User Authentication from Mouse Movement Data Using SVM Classifier

Conference Paper *in* Lecture Notes in Computer Science · November 2016

DOI: 10.1007/978-3-319-48965-0_47

Bashira Akter Anima
University of Nevada, Reno

Mahmood Jasim
University of Dhaka

Abir Rahman
Daffodil International University

M. Hasanuzzaman
University of Dhaka

User Authentication from Mouse Movement Data Using SVM Classifier

Bashira Akter Anima[1], Mahmood Jasim[1],
Khandaker Abir Rahman[2](✉), Adam Rulapaugh[2],
and Md Hasanuzzaman[1]

[1] University of Dhaka, Dhaka, Bangladesh
anima_csedu@yahoo.com, mahmood.jasim.0@gmail.com,
hzamancsdu@yahoo.com
[2] Saginaw Valley State University, University Center, MI, USA
{krahman,ajrulapa}@svsu.edu

Abstract. This paper presents a robust user authentication system by gleaning raw mouse movement data. The data was collected using a publicly available tool called Recording User Input (RUI) from 23 subjects analyzed for three types of mouse actions - Mouse Move, Point-and-Click on Left or Right mouse button, and Drag-and-Drop. Samples are broken down to unit blocks comprising a certain number of actions and from each block seventy-four features are extracted to construct feature vectors. The proposed system was rigorously tested against public benchmark data. Experiment results generated by using the Support Vector Machine (SVM) classifier shows a False Rejection Rate (FRR) of 1.1594 % and a False Acceptance Rate (FAR) of 1.9053 % when the block size was set for 600 actions. After reducing dimensions using Principle Component Analysis (PCA), SVM classifier shows FRR of 1.2081 % and FAR of 2.3604 %. Compared with the existing methods based on mouse movements, our method shows significantly lower error rates, which we opine are viable enough to become an alternate to conventional authentication systems.

Keywords: Biometric · Cyber behavioral biometrics · Mouse dynamics · Person identification · SVM

1 Introduction

One of the preliminary tasks in the field of information security is to make sure that the person who is accessing the system which may contain sensitive and confidential information, is the right person. To ensure so, a person can be classified genuine or intruder by the method of user authentication which in general falls into two categories - (1) to authenticate a person by something he/she possesses such as tokens, ID and (2) to authenticate by something he/she knows, for example, by knowing a password or PIN number. However, there are limitations in these traditional approaches. For instance, tokens or IDs can be lost, stolen or misplaced and a person may forget his PIN number or password. Alternatively, it is possible that an intruder may acquire one's password using automated password cracking tools. To deal with these issues,

S. Foresti and G. Persiano (Eds.): CANS 2016, LNCS 10052, pp. 692–700, 2016.
DOI: 10.1007/978-3-319-48965-0_47

biometrics [1] are introduced to identifies a person by using unique physical or behavioral characteristics that the person possesses.

Although a physical biometric system such as fingerprint, retina, and iris scan provide stronger security, it also requires expensive hardware to record user's biometric data. On the other hand, cyber behavioral biometric such as keystroke or mouse dynamics which are generated naturally when a user interacts in cyberspace; (1) do not require specialized hardware and therefore, is inexpensive and (2) unobtrusive. For these reasons research in these fields has been gaining momentum in recent days.

In this research work, we focused on mouse dynamics that means the characteristics of a user which are collected by analyzing the inputs performed by a pointing device such as mouse. In this system, only the availability of a mouse is required. Based on a user's mouse actions, some features are extracted and stored for every user profile. When the user uses the system again, the system matches his actions with his profile and determines whether it is a genuine user or an intruder.

Contribution of the paper follows:

- 48 new features are proposed and 74 total features has been defined and processed for the experiments. This rich feature set, combined with the data processing and classification methods we adopted, was the key to achieving impressively low FRR of 1.1594 % and FAR of 1.9053 %.
- Performances comparison (see in Sect. 3) between our method and other existing methods has been compiled. The comparison clearly indicates the merits of our system.

The rest of the paper is organized as follows: Sect. 2 describes the proposed system. Section 3 presents experimental results with performances of the proposed system and Sect. 4 describes the contributions, limitations and future plan for improvement.

2 Proposed System Description

The proposed system is divided into three major components. The components are (I) Data Acquisition, Processing, and Segmentation, (II) Feature Extraction and Normalization, and (III) Training and Classification.

