Hindawi Publishing Corporation
ISRN Signal Processing
Volume 2013, Article ID 565183, 7 pages
http://dx.doi.org/10.1155/2013/565183

Research Article

Keystroke Dynamics User Authentication Based on Gaussian Mixture Model and Deep Belief Nets

Yunbin Deng and Yu Zhong

6 New England Executive Park, Burlington, MA 01803, USA

Correspondence should be addressed to Yunbin Deng; yunbindeng@gmail.com

Received 30 June 2013; Accepted 18 August 2013

Academic Editors: S. Kwong and K. Wang

User authentication using keystroke dynamics offers many advances in the domain of cyber security, including no extra hardware cost, continuous monitoring, and nonintrusiveness. Many algorithms have been proposed in the literature. Here, we introduce two new algorithms to the domain: the Gaussian mixture model with the universal background model (GMM-UBM) and the deep belief nets (DBN). Unlike most existing approaches, which only use genuine users' data at training time, these two generative model-based approaches leverage data from background users to enhance the model's discriminative capability without seeing the imposter's data at training time. These two new algorithms make no assumption about the underlying probability distribution and are fast for training and testing. They can also be extended to free text use cases. Evaluations on the CMU keystroke dynamics benchmark dataset show over 58% reduction in the equal error rate over the best published approaches.

1. Introduction

With the ever increasing demand for more secure access control in many of today's security applications, traditional methods fail to keep up with the challenges because pins, tokens, and passwords are too many to remember. Even carefully crafted user name and password can be hacked, which compromises the system security. On the other hand, biometrics [1–5] based on "who" the person is or "how" the person acts, as compared with what the person has (key) and knows (password), presents a significant security advancement to meet these new challenges. Among them, keystroke dynamics [6] provides a natural choice for secure "password-free" computer access with no additional hardware required. Keystroke dynamics refers to the habitual patterns or rhythms an individual exhibits while typing on a keyboard input device. These rhythms and patterns of tapping are idiosyncratic, [7] the same way as handwritings or signatures are, due to their similar governing neurophysiological mechanisms. Back in the 19th century, telegraph operators could recognize each other based on one's specific tapping style [8]. Recently, it is shown that typing text can be deciphered simply based on the sound of key typing [9]. As such, it is believed that the keystroke dynamics contains enough information to be a good biometrics to ascertain a user at the keyboard.

Compared with other biometrics, keystroke biometrics has additional attractiveness for its user-friendliness and nonintrusiveness. Keystroke dynamics data can be collected without a user's awareness. Continuous authentication is possible using keystroke dynamics just as a mere consequence of using the keyboard. Unlike many other biometrics, the timing information of keystrokes can be collected using software only without additional hardware. In summary, keystroke dynamics biometrics enables an emerging cost-effective, user-friendly, and continuous user authentication modality.

Although keystroke dynamics is governed by a person's neurophysiological pathway to be highly individualistic, it can also be influenced by his or her physical and psychological state. As a "behavioral" biometrics [10], keystroke dynamics exhibits instabilities due to transient factors such as fatigue, emotions, stress, and drowsiness. It also depends on some external factors, such as the specific keyboard device used, possibly due to different layout of the keys. As such, keystroke data need to be collected at multiple sessions to assess the robustness of various approaches. It was shown that the frequently used word tends to have better typing timing consistency [6]. Our experiments show that accurate keystroke biometrics can be achieved with a single word, and thus keystroke has the potential to be a highly accurate biometric modality with sufficient well-chosen testing data.

Keystroke biometrics can use “static text,” where a prespecified fixed text, such as a password, is analyzed at a certain time, for example, during the log-on process. For more secure applications, “free text” with arbitrary input text and language should be used to continuously authenticate a user. Although this work is focused on a “static text” task, the proposed algorithms can be easily extended to the “free text” real application domain.

The rest of the paper is organized as follows. Section 2 gives a short review of the current state of keystroke biometric techniques. Section 3 details our new approaches to the problem of accurate and discriminative keystroke dynamics model. Section 4 describes our user verification experiments and performance of the proposed algorithms. Section 5 discusses the future work and extension of the proposed approaches to “free text” applications.

2. The Literature Survey

Recently, keystroke dynamics has become an active research area due to the increasing concerns of cyber security and access control. Some researchers have made their datasets available to the public, including works of Allen, Bello, Giot, Jugurta, and Maxion [11, 13–16]. Most of the existing datasets are based on “static text,” with only one exception of the BioChaves dataset, which contains both static and free text [16]. Compared with most mature biometrics, keystroke dynamics is still at its very early stage, and various existing public keystroke datasets containing subject only range from over a dozen to over one hundred.

