Continuous Keystroke Dynamics-Based User Authentication Using Modified Hausdorff Distance

Maksim Zhuravskii, Maria Kazachuk, Mikhail Petrovskiy(✉), and Igor Mashechkin

Lomonosov Moscow State University, Vorobjovy Gory, Moscow 119899, Russia
paperlark@icloud.com, {mkazachuk,michael,mash}@cs.msu.ru

Abstract. Continuous keystroke dynamics-based user authentication methods are one of the most perspective means of user authentication in computer systems. Such methods do not require specialized equipment and allow detection of user change anytime during a user session. In this paper, we explore new approaches to solving the problem based on Hausdorff distance and its modification, including a new method, the sum of maximum coordinate deviations. We compare proposed methods to existing ones that are based on distance functions defined in feature space, statistical criteria, and neural networks. Based on the experiments, we observe that the proposed method based on the sum of maximum coordinate deviations with k nearest feature vector selection reports the highest accuracy of all reviewed methods.

Keywords: Outlier detection · Continuous authentication · Keystroke dynamics · Hausdorff distance · SMCD

1 Introduction

In recent decades, cloud services have gained much attention. They are used for accomplishing both personal and enterprise tasks. Thus, it is crucial to ensure the security of information processed by the cloud services. One of the key tasks in ensuring information security is authentication, a process by which subjects, normally users, establish their identity to a system [4]. For authentication purposes, subjects provide an identifier to the system.

Authentication methods differ by the nature of the identifier used in the process. Although methods that employ secret knowledge, such as a password, or an identification object, such as a magnetic card, are easy to implement, they are vulnerable to identifier compromise. In this regard, the most promising methods are those that use user's biometrics for identification. Biometrics is divided into two categories: physiological samples and behavioral ones.

Physiological samples represent the physiological characteristics of a person which remain with him throughout his life. These include fingerprints, iris, and

C. S. Jensen et al. (Eds.): DASFAA 2021 Workshops, LNCS 12680, pp. 223–236, 2021.
https://doi.org/10.1007/978-3-030-73216-5_16

facial geometry. A significant drawback of such methods is that they depend on specialized equipment.

Behavioral patterns represent the behavioral characteristics of a person, such as voice, gait, and handwriting. These methods do not require specialized equipment. However, existing approaches are less stable than those based on physiological samples.

As for authentication frequency, continuous authentication is preferred as it eliminates the possibility of an attacker gaining access to the system sometimes after a legitimate user passed authentication.

Thus, the most promising task is continuous user authentication based on behavioral biometric characteristics. As such, we can consider keystroke dynamics, i.e. characteristics of a person's dynamics when working with a computer keyboard. Since the keyboard is one of the primary means of human interaction with a computer, keystroke dynamics can be effectively used for continuous user authentication.

In this paper, we discuss continuous keystroke dynamics-based user authentication as an outlier detection problem [9]. The goal of the research was to improve authentication effectiveness when a small dataset is used to train a user's model. With this goal in mind, we propose a new method, the sum of maximum coordinate deviations, or SMCD. Along with this method, we explore new approaches based on the Hausdorff distance [7] and its modifications. Our hypothesis was that these approaches would achieve higher efficiency when compared to other ones.

This article has the following structure. Section 2 provides an overview of existing continuous keystroke dynamics-based authentication methods. Section 3 describes methods based on the Hausdorff distance and the sum of maximum coordinate deviations. Section 4 is devoted to an experimental study of the proposed methods and their comparison with existing approaches to solving the problem. Section 5 summarises the obtained results.

2 Related Work

When considering authentication methods, we should pay attention to what is considered features of keystroke dynamics in a specific method, how these features are preprocessed, which algorithm is used to decide the user's legitimacy, and what accuracy the method has. In this section, we discuss each of these steps in detail.

2.1 Feature Extraction and Preprocessing

Most studies use single keystroke characteristics [2,3,10,12,13,17] and digraph characteristics [2,3,8,10,13,17] as features of user's keystroke dynamics. Here, a digraph is a combination of two consecutive user keystrokes. The duration of keystrokes is considered as a characteristic of single clicks. As for digraphs, all possible intervals between digraph keystrokes are used as its features.

