Development of the Keystroke Dynamics Recognition System

E A Kochegurova[1], **E S Gorokhova**[1], **A I Mozgaleva**[1]

[1] Tomsk Polytechnic University, 30, Lenina Ave., Tomsk, 634050, Russia

E-mail: GorokhovaES@mail.ru, kocheg@tpu.ru

Abstract. The paper is related to creating an algorithm for keystroke dynamics recognition and development of software, which is able to identify users according to their keystroke dynamics. Different characteristics of keystroke dynamics are considered. Probabilistic-statistical methods are compared with neural network algorithms for recognition. The algorithm for recognition was created and implemented. The software was tested with the help of some users. Their keystroke dynamics was analyzed in order to determine an efficiency of the created algorithm.

1. Introduction

Nowadays, the importance of information is difficult to overstate. Users should be identified in order to delimit access to it. The problem of user authentication traditionally belongs to a field of interest of cryptolography, different directions of which are still developing [1-4]. It is necessary to notice that standard authentication tools, such as a user name and a password, cannot already provide the required stage of protection, because passwords may be stolen or hacked. That is why, biometric methods of authentication, which are based on the biological characteristics of a specific individual, are becoming increasingly popular [5]. This group of methods includes:

- Voice analysis;
- Facial recognition;
- Iris recognition;
- Fingerprint recognition;
- Keystroke dynamics recognition.

A keystroke dynamics recognition method is convenient for users and security departures due to the fact that keystroke dynamics monitoring is made in a hidden mode and does not requires any extra actions from users. One of the main advantages of this method is low cost of necessary hardware, because almost every computer has a keyboard. That is why, we should make a point of methods and algorithms of keystroke dynamics recognition.

2. Algorithm description

Keystroke dynamics is the detailed timing information that describes exactly when each key was pressed and when it was released when concrete person is typing at a keyboard of a computer, gadget etc [6]. The most common features, which are used to characterize keystroke dynamics, include dwell time, intervals between key presses and overlapping of key presses. Below there is an explanation of the content of these characteristics.

Dwell time is a period, during which a key is in the pressed state. It is usually measured in milliseconds.

Overlapping occurs when one key is not up and another key is already on. The increase of speed leads to the increase of a number of overlapping of key pressed.

Pause is the period between key-up and key-on moments of two keys.

Figure 1 illustrates characteristics of keystroke dynamics, described above. For example, pieces 1 and 2 mean dwell time, piece 3 is overlapping and piece 4 is interval.

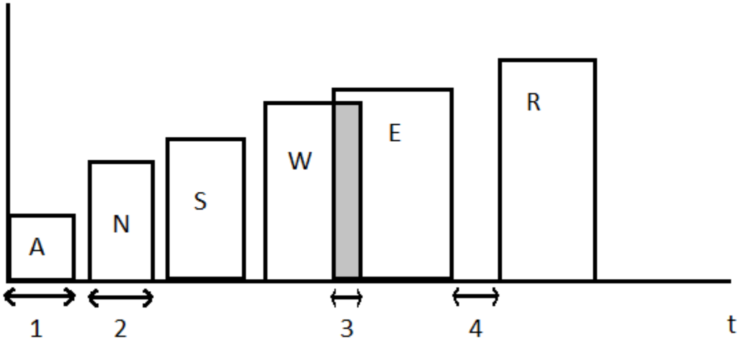

Figure 1. A time diagram of the input word "answer"

Keystroke dynamics may be compared using different kinds of methods, especially bio-inspired or neural networks-based algorithms and other probabilistic-statistical methods of decision making [7, 8].

Neural network based methods, providing high precision, are considered. On the other hand, they require high computation performance as well and cannot be used in real-time computations. Moreover, the learning stage for this kind of systems may run over time. One more problem occurs due to the fact that system cannot get learning examples for all "illegal" users [9].

Probabilistic-statistical methods assume calculating of some statistical characteristics (mean, variance) on the basis of a limited sample of the dynamic. Then the obtained characteristics are compared with a control sample for each user [10]. These methods become quite efficient when significant volume of statistics is collected. Therefore it was decided to develop a probabilistic-statistical algorithm for keystroke dynamics recognition.

