CHAPTER 4

Keystroke Dynamics Advances for Mobile Devices Using Deep Neural Network

Yunbin Deng and Yu Zhong

Recent popularity in mobile devices has raised concerns on mobile technology security, as not only sensitive and private data are being stored on mobile devices, but also allowing remote access to other high value assets. This drives research efforts to new mobile technology security methods. Fortunately, new mobile devices are equipped with advanced sensor suite, enabling a multi-modal biometrics authentication solution, to include voice, face, gait, signature, and keystroke authentication, among others. Compared with other modalities, keystroke authentication offer some very attractive features: 1) non-intrusive, either password or free-text typing keystroke authentication can be applied without affecting users' daily user of the device; 2) it can work on continuous authentication mode for free typing; 3) it can leverage a unique set of advanced build in sensors, including accelerometer and gyroscope to capture rich typing information than raw timing pattern. We present a deep learning approach

Yunbin Deng and Yu Zhong
BAE