiers. We combined all data (given name, family name, email, nationality and ID number) at feature level by concatenation of feature vectors before the classification. The results (see Table 3) show a clear difference between the Hold Time and the latencies (differences range from 3% to 5%).

Despite the poor performance of the Hold Time as an individual feature, its combination with other features suggests certain complementarity. The last row of Table 3 shows the performance of the classifiers when RP-Latency (also called flight time) and Hold Time are combined at feature level. The results show a consistent performance improvement for most of the classifiers which suggests such a complementarity. Note that this latency is the only one that does not includes the hold time in the timing calculation.

4.2. Data Field Comparison

This experiment analyzes the performance of each data field as an independent sequence of keystroke dynamics. The idea is to assess the discriminative ability of each data field and their complementarity when they are combined. Table 4 shows the individual performance of each data field. The results suggest clear differences between data field (e.g. name and email). The poor performance of the name is closely related to the short length of this input. It is well known that the performance of keystroke dynamics is related to the amount of information [4] and the differences between performances of given and family names complies with this fact (70% of the users have two family names).

Table 4 shows the superior performance of the ID number and email. This superior performance could be caused by the mixed nature of the email address of users (most of the emails are combination of given name, family name and few extra characters). The email can be seen as a mixture between personal data and passwords. The ID is a sequence of numbers which is difficult to memorize for impostors. It is remarkable that 25% of the users show

Table 3: Feature performance in terms of Average EER (std dv). Combination at feature level of all personal data. Last row includes the performance of the combination of RP and HT features.

	d_1	d_2	d_3	d_4
HT	0.110 (0.12)	0.140 (0.13)	0.159 (0.13)	0.086 (0.12)
PR	0.063 (0.10)	0.130 (0.12)	0.121 (0.13)	0.048 (0.08)
RR	0.074 (0.11)	0.101 (0.11)	0. 104 (0.14)	0.054 (0.09)
PP	0.069 (0.10)	0.116 (0.11)	0.110 (0.11)	0.045 (0.08)
RP	0.073 (0.10)	0.097 (0.11)	0.106(0.12)	0.042 (0.08)
RP,HT	0.044 (0.07)	0.097 (0.11)	0.089 (0.11)	0.023 (0.06)

Table 4: Personal Data performance in terms of Average EER (std dv) using the feature combination (HT+RP); G=Given Name; F= Family Name; E=Email; N=Nationality; ID=ID number.

	d_1	d_2	d_3	d_4
G	0.167 (0.14)	0.224 (0.16)	0.226 (0.16)	0.161 (0.14)
F	0.130 (0.14)	0.241 (0.18)	0.240 (0.16)	0.124 (0.12)
E	0.114 (0.14)	0.295 (0.17)	0.246 (0.15)	0.089 (0.12)
N	0.140 (0.14)	0.201 (0.16)	0.201 (0.16)	0.142 (0.15)
ID	0.115 (0.13)	0.203 (0.17)	0.175 (0.16)	0.098 (0.11)

Table 5: Performance of personal data combination (BX indicates the combination of best X fields according to Table 4) in terms of Average EER (std dv). Combination at feature level using the feature combination (HT+RP).

	d_1	d_2	d_3	d_4
B2	0.074 (0.10)	0.218 (0.15)	0.156 (0.15)	0.044 (0.08)
B3	0.055 (0.09)	0.156 (0.13)	0.161 (0.15)	0.031 (0.07)
B4	0.050 (0.08)	0.122 (0.12)	0.120 (0.14)	0.026 (0.07)
B5	0.044 (0.07)	0.097 (0.11)	0.089 (0.11)	0.023 (0.06)

Table 6: Performance according the number of training samples in terms of Average EER (std dv) using the feature combination (HT+RP) and all fields available (B5 showed in Table 5). Bold font highlights the low degradation of d_4.

	d_1	d_2	d_3	d_4
6	0.044 (0.07)	0.097 (0.11)	0.089 (0.11)	**0.023 (0.06)**
5	0.055 (0.10)	0.137 (0.13)	0.128 (0.13)	**0.023 (0.08)**
4	0.073 (0.10)	0.166 (0.14)	0.162 (0.14)	**0.039 (0.09)**
3	0.088 (0.11)	0.216 (0.13)	0.195 (0.16)	**0.038 (0.09)**
2	0.174 (0.16)	0.295 (0.16)	0.260 (0.16)	**0.040 (0.08)**

EER=0% using only the name and d_4 distance as keystroke sequence and classifier respectively.

Table 5 shows the performance of the four classifiers when data is combined at feature level (concatenating feature vectors). The combination is made according to the individual performance of the data (using d_4 as reference). As an example, B2 represents the combination of the best two fields (email and ID number) and B3 represents the combination of the best three fields (email, ID number and family name). In most of the cases, the combination improves the performance. This is an expected results motivated by the complementarity of the data. However, there some interesting results when we compare the improvements of the different classifiers. The Mahalanobis + Nearest Neighbor classifier (d_3) shows a scarce improvement when the data is combined while the other classifiers show improvements ranging between 30% and 50%.

The best performance achieved in this work (EER=2.4%) is obtained combining all the 5 fields (name, email, ID number, family name and nationality), Hold Time and RP-Latency with the classifier based on the Modified Scaled Manhattan distance d_4. It is remarkable the performance achieved (EER=3.1%) using only three data (ID number, email and family name).