In a simplified form, the keystroke dynamics recognition algorithm may be divided into 2 stages. At the first stage, a program collects a set of statistics from keystroke dynamics characteristics of a user. The second stage is for comparison of the obtained user's keystroke characteristics with standard values for this user. The comparison may be made with the help of any proximity measure. Here we use Euclidean distance, which may be calculated as follow:

$$P = \sqrt{\sum_{i=1}^{N} (t_{et} - t_{cur})^2} \ ,$$

where N – the amount of different characters,

t_{st} – the standard dwell time for a key;

t_{cur} – the current dwell time for this key.

In this work, the dwell time for cases with overlapping keys is evaluated separately from cases without overlapping. Only characters from Russian and English alphabets were considered.

The server component of the created software is responsible for storing and processing data about keyboarding characteristics of different users. The client component of this software is used for collecting necessary data about keyboarding. The process of getting such kind of data happens while the user presses keys. Moreover, the client component can show examples of keyboarding of known users as a graph or in a table. Necessary data are transferred between a client and server components via TCP-sockets. Another feature of the created software is an ability to log a file with records about time when every user works with a computer.

An algorithm for the first stage of recognition system's work is represented in figure 2. This stage aims to learn the standard of a keystroke for different users and collect some statistics. It includes a key name, description of a keyboard event, e.g. a key may be up or down, and the time when the event occurs.

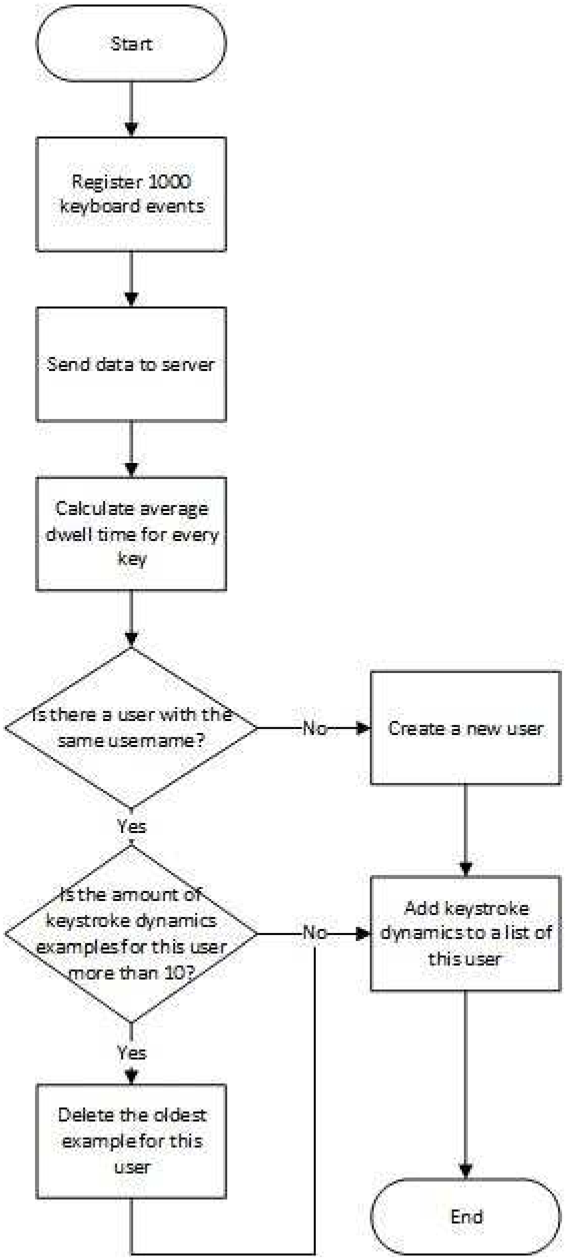
Start
Register 1000 keyboard events
Send data to server
Calculate average dwell time for every key
Is there a user with the same username?
No
Create a new user
Yes
Is the amount of keystroke dynamics examples for this user more than 10?
No
Add keystroke dynamics to a list of this user
Yes
Delete the oldest example for this user
End

Figure 2. The first stage of the recognition algorithm

A number of typing mistakes are calculated separately. It is considered to be equal to the amount of presses of such keys as “Delete” and “Backspace”.

This kind of data is obtained with a client application and then proceeds to a server application for further analysis.

After that, the average dwell time is calculated for every key, and overlapping presence is detected.