The most common keystroke dynamics features are based on the timing information of the key down/hold/up events, although some custom commercial keyboards can collect pressure information. The hold time or the dwell time of individual keys and the latency between two keys, that is, the time interval between the release of a key and the pressing of the next key, also called the flight time, are typically exploited. “Digraphs,” which are the time latencies between two successive key down presses, are commonly used. “Trigraphs,” which are the time latencies between every three consecutive key down presses, and, similarly, n-graphs have been investigated as well. In a keystroke analysis using free text, Sim and Janakiraman [17] investigated the effectiveness of digraphs and more generally n-graphs for free text keystroke biometrics, and they concluded that n-graphs are discriminate only when they are word-specific.

In addition, the total durations taken to type some certain string can be used as features. The relative order of duration times for different n-graphs was found to be more robust to the variations than the absolute timing [18, 19]. The relative feature, when combined with absolute timing features, improved the authentication performance under free text scenario. Furthermore, derived features, including first- and second-order statistics and entropy of the basic lower-level features, are investigated [20].

The use of keystroke dynamics for verification and identification has a long history and can be dated back to the 1970s [21, 22]. Since then, numerous template-matching and machine-learning algorithms have been reported to tackle the classification problem. These approaches can be broadly classified into four categories: statistical method based on distance metrics, neural networks, statistical machine-learning methods, and many other algorithms [23].

The first category uses the first- and second-order statistics of the basic feature and applied various distance metrics and hypothesis testing. For example, Gaines et al. [24] did a preliminary study on keystroke-dynamics-based authentication using t-test on digraph features. Monrose and Rubin [6] later extracted keystroke features using the mean and variances of digraphs and trigraphs. Using the Euclidean distance metric with the Bayesian-like classifiers, they reported a correct identification rate of 92% for their dataset. Different distance metrics, such as the Euclidean distance [6, 25], the Mahalanobis distance [25, 26], and the Manhattan distance [27, 28], were explored. Recently, we proposed a new distance metric to combine the fundamentals of both the Mahalanobis distance and the Manhattan distance metrics [12].

Various artificial neural networks (ANNs) have been applied to the keystroke classification problem, including perceptron, backpropagation neural network, and Art-2 neural network [23, 29, 30]. Neural networks are well known to be capable of learning nonlinear models of data. However, they often suffer slow speed of model training, hand selection of model architecture and tuning of parameters, and poor generalization capabilities.

Statistical machine-learning algorithms, ranging from the simple K-nearest neighbors (KNN) classifiers [26] to the Bayesian classifiers [6], and support vector machines (SVMs) [31] have been applied to the keystroke classification. The SVMs have been shown to work well under both identification and verification tasks. Compared with ANN approaches, SVMs have fewer parameters to tune and can be highly efficient in both training and testing.

The long history of keystroke research has also resulted in other various approaches, including K-means methods [32], fuzzy logic [29], fuzzy ARTMAP, histogram equalization of time intervals, Gaussian mixture model (GMM) [33], hidden Markov model (HMM), and genetic algorithms.

Various studies have reported a wide range of performances as most studies used their own datasets. To address this issue, Killourhy and Maxion collected and published a keystroke dynamics benchmark dataset [11]. Furthermore, they evaluated fourteen existing keystroke dynamics algorithms on this dataset, including neural networks [26], K-means [32], fuzzy logic [29], KNN, outlier elimination [29], and SVMs [31]. Distance metrics including the Euclidean distance [25], the Manhattan distance [27, 28], and the Mahalanobis distance [25, 26] were used. This keystroke dataset with the evaluation methodology and the performances of the state-of-the-art algorithms provides a good benchmark to objectively assess progress of new keystroke biometric algorithms.

3. New Keystroke Modeling Approaches

Most existing studies on keystroke authentication only use genuine users' data at training time to build a model for each genuine user and apply a user-specific threshold at testing time for decision making on unforeseen data. The ideal of GMM-UBM is to take advantage of large amount of data from many subjects to enhance discriminative capability over the GMM trained on genuine user alone. Although the imposters data were never seen and were not used in UBM training, the GMM-UBM approach shows great success in the state- of-the-art speaker verification system [34]. Here, we apply this method to the domain of keystroke authentication.

Although many purely discriminative model approaches exist, such as ANNs and SVMs, models trained on large amount of background users, without access to real imposter's data at training time, do not guarantee better performance to unforeseen imposters. Recently, DBN was proposed in the machine-learning community as a generative-discriminative hybrid approach [35]. The generative step training grants the model with good generalization capabilities to unforeseen test data, while the discriminate fine-tune step endows the model with super classification accuracy. It has achieved better performance than those of ANNs and SVMs in many well-defined tasks, including hand-writing digits recognition, speech recognition, and speaker identification. Here, we apply the DBN modeling approach to the problem of keystroke.

