The impact of database quality on keystroke dynamics authentication

Piotr Panasiuk, Mariusz Rybnik, Khalid Saeed, and Marcin Rogowski

Citation: [AIP Conference Proceedings](#) **1738**, 180020 (2016); doi: 10.1063/1.4951967

View online: <http://dx.doi.org/10.1063/1.4951967>

View Table of Contents: <http://scitation.aip.org/content/aip/proceeding/aipcp/1738?ver=pdfcov>

Published by the [AIP Publishing](#)

Articles you may be interested in

[Modular logic of authentication using dynamic keystroke pattern analysis](#)

AIP Conf. Proc. **1738**, 180012 (2016); 10.1063/1.4951959

[A Database Schema for the Analysis of Global Dynamics](#)

AIP Conf. Proc. **1168**, 918 (2009); 10.1063/1.3241632

[Marine wildlife behavior database for estimating environmental impacts](#)

J. Acoust. Soc. Am. **122**, 3059 (2007); 10.1121/1.2942907

[Impact of ultraviolet light on coercive field, poling dynamics and poling quality of various lithium niobate crystals from different sources](#)

J. Appl. Phys. **96**, 2816 (2004); 10.1063/1.1776324

[Database of open-access journals tightens its quality control](#)

Phys. Today

The Impact of Database Quality on Keystroke Dynamics Authentication

Piotr Panasiuk*, Mariusz Rybnik[†], Khalid Saeed** and Marcin Rogowski[‡]

*Faculty of Mathematics and Information Science, Warsaw University of Technology,
ul. Koszykowa 75, 00-662 Warsaw, Poland p.panasiuk@mini.pw.edu.pl

[†]Faculty of Mathematics and Computer Science, University of Białystok,
ul. Ciołkowskiego 1M, 15-245 Białystok, Poland mariuszrybnik@wp.pl

**Faculty of Computer Science, Białystok University of Technology,
ul. Wiejska 45A, 15-351 Białystok, Poland k.saeed@pb.edu.pl

[‡]King Abdullah University of Science and Technology,
Thuwal 23955-6900, Kingdom of Saudi Arabia marcin.rogowski@kaust.edu.sa

Abstract. This paper concerns keystroke dynamics, also partially in the context of touchscreen devices. The authors concentrate on the impact of database quality and propose their algorithm to test database quality issues. The algorithm is used on their own <KDS database> as well as the well-known <Keystroke Dynamics – Benchmark Data Set database>. Following specific problems were researched: classification accuracy, development of user typing proficiency, time precision during sample acquisition, representativeness of training set, sample length.

Keywords: access control, authentication, behavioral biometrics, keystroke dynamics, computer security, identification, touchscreen

PACS: 89.20.Ff

INTRODUCTION

Computers, and especially mobile devices, become more and more ubiquitous nowadays. Computer security however, is not given enough attention. The most common and important tasks in security are *authentication* – determining if the specific person (or the remote computer in the network) identity conforms to its claim, and *identification*: determining the user's identity, without a hint who he is. Expanding the Wood's classification [1] we distinguish three group of methods: proof by knowledge, proof by possession and biometrics. *Keystroke dynamics* is a *behavioral biometric* method that is among many physical and behavioral characteristics currently researched.

A case when the particular biometric feature cannot be obtained may happen (a simple example is finger injury altering the fingerprint), thus multi-biometric systems based on more than one feature are desirable. In [2] voice, hand geometry and face image are used together. [3] correctly points out that the researchers often overlook an important disadvantage of many biometric methods – *acceptability*. Obtaining fingerprints or an iris scan may be considered insulting by some people.

Keystroke dynamics, like *gait analysis*, has significant advantages over other biometric features. It is non-invasive, highly acceptable and – in its basic form – it does not need specialized hardware. There are also some disadvantages of *keystroke biometrics*: (i) efficient interpretation of features can be problematic; (ii) limitations of devices, operating systems and communication protocols can affect the data quality. This paper is the continuation of an earlier conference paper [4], with addition of touchscreen devices issues.