2.1 Data Acquisition, Processing, and Segmentation

Mouse data are collected by using a publicly available logging tool named Recording User Input (RUI) [2] where different mouse actions are observed and recorded for 23 volunteers. The dataset contains 284 h of raw mouse data with an average of 45 sessions per user. Users are given with an individual choice of operating environments and applications. Users were asked to use their computer and mouse in a normal, everyday fashion.

For each action (listed below), data are formatted as Elapsed Time (in milliseconds), Action Type, X-Coordinate, and Y-Coordinate. Elapsed time means the time difference in milliseconds between the start time of monitoring the system and the time after the specific action has occurred. Action types are:

(I) Mouse Move, (II) Press Left Button, (III) Release Left Button, (IV) Press Right Button and (V) Release Right Button. X-Coordinate and Y-Coordinate are pixel location values of x and y coordinates of the mouse on the screen respectively. Table 1 shows four sample actions recorded by the tool RUI. Raw mouse data are then processed into three upper level mouse actions: Mouse Move, Point-and-Click on left or right mouse button and Drag-and-Drop.

Table 1. Example of four mouse action instances recorded by the mouse logging tool RUI.

Elapsed time (in ms)	Action	X-coordinate (in pixels)	Y-coordinate (in pixels)
0.33	Moved	204	492
0.338	Moved	206	479
0.354	Pressed Left	206	479
0.394	Released Left	206	479

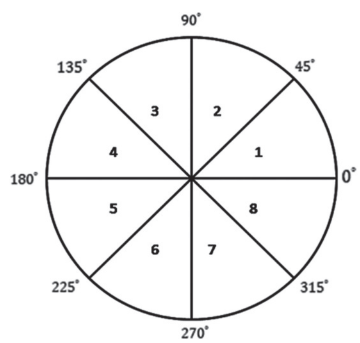

Fig. 1. Direction of mouse movement divided by octants of 45° intervals.

In segmentation step, the processed data is divided into different block sizes based on the number of mouse actions. A block consists of a set of aforementioned mouse actions. Block sizes of 350, 400, 450, 500, 550, and 600 are used. From each block, a set of features are extracted.

2.2 Feature Extraction and Normalization

In this step, features are extracted from the preprocessed dataset. Features are selected in a way that makes the system compact, efficient and at the same time consist of some unique characteristics of an individual.

For each action type, twenty-two features are calculated from each block. These are; Mean and Standard Deviation of time (in milliseconds) to perform a specific type of action in a block, Mean and Standard Deviation of travel distance (in pixels) to perform a specific type of action in a block, Number of a specific type of mouse action (N) in a block, Ratio of number of mouse actions (N) and total number of actions in

Table 2. List of features extracted from each block.

Features	Number of features
Mean of Time	3
Standard Deviation of Time	3
Mean of Travel Distance	3
Standard Deviation of Travel Distance	3
Number of Mouse Actions	3
Ratio of Mouse Action and Total Number of Actions	3
Direction Specific Mean Time	24
Direction Specific Mean Mouse Movement Distance	24
Total Mouse Movement Distance in each direction	8
Total Features	74

block (*NB*), proposed direction specific mean time ($\bar{X}_{tj}^{K}$) and proposed direction specific mean mouse movement distance ($\bar{X}_{dj}^{K}$). Here, direction of the mouse movement is described by octant of 45° intervals with 0° to 360° spans [see in Fig. 1] for every mouse action. Thus, there are 66 features for three mouse action type. The newly proposed features are described below.

Proposed direction specific mean time to perform a specific type of action in a block ($\bar{X}_{tj}^{K}$) is a ratio between total time to perform a type of action in K direction and total time to perform the same type of action throughout the block.

$$\overline{X}_{tj}^{K} = \frac{\sum_{j=1}^{M} X_{tj}^{K}}{\sum_{i=1}^{N} X_{ti}} \tag{1}$$

X_{tj}^{K} is the time to perform an action of $\mathrm{J}(1, 2, \ldots, M)$ samples in $\mathrm{K}(1, 2, \ldots, 8)$ directions, X_{ti} is the time to perform an action of $\mathrm{I}(1, 2, \ldots, N)$ samples.

Proposed direction specific mean mouse movement distance to perform a specific type of action in a block ($\bar{X}_{dj}^{K}$) is a ratio between total travel distance to perform a type of action in K direction and total travel distance to perform the same type of action throughout the block.

$$\overline{X}_{dj}^{K} = \frac{\sum_{j=1}^{M} X_{dj}^{K}}{\sum_{i=1}^{N} X_{di}} \tag{2}$$

Where X_{dj}^{K} is the mouse movement distance of $\mathrm{J}(1, 2, \ldots, M)$ samples in $\mathrm{K}(1, 2, \ldots, 8)$ directions, X_{di} is the mouse movement distance of $\mathrm{I}(1, 2, \ldots, N)$ samples.

