Study on the BeiHang Keystroke Dynamics Database

Yilin Li, Baochang Zhang+, Yao Cao
Science and Technology on Aircraft Control Laboratory
School of Automation Science and Electrical Engineering
Beihang University
Beijing,100191,China
+correspondence:bczhang@buaa.edu.cn

Sanqiang Zhao, Yongsheng Gao
IIIS, Griffith University
Brisbane, Australia

Jianzhuang Liu
Shenzhen Institutes of Advanced Technology
Chinese Academy of Sciences/The Chinese University of Hong Kong

Abstract

This paper introduces a new BeiHang (BH) Keystroke Dynamics Database for testing and evaluation of biometric approaches. Different from the existing keystroke dynamics researches which solely rely on laboratory experiments, the developed database is collected from a real commercialized system and thus is more comprehensive and more faithful to human behavior. Moreover, our database comes with ready-to-use benchmark results of three keystroke dynamics methods, Nearest Neighbor classifier, Gaussian Model and One-Class Support Vector Machine. Both the database and benchmark results are open to the public and provide a significant experimental platform for international researchers in the keystroke dynamics area.

1. Introduction

1.1. Background

Internet has greatly changed our lives, making our work and daily lives more convenient. However, it also brings us a big concern in information security. Our private information faces more serious intrusion than before. Currently, password is extensively used to prevent user accounts from being intruded. However, too many methods can be used to decipher password, and once a password is decoded, it will probably lead to a significant financial loss of the user. Such crimes on internet are able to cause a wide range of serious damages, and yet difficult to be prevented. Therefore, to cope with such problems, we urgently need a more reliable way to protect our privacy.

1.2. Keystroke Dynamics

Keystroke dynamics utilizes the rhythm and manner in which an individual types characters on a keyboard. The original keystroke data contain the time of the key press and release (shown in Figure 1), from which two kinds of features are extracted, *flight time* and *dwelling time*. The flight time is defined as the time difference between one key release and the following key press. The dwelling time is the time difference between the press and release of one key.

Keystroke dynamics is still an on-going research topic [1–3]. Researchers had proposed many methods for keystroke dynamics [4–14]. The first research paper on keystroke dynamics [4] was done by Rand Corporation and published in 1980, which proves that professional typists have distinguishable ”styles” of typing as measured by patterns of expected times to type certain digraphs. In [5], Young and Hammon conducted an experiment to build a template from an individual’s typing manner. Monrose and Rubin [6] constructed an identification system based on template matching and Bayesian likelihood models. Hu et al. [7]proposed a K-Nearest neighbor based authentication method, which focuses on improving the efficiency while maintaining the performance as other methods. In [8], researchers presented a method based on Hidden Markov Model, which achieved a reasonable performance. Some researchers applied neural network to keystroke dynamics [9, 10]. In [11], researchers developed a pressure-based user authentication system, and the discrete time signal is transformed into frequency domain by using FFT. In [12–14], SVM was studied for keyboard dynamics, whose performance and efficiency is better than those based on neural networks.

Figure 1. The dwelling time and flight time of keystroke dynamics.

1.3. Keystroke Database

Similar to other biometrics techniques, databases are very important to researches in the field of keystroke dynamics. However, to the best of our knowledge, there is no open keystroke dynamics database from a real commercialized system, while the benchmark databases in other biometrics problems are generally ample. For example, the FERET and FRGC databases are commonly used in face recognition [15, 16]. In Palmprint recognition, the PolyU dataset [17]is the open platform for comparative evaluation of different algorithms.

The aim of database is to allow different researchers to test their own algorithms based on the same dataset. In the same experimental environment, the comparison between different algorithms would be more reasonable. In [4], researches on the keystroke dynamics were based on a long text. Several people were asked to type a same paragraph of words. These experiments can prove the uniqueness of keystroke behavior; however, they could not be used for the practical application.

