On the Comparison of the Keystroke Dynamics Databases

Piotr Panasiuk[1], Marcin Dąbrowski[2], Khalid Saeed[3,4],
and Katarzyna Bocheńska-Włostowska[5]

[1] DCC Labs, Warsaw, Poland
[2] University of Finance and Management in Bialystok, Elk Branch, Poland
[3] AGH University of Science and Technology
[4] Bialystok Technical University, Bialystok, Poland
[5] The Maria Sklodowska-Curie Warsaw Academy, Warsaw, Poland
piotr@panasiuk.org, marcin.dabrowski@poczta.fm,
saeed@agh.edu.pl, katarzyna.bochenska@uwmsc.pl

Abstract. This paper concerns about Keystroke Dynamics database quality which can vary depending on researchers' approach. Database classification has been presented and the most popular publicly available databases are introduced. Authors' database is presented and compared with the others. This paper introduces new database and compares the results of the same algorithms obtained on two almost identical databases. The results of comparison are discussed in terms of keystroke dynamics dataset quality. It has been proven that different methods can produce results of unanticipated kind.

Keywords: Keystroke dynamics, verification, behavioral biometrics, database acquisition, computer security.

1 Introduction

Biometrics is a field of science that focuses on measuring live beings in order to recognize the entity. Everyone is different. There are no two individuals who would be undistinguishable. There are two kinds of features that could be measured. The first one is basing on physical features of organism like fingerprint, retina scan, DNA, vein pattern and other resulting from how the organisms are built. This kind of biometric is called physical biometrics. The second group of biometrics is called behavioral because those features originate from how one do things. The most common biometrics in this group are voice, handwritten signature, gait and the subject of this paper – keystroke dynamics. This kind of biometrics is not related to genetics, so even twins with the same DNA would have different characteristics of writing, walking or typing on a computer keyboard.

Keystroke dynamics accounts to behavioral biometrics. It has been originated from the telegraph. Soon after the telegraph was invented, operators developed ability to recognize each other by a timing pattern of pressing dots and dashes of messages sent in Morse code. Keystroke dynamics seems to be perfect for securing personal com-

K. Saeed and V. Snášel (Eds.): CISIM 2014, LNCS 8838, pp. 122–129, 2014.

puters due to similarity to the telegraph in sending electric signals [1]. Additionally each signal is supplied with the information of which key has been pressed or released and a timestamp of the event in milliseconds. This feature can be compared to the handwritten signature but written on the keyboard.

As an advantage, keystroke dynamics implementations does not require expensive hardware. Most researches have been conducted on simple PC with the most common keyboard. The greatest part of keystroke dynamics analysis is done by a software, which is the subject of this paper. Plain keyboard allows researchers to obtain such characteristics as press and release times of specific keys. There were few approaches to find some new characteristics to analyze. However those researches required specialized (often custom made) hardware such as pressure sensitive keyboards [2,3,4] or touchscreens [5,6,7] for analyzing field below finger during key press. There are also approaches basing on alternative text input methods like Swype on mobile phones [8].

2 Previous Approaches

Our teams' first documented research on keystroke dynamics is dated back on 2008 [9], where authors presented promising results with the use of simple classification techniques and analyzing only “dwell” and “flight” times. Every single user had to enter three different and independent samples (without any repetitive words), where two of them were 110 keystrokes each (used as reference) and last one had length of about 55 keystrokes (used for validation). Keys other than letters and “space” were ignored. These include: “shift”, cursor keys, “delete”, “backspace” and other non-alphanumeric keys. Algorithm was basing on 1-NN classifier and resulted with accuracy of 75.68% on a group of 37 individuals.

The purpose of the next approach [10] was to further analyze new methods. In this paper authors presented authentication experiments on one single-word phrase and two longer phrases in Polish and English languages respectively. In contrast to our teams' previous work, the letter or number represented by keystroke and position in phrase was taken into account this time. From that work one can deduct that acquiring and analyzing more samples from single user results in better classification accuracy. Also, longer phrase (28 keystrokes) allowed to obtain much better accuracy (90,83%) than the shorter one (9 keystrokes, 68,7%). These results were obtained for 21 and 23 users respectively.