Eight more features are also calculated which are the total mouse movement distance in each direction, $\sum_{j=1}^{M} X_{dj}^{K}$ where X_{dj}^{K} is the mouse movement distance of $\mathrm{J}(1, 2, \ldots, M)$ samples in $\mathrm{K}(1, 2, \ldots, 8)$ directions. Therefore, the total number of features is 74 where the total number of proposed features is 48 for three mouse action type. See Table 2 for the full list of features. These features are used to construct a feature vector for each user. The dimension of each feature vector is the number of selected features which is 74. Before classifying, data of the feature vector are normalized in a scale. This helps to avoid attributes in greater numeric ranges overshadowing those in smaller numeric ranges. By doing this, training and testing data will be in the same scale. In this proposed system, data is normalized into the scale of zero to one by using Min-Max Normalization.

2.3 Training and Classification

To analyze how the classifier is checking a genuine user, at first the classifier is trained with a set of randomly selected data for a selected user from the dataset. The training data pattern contains patterns of the legitimate user. The classifier is also trained with imposter patterns labeled with the legitimate patterns. Then the other portions of the dataset which are treated as testing patterns are applied to the classifier. After testing, it is analyzed that how the system is classifying genuine data by examining the predicted label.

In this proposed system, Support Vector Machine (SVM) [3] classifier is used for training and testing purposes. We adopted the classifier SVM since it has been widely used in the field of object recognition, speech recognition, biometrics, image retrieval, image regression etc. It is highly accepted classifiers since it offers a result with good performances. Sometimes it outperformed other classifiers, such as neural network.

In case of SVM, two techniques are applied. One is using original feature vector (with 74 features) and the other is using dimensionally reduced feature vector by applying Principal Component Analysis (PCA) [4]. PCA is a mathematical technique of matching patterns in high dimensions of data. It helps to reduce the dimension of the data, so when the dataset is larger, PCA plays an important role by reducing the dimensions and selecting a subset.

To implement the system using SVM classifier, an open source package LIBSVM [5] is used. The popular choice of Kernel function is Gaussian Radial Basis Function (RBF). Kernel parameters are obtained by applying fivefold cross validation technique. The system applies SVM on original feature space as well as SVM on dimensionally reduced feature space using PCA.

3 Experimental Results and Discussion

The proposed system is implemented in a Windows 7 system with 1.70 GHz Intel Core i3 4005U CPU with 4.00 GB of RAM. Other remaining part of the system such as processing, segmentation, scaling, and classification were performed with MATLAB R2013a.

The proposed system is tested by using a public benchmark data [6, 7]. In the public benchmark dataset, four types of actions are defined which are; (1) Mouse Movement (MM) which means normal mouse movement, (2) Silence which means the time when the mouse does not move, (3) Point and Click (PC) which defines mouse movement which is followed by mouse button press and release, and (4) Drag and Drop (DD) which relates with the combination of mouse actions such as mouse movement, mouse button press and then release sequentially. Before experimenting data for silence action are deducted from the benchmark dataset. Note that from these four actions, three upper level actions are derived as mentioned in Sect. 2.1.

Performance is measured by computing False Acceptance Rate (FAR) and False Rejection Rate (FRR).

3.1 Results of Classification

Experiments are performed on different sizes of blocks (350, 400, 450, 500, 550, and 600 actions) each with 74 features derived from the public dataset. Table 3 shows that among different block sizes of actions, block size of 600 actions provides better result. In case of block size of 600 actions, SVM and SVM (+PCA) show FRR of 1.1594 % and 1.2081 % respectively. Again, for block size of 600 actions, SVM and SVM (+PCA) show FAR of 1.9053 % and 2.3604 % respectively.

Table 3. Performance for different block sizes using SVM and SVM (+PCA).