Then, if a user works with the system for the first time, characteristics of his keystroke dynamics are saved for a new user with a specified username. Otherwise, the current keystroke dynamics are added to a list for this user. In case of a too long list – more than 10 records for one person – the oldest one is deleted by the system.

Such approach takes into account the fact that the keystroke dynamics of one person may vary in different psycho-emotional states or at different times of the day. For example, researches noticed [11] that tired persons usually type more slowly and make more mistakes.

The next stage of authentication is recognition. The algorithm for keystroke dynamics recognition is shown in figure 3.

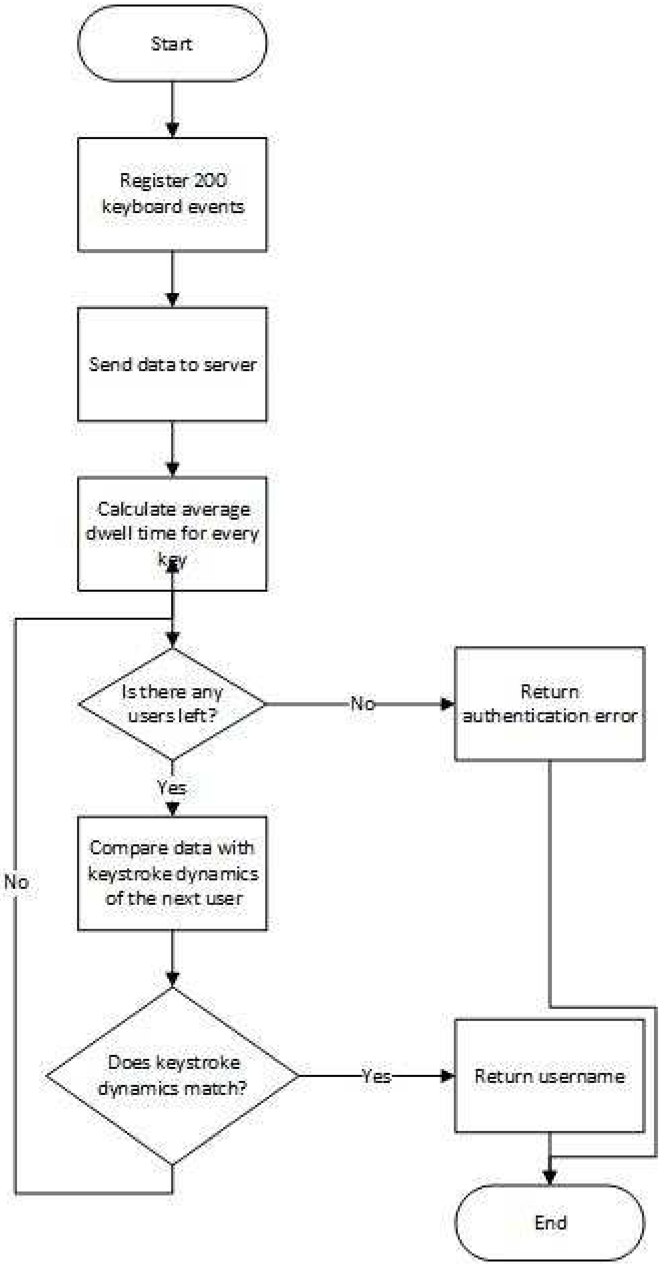
Start
Register 200 keyboard events
Send data to server
Calculate average dwell time for every key
Is there any users left?
No
Return authentication error
Yes
Compare data with keystroke dynamics of the next user
No
Does keystroke dynamics match?
Yes
Return username
End

Figure 3. The second stage of the recognition algorithm

The algorithm of recognition starts similarly as an algorithm of learning, which was described above. The main difference between them is a number of analyzed characters, which decrease from 1000 for the learning algorithm to 200 for the recognition algorithm. It was done in order to make the process of recognition faster and reveal the change of the user as soon as possible. But the volume of the data set must be statistically significant. The result of the identification process is the username of that person, whom keystroke dynamics is the most matching (according to Euclidean distance and confidence level more than 0.9). In case none of the known keystroke dynamics matches, the system reports about an authentication error.