4.3. Performance vs. Training Data

The last experiment shows the degradation of the performance when the number of enrollment samples decreases. Table 6 shows the performance of the four classifiers for different sizes of the enrollment set. The results show different degree of degradation depending on the classifier. However, it is surprising the performance achieved by some of the classifiers (d_4) with only two enrollment samples. The results suggest the stability of the keystroke patterns obtained from personal data (no need of large learning curves). We would like to highlight also the difference between the Scaled Manhattan (d_1) and the Modified Scaled Manhattan (d_4). When the number of enrollment samples is low, the normalization techniques (as those employed in d_1, d_2 and d_3) have a large negative impact on the performances and it is recommendable to adopt other solutions as the standard deviation used in d_4.

5. Reproducibility

This work complies with the Open knowledge and Reproducible Research guidelines. The data and code used to generate all the results included in this work are free to

use, reuse and redistribute among the research community.

There are some interesting efforts to open up public research results including datasets, benchmarks, and other tools (e.g. OpenBR http://openbiometrics.org/, OpenAIRE https://www.openaire.eu and FOSTER https://www.fosteropenscience.eu/). Among these platforms, the BEAT – Biometrics Evaluation and Testing platform (https://www.beat-eu.org/platform) offers an experimental framework for reproducible research of biometric technologies. The benchmark proposed in this paper is available in the BEAT platform. The researchers have free access to the code, the database and the results[2]. This platform allows to easily reproduce the experiments and propose further improvement based on the baseline included in this work. In addition, the material to reproduce all the experiments of this work is fully available at a public scientific repository[3].

6. Conclusions

This work proposes new benchmarks for keystroke dynamics recognition developed in the basis of Open Knowledge guidelines. The benchmarks include a new dataset composed by keystroke dynamics obtained from personal data of 63 users and source code including four keystroke dynamics classifiers.

The results show promising discriminative ability of the keystroke patterns obtained from personal data with a best EER of 2.3%. The experiments include an analysis of the performance of the proposed approach in terms of features, classifiers, number of training samples and nature of the data. In comparison with other benchmarks [5][8][13], the dataset presented in the present work includes new characteristics such as the passwords based on different personal data fields from users. From the experimental results we have observed: i) the performance is related to the nature of the data used (e.g. performances achieved for email, family name or ID); ii) the superior performance of the feature combination (HT+RP) and iii) the large difference in terms of performance between keystroke dynamics classifiers which report similar performances in other benchmarks such as CMU [5]. These results highlight the necessity of experimentation in multiple databases and different scenarios to better understand the real performance and advance the challenges encountered in keystroke authentication systems.

Acknowledgment

A.M. is supported by a post-doctoral Juan de la Cierva contract by the Spanish MECD (JCI-2012-12357). M.F. was supported by a scholarship provided by University of Naples Federico II and Compagnia di San Paolo. This work has been partially supported by projects: Bio-Shield (TEC2012-34881) from Spanish MINECO, BEAT (FP7-SEC-284989) from EU, CECABANK and Cátedra UAM Telefónica.

[2] https://www.beat-eu.org/platform/attestations/1839141191/

[3] http://atvs.ii.uam.es/keystroke_db.html

References

[1] M. McDowell, S. Hernan, J. Rafail. Choosing and Protecting Passwords. US CERT. 2013.

[2] A. Peacock, X. Ke, M. Wilkerson. Typing patterns: A key to user identification. IEEE Security and Privacy, 2(5):40–47, 2004.

[3] D. Gunetti and C. Picardi. Keystroke analysis of free text. ACM Transactions on Information and System Security, 8(3):312–347, 2005.

[4] J. Montalvão, E. O. Freire, M. A. Bezerra Jr,, R. Garcia. Contributions to empirical analysis of keystroke dynamics in passwords. Pattern Recognition Letters, 52: 80-86, 2015.

[5] K. S. Killourhy and R. A. Maxion. Comparing Anomaly Detectors for Keystroke Dynamics. In Proc. of the 39th Annual Intl. Conf. on Dependable Systems and Networks, pp. 125-134, Estoril, Portugal, 2009.

[6] C. Loy, W. K. Lai, C. Lim. Keystroke patterns classification using the artmap-fd neural network. In Proc. of Third Intl. Conf. on Intelligent Information Hiding and Multimedia Signal Processing, pp. 61–64, 2007.

[7] E. Vural, J. Huang, D. Hou, S. Schuckers. Shared Research Dataset to Support Development of Keystroke Authentication. In Proc. of Int. Joint Conf. on Biometrics, pp. 1-8, 2014.

[8] R. Giot, M. El-bed, R. Christophe. Greyc keystroke: a benchmark for keystroke dynamics biometric systems. In Proc. of IEEE Intl. Conf. on Biometrics: Theory, Applications and Systems, pp. 1-6, 2009.

[9] Y. Li, B. Zhang, Y. Cao, S. Zhao, Y. Gao, J. Liu. Study on the Beihang Keystroke Dynamics Database. In Proc. of Intl. Joint Conf. on Biometrics, pp. 1-5, 2011.

[10] L. C. F. Araujo, L. H. R. Sucupira, M. G. Lizarraga, L. L. Ling, J. B. T. Yabu-uti. User Authentication Through Typing Biometrics Features. IEEE Trans. On Signal Processing, 53(2):851:855, 2005.

[11] Y. Zhong, Y. Deng, and A. K. Jain. Keystroke dynamics for user authentication. In Proc. Of IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 117-123, 2012.

[12] S. Cho, C. Han, D. H. Han, H. Kim. Web-based keystroke dynamics identity verification using neural network. Journal of Organizational Computing and Electronic Commerce, 10(4):295–307, 2000.

[13] J. V. Monaco, G. Perez, C. C. Tappert, P. Bours, S. Mondal, S. Rajkumar, A. Morales, J. Fierrez and J. Ortega-Garcia,. One-handed Keystroke Biometric Identification Competition. In Proc. IEEE/IAPR Int. Conf. on Biometrics, ICB, Phuket (Thailand), pp. 1-7, 2015.