Block size (number of action)	SVM		SVM (+PCA)	
	FRR (%)	FAR (%)	FRR (%)	FAR (%)
350	1.4631	2.3358	1.5291	2.6496
400	1.3685	2.2234	1.4616	2.5512
450	1.2917	2.2114	1.3746	2.4789
500	1.1902	2.0379	1.3030	2.3574
550	1.1619	2.0020	1.2327	2.3941
600	1.1594	1.9053	1.2081	2.3604

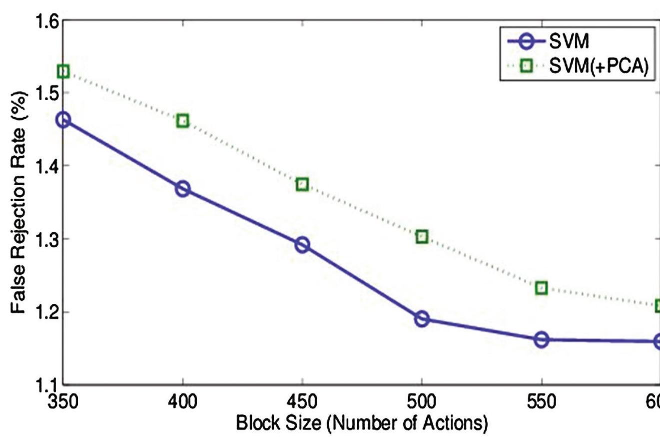

Fig. 2. Comparison of SVM and SVM (+PCA) Classifiers based on FRR.

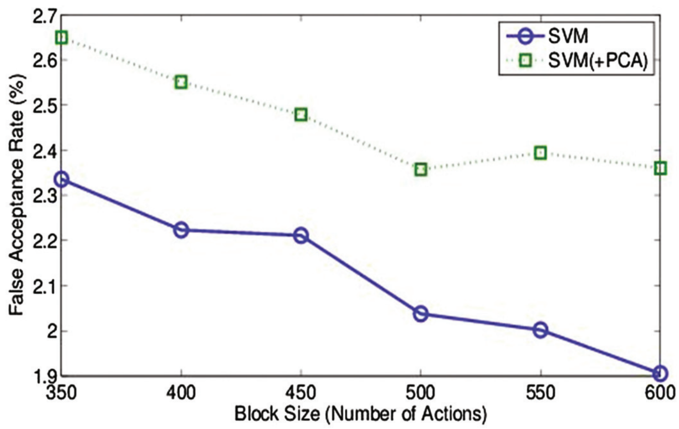

Fig. 3. Comparison of SVM and SVM (+PCA) Classifiers based on FAR.

After studying the performance result for different classification techniques, it is observed that the performance rate of the SVM with original feature space offers better result.

The comparison based on the performance rate of FRR and FAR shown in Figs. 2 and 3 respectively.

3.2 Comparison with Related Works

The results found in our experiments are compared with the results found by Ahmed et al. in [7], which is considered as benchmark in the field of mouse dynamics. Features of an existing system by Ahmed et al. [7] are extracted from the public benchmark dataset and applied to the proposed system. These features are Movement Speed compared to Travelled Distance (MSD) curve, Average Movement Speed per Movement Direction (MDA), Movement Direction Histogram (MDH), Average Movement Speed per Type of Action (ATA), Action Type Histogram (ATH), Travelled Distance Histogram (TDH) and Movement elapsed Time Histogram (MTH). Twelve points are computed through periodic sampling over the MSD curve. In case of TDH, values in the range of 0–100 pixels and 100–200 pixels are used. In case of MTH, values within the range of 0.0–0.5 s, 0.5–1.0 s, and 1.0–1.5 s are collected. In total, the number of features is 39.

For block size of 600 actions, SVM and SVM (+PCA) offer FRR of 1.6001 % and 1.7851 % respectively by using existing set of features proposed in [7] which are higher than FRRs showed by our proposed system with the same set of data and block size. Likewise, for block size of 600 actions, SVM and SVM (+PCA) offer FAR of 2.9798 % and 2.9042 % respectively by using existing features in [7] which are higher than ours. This clearly indicates the merits of our newly proposed features.

Several other researches showed impressive results in recent times. Below we mention the notable works and compare their outcomes with ours.

(1) In the work of Ahmed et al. [7], they offer FRR of 2.4614 % and FAR of 2.4649 %. To gain this performance the number of required actions is 2000 where the actions include point and click, drag and drop, mouse move and silence.
(2) Nakkabi et al. [6] also show FRR of 0.36 % and FAR of 0 for same number of mouse actions. However, the number of mouse action is large and not always practical to play a tile game to use the system.
(3) Pusara and Bordley [8] offered a web based authentication system where decision tree is used as a classifier. It shows good result where false negative rate is 1.75 % and false positive rate is 0.43 %. However, it only consists of eleven users' involvement.
(4) In the works of Muthumari et al. [9] they proposed 6.25 % FRR and 7.25 % FAR using Learning Vector Quantization (LVQ) method.
(5) In their other work [10], Kernel Principle Component Analysis (KPCA) method is used to reduce the dimension of the feature vector and one class support vector machine is used as a classifier which offered 8.25 % FRR and 8.98 % FAR.
(6) In the method of Lakshmipriya et al. [11], holistic and procedural features are used and Nearest Neighbor Algorithm is applied to extract the features. It offers FRR of 7.70 % and FAR of 8.75 %.
(7) In the method of Rahman et al. [12], similarity score method has been used which is based on statistical normal distribution. They found equal error rate (EER) to be 6.7 %.