The mobile phones market is switching towards touch-screen and smartphones, therefore new opportunities can be seen with such hardware. The displays of these devices are not as big as traditional keyboards, they are certainly bigger than the 12-key keyboards and miniature QWERTY keyboards used previously in mobile phones. The small size was the main factor that caused these keyboards to be thumb-typed and most likely that is the reason why authors in [5] found single key dwell time an unreliable characteristic. On bigger touch-screens, multiple fingers are more likely to be used, which will probably result in a similar number of discriminative features available to full-size PC keyboards. The most widely used technology nowadays – capacitive touchscreens – does no longer rely on the information about pressure to work as in previously used resistive displays. However, with the exact position of fingers on the screen, the area under the finger can be calculated thus both pressure and finger size can be estimated.

THE AUTHORS' APPROACH

In this section, the authors describe their *identification* algorithm operating on fixed-text samples. All samples containing errors are disregarded. The goal is to compare *KDS database* to *Keystroke Dynamics – Benchmark Data Set*. We use *k-Nearest Neighbors* classifier. We choose the k value and then build a training dataset where the amount of samples per each user cannot be less than k . The remaining user samples are assigned to the testing dataset, so $k+1$ samples per user are needed. If this condition is not satisfied, the user is not taken into account. The authors' approach in [6] was to calculate initial weights for all expected key events.

During the classification the distance to all training samples is calculated for each test sample using Manhattan metrics between corresponding keyboard event times. Additionally, absolute times are processed into *flight* times and *dwell* times. Flight times are the times between releasing one key and pressing another. Dwell time is the time when a key is in the pressed state. The reason the authors (as well as many others) convert simple event times into those two characteristics is because they are more stable. When the user makes a mistake or hesitates on some key, this would only affect the next two keys and not all the remaining times. Partial distances for two given samples were calculated using Manhattan distance (for corresponding dwells and flights). Both flight and dwell times are important and if we take only one of them into account, the rate of proper classification decreases significantly, as the authors presented in [7].

After the calculation of all distances between the given sample and each training sample, we have k users results marked with the training author ID that were assigned a rank. We evaluate only closed-world case, that is we do not consider users outside of the system (when taking into account impostors from the outside world, a mixed approach – both rank and threshold basis – can be applied). Among all the results we take the k -Nearest Neighbors best ones and then conduct voting procedure on users. The shortest distance gets the highest score of k -Nearest Neighbors [8], the longest distance gets the lowest score of 1. If there is more than one vote on a particular user, we merge those adding scores, so finally there is only one vote per user. The winner is the user with the greatest score.

RESULTS COMPARISON

Classification Efficiency

With their algorithm, the authors of this work have tried each of their previous approaches and many combinations for the initial parameter values. In the beginning it was tested with the authors' newest method, which calculates the weights of each keyboard event based on Fisher's discriminant before classification. Those weights are later used in the calculation of the distances between test and training samples. However, this method gave worse results on *Keystroke Dynamics – Benchmark Data Set* than our simplest solution. This revealed also interesting correlation of data samples "quality" with the acquisition time.

For the comparison the authors also tested this algorithm on *KDS database* after proper data preparation. For this experiment the authors used the same amount of users in both databases, number of characters in a phrase (reduced to the phrase "kaloryfer"), amount of training samples and algorithm. In both experiments the training data sets were created using random samples, all the parameters were the same for both databases, $k=2$, training set containing 6 samples for each user and 51 classes. Fig. 1 shows the results of this comparison where the flight-to-dwell importance (ratio of corresponding weights for distance [7]) is presented in horizontal axis.