The following subsections detail the basic theory of these two new approaches.

3.1. The Gaussian Mixture Model with the Universal Background Model (GMM-UBM)

3.1.1. GMM. The Gaussian mixture model was widely used in many statistical modeling tasks. It is a parametric model in the sense that it is parameterized by mean vectors and covariance matrixes of the Gaussian distributions and weights of all of the Gaussian components. It is a nonparametric model in the sense that the real distribution of the data can be unknown. In theory, it can be shown that, with sufficient number of mixtures, the GMM can approximate arbitrary probability distribution. However, the higher number of mixtures required more training data to achieve a well-trained model. In practice, the number of mixtures is determined by the amount of the training data, the complexity of the real distribution, and the computation capacity the system can handle.

A GMM is a weighed sum of M multivariate Gaussian functions [1]. The probability of a feature vector under the GMM is given by

$$p(x \mid \lambda) = \sum_{i=1}^{M} p_i b_i(x), \quad (1)$$

where x is a D-dimensional feature vector, $\lambda = \{p_i, \mu_i, \Sigma_i\}$ is the model parameter, p_i is the mixture weights for the multivariant Gaussian component densities $b_i(x)$, and μ_i, Σ_i are the mean vector and the covariance matrix for the multi-variant normal distribution. The covariance matrixes are typically assumed to be diagonal to dramatically reduce the number of parameters that need to be estimated. These parameters can be trained from data through a maximum-likelihood estimation principle, implemented by the EM (expectation maximization) algorithm. We performed incremental GMM model training procedure; that is, start with a single-mixture Gaussian model and train its parameter from the data. The single-mixture Gaussian model is then split into a two mixture Gaussian models, and its parameters are reestimated from the training data using EM algorithm. This process repeats till the final desired number of mixtures is achieved. An important parameter in the GMM is the variance floor. When a certain mixture of the Gaussian model has little sample in the training data, the estimated variance will be very small; thus, it is not a good estimate of the true variance. In this case, we use a floor number to replace the estimated variance, and this grants the model with better generalization capability.

3.1.2. GMM-UBM. Existing works on applying GMM to keystroke authentication are concerned with training a GMM for each genuine user. At testing time, a keystroke feature is evaluated against the genuine user's GMM, and a threshold is applied to the likelihood of the feature vector to make the decision [33].

The idea of GMM-UBM is to train another GMM from a large pool of the so-called background subjects (except for the genuine user and the testing subjects), in addition to a GMM for each genuine subject. When the background subject pool is large enough, the UBM will have a good chance to reasonably represent an imposter's data. Thus, the imposter can have a relatively high likelihood score under UBM, as compared with the genuine user's GMM. On the other hand, as the UBM is trained from a large pool of subjects, it is a poor model for the genuine user, as compared with the genuine user's GMM, which is only trained from the genuine user. Thus, the genuine user's data have a relatively high score on his/her own model as compared with the UBM. A likelihood ratio test can then be performed based on scores from these two models to make the authentication decision.

3.2. Deep Belief Nets (DBN).

The deep belief nets are probabilistic generative models that are composed of multiple layers of hidden variables. The hidden variables typically have binary values and are called feature detectors. These hidden layers can be trained one layer at a time, with the output of the lower-level layer serving as an input to the higher-level layer. The idea is to build a hierarchical generative model, so that each higher-level layer captures more complex nonlinear features in the data. These pretrained generative models then collapse and serve as an initialized ANN for further discriminative parameter fine tuning. The pretraining of a generative model is important for the generalization capability of the final model. It also facilitates the fine tuning of the ANN, as it is well known that ANN is sensitive to the model parameter initialization and can easily fall to local optimal. The DBN pretraining not only avoids the random initialization of ANN parameters but also significantly speeds up the ANN training process.

3.2.1. Pretraining of RBMs. Specifically, the first step of DBN training performs a layer-wise unsupervised pretraining of the restricted Boltzmann machines (RBMs). An RBM is one type of the Markov random field that has two layers: the visible layer and the hidden layer. The units in the visible layers (v) are connected to all units in the hidden layer (h) with associated weights W. There is no connection within each layer. The units in the visual layer can be real value, integer, or binary depending on the type of input data. The hidden units are typically binary stochastic variables; that is, $h \in \{0, 1\}$. The Gaussian RBM is chosen for the first layer of RMB to model the real value of keystroke timing features. The energy of the state $\{v, h\}$ is defined as [36]

$$E(v, h; \theta) = \sum_{i=1}^{D} \frac{(v_i - b_i)^2}{2\sigma_i^2} - \sum_{i=1}^{D} \sum_{j=1}^{F} W_{ij} \frac{v_i h_j}{\sigma_i} - \sum_{j=1}^{F} a_j h_j, \quad (2)$$

where $\theta = \{W, a, b, \sigma\}$ are parameters specifying the RBM. D is the number of input units, which is equal to the keystroke feature dimension. F is a user-defined parameter specifying the number of hidden units. a is a weight vector for the hidden units, while b and σ are parameters for the input layer. The binary output of the first layer Gaussian RBM further serves as an input for higher-level RBMs to capture more complex nonlinear structures embedded in the data. This process is also known as automatic feature engineering.