Here and further to facilitate notation we will denote a set $\{i \in N | a \leq i \leq b\}$ where $a, b \in N$ with $\overline{a, b}$. With this in mind, let us denote $t_i^{down}, i = \overline{1, 2}$ the time when the i-th digraph key is pressed, $t_i^{up}, i = \overline{1, 2}$ the time when the i-th digraph key is released (see Fig. 1). Then, we can define four digraph characteristics as follows:

1. DD-time, or duration of the interval between keypresses: $t_2^{down} - t_1^{down}$;
2. DU-time, or digraph input duration: $t_2^{up} - t_1^{down}$;
3. UU-time, or duration of the interval between key releases: $t_2^{up} - t_1^{up}$;
4. UD-time, or duration of the jump between keystrokes: $t_2^{down} - t_1^{up}$.

To construct feature vectors a sliding window can be used [8]. In this case, we select a window of a certain size from the sequence of user keystrokes and calculate features based on the keystrokes contained in the window. When the user clicks, the sliding window shifts, and authentication is performed again (see Fig. 2). This means that the user is authenticated continuously while he is using the keyboard.

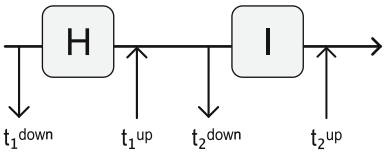

Fig. 1. Digraph features.

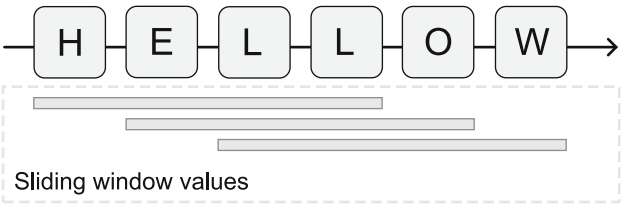

Fig. 2. Sliding window operation.

Using a sliding window allows to use aggregate feature values, such as their empirical means [3,8,10,12,17] and standard deviations [3,8,12,17]. This increases the stability of the method to possible outliers in individual feature values. It is worth noting, however, that a smaller sliding window makes a method more responsive to a possible user change. Hence, smaller window size is preferable.

Modern keyboards contain more than $n = 70$ different keys. As a result, the dimension of the feature space of all possible digraphs and all possible single keystrokes exceeds $n^2 + n = 4970$. To reduce the dimensionality of the feature space, in studies [8,10,17] it is proposed to split keyboard keys into groups based on their physical location or functional purpose. In this case, features of digraphs or individual keystrokes are calculated only within the group and between different groups. As a result, we can decrease the number of considered features, and increase the number of their values observed in the sliding window.

2.2 User Model Construction

As mentioned earlier, we consider the problem of continuous user keystroke dynamics-based authentication an outlier detection problem [9]. In study [8], various classification methods are considered in application to the problem of deciding the user's legitimacy based on their keystroke dynamics. The study covers methods based on statistical criteria, distance functions defined in features space, and machine learning methods, such as SVM, k nearest neighbors, and SVDD. According to the results obtained, the method based on the Kolmogorov-Smirnov statistics [15] reports to be the most effective for small sliding window size.

This method considers random variables ξ and η that correspond to some feature of a legitimate user and a tested one respectively. Based on the feature values obtained during two users were interacting with the keyboard, we construct empirical cumulative distribution functions $F_\xi(x)$ and $F_\eta(x)$. The Kolmogorov-Smirnov statistic is:

$$D_{\xi,\eta} = \sup_{x \in \mathbb{R}} \mid F_\xi(x) - F_\eta(x) \mid . \tag{1}$$

The decision on the legitimacy of the user being tested is made based on comparing the statistic value with a certain threshold value. The threshold value can be calculated based on a given significance level or selected experimentally.

Some works explored the possibility of using neural networks for creating a model of legitimate user's behaviour [11].

3 Proposed Approach

This section describes the proposed approaches to continuous keystroke dynamics-based user authentication.

3.1 Feature Extraction

To build features based on the available data, we use the sliding window method described in Sect. 2. We use sliding windows of size from 100 to 300 events, where each event describes either a key press or a release. The event description

consists of its type: press or release, a key code, and a timestamp of the moment when the event occurred.