The following experiments brought even greater accuracy by using improved k-NN algorithm [11]. An improvement was made by calculating weights for each flight and dwell time in training dataset. This was possible due to use of fixed-text phrase. Firstly the mean values and variances for each key in every peculiar user's sample were used to calculate the weights. For each key the mean set separations for users were then calculated using Fisher's discriminant. In the end the mean value from all separation factors for each key were calculated, and followed by normalization. Those results were used as weights for each time. This improvement increased the classification accuracy to 98.78% within 16 users.

Unfortunately, those results were calculated on different phrases with different amount of classes, so they are hardly comparable and their purpose is just informative how efficient keystroke dynamics authentication can be. The same issue appears when one wants to compare two different approaches presented by some researchers but their results have been processed on different databases collected in different way and at different conditions (supervision, classes, samples strength etc.). This paper is supposed to compare two almost identical databases and show if the results could be comparable.

3 Database Gathering

3.1 Database Classification

It has been shown that keystroke dynamics authentication results highly depend on the database quality [12,13]. Viable algorithms should deal with noisy samples: the ones with typos or random pauses in user typing. Among the databases the authors can distinguish ones collected in a supervised way, meaning every test subject was individually instructed by a supervisor before the start of the samples acquisition process. The supervisor can also make notes on how the subject types and what influences him. It guarantees samples of good quality. This type of database, however, usually does not reflect real world situations. Databases may have accounts duplicated, for example if the user forgets his password or just wants to have multiple accounts. When typing pattern is duplicated by some user, it could decrease the identification accuracy and in hybrid (rank-threshold) based verification methods it may increase the FRR. On the other hand FAR can be increased by typing with unnatural manner by a user.

Another factor is the purpose for which the database is gathered. Authentication requires user ID attached to keystroke data. Simulation of hacking requires the same text typed by many users. Passwords are usually short phrases often consisting additional characters like capital letters (that involve shift key), dots, semicolons, numbers and symbols. For identification, samples should be preferably longer, as this application is more complex.

There can be two additional approaches to keystroke data acquisition. The first is based on a fixed text. The second way is to use free-text authorization [13] to continuously monitor user's workstation while trying to authorize him/her. There are the following problems with free-text authorization: (i) how often user authentication algorithm should be run, (ii) more difficulty with data collection, (iii) more samples are needed for learning of the recognition algorithm. Additionally potential noise can be a unique feature that helps to recognize users, so removing it completely – without deeper analysis – would be a loss of valuable information.

3.2 Publicly Available Databases

Most of works were done basing on closed private datasets gathered for a specific research purpose only. This situation makes the results incomparable because amount of information carried by the sample is variable and depends on the length and used key sequences. Sample value depends on user proficiency in given language, occurrence of special characters, digits or case-sensitiveness. What is more each database and each experiment use different amount of users and different count of samples which also influences overall results. Although most of papers describe results calculated on closed databases, researchers' awareness is growing and more and more databases are released to the public. There are few databases available online currently. Below the most interesting ones are described.

The oldest database in this comparison has been built and released by Montalvao et al. [14]. Database has been stored in four archives. Sample data consist of press-press intervals only. Database A and B carry data of four fixed English phrases. Database C contains two fixed Portuguese words and database D keeps freely typed rows of text. Each data package was gathered in a different manner. For more details and knowledge about the database one can refer to:

`http://www.biochaves.com/en/download.htm`

One of the most interesting is the database built by Maxion et al. [15]. This database contains great quality samples as it has been gathered under supervised conditions. Collected samples contain lower-case, upper-case letters, digits and a special character. Samples were measured using external reference clock to get the time precision on the level impossible to get on common PC. The accuracy of the clock was +- 200 µs. There are 51 users registered in the database who left overall 400 samples each in 8 sessions. Maxion's database is available online for public use at:

`http://www.cs.cmu.edu/~keystroke/`

Database provided by Giot et al. [16] contains 133 users, 100 of which provided at least five sessions of samples. Session consisted of typing "greyc laboratory" twelve times on two different keyboards. It is stored in SQLite database file and contains both dwell and flight times. It is available online at:

`http://www.ecole.ensicaen.fr/~rosenber/keystroke.html`

Special attention deserves a database built by Allen [17], who additionally to press and release times made it possible to collect pressure force data while typing specific key. Amount of registered users is 104 which is quite a lot, however samples count left by users varies from 3 to 504 so many of them may occur unusable for some experiments. Each sample consists of three phrases. Database is available at:

`http://jdadesign.net/2010/04/pressure-sensitive-keystroke-dynamics-dataset/`

One more database has been announced to become public in a short time. Idrus et al. [18] collected database of 110 individuals which 70 of them were located in France and 40 in Norway. What is additionally interesting subjects originated from 24 different countries. Every user left 20 samples under supervised conditions consisting of 5 phrases. Users located in France used keyboard with French layout and those located in Norway with Norwegian.

3.3 Authors' Database

Authors' dataset is based on Maxion's phrase ".tie5Roanl". In the opposite to Maxion's however data were gathered in unsupervised conditions and with the use of commonly available devices and technologies. The goal was to simulate real-life scenario. Authors wanted to verify how the results will change if the keystroke dynamics algorithm would be implemented in some web, browser-based application using JavaScript to gather key timing information. This question is valid since keyboard event timing may be affected by OS process queuing clock. While Maxion's database was built using arbitrary waveform generator [19] with accuracy of 200μs, in authors' database accuracy was limited to OS event clock precision, which is 15.625 ms (64 Hz) using MS Windows and 10 ms using most Linux distributions. Data gathering schema was similar to Maxion's. Each user taken under consideration in this study had to leave 400 samples in 8 sessions. The sample itself was identical to the one proposed by Maxion in his research and consisted of ".tie5Roanl" with Enter key at the end of the input. There are 45 users satisfying the condition of 400 valid samples. Database was gathered under unsupervised conditions, however collecting algorithm disallows corrections. In such case after making a mistake the sample was cleared and the user had to type it once again from the beginning.

4 Database Comparison in Practice

The goal of this experiment was to compare two similar databases collected under different conditions. In comparison we used algorithms provided by Maxion available online at: `http://www.cs.cmu.edu/~keystroke/`

Eight anomaly detection algorithms were chosen to examine gathered data. These include: Euclidean, Manhattan, Mahalanobis, Chebyshev, Canberra, Scaled Manhattan, k-NN and k-Means. Most of them were provided by Roy Maxion and Kevin Killourhy at: `http://www.cs.cmu.edu/~keystroke/ksk-thesis/`, while the others are authors' own implementation.

The experiment was conducted on two identically formed databases and with the same evaluation scripts. In order to keep the same amount of information the hold time of "Return" key had to be removed from Maxion's database. This was necessary because while author's mechanism was triggered by "Return" key press the release time was not stored in the database under some web browsers. Also count of identities in Maxion's database had to be trimmed because in author's database there are only 45 users satisfying the condition of 400 samples per user. This way the same amount of users can be found in both databases. Another unexpected issue was occurrences of zero values. In some cases two following key events were so close that they fell into the same operating system clock window. This resulted in zero distance between them. Because some metrics cannot handle zero values well (e.g. Canberra distance where division by zero may occur), we had to replace zero values with values near to zeros (in our case mean value of 10^{-7}). Both databases consisted of 45 users who typed phrase ".tie5Roanl" 400 times in 8 sessions. Figure 1 shows mean EER values.

Figure 2, however, presents the EER standard deviation.