Compared with the above existing methods, our method shows significantly lower error rates by processing even fewer number of actions (maximum 600 for instance). The works which show lower error rates than ours, suffers from either inadequate population size (such as in [8]) or impractical due to restricted testing environment (see in [6]).

4 Conclusion

In this system, three types of mouse actions: Mouse Move, Point-and-Click on left or right mouse buttons and Drag-and-Drop are obtained. The processed data is divided into blocks where block means a set of specific number of mouse actions. Seventy-four features are extracted from each block to form a feature vector where the number of new features is forty-eight. For each type of mouse action, the features are calculated from mean and standard deviation of travel distance, mean and standard deviation of elapsed time to perform an action, mean number of mouse actions, proposed direction specific mean time of an action and direction specific mean travel distance. The direction of the mouse movement action is described by an octant of 45° intervals. Using these features a person's mouse movement distance and total time to perform an action are described with eight values instead of one direction. The data of the feature vector is normalized into the scale of zero to one. After normalizing the feature vector is applied to classifiers. Support Vector Machine (SVM) with original feature space and Support Vector Machine (SVM) with dimensionally reduced feature space by Principal Component Analysis (PCA) are used in the system. To test the system, public benchmark dataset is used. Performances are measured and analyzed for six different block sizes. After experimenting it is observed that the system provides better performance of the block size of 600. Experiment result shows that, in case of original feature space SVM offers 1.1594 % FRR and 1.9053 % FAR. In case of dimensionally reduced feature space by PCA, SVM classifier offers 1.2081 % FRR and 2.3604 % FAR.

This system did not consider some actions due to inadequacy of benchmark dataset. In future, more types of actions such as Double Click, Mouse Wheel etc., will be considered. A larger dataset is expected to be gathered and tested against our system. With some impressive initial results, we believe this system could be used with other conventional authentication systems to build a multi-modal authentication system.

Acknowledgements. This work was done under the assistance of Ministry of Posts, Telecommunications and Information Technology Fellowship given by the Information and Communication Technology division of Ministry of Posts, Telecommunications and Information Technology, Government of the People's Republic of Bangladesh.

References

1. Jain, A.K., Pankanti, S.: Biometric identification. Commun. ACM **43**, 91–98 (2000)
2. Kukreja, U., Stevenson, W.E., Ritter, F.E.: RUI: recording user input from interfaces under Windows and Mac OS X. Behav. Res. Methods **38**(4), 656–659 (2011)
3. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines: And Other Kernel-based Learning Methods. Cambridge University Press, New York (2000)
4. Jolliffe, I.: Principal Component Analysis. Springer, New York (1986)
5. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**, 1–27 (2011). Article No. 27
6. Nakkabi, Y., Traoré, I., Ahmed, A.A.E.: Improving mouse dynamics biometric performance using variance reduction via extractors with separate features. Trans. Sys. Man Cyber. Part A **40**(6), 1345–1353 (2010)
7. Ahmed, A.A.E., Traoré, I.: A new biometric technology based on mouse dynamics. IEEE Trans. Dependable Sec. Comput. **4**(3), 165–179 (2007)
8. Pusara, M., Brodley, C.E.: User re-authentication via mouse movements. In: ACM Workshop on Visualization and Data Mining for Computer Security. ACM Press (2004)
9. Muthumari, R.S., Pepsi, M.B.B.: Mouse gesture based authentication using machine learning algorithm. In: International Conference on Advanced Communication Control and Computing Technologies (2014)
10. Muthumari, G., Shenbagaraj, R., Pepsi, M.B.B.: Authentication of user based on mouse-behavior data using classification. In: IEEE International Conference on Innovations in Engineering and Technology (ICIETŠ) (2014)
11. Lakshmipriya, D., Balakrishnan, J.R.: Holistic and procedural features for authenticating users **16**, 98–101 (2014)
12. Rahman, K.A., Moormann, R., Dierich, D., Hossain, M.: Continuous user verification via mouse activities. In: Dziech, A., et al. (eds.) MCSS 2015. CCIS, vol. 566, pp. 170–181. Springer, Heidelberg (2015). doi:10.1007/978-3-319-26404-2_14