Higher-level RBMs in the hierarchical generative model are all defined as binary RBMs; that is, both the visible and the hidden layers contain only binary units. Their energy functions are defined as

$$\begin{aligned} E(v, h; \theta) &= -v^T W h - b^T v - a^T h \\ &= -\sum_{i=1}^{D} \sum_{j=1}^{F} W_{ij} v_i h_j - \sum_{i=1}^{D} b_i v_i - \sum_{j=1}^{F} a_j h_j. \end{aligned} \quad (3)$$

The joint distribution of visible and hidden units is defined by

$$P(v, h; \theta) = \frac{\exp(-E(v, h; \theta))}{z(\theta)}, \quad (4)$$

where $z(\theta)$ is the normalization factor, known as the partition function, and can be defined as

$$z(\theta) = \sum_{v} \sum_{h} \exp(-E(v, h; \theta)). \quad (5)$$

The likelihood of the training data is then specified as

$$P(v; \theta) = \frac{\sum_h \exp(-E(v, h; \theta))}{z(\theta)}. \quad (6)$$

However, the exact layer-wise maximum likelihood training of the RBM is intractable as the computation takes time that is exponential to the dimensions of D and F. The approximate solution is provided by a technique known as "contrastive divergence" [37].

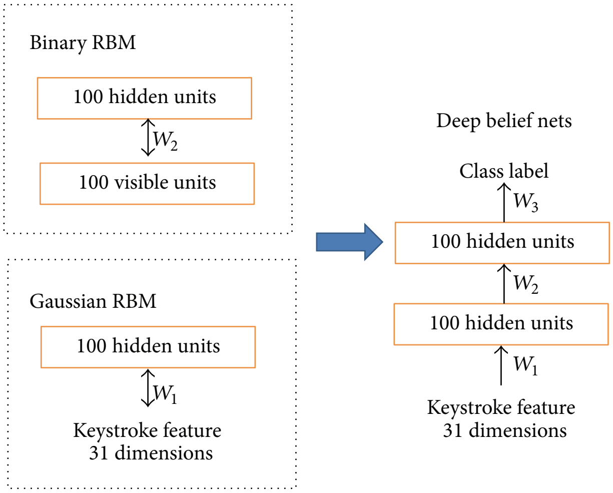


FIGURE 1: Training of DBN for keystroke dynamics authentication in two steps. Left: unsupervised training of RBMs. Right: converting RBMs into DBN.

3.2.2. Fine Tuning of DBN. The output of the unsupervised pretraining step is decks of RBMs, which can be stacked together and can be added with a final classification layer to form an initialized ANN. This is conceptually illustrated in Figure 1, where two RBMs collapse by sharing the middle units and a final layer is added to perform keystroke classification. The parameter of the final classification layer can be trained the same way as training a typical ANN with backpropagation.

4. Experiments

4.1. Dataset and Experimental Setup. We evaluated the proposed keystroke biometric algorithms using the CMU keystroke dynamics benchmark dataset [11] because it came with the performance numbers of a range of existing keystroke dynamics algorithms for objective comparisons.

The CMU benchmark dataset contains keystroke dynamics consisting of the dwell time for each key and the latencies between two successive keys for static password string ".tie5Roanl." There were 51 subjects in the dataset. For each subject, there were eight data-collection sessions with at least one day apart between two sessions. 50 repeated keystroke strings were collected in each session, resulting in a total of 400 samples for each subject. Some data samples are shown in Figure 2.