It should be noted that most of the previous experiments collected the test samples in pure laboratory context [14]. Therefore those datasets cannot represent a more general situation. So creating an open and comprehensive database from a commercialized system for keystroke dynamics becomes a very important issue.

The rest of the paper is organized as follows. In Section 2, we describe the commercialized keystroke dynamics system used to establish the proposed database. The details of the BH Keystroke Dynamics database and benchmark experiments are introduced in Sections 3 and 4, respectively. Section 5 gives some discussions with future work.

2. The Keystroke Dynamics system

In this section, we introduce the preprocessing, feature reduction and benchmark recognition methods used in the commercialized keystroke dynamics system.

2.1. Preprocessing Procedure

Supposed a password is represented by the following vector:

$$P_1, R_1, P_2, R_2, ..., P_n, R_n \tag{1}$$

where P_i and R_i represent the press and release time of the $i-th$ keystroke of a password. The elements of the feature vector extracted from original keystroke information are classified into two categories: *dwelling time* and *flight time*. The *dwelling time* is calculated by $R_i - P_i$, and the *flight time* is calculated by $P_i - R_{i-1}, P_i - P_{i-1}, R_i - R_{i-1}$.

Therefore, the extracted feature from the original vector is represented as:

$$F = (R_1 - P_1, P_2 - R_1, P_2 - P_1, R_2 - R_1, R_2 - P_2, ..., P_n - R_{n-1}, P_n - P_{n-1}, R_n - R_{n-1}, R_n - P_n) \tag{2}$$

The number of the registration samples collected in the training procedure is 4 or 5, which is not enough to get an effective model. So we augment the training set by using the mean sample of each pair of samples.

2.2. Feature reduction by variance

In the registration procedure, to collect the training dataset, some key press or release events may suffer from unpredicted disturbance, and so it may bring noise to the training process. Therefore, feature reduction is a very important step towards a system with a high performance. In this study, the variance is considered as the selection criterion to select features which can eliminate the noise from training dataset. The feature extracted in the preprocessing procedure is represented by F. For each element F_i , we calculate its variance Var_i , and a threshold (*e.g.* $Var_i < 0.1$) is used to decide whether a certain feature can be reserved.

2.3. Benchmark Recognition Methods

Support Vector Machine (SVM) is an efficient classifier in machine learning. One Class SVM (OC-SVM) as a variant of SVM can train a classification model from one class without negative samples. OC-SVM can also be viewed as a regular two-class SVM where all the training data lie in the first class. The keystroke dynamics is basically a single-class problem, while in this paper we exploit the nonlinear version of OC-SVM algorithm which maps the input data into a high dimensional feature space (via a kernel) as our first benchmark algorithm.

We also exploit Nearest Neighbor (NN) classifier as our second benchmark algorithm on the collected database. In the NN classifier, the Euclidean distance is used as the measure with given threshold for final classification.

In probability theory, the Gaussian distribution is a continuous probability distribution that is often used as a first approximation to describe real-valued random variables that tend to cluster around a single mean value. We suppose that the keystroke data follow Gaussian distribution, therefore we can build a Gaussian model using the training data. This model is used as our third benchmark algorithm for keystroke dynamics authentication in the format of the following Gaussian probability function:

$$p(x) = \frac{1}{2\pi^{d/2}|\Sigma|^{1/2}} exp\{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\} \quad (3)$$

where x is the test sample with Σ, μ as the covariance matrix and the mean vector of the Gaussian model, respectively.

3. The BH Keystroke Dynamics Database

We have collected a database by using a commercialized system on keystroke dynamics. It can be used by different researchers to test their algorithms and can eventually boost the development of keystroke dynamics.