Feature values that correspond to long pauses when the user stops typing are abnormal for the legitimate user model. Their presence in the sliding window degrades the quality of the authentication method. Thus, to prevent this we split the window if the pause between consecutive keystrokes exceeds the 40-s threshold. The optimal values of the sliding window parameters were selected experimentally.

It is worth noting that the space of keystroke dynamics features is sparse. This is because only a small subset of all possible digraphs and single keystrokes can occur in the sliding window. To select only the most significant features from the set of all features and reduce the dimensionality of feature space, in [10] we proposed to reduce the dimension of the feature space by selecting a fixed number of single keystrokes and digraphs that are most common in the observed window. This way, in each window we select 37 most frequently encountered keys and 100 most frequently encountered digraphs. For selected keystrokes, we calculate their keystroke duration, and for selected digraphs, we calculate the average duration of the UU and DU intervals. The optimal ratio of single keypress and digraph features was found during a preliminary series of experiments.

Along with the features described above we also consider the following features: average keystroke duration by groups (see Fig. 3), average frequency of keystrokes in the sliding window, average frequency of command keys used by the user in the sliding window.

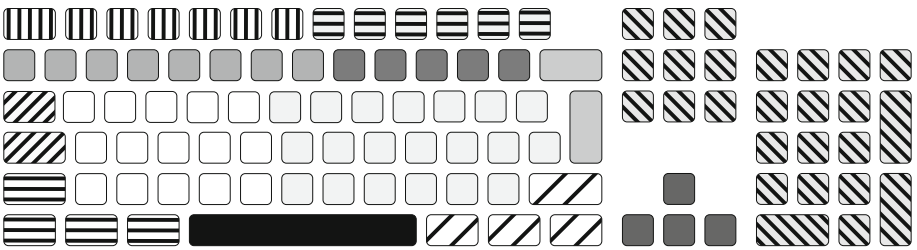

Fig. 3. Proposed key groups for feature extraction.

3.2 Quantile Discretization

To preprocess the obtained feature values we propose to use quantile discretization [10].

Let $\mathbb{X}$ be a training sample of values of some feature η. Using sample $\mathbb{X}$ we calculate empirical quantiles q_i of orders $\frac{i}{k}, i = \overline{1, k-1}$, where number k is an algorithm hyper-parameter. Then we replace every value $\hat{\eta}$ of feature η with the number j such that:

$$j = \begin{cases} 1 & \text{if } \hat{\eta} \in (-\infty, q_1]; \\ k & \text{if } \hat{\eta} \in (q_{k-1}, +\infty); \\ t & \text{if } t \in \overline{2, k-1} \text{ and } \hat{\eta} \in (q_{t-1}, q_t]. \end{cases} \tag{2}$$

Quantile discretization of features allows mitigating small fluctuations in values of continuous features by discretizing their values. In addition to that, it was shown in [10] that the distribution of the considered features is multimodal. By applying quantile discretization it is possible to smoothen feature distributions and thereby improve the quality of machine learning methods designed to work with more homogeneous data.

Based on the results of a series of experiments, the optimal number of sampling intervals is $k = 7$.

3.3 Hausdorff Distance

We propose to use the Hausdorff distance [7] for outlier detection. Let $\mathbb{X} = \{x^1, \ldots, x^m\}$ and $\mathbb{Y} = \{y^1, \ldots, y^l\}$ be two sets of vector from a metric space $\mathbb{M}$ with the distance function ρ. The Hausdorff distance between sets of vectors in $\mathbb{X}$ and $\mathbb{Y}$ is defined as:

$$\begin{aligned} \rho_H(\mathbb{X}, \mathbb{Y}) &= \max\{h(\mathbb{X}, \mathbb{Y}), h(\mathbb{Y}, \mathbb{X})\}; \\ h(\mathbb{X}, \mathbb{Y}) &= \max_{i \in \overline{1,m}} \min_{j \in \overline{1,l}} \rho(x^i, y^j); \\ h(\mathbb{Y}, \mathbb{X}) &= \max_{j \in \overline{1,l}} \min_{i \in \overline{1,m}} \rho(x^i, y^j). \end{aligned} \tag{3}$$