We used the exact same evaluation methodology as in [11] to ensure objective performance comparisons. For each subject, we used the first 200 feature vectors as the training data. The remaining 200 feature vectors were used as positive test data, and the first 5 samples from the remaining 50 subjects are used to form 250 negative feature vectors as imposters in the authentication phase for this user. To show the advantage of using UBM, simple GMM (without UBM) experiment is also conducted. For the GMM-UBM and DBN the experiments, the first four samples from background users were also included in the training set, resulting in

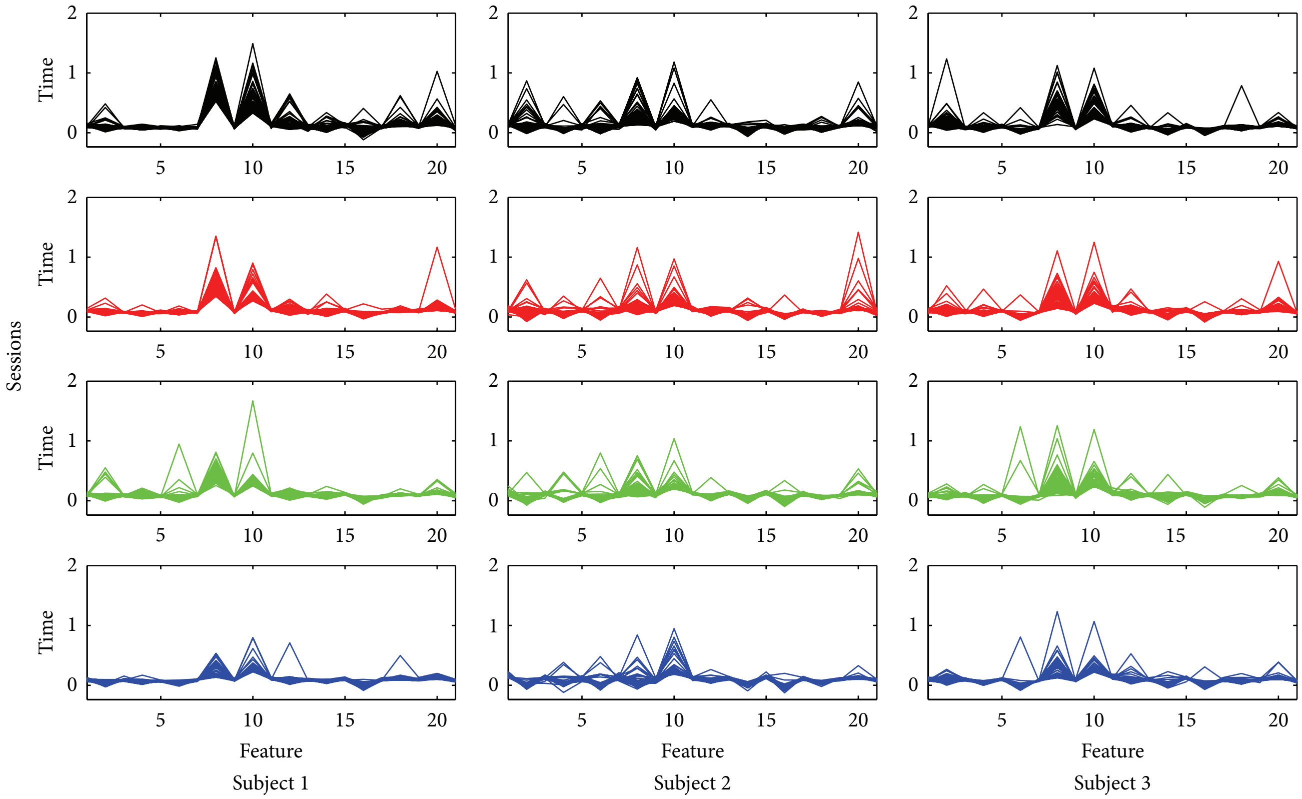


Figure 2: Keystroke dynamics features for static key string ".tie5Roanl" from the CMU keystroke dynamics benchmark dataset [11]. The dwell time and digraphs for the first four data-collection sessions for three subjects are shown. Although the keystroke features provide sufficient distinguishing patterns for each subject, they are highly correlated with large-scale variations and typical of noise and outliers. We have previously proposed a new distance metric to effectively handle these challenges that are intrinsic to keystroke dynamics data [12]. In this work, we show that GMM-UBM and DBN-based approaches perform even better to model large variations and correlations in the data.

additional 196 training samples from the negative class. Note that the imposters' samples were never seen during the training time. It requires a total of 51∗51 sets of experiments; each used different subjects' data for training and testing. Under the simple GMM case (without UBM), only 51 sets of experiments are required.

The authentication accuracy is evaluated using the equal error rate (EER) where the miss rate and the false alarm rate are equal. The evaluation is performed for each subject. In the GMM-UBM and the DBN experiments, for each genuine user, a single threshold is applied to 51 sets of experiments. The mean and standard deviation of the equal error rates for the 51 subjects are reported.

4.2. The GMM, GMM-UBM, and DBN Modeling Setup. For the GMM experiment, we build a 32-mixture Gaussian model for each genuine user. The variance floors for all feature dimensions and all Gaussian components are set to 0.01 to avoid poorly trained parameters. Each genuine user applies its own threshold value to the likelihood scores of all test samples to compute the EER.