FIGURE 1. Classification rates for the authors approach on both databases under the same circumstances

Fig. 1 shows that even if we have databases with the same amount of data and similar supervised texts, the results may be different. It leads to the claim that the results are not comparable even if we test the algorithm on similar databases. This supports the conclusion from [9] that all the results obtained by research teams on their own databases may be incomparable in practice.

On the Increase in the User's Typing Proficiency

In the *Keystroke Dynamics – Benchmark Data Set* users were asked to type 50 samples each time in 8 sessions separated by at least 24 hours. The authors wondered how the learning process influences the results, so we tried to test a number of configurations. For the first experiment the authors took 8 random samples from all sessions per each user. In the second experiment the authors selected the first sample from each session. In the third experiment only the first 8 samples from the first session were taken into account. In the fourth experiment the last 8 samples from the last session were selected for training. We always used 8 training samples per user profile and we set k value to 8 in our algorithm.

As one can conclude, using the samples collected early does not result in satisfactory accuracy. The characteristics obtained from them are distorted by the fact that the user was still unfamiliar with the password. The first samples from each session also are not a very trustworthy training dataset because the user had a long break between inserting them and they differ from the average user's characteristics. The last inserted samples are better, however, the user seemed to be tired typing so many samples and they may not be as stable as the samples from the middle of the session.

Many of keystroke dynamics methods are based on Artificial Neural Network (ANN) algorithms [10]. The authors' experiments show that the first samples of the user are the noisiest ones, and using them to train the ANN yields poor results similar to those shown in [9].

FIGURE 2. The mean value of the standard deviation of keystrokes in all samples in each session for all users in the database

One can observe how the standard deviation of each keystroke changes over time. In Fig. 2 it can be seen that with every session the standard deviation is decreasing. This means that gradually the users start to insert samples in a more consistent manner. Inner-class differences decrease, what helps in discriminating the class. It also reduces FAR in verification systems.

On the Meaning of Time Precision

In [11] keyboard functioning using 15 MHz function and arbitrary waveform generator was examined. It was noticed that 18.7% of keystrokes were acquired with a 200 microsecond error. Therefore, the keyboard was calibrated and the database was collected using higher precision. Data have been gathered with an accuracy of $100\mu s$. This experiment has shown that databases gathered using different machines may not be comparable because of the lack of the main bus clock calibration. What is more, when the CPU(s) is under load, the delays in keystroke acquisition occur as they are usually handled by the message queue. There is also a difference between the lengths of keyboard clock frames of Linux/Unix (10 ms frame) and Windows (15 ms frame, 64 ticks per second) operating systems. The influence of time resolution on the algorithm classification accuracy was tested using the authors' approach.

CONCLUSIONS AND FUTURE WORK

There has been a lot of research done in the field of keystroke dynamics since the beginning of the 1980s. Unfortunately, the results are hardly comparable due to the use of various procedures or not published databases. Researchers have been trying to find new methods to improve keystroke dynamics accuracy, in particular extracting new features. Keystroke dynamics cannot be used for forensic purpose on their own, as the method does not meet the European access control standards such as EN-50133-1. It specifies that FRR (False Rejection Rate) should be less than 1% and FAR (False Acceptance Rate) should be no more than 0.001% [12]. However, when including other features, the keystroke dynamics feasibility will potentially improve.

The authors' *KDS database* has slightly worse time precision than *Keystroke Dynamics – Benchmark Data Set* due to the use of non-specialized hardware and use of JavaScript. However, the way of collecting the data represents real-world situations more closely and the database is more suitable for the user *identification* because of the longer samples. It is also worth mentioning that *KDS database* contains users' mistakes and corresponding methods of correction, which could be used in further analysis and experiments.

As the authors have proven, there is a correlation between the chosen training dataset and the algorithm accuracy. The samples obtained later with greater users' proficiency are of better quality and they allow distinguishing users more reliably. An algorithm for updating the training set should be considered, as using only the initial samples collected early would affect the system accuracy.