Let us consider the feature space as a metric space $\mathbb{M}$. Let $\mathbb{X}$ be a training sample of feature vectors of a legitimate user and $\mathbb{Y}$ be a set containing a single feature vector y of a tested user. In that case, i.e. for $l = 1$, Formula 3 can be rewritten as follows:

$$\rho_H(\mathbb{X}, \mathbb{Y} \mid l = 1) = \max_{i \in \overline{1,m}} \rho(x^i, y). \tag{4}$$

To decide the legitimacy of the tested user, we compare the value of Hausdorff distance $\rho_H(\mathbb{X}, \mathbb{Y})$ to a threshold value set a priori. If $\rho_H(\mathbb{X}, \mathbb{Y})$ exceeds the threshold, the hypothesis about the legitimacy of the current user is rejected, and the user's session is suspended. Otherwise, the user continues working in the system.

Let us also consider a method based on a modified Hausdorff distance [5]. The modified Hausdorff distance, or MHD, is defined as:

$$\begin{aligned}\rho_{\text{MHD}}(\mathbb{X},\mathbb{Y}) &= \max\{h_M(\mathbb{X},\mathbb{Y}), h_M(\mathbb{Y},\mathbb{X})\};\\ h_M(\mathbb{X},\mathbb{Y}) &= \frac{1}{m}\sum_{i=1}^{m}\min_{j=\overline{1,l}}\rho(x^i,y^j);\\ h_M(\mathbb{Y},\mathbb{X}) &= \frac{1}{l}\sum_{j=1}^{l}\min_{i=\overline{1,m}}\rho(x^i,y^j).\end{aligned} \tag{5}$$

The modified Hausdorff distance is less sensitive to the presence of outliers in the sets $\mathbb{X}$ and $\mathbb{Y}$. For $l = 1$, Formula 5 is equivalent to:

$$\rho_{\text{MHD}}(\mathbb{X},\mathbb{Y} \mid l=1) = \frac{1}{m}\sum_{i=1}^{m}\rho(x^i,y). \tag{6}$$

We also investigate the method based on interpolated modified Hausdorff distance [14]. Interpolated modified Hausdorff distance, or IMHD, is defined as follows:

$$\begin{aligned}\rho_{\text{IMHD}}(\mathbb{X},\mathbb{Y}) &= \max\{h_I(\mathbb{X},\mathbb{Y}), h_I(\mathbb{Y},\mathbb{X})\};\\ h_I(\mathbb{X},\mathbb{Y}) &= \frac{1}{l}\sum_{j=1}^{l}\min_{i=\overline{2,m}}\rho\left(\frac{x^i+x^{i-1}}{2},y^j\right);\\ h_I(\mathbb{Y},\mathbb{X}) &= \frac{1}{m}\sum_{i=1}^{m}\min_{j=\overline{2,l}}\rho\left(\frac{y^j+y^{j-1}}{2},x^i\right).\end{aligned} \tag{7}$$

For $l = 1$, Formula 7 is equivalent to:

$$\rho_{\text{IMHD}}(\mathbb{X},\mathbb{Y} \mid l=1) = \min_{i=\overline{2,m}}\rho\left(\frac{x^i+x^{i-1}}{2},y\right). \tag{8}$$

3.4 Sum of Maximum Coordinate Deviations

As a modification of the method based on the Hausdorff distance, we propose a method based on the sum of maximum coordinate deviations.

Let us denote $\mathbb{M}$ a n – dimensional feature space and let $\mathbb{X} = \{x^1,\ldots,x^m\}$ be a set of m vectors from space $\mathbb{M}$ where $x^i = (x_1^i,\ldots,x_n^i), i = \overline{1,m}$. The sum of the maximum coordinate deviations, or SMCD, between the vector $y = (y_1,\ldots,y_n) \in \mathbb{M}$ and the set $\mathbb{X}$ is defined as:

$$\rho_S(y,\mathbb{X}) = \frac{1}{n}\sum_{i=1}^{n}\max_{j\in\overline{1,m}}|x_i^j - y_i|. \tag{9}$$

This means that the greater the value of the sum of the maximum coordinate deviations is the greater is the difference between the vector y and the vectors from the set $\mathbb{X}$.