3. Experiments and simulation

The keystroke dynamics recognition system now is at the stage of testing. In order to do this, the keystroke dynamics examples of 10 users were used. The keystroke dynamics of these users were analyzed and compared. While the experiment proceeded, the users were typing different texts as usual. The program collected the statistics of the keystroke dynamics and put it in the database. After that the results were analyzed as it is shown below.

A graph in figure 4 shows the dwell time for 5 examples of the keystroke belonging to the same user. Line 1 means the dwell time for the first example of the user's keystroke dynamics. Line 2 relatively means the dwell time for the second example and so on. The dwell time is shown for the most frequency used letters of the Russian alphabet, because users typed texts in Russian. This picture shows that the dwell time for every letter stays quite similar for different examples of keystroke dynamics.

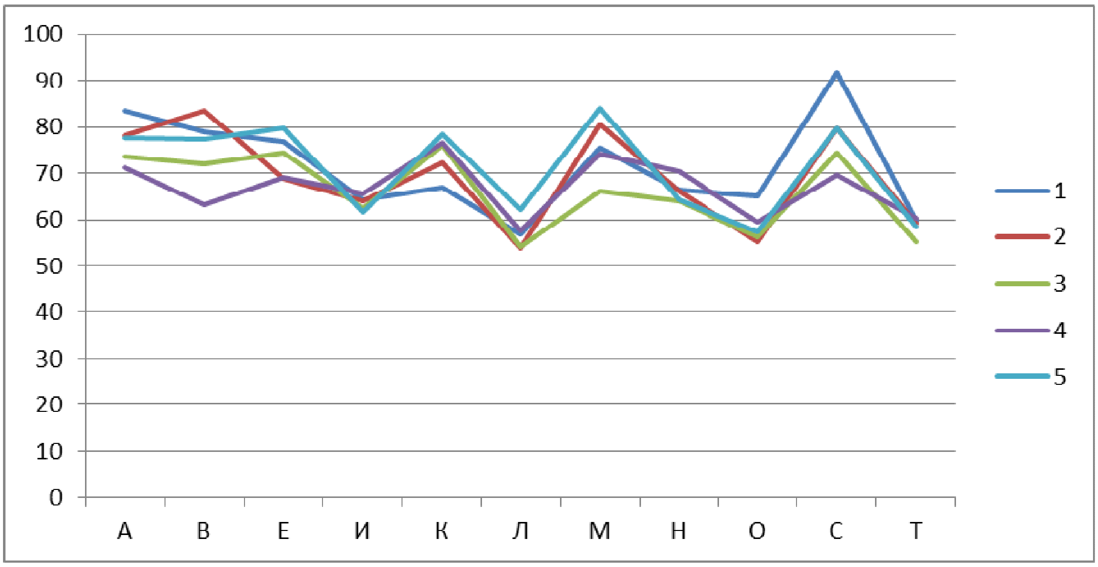

Figure 4. Comparison of the dwell time for one user

The next testing stage includes comparing keystroke dynamics of 2 different users.

Figure 5 contains a graph, which represents the dwell time for 2 examples of keystroke dynamics for two different users.