3.1. Data collection

Generally, several kinds of methods are used to capture the keystroke event. The kernel based method is very powerful and can gain any information typed on keyboard as it reaches the operation system. However, the synchronization problem is not well solved in multi-kernel computers, though it is really effective and difficult to be detected by user-mode applications in the single-kernel computers. Another widely used method is based on the HOOK keyboard APIs. It includes a series of functions which reveal the status of key press or release event. However, the HOOK function is generally based on a lot of APIs which can lead to an increase in CPU usage. There are also some methods based on web browsing, but they are not secure as other methods in keystroke event detection. To deal with the above problems, we design an instance stream to capture the key press and release events as shown in Figure 2. The proposed method is effective since the instance stream can be a complement to the traditional HOOK function.

A commercialized system following the above work principle was deployed to different environments, such as cybercafe and laboratories. It involved a variety of individuals whose registration and log-in keystroke information was collected. Each user was asked to type in his/her username for one time and his/her password for 4 or 5 times in order to create a new account. Some false data from users' misuse of the system were included in the primary dataset. By using a filter, those error data were abandoned and finally we got our BH database.

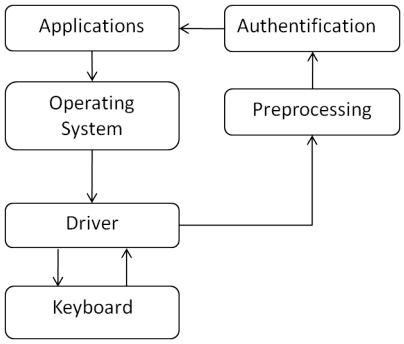

Figure 2. Flowchart of the keystroke dynamics system.

3.2. Description of Database

The BH database includes 2057 test samples and 556 training samples from 117 subjects. The whole database is divided into two subsets, Dataset A and Dataset B, collected from two different environments. DatasetA was collected in cybercafe environment. The commercial system was embedded into the login system of an online application. DatasetB was collected online. The commercial system is open to the public. Each subject includes registration data from genuine users which is used as training samples, log-in data from genuine users and log-in data from intruders. The three sub-directories are named as "training set", "positive set" and "negative set", respectively. All the data are stored in text format.

4. Expermental results

In this section, we present the benchmark experimental results of three kinds of keystroke dynamics methods conducted on the BH database.

4.1. Evaluation Criteria

In the experiment, we use False Positive Rate and True Positive Rate for evaluation. The False Positive Rate is the percentage of intruders who can enter the account by imitating the typing manner of genuine users. The True Positive Rate is the percentage of genuine users who can successfully login the system with right keystroke manner. By changing the threshold in the classification procedure, we get a series of True Positive Rate and False Positive Rate, and then we use these results to draw a ROC curve. The ROC curve is used for evaluation of an algorithm. Besides these two indicators and the ROC curve, we also use the Equal Error Rate to compare the performance of different methods. The Equal Error Rate is the percentage where the False Positive Rate equals the False Negative Rate.

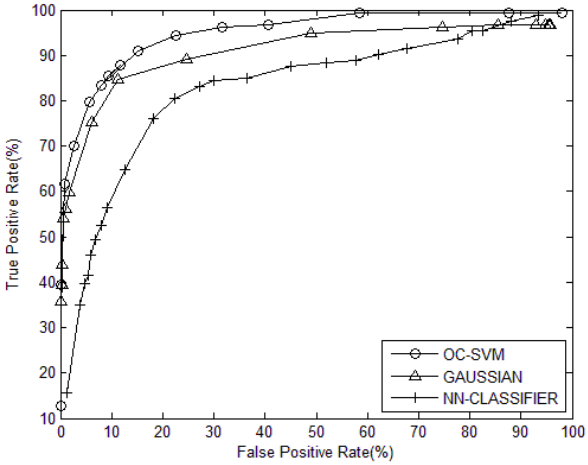

Figure 3. Comparison of the ROC results of the three methods based on BH Dataset A.