Under the GMM-UBM setting, for each genuine user, 51 sets of experiments were conducted, one for each test subject, to exclude the test subject from the UBM subject set. The UBM is also modeled with 32 mixtures of the Gaussian modl. For each testing sample, the log likelihood ratio is computed for the genuine user model and the UBM model. A single likelihood ratio threshold is applied to the 51 sets of experiment to compute the EER.

To apply DBN to keystroke model, we first build a Gaussian RBM, with 31 visible units and 100 hidden units, and a binary RBM, with 100 visible units and 100 hidden units. The ANN parameter fine tuning stops when the training error is less than 1%.

4.3. Experimental Results. The performances, measured in mean and standard deviation of equal error rate (EER), of the proposed GMM-UBM and DBN approaches are listed in Table 1. For comparison, some of the best published results on the same dataset are also included in the table. The results showing that the simple GMM-based approach performs very close to our recently reported best results based on the combined Mahalanobis and Manhattan distance all outperform all of the 14 published algorithms on the task, with the results of three well-known algorithms shown in the table [11].

When we include background users' data in the keystroke model, the GMM-UBM approach reduced the EER significantly compared with the simple GMM approach. The best performance is achieved using the DBN approach. Compared with the best reported EER of 8.4%, the DBN reduced the EER

Table 1: Performance comparison between the proposed approaches and the existing best reported algorithms on the same CMU dataset. Mean and standard deviation are shown for the equal error rate (EER). The proposed approaches significantly outperform the state-of-the-art approaches.

Algorithm	EER
Neural network (auto-assoc) [11]	0.161 (0.080)
SVM (one-class) [11]	0.102 (0.065)
Manhattan (scaled) [11]	0.096 (0.069)
Combined Mahalanobis and Manhattan distance [12]	0.084 (0.056)
GMM	0.087 (0.058)
GMM-UBM	**0.055 (0.052)**
DBN	**0.035 (0.027)**

to 3.5%, or a 58% relative error rate reduction. This dramatic improvement is due to DBN's generative and discriminative model nature. The RBM generative modeling is not only better suited to capture keystroke nonlinear statistics, but also ensures better generalization to sample from new users. The DBN discriminative parameter fine tuning step further boosted classification accuracy.

5. Discussion and Future Work

We studied the characteristics of keystroke dynamics for computer user authentication and proposed a couple of approaches featuring generative model with discriminative power. The proposed GMM-UBM and DBN approaches outperform a spectrum of top performing keystroke dynamics classifiers using traditional distance metrics statistics and machine-learning algorithms.

The achieved performance of 3.5% EER is very encouraging, because this is achieved based on a single word. As a continuous user authentication modality, the keystroke dynamic can conveniently acquire a large set of training and verification data from a specific user; thus, highly accurate keystroke authentication is achievable.

Although the proposed GMM-UBM and DBN approaches are only applied to keystroke dynamics using static text, they can be easily extended to the free text use cases. Given a large pool of free typing texts from a large set of subjects, the discriminative power of each word and subword string can be discovered. The future research should work toward making a large set of keystroke datasets available to the research community, investigating the more challenging problem of keystroke biometrics using free text, developing richer keystroke features, studying context-dependent subword and across-word models, seamlessly integrating language model score, that is, the authorship, into the keystroke dynamic system, and mitigating the effect of different hardware and network delay for remote-access applications.

References

[1] A. K. Jain, R. Bolle, and S. Pankanti, Eds., *Biometrics: Personal Identification in Networked Society*, Kluwer Academic, 1999.

[2] A. K. Jain, S. Pankanti, S. Prabhakar, H. Lin, and A. Ross, "Biometrics: a grand challenge," in *Proceedings of the 17th International Conference on Pattern Recognition (ICPR '04)*, pp. 935–942, August 2004.

[3] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 4–20, 2004.

[4] S. Prabhakar, S. Pankanti, and A. K. Jain, "Biometric recognition: security and privacy concerns," *IEEE Security and Privacy*, vol. 1, no. 2, pp. 33–42, 2003.

[5] J. D. Woodward, N. M. Orlans, and P. T. Higgins, *Biometrics: Identity Assurance in the Information Age*, McGraw-Hill, New York, NY, USA, 2003.

[6] F. Monrose and A. D. Rubin, "Keystroke dynamics as a biometric for authentication," *Future Generation Computer Systems*, vol. 16, no. 4, pp. 351–359, 2000.

[7] A. Dvorak, N. Merrick, W. Dealey, and G. Ford, *Typewriting Behavior*, American Book Company, New York, NY, USA, 1936.

[8] J. Leggett and G. Williams, "Verifying identity via keystroke characteristics," *International Journal of Man-Machine Studies*, vol. 28, no. 1, pp. 67–76, 1988.