Let us introduce a feature space $\mathbb{M}$, a training sample of the feature vectors $\mathbb{X}$, and a feature vector y of the tested user in the same way as they were introduced

in the previous subsection. As before, the decision on the legitimacy of the tested user can be made based on the result of comparing the value of the sum of the maximum coordinate deviations $\rho_S(y, \mathbb{X})$ with a threshold value set a priori. The current user's legitimacy hypothesis is rejected if the value $\rho_S(y, \mathbb{X})$ exceeds the specified threshold value.

Note that the sum of the maximum coordinate deviations is related to the Hausdorff distance. Let $\mathbb{M}_{l1}$ be an n-dimensional linear metric space with norm $\|a\| = \frac{1}{n}\sum_{i=1}^{n}|a_i|, a \in \mathbb{M}_{l1}$ and $\mathbb{X} = \{x^1, ..., x^m\}$, $\mathbb{Y} = \{y\}$ be two sets of vectors from $\mathbb{M}_{l1}$. Then, taking into account Formula 4, the following is true:

$$\rho_H(\mathbb{X}, \mathbb{Y}) = \frac{1}{n}\max_{j=\overline{1,m}}\sum_{i=1}^{n}|x_i^j - y_i|. \tag{10}$$

In a linear metric space $\mathbb{M}_{l1}$ for an arbitrary set of vectors $\mathbb{Z} = \{z^1, \ldots, z^m\}$ is the following true:

$$\max_{j=\overline{1,m}}\sum_{i=1}^{n}|z_i^j| = \sum_{i=1}^{n}|z_i^{j_0}| \le \sum_{i=1}^{n}\max_{j=\overline{1,m}}|z_i^j|, \tag{11}$$

where $z^{j_0} \in \mathbb{Z}$ is the element of the set $\mathbb{Z}$ that provides the maximum in the left part of the equation.

Combining Formulas 9, 10, and 11 we conclude that the following is true:

$$\rho_H(\mathbb{X}, \mathbb{Y}) \le \rho_S(y, \mathbb{X}). \tag{12}$$

Thus, sum of maximum coordinate deviations serves as an upper boundary of the Hausdorff distance in the linear space $\mathbb{M}_{l1}$.

3.5 K Nearest Neighbors

Along with other methods, an outlier detection method based on the k nearest neighbors search method is considered in study [8].

Let us introduce the feature space $\mathbb{M}$, a training sample of feature vectors $\mathbb{X}$, and the feature vector y of the tested user as in the previous subsection. Let $\rho(.,.)$ be a distance function defined in space $\mathbb{M}$. Then, for the vector y and a given value k there is a subset $\hat{\mathbb{X}} = \{\hat{x}^1, \ldots, \hat{x}^k\} \subset \mathbb{X}$ containing k nearest to the vector y of vectors from the set $\mathbb{X}$ based on the distance function $\rho(.,.)$. The decision on the legitimacy of the tested user is made based on comparing the average distance from the vector y to the vectors from $\hat{\mathbb{X}}$:

$$d(y, \mathbb{X}) = \frac{1}{k}\sum_{j=1}^{k}\rho(y, \hat{x}^j). \tag{13}$$

As before, if the value is $d(y, \mathbb{X})$ exceeds a certain threshold set a priori, and the hypothesis about the legitimacy of the tested user is rejected.

In study [8], the L2 distance function is defined in feature space. However, other distance functions can also be used. In this regard, we also considered a variation of this method that defines a cosine distance in feature space.

Also, we propose a modification of this method. As before, we find a subsample $\hat{\mathbb{X}}$ of k vectors closest to the vector y in the sample $\mathbb{X}$. To decide the legitimacy of the tested user, we will use the value of the Hausdorff distance $\rho_H(\mathbb{X}, \{y\})$ (see Formula 3), modified Hausdorff distance $\rho_{\text{MHD}}(\mathbb{X}, \{y\})$ (see Formula 5), interpolated modified Hausdorff distance $\rho_{\text{IMHD}}(\mathbb{X}, \{y\})$ (see Formula 7), or the sum of maximum coordinate deviations $\rho_S(y, \hat{\mathbb{X}})$ (see Formula 9).