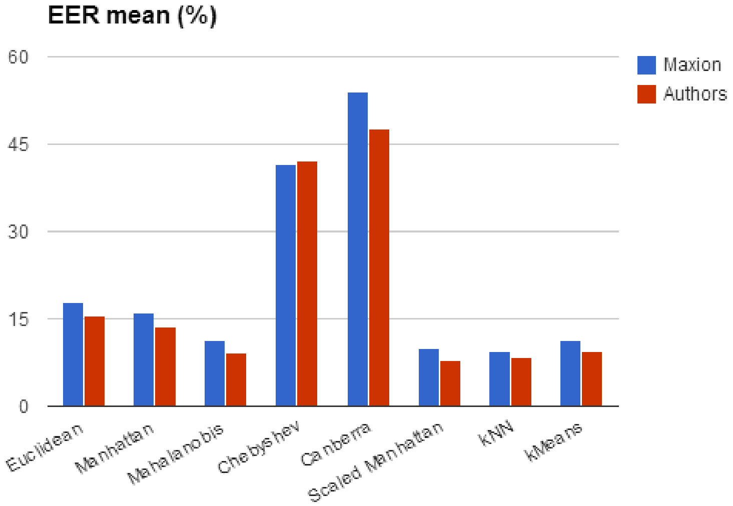

Fig. 1. Results of anomaly detectors on both databases. Mean EER values.

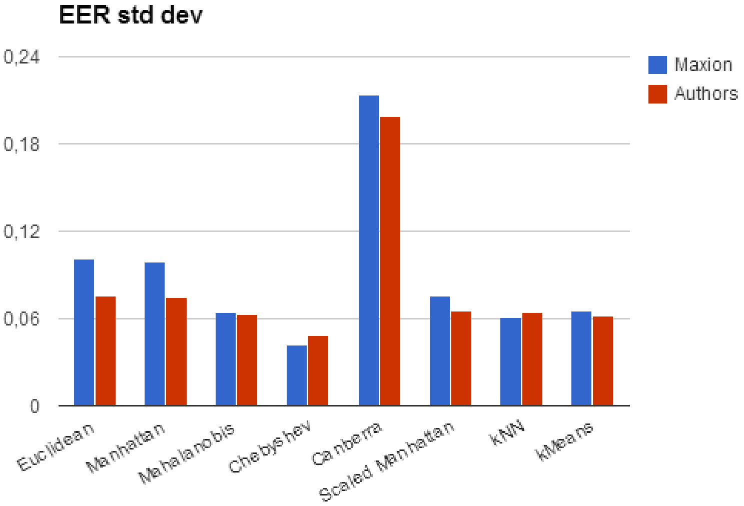

Fig. 2. Results of anomaly detectors on both databases. Standard deviation EER values.

5 Conclusions

The main goal of this paper was to verify if the same algorithm run on two theoretically identical databases, fulfilling the criteria of the same phrase, the same amount of keystrokes and the same count of classes will provide the same results.

As one can see the results vary depending on which metric was used. However despite very similar data, exactly the same amount of samples and using the same evaluation algorithms there are differences in the results ranging up to about 30%. This makes the results incomparable and leads to conclusion that each database is different and every new keystroke dynamics algorithm should be tested on public reference one.

Moreover, awareness of researchers on this matter is growing and hopefully more and more databases will become publicly available. However, this may lead to another threat, that many experiments would be executed on different databases. What is more every database fits to different purpose. Identification database will not be the best match to the verification algorithm. The same as a fixed-text algorithm would not work well on a free-text database. Better insight into this matter is needed. The databases should be classified according to their purpose and reference ones should be preferred by the research community.

What is also worth noting is that surprisingly authors' database gives better results in most cases although it was collected using technologies charged with higher inaccuracy. This was opposite to our expectations and is to further examination.

Acknowledgement. The research was partially supported by grant no. 11.11.220.01, AGH University of Science and Technology in Krakow.