[9] L. Zhuang, F. Zhou, and J. D. Tygar, "Keyboard acoustic emanations revisited," *ACM Transactions on Information and System Security*, vol. 13, no. 1, article 3, 2009.

[10] R. V. Yampolskiy and V. Govindaraju, "Behavioral biometrics: a survey and classification," *International Journal of Biometrics*, vol. 1, no. 1, pp. 81–113, 2008.

[11] K. S. Killourhy and R. A. Maxion, "Comparing anomaly-detection algorithms for keystroke dynamics," in *Proceedings of the IEEE/IFIP International Conference on Dependable Systems and Networks (DSN '09)*, pp. 125–134, Lisbon, Portugal, July 2009.

[12] Y. Zhong, Y. Deng, and A. K. Jain, "Keystroke dynamics for user authentication," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW '12)*, pp. 117–1123, Providence, RI, USA, 2012.

[13] J. D. Allen, *An analysis of pressure-based keystroke dynamics algorithms [M.S. thesis]*, Southern Methodist University, Dallas, Tex, USA, 2010.

[14] L. Bello, M. Bertacchini, C. Benitez, J. C. Pizzoni, and M. Cipriano, "Collection and publication of a fixed text keystroke dynamicsdataset," in *Proceedings of the (CACIC '10)*, October 2010.

[15] R. Giot, M. El-Abed, and C. Rosenberger, "GREYC keystroke: a benchmark for keystroke dynamics biometric systems," in *Proceedings of the IEEE 3rd International Conference on Biometrics: Theory, Applications and Systems (BTAS '09)*, Washington, DC, USA, September 2009.

[16] J. R. Montalvão Filho and E. O. Freire, "On the equalization of keystroke timing histograms," *Pattern Recognition Letters*, vol. 27, no. 13, pp. 1440–1446, 2006.

[17] T. Sim and R. Janakiraman, "Are digraphs good for free-text keystroke dynamics?" in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '07)*, pp. 17–22, Minneapolis, Minn, USA, June 2007.

[18] F. Bergadano, D. Gunetti, and C. Picardi, "User authentication through Keystroke dynamics," *ACM Transactions on Information and System Security*, vol. 5, no. 4, pp. 367–397, 2002.

[19] D. Gunetti and C. Picardi, “Keystroke analysis of free text,” *ACM Transactions on Information and System Security*, vol. 8, no. 3, pp. 312–347, 2005.

[20] F. Monrose, M. K. Reiter, and S. Wetzel, “Password hardening based on keystroke dynamics,” in *Proceedings of the 6th ACM Conference on Computer and Communications Security (ACM CCS ’99)*, pp. 73–82, November 1999.

[21] G. Forsen, M. Nelson, and R. Staron Jr., “Personal attributes authentication techniques,” Tech. Rep. RADC-TR-77-333, Rome Air Development Center, 1977.

[22] R. Spillane, “Keyboard apparatus for personal identification,” *IBM Technical Disclosure Bulletin*, vol. 17, no. 3346, 1975.

[23] S. P. Banerjee and D. L. Woodard, “Biometric authentication and identification using keystroke dynamics: a survey,” *Journal of Pattern Recognition Research*, vol. 7, pp. P116–P139, 2012.

[24] R. Gaines, W. Lisowski, S. Press, and N. Shapiro, “Authentication by keystroke timing: some preliminary results,” Rand Rep. R-2560-NSF, Rand Corporation, 1980.

[25] S. Bleha, C. Slivinsky, and B. Hussien, “Computer-access security systems using keystroke dynamics,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 12, pp. 1217–1222, 1990.

[26] S. Cho, C. Han, D. H. Han, and H.-I. Kim, “Web-based keystroke dynamics identity verification using neural network,” *Journal of Organizational Computing and Electronic Commerce*, vol. 10, no. 4, pp. 295–307, 2000.

[27] L. C. F. Araújo, L. H. R. Sucupira, M. G. Lizárraga, L. L. Ling, and J. B. T. Yabu-uti, “User authentication through typing biometrics features,” in *Proceedings of the 1st International Conference on Biometric Authentication (ICBA ’04)*, vol. 3071 of *Lecture Notes in Computer Science*, pp. 694–700, Springer, Berlin, Germany, 2004.

[28] R. Joyce and G. Gupta, “Identity authentication based on keystroke latencies,” *Communications of the ACM*, vol. 33, no. 2, pp. 168–176, 1990.

[29] S. Haider, A. Abbas, and A. K. Zaidi, “Multi-technique approach for user identification through keystroke dynamics,” in *Proceedings of the IEEE Interantional Conference on Systems, Man and Cybernetics*, pp. 1336–1341, October 2000.