4 Results

We tested the proposed methods on a dataset used in studies [12,16]. It contains collected data for 144 users while they were working with the computer keyboard. The dataset contains over 1,345 registered clicks for each user as well as the following information about each user: platform used by the user: desktop or laptop, user's gender, user's age group, user's dominant hand, user's awareness of the data collection.

In our study we use only the following information about user's actions: code of the pressed key, keypress start timestamp, keypress end timestamp.

When testing each method, we used 80% of the legitimate user's sample to fit the method, and the remaining 20% as well as other users' samples to evaluate the method.

To assess each method we only used those users for whom we obtained more than 10 feature vectors during feature extraction. We do so to mitigate the fact that a smaller training sample may result in poor model fit and, as a result, poor accuracy of the tested authentication method.

To compare the effectiveness of the methods under consideration, we used ROC AUC, which is equal to the value of the area under the ROC curve. It can be interpreted as the probability that the model ranks a randomly chosen positive instance higher than a randomly chosen negative instance [6]. Hence, it can be used to evaluate the quality of classification without setting an exact threshold for deciding the user's legitimacy.

Along with ROC AUC, we assessed methods' effectiveness based on equal error rate (EER) [8] and average precision (AP) [18].

As a baseline model, we consider a one-class SVM with an RBF core [10]. This model is often used in outlier detection problems. We also consider the Fuzzy method [10] and neural network models.

We considered different neural network models based on the autoencoder architecture [1], including fully connected, recurrent, and fully convolutional. These models are trained to encode feature vectors of the legitimate user into vectors of reduced dimensionality. The decision on the legitimacy of the user being tested is made based on comparing the Euclidean distance between the initial and restored vectors with a certain threshold value set a priori.

Based on the results of preliminary experiments a fully convolutional autoencoder (see Fig. 4). It is worth noting that deep neural networks require a larger dataset to get an optimal model. Hence, to assess the method based on the fully convolutional autoencoder we only used those users for whom we obtained more than 100 feature vectors. This drawback of neural network models is the reason why we did not focus our research on these models.

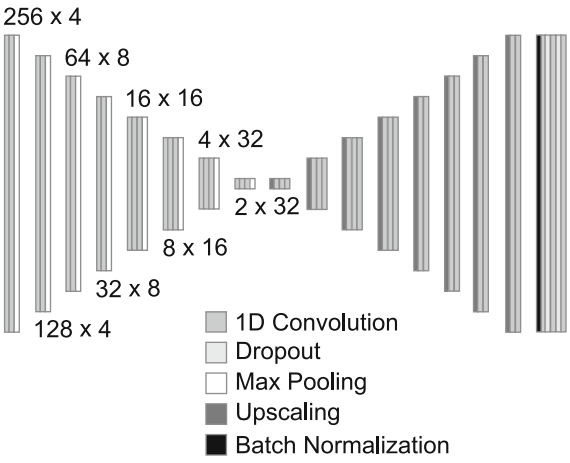

Fig. 4. Fully convolutional autoencoder model.

In this study we evaluated the existing methods based on Kolmogorov-Smirnov statistic, a one-class SVM with RBF core, the k nearest neighbors algorithm, the Fuzzy method, the approach based on a fully convolutional autoencoder, the proposed approaches based on Hausdorff distances, and the proposed methods based on SMCD. Every algorithm was tested both with and without quantile discretization. Table 1 contains descriptions of algorithms with their optimal hyper-parameter values that were found using grid search method. Table 2 shows the experimental results.

According to the results, the proposed methods of continuous authentication based on SMCD are more effective than all other methods. Meanwhile, methods based on the composition of the k nearest neighbors method and the sum of maximum coordinate deviations report the best accuracy. It is worth noting that the proposed method based on the sum of maximum coordinate deviations surpasses methods based on other considered modifications of the Hausdorff distance.

Table 1. Description of tested algorithms.