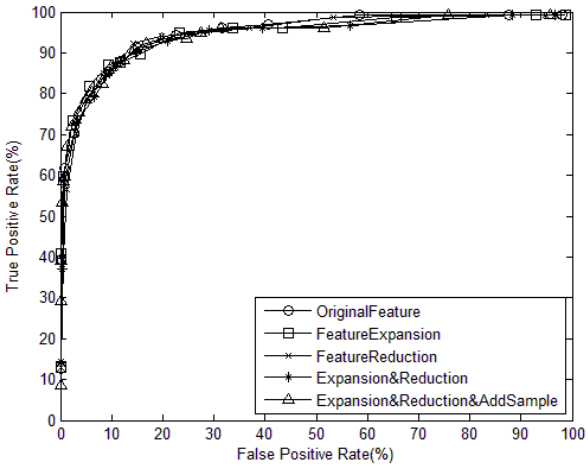

Figure 4. Comparison of different prepossessing methods based on BH Dataset A.

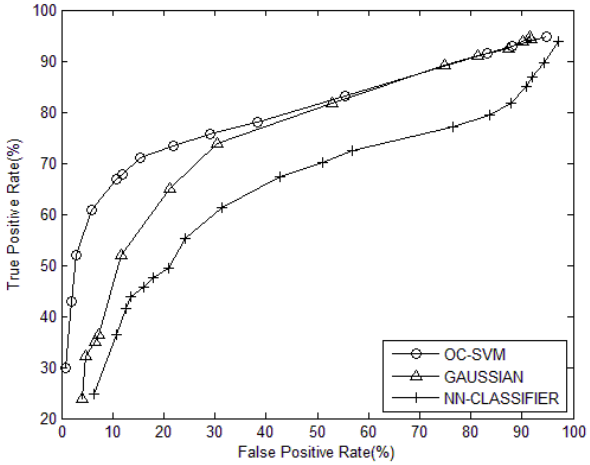

Figure 5. Comparison of the ROC results of the three methods based on BH Dataset B.

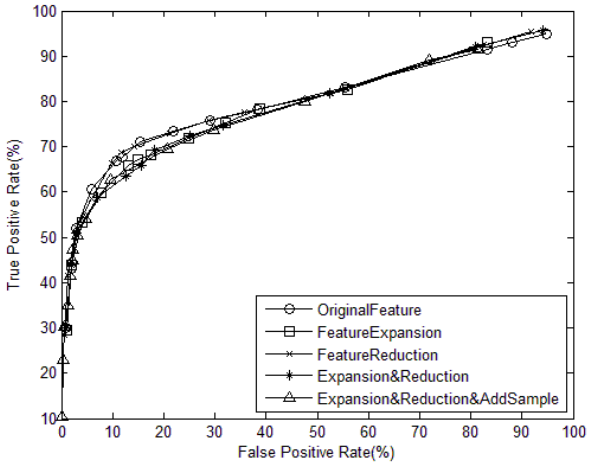

Figure 6. Comparison of different prepossessing methods based on BH Dataset B.

4.2. Experimental Result

In Figure 3 and 4, we display the experimental results of OC-SVM, Gaussian Model and NN classifier obtained from BH Dataset A and Dataset B, respectively. It can be seen from the ROC results on the figures that the performance of OC-SVM is better than the Gaussian Model and NN classifier.

We also have tested different methods which are discussed in section 2.1 and 2.2 in the preprocessing procedure such as feature reduction and feature expansion based on the same classify method, OC-SVM. The results of this part of experiment are shown in Figure 5 and 6.

The experimental results of the Equal Error Rate is shown in Table 1. This to compare the performance of different methods. A similar conclusion as in Figure 3-6 can be observed.

5. Conclusion and Future Work

The BeiHang keystroke dynamics database is open to public use. Researchers who are interested in this database may send a request to the corresponding author. It can be found at the following two Chinese websites:

`http://www.microdone.cn/ballet_login.php`
`http://www.u1ge.com/help/passdancer.`

Our future work will focus on collecting a much larger database. The proposed keystroke dynamics system has already been commercialized, but we plan to test other new methods as well.