[30] C. L. Chen, K. L. Weng, and P. L. Chee, “Keystroke patterns classification using the ARTMAP-FD neural network,” in *Proceedings of the 3rd International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP ’07)*, pp. 61–64, Kaohsiung, Taiwan, November 2007.

[31] E. Yu and S. Cho, “Ga-SVM Wrapper approach for feature subset selection in keystroke dynamics identity verifcation,” in *Proceedings of the International Joint Conference on Neural Networks*, pp. 2253–2257, IEEE Press, 2003.

[32] P. Kang, S. Hwang, and S. Cho, “Continual retraining of keystroke dynamics based authenticator,” in *Proceedings of the 2nd International Conference on Biometrics (ICB ’07)*, pp. 1203–11211, Springer, Berlin, Germany, 2007.

[33] D. Hosseinzadeh and S. Krishnan, “Gaussian mixture modeling of keystroke patterns for biometric applications,” *IEEE Transactions on Systems, Man and Cybernetics Part C*, vol. 38, no. 6, pp. 816–826, 2008.

[34] D. A. Reynolds, “Comparison of Background Normalization Methods for Text-independent Speaker Verification,” Euro-Speech, 1997.

[35] G. E. Hinton, S. Osindero, and Y.-W. Teh, “A fast learning algorithm for deep belief nets,” *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[36] R. Salakhutdinov, *Learning deep generative models [Ph.D. thesis]*, University of Toronto, 2009.

[37] G. E. Hinton, “Training products of experts by minimizing contrastive divergence,” *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, 2002.

The Scientific World Journal

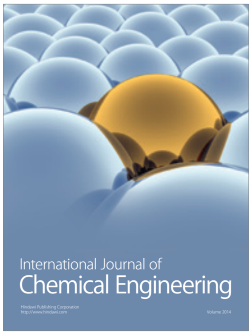
International Journal of
Rotating Machinery

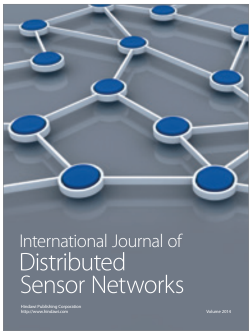
Journal of
Robotics

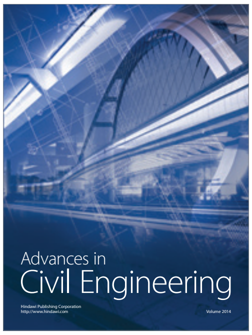
Journal of
Sensors

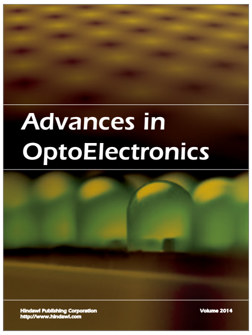
Advances in
Mechanical Engineering

Journal of
Engineering


International Journal of
Chemical Engineering

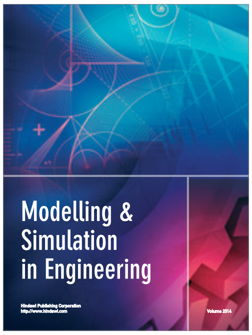

International Journal of
Distributed Sensor Networks

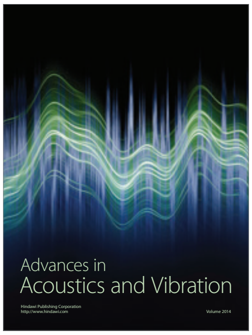

Advances in
Civil Engineering


Advances in
OptoElectronics

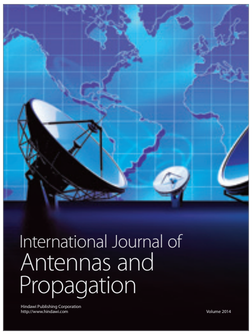
VLSI Design

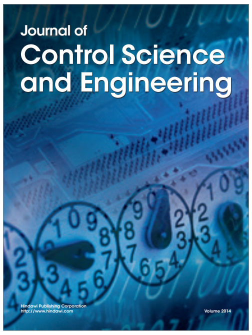
International Journal of
Navigation and Observation

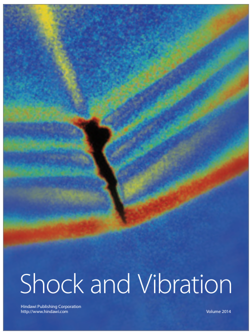

Modelling & Simulation in Engineering


Advances in
Acoustics and Vibration

Active and Passive Electronic Components


International Journal of
Antennas and Propagation


Journal of
Control Science and Engineering


Shock and Vibration

Journal of
Electrical and Computer Engineering