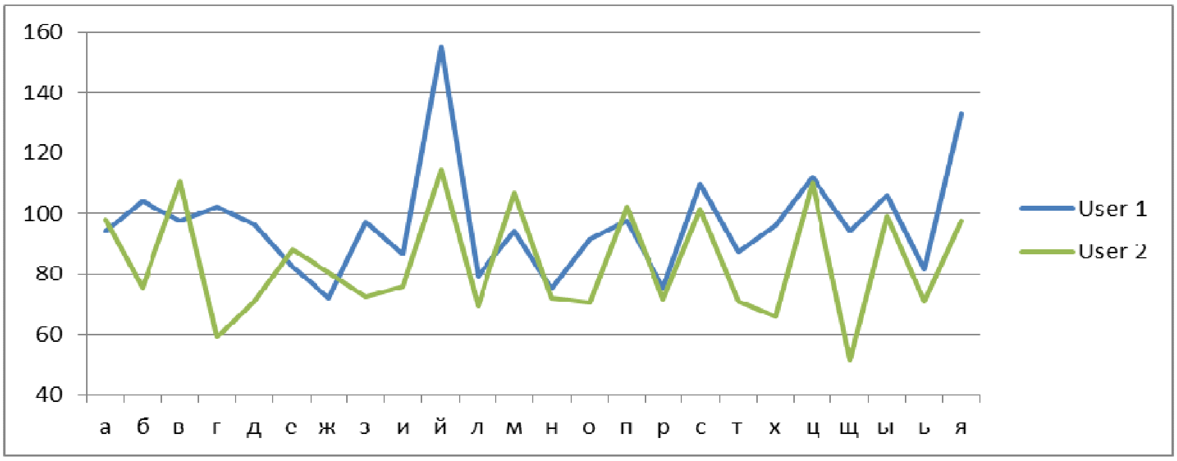

Figure 5. Comparison of the dwell time for two different users

In this case, a range of the dwell time for the first user is from 72 ms to 155 ms and for the second user it is from 51.5 ms to 114.29 ms. The average dwell time for all keys is equal to 96.67 ms for the first user and 83.59 ms – for the second one. Here, the difference is more than 10 ms (>10%), as it is distinct from the case of comparing examples of keystroke dynamics, which belong to one user, where the difference was less than 1 ms. The average difference for all keys is 13.07 ms.

The results are summarized in table 1.

Table 1. Keystroke dynamics comparison

	t_{max}, ms	t_{min}, ms	t_{av}, ms
User1 , example 1	123.67	83.14	95.13
User1 , example 2	116.33	79	95.32
User 2	155	72	96.67
User 3	114.29	51.5	83.59

Testing results allow concluding that every person has their own keystroke dynamics. At the same time, keystroke dynamics of different users differ more than by 10% and may be recognized statistically. That is why, authentication based on keystroke dynamics recognition will be efficient.

The final testing stage is focused on the system's ability to recognize users. Two users tried to authorize themselves in the system using their keystroke dynamics. The first user succeeds 17 times out of 20 attempts. The second user succeeds 18 times out of 20 attempts. This result shows 0.875% accurancy of the recognition algorithm.

4. Conclusion

The aim of this work is to develop the software and the algorithm for the keystroke dynamics recognition system. The algorithm was created based on the probabilistic-statistical method. It enables the system to save examples of users keyboarding and compare them for the authentication purpose.

The system was tested with the help of some users. The analysis of the results shows that the keyboarding recognition is an efficient authentication tool.

References

[1] Basavegowda R A 2013 Secret Code Authentication Using Enhanced Visual Cryptography *Emerging Research in Electronics, Computer Science and Technology* (Springer India) pp.69-76

[2] Boyd C and Mathuria A 2003 Authentication and Key Transport Using Public Key Cryptography *Protocols for Authentication and Key Establishment* (Springer Berlin Heidelberg) pp. 107-135

[3] Chourasia J 2013 Identification and authentication using visual cryptography based fingerprint watermarking over natural image *CSI Transactions on ICT* **1 (4)** 343-348

[4] Smart N J 2003 *Cryptography: An Introduction* (McGraw-Hill) pp. 257-271

[5] Jain A, Flynn P and Ross A A 2008 *Handbook of Biometrics* (Springer US) pp. 1-2

[6] Moskovitch R, Feher C, Messerman A, Kirschnick N, Mustafić T, Camtepe A, Löhlein B, Heister U, Möller S, Rokach L and Elovici Y 2009 Identity Theft, Computers and Behavioral Biometrics *Proc. of the 2009 IEEE Int. Conf. on Intelligence and security informatics* (IEEE Press Piscataway, NJ, USA) pp. 155-160

[7] Kochegurova E A and Gorokhova E S 2016 Optimizing Urban Public Transportation with Ant Colony Algorithm *Lecture notes in artificial intelligence* **9875** 489-497

[8] Cherkashina Yu A and Gerget O M 2016 Regression analysis for solving diagnosis problem of children's health *Materials Science and Engineering* **124(1)** 1-7

[9] Brown M and Rogers S J 1993 User identification via keystroke characteristics of typed names using neural networks. *Int J Man-Machine* **39 (6)** 999-1014

[10] Monrose F and Rubin A D 1999 Keystroke dynamics as a biometric for authentication *Future Generation Computer Systems* **16 (9)** 351-359

[11] Nazmul H N Alam J M Mahmud H and Hasan K 2014 Identifying emotion by keystroke dynamics and text pattern analysis *Behaviour and Information Technology* **33(9)** 987-996