References

1. Checco, J.C.: Keystroke Dynamics and Corporate Security. WSTA Ticker (2003)
2. Dietz, P.H., Eidelson, B., Westhues, J., Bathiche, S.: A practical pressure sensitive computer keyboard. In: Proc. of the 22nd Annual ACM Symposium on User Interface Software and Technology, New York (2009)
3. Saevanee, H., Bhattarakosol, P.: Authenticating User Using Keystroke Dynamics and Finger Pressure. In: Consumer Communications and Networking Conference, Las Vegas, NV, pp. 1–2 (2009)
4. Loy, C.C., Lai, W.K., Lim, C.P.: Keystroke Patterns Classification Using the ARTMAP-FD Neural Network. In: Intelligent Information Hiding and Multimedia Signal Processing, Kaohsiung, pp. 61–64 (2007)
5. Clarke, N.L., Furnell, S.M.: Authenticating mobile phone users using keystroke analysis. International Journal of Information Security 6(1) (2006)
6. Campisi, P., Maiorana, E., Lo Bosco, M., Neri, A.: User authentication using keystroke dynamics for cellular phones. IET Signal Processing 3(4) (2009)
7. Karatzouni, S., Clarke, N.L.: Keystroke Analysis for Thumb-based Keyboards on Mobile Devices. In: Venter, H., Elofif, M., Labuschagne, L., Elofif, J., von Solms, R. (eds.) New Approaches for Security, Privacy and Trust in Complex Environments. IFIP, vol. 232, pp. 253–263. Springer, Boston (2007)
8. Trojahn, M., Ortmeier, F.: Toward Mobile Authentication with Keystroke Dynamics on Mobile Phones and Tablets. In: Advanced Information Networking and Applications Workshops (WAINA), pp. 697–702. IEEE (2013)
9. Rybnik, M., Tabedzki, M., Saeed, K.: A Keystroke Dynamics Based System for User Identification. In: 7th Computer Information Systems and Industrial Management Applications, CISIM 2008, pp. 225–230. IEEE (2008)

10. Rybnik, M., Panasiuk, P., Saeed, K.: User Authentication with Keystroke Dynamics Using Fixed Text. In: IEEE-ICBAKE 2009 International Conference on Biometrics and Kansei Engineering, Cieszyn, Poland, pp. 70–75 (2009)
11. Panasiuk, P., Saeed, K.: A Modified Algorithm for User Identification by His Typing on the Keyboard. In: Choraś, R.S. (ed.) Image Processing and Communications Challenges 2. AISC, vol. 84, pp. 113–120. Springer, Heidelberg (2010)
12. Killourhy, K.S., Maxion, R.A.: Comparing Anomaly-Detection Algorithms for Keystroke Dynamics. In: Dependable Systems & Networks, Lisbon, Portugal, pp. 125–134 (2009)
13. Panasiuk, P., Saeed, K.: Influence of Database Quality on the Results of Keystroke Dynamics Algorithms. In: Chaki, N., Cortesi, A. (eds.) CISIM 2011. CCIS, vol. 245, pp. 105–112. Springer, Heidelberg (2011)
14. Montalvao, J., Almeida, C.A.S., Freire, E.O.: Equalization of keystroke timing histograms for improved identification performance. In: International Telecommunications Symposium, pp. 560–565. IEEE (2006)
15. Killourhy, K.S., Maxion, R.A.: Comparing Anomaly-Detection Algorithms for Keystroke Dynamics. In: Dependable Systems & Networks, Lisbon, Portugal, pp. 125–134 (2009)
16. Giot, R., El-Abed, M., Rosenberger, C.: GREYC Keystroke: a Benchmark for Keystroke Dynamics Biometric Systems. In: IEEE Third International Conference on Biometrics: Theory, Applications and Systems (BTAS), Washington, DC, USA, pp. 28–30 (2009)
17. Allen, J.D.: An Analysis of Pressure-Based Keystroke Dynamics Algorithms, Master's thesis, Southern Methodist University, Dallas, TX, USA (2010)
18. Idrus, S.Z.S., Cherrier, E., Rosenberger, C., Bours, P.: Soft Biometrics Database: A Benchmark for Keystroke Dynamics Biometric Systems. In: International Conference of the Biometrics Special Interest Group (BIOSIG), pp. 1–8. IEEE (2013)
19. Killourhy, K.S., Maxion, R.A.: The Effect of Clock Resolution on Keystroke Dynamics. In: Lippmann, R., Kirda, E., Trachtenberg, A. (eds.) RAID 2008. LNCS, vol. 5230, pp. 331–350. Springer, Heidelberg (2008)