EER/%	SVM-basic	SVM-expansion	SVM-reduction	SVM-expansion-reduction	SVM-expansion-reduction-addsample	Gaussian	NN-classifier
DataBase A	12.1886	12.0107	11.8327	11.8327	12.0107	14.1459	20.7295
DataBase B	25.3559	26.9840	26.9840	26.6014	26.9840	28.2028	49.9110

Table 1. Equal Error Rate result based on different preprocessing methods.

Acknowledgements

This work was supported in part by the Natural Science Foundation of China, under Contracts 60903065 and 61039003, in part by the Ph.D. Programs Foundation of Ministry of Education of China, under Grant 20091102120001, in part by the Fundamental Research Funds for the Central Universities, and in part by Shenzhen Key Laboratory for Computer Vision and Pattern Recognition.

References

[1] Fabian Monrose, Michael K. Reiter, and Susanne Wetzel. Password hardening based on keystroke dynamics. In *International Journal of Information Security*, volume 1, pages 69–83, 2002.

[2] R Joyce and G Gupta. Identity authentication based on keystroke latencies. *Communications of the ACM*, 33(2), 1990.

[3] MS Obaidat and D T Macchairolo. An on-line neural network system for computer access security. *IEEE Trans.Industrial Electronics*, 40:235–241, 1993.

[4] R.Gaines, W.Lisowski, S.Press, and N. Shapiro. Authentication by keystroke timing: some preliminary results. Technical report, Rand Corporation, 1980.

[5] J.R. Young and R.W. Hammon. Method and apparatus for verifying an individual's identity. *Patent No. 4,805,222, U.S. Patent and Trade, Mark Office*, 1989.

[6] F. Monrose and A.D. Rubin. Keystroke dynamics as a biometric for authentication. *Future Generation Computer Systems*, pages 351–359, 1980.

[7] J. Hu, D. Gingrich, and A. Sentosa. A k-nearest neighbor approach for user authentication through biometric keystroke dynamics. In *Proceedings of IEEE International Conference on Communications*, pages 1556–1560, 2008.

[8] Ricardo N.Rodrigues *et al*. Biometric access control through numerical keyboards based on keystroke dynamics. In *Proceedings of the International Conference on Biometrics*, pages 640–646, 2005.

[9] Daw-Tung Lin, Chung-Hua, and Hsinchu. Computer-access authentication with neural network based keystroke identity verification. *Internaltional Congerence on Neural Networks*, 1997.

[10] M.S. Obaidat and D.T. Macchairolo. A multilayer neural network system for computer access security. In *IEEE Transactions on Systems, Man and Cybernetics*, pages 806–813, 1994.

[11] Chen Change Loy, Dr. Weng Kin Lai, and Dr. Chee Peng Lim. Development of a pressure-based typing biometrics user authentication system. *ASEAN Virtual Instrumentation Applications Contest Submission*, 2005.

[12] E. Yu and S. Cho. Keystroke dynamics identity verification: its problems and practical solutions. *Computers and Security*, 23:428–440, 2004.

[13] Y. Sang, H. Shen, and P. Fan. Novel impostors detection in keystroke dynamics by support vector machine. *Parallel and Distributed Computing:Applications and Technologies*, pages 666–669, 2005.

[14] Romain Giot, Mohamad El-Abed, and Christophe Rosenberger. Greyc keystroke: a benchmark for keystroke dynamics biometric systems. In *IEEE International Conference on Biometrics: Theory, Applications and Systems*, 2009.

[15] P. J. Phillips, H. Moon, P. J. Rauss, and S. Rizvi. The feret evaluation methodology for face recognition algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(10), October 2000.

[16] Hoffman, J. Marques, J. Min, and W. Worek. Overview of the face recognition grand challenge. *Proc. Computer Vision and. Pattern Recognition*, 2005.

[17] Biometric Research Centre at The Hong Kong Polytechnic University, http://www.comp.polyu.edu.hk/ biometrics/. *PolyU Palmprint Database*.