All the observations described above lead to the conclusion that the selection of the data for the tests has the vital meaning for the results' reliability. All the tests should be run on the same database without any further modifications and with a fixed method of training data set building. If the conditions are not satisfied, the obtained results cannot be compared with the others. One should notice that even when using similar database (the amount of users and samples), the results can vary, often greatly.

ACKNOWLEDGMENTS

This work is supported by the Rector of Bialystok University of Technology (grant no. S/WI/1/2013) and partially by the Rector of Warsaw University of Technology.

REFERENCES

1. H. M. Wood, "The Use of Passwords for Controlling Access to Remote Computer Systems and Services," in *Proceedings of the National Computer Conference*, AFIPS '77, ACM, New York, NY, USA, 1977, pp. 27–33.
2. K. Veeramachaneni, L. Osadcw, and P. Varshney, "An adaptive multimodal biometric management algorithm," in *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 2005, vol. 35, pp. 344–356.
3. X. Li, S. Maybank, S. Yan, D. Tao, and D. Xu, "Gait Components and Their Application to Gender Recognition," in *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 2008, vol. 38, pp. 145–155.
4. M. Rybnik, P. Panasiuk, K. Saeed, and M. Rogowski, "Advances in the Keystroke Dynamics: The Practical Impact of Database Quality," in *Computer Information Systems and Industrial Management*, Springer Berlin Heidelberg, 2012, vol. 7564, pp. 203–214, ISBN 978-3-642-33259-3.
5. S. Karatzouni, and N. Clarke, "Keystroke Analysis for Thumb-based Keyboards on Mobile Devices," in *New Approaches for Security, Privacy and Trust in Complex Environments*, edited by H. Venter, M. Eloff, L. Labuschagne, J. Eloff, and R. Solms, Springer US, 2007, vol. 232 of *IFIP International Federation for Information Processing*, pp. 253–263.
6. P. Panasiuk, and K. Saeed, "A Modified Algorithm for User Identification by His Typing on the Keyboard," in *Image Processing and Communications Challenges 2*, edited by R. S. Choraś, Springer Berlin Heidelberg, 2010, vol. 84 of *Advances in Intelligent and Soft Computing*, pp. 113–120, ISBN 978-3-642-16294-7.
7. M. Rybnik, P. Panasiuk, and K. Saeed, "User Authentication with Keystroke Dynamics Using Fixed Text," in *International Conference on Biometrics and Kansei Engineering, 2009. ICBACE 2009.*, 2009, pp. 70–75.
8. S. A. Dudani, "The distance-weighted k-nearest-neighbor rule," in *Systems, Man and Cybernetics, IEEE Transactions on*, 4, IEEE, 1976, pp. 325–327.
9. K. S. Killourhy, and R. A. Maxion, "Comparing anomaly-detection algorithms for keystroke dynamics," in *DSN*, IEEE, 2009, pp. 125–134, URL <http://dblp.uni-trier.de/db/conf/dsn/dsn2009.html#KillourhyM09>.
10. S. Cho, C. Han, D. H. Han, and H.-I. Kim, "Web-based keystroke dynamics identity verification using neural network," in *Journal of organizational computing and electronic commerce*, Taylor & Francis, 2000, vol. 10, pp. 295–307.
11. K. R. Killourhy, "The Effect of Clock Resolution on Keystroke Dynamics," in *Recent Advances in Intrusion Detection*, edited by R. Lippmann, E. Kirda, and A. Trachtenberg, Springer Berlin Heidelberg, 2008, vol. 5230 of *Lecture Notes in Computer Science*, pp. 331–350, ISBN 978-3-540-87402-7, URL http://dx.doi.org/10.1007/978-3-540-87403-4_18.
12. CENELEC, Access control systems for use in security applications, Standard Number EN 50133-1:1996/A1:2002, Technical Body CLC/TC 79, European Committee for Electrotechnical Standardization, Tech. rep. (2002).