Authentication method	Method parameter	Parameter value
One-class SVM	Feature preprocessing	Quantile discretization
	Kernel function	RBF
	Kernel width (γ)	0.000837
	Error fraction in training set (ν)	0.00144
Kolmogorov-Smirnov statistic	Feature preprocessing	–
Hausdorff distance	Feature preprocessing	–
	Distance function	L1
Modified Hausdorff distance	Feature preprocessing	Quantile discretization
	Distance function	L1
Interpolated modified Hausdorff distance	Feature preprocessing	Quantile discretization
	Distance function	L1
SMCD	Feature preprocessing	Quantile discretization
k nearest neighbors	Feature preprocessing	Quantile discretization
	Distance function	L2
	k	74
Hausdorff distance with k nearest neighbors selection	Feature preprocessing	Quantile discretization
	Distance function	L1
	k	72
MHD with k nearest neighbors selection	Feature preprocessing	Quantile discretization
	Distance function	L1
	k	73
IMHD with k nearest neighbors selection	Feature preprocessing	Quantile discretization
	Distance function	L1
	k	73
SMCD with k nearest neighbors selection	Feature preprocessing	Quantile discretization
	Distance function	L2
	k	58
Fuzzy	Feature preprocessing	Quantile discretization
	Kernel function	RBF
	Kernel width (γ)	1e−06
	Error fraction in training set (k)	0.001
	Affiliation level decrease rate (m)	12.5
Fully convolutional autoencoder	Feature preprocessing	Quantile discretization
	Optimization algorithm	Adam
	Loss function	LogCosh
	Learning rate	0.01
	Epochs	300

Table 2. Experiments results.

Authentication method	Median ROC AUC	IQR ROC AUC	Median AP	Median EER
One-class SVM	0.961	0.103	0.186	0.111
Kolmogorov-Smirnov statistic	0.558	0.181	0.351	0.500
Hausdorff distance	0.837	0.242	0.005	0.333
Modified Hausdorff distance	0.857	0.246	0.011	0.214
Interpolated modified Hausdorff distance	0.935	0.152	0.077	0.158
SMCD	0.950	0.091	0.305	0.064
k nearest neighbors	0.977	0.129	0.315	0.089
Hausdorff distance with k nearest neighbors selection	0.929	0.188	0.090	0.151
MHD with k nearest neighbors selection	0.893	0.235	0.046	0.173
IMHD with k nearest neighbors selection	0.960	0.156	0.176	0.130
SMCD with k nearest neighbors selection	**0.987**	**0.030**	**0.503**	**0.065**
Fuzzy	0.972	0.079	0.183	0.102
Fully convolutional autoencoder	0.908	0.128	0.095	0.185

5 Conclusion

Continuous keystroke dynamics-based user authentication is one of the most promising areas of authentication methods development. We propose new approaches to this problem that are based on the Hausdorff distance and its modifications as well as new methods based on the Hausdorff distance modification, the sum of maximum coordinate deviations.

According to the results of experimental research, the proposed method based on the sum of the maximum coordinate deviations with k nearest keyboard vectors selection reports the median value of the ROC AUC equal to 0.987. Thus, this method surpasses all other considered approaches, including the existing method based on Kolmogorov-Smirnov statistics, which is considered one of the most effective methods of one-class user authentication using keystroke dynamics, and a method based on a fully convolutional autoencoder.

Despite favorable experimental results, there are ways to extend the research. First of all, to guarantee language independence, the proposed method's performance should be evaluated on datasets in other languages. Secondly, with the proposed approach the size of the feature vector strongly depends on the sliding window size, i.e. in case of a smaller window fewer significant features can be extracted. Thus, to use a smaller sliding window different methods of feature extraction should be explored. Also, we plan to make a more detailed comparison with statistical and deep learning methods.

References

1. Badrinarayanan, V., Kendall, A., Cipolla, R.: SegNet: a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. **39**(12), 2481–2495 (2017). https://doi.org/10.1109/TPAMI.2016.2644615
2. Bicakci, K., Salman, O., Uzunay, Y., Tan, M.: Analysis and evaluation of keystroke dynamics as a feature of contextual authentication. In: 2020 International Conference on Information Security and Cryptology (ISCTURKEY), pp. 11–17 (2020). https://doi.org/10.1109/ISCTURKEY51113.2020.9307967
3. Bours, P., Mondal, S.: Performance evaluation of continuous authentication systems. IET Biometr. **4**(4), 220–226 (2015). https://doi.org/10.1049/iet-bmt.2014.0070
4. Butterfield, A., Ngondi, G.E., Kerr, A. (eds.): A Dictionary of Computer Science. Oxford University Press, Oxford (2016). https://doi.org/10.1093/acref/9780199688975.001.0001
5. Dubuisson, M.P., Jain, A.: A modified Hausdorff distance for object matching. In: Proceedings of 12th International Conference on Pattern Recognition, vol. 1, pp. 566–568 (1994). https://doi.org/10.1109/ICPR.1994.576361
6. Fawcett, T.: An introduction to ROC analysis. Pattern Recogn. Lett. **27**(8), 861–874 (2006). https://doi.org/10.1016/j.patrec.2005.10.010
7. Huttenlocher, D.P., Klanderman, G.A., Rucklidge, W.J.: Comparing images using the Hausdorff distance. IEEE Trans. Pattern Anal. Mach. Intell. **15**(9), 850–863 (1993). https://doi.org/10.1109/34.232073
8. Kang, P., Cho, S.: Keystroke dynamics-based user authentication using long and free text strings from various input devices. Inf. Sci. **308**, 72–93 (2015). https://doi.org/10.1016/j.ins.2014.08.070
9. Kazachuk, M., Petrovskiy, M., Mashechkin, I., Gorokhov, O.: Outlier detection in complex structured event streams. Moscow Univ. Comput. Math. Cybern. **43**(3), 101–111 (2019). https://doi.org/10.3103/S0278641919030038
10. Kazachuk, M., et al.: One-class models for continuous authentication based on keystroke dynamics. In: Yin, H., et al. (eds.) IDEAL 2016. LNCS, vol. 9937, pp. 416–425. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46257-8_45
11. Kim, J., Kang, P.: Recurrent neural network-based user authentication for freely typed keystroke data. CoRR abs/1806.06190 (2018). http://arxiv.org/abs/1806.06190. Withdrawn
12. Monaco, J.V., Bakelman, N., Cha, S., Tappert, C.C.: Developing a keystroke biometric system for continual authentication of computer users. In: 2012 European Intelligence and Security Informatics Conference, pp. 210–216 (2012). https://doi.org/10.1109/EISIC.2012.58
13. Raul, N., Shankarmani, R., Joshi, P.: A comprehensive review of keystroke dynamics-based authentication mechanism. In: Khanna, A., Gupta, D., Bhattacharyya, S., Snasel, V., Platos, J., Hassanien, A.E. (eds.) International Conference on Innovative Computing and Communications. AISC, vol. 1059, pp. 149–162. Springer, Singapore (2020). https://doi.org/10.1007/978-981-15-0324-5_13
14. Shao, F., Cai, S., Gu, J.: A modified Hausdorff distance based algorithm for 2-dimensional spatial trajectory matching. In: 2010 5th International Conference on Computer Science Education, pp. 166–172 (2010). https://doi.org/10.1109/ICCSE.2010.5593666

15. Shiryayev, A.N.: 15. On the empirical determination of a distribution law. In: Shiryayev, A.N. (ed.) Selected Works of A. N. Kolmogorov. Mathematics and Its Applications (Soviet Series), vol. 26, pp. 139–146. Springer, Dordrecht (1992). https://doi.org/10.1007/978-94-011-2260-3_15
16. Tappert, C.C., Villani, M., Cha, S.H.: Keystroke biometric identification and authentication on long-text input. In: Behavioral Biometrics for Human Identification: Intelligent Applications, pp. 342–367. IGI global (2010)
17. Villani, M., Tappert, C., Ngo, G., Simone, J., Fort, H., Cha, S.: Keystroke biometric recognition studies on long-text input under ideal and application-oriented conditions. In: 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW 2006), p. 39 (2006). https://doi.org/10.1109/CVPRW.2006.115
18. Zhang, E., Zhang, Y.: Average Precision. Springer, Boston (2009). https://doi.org/10.1007/978-0-387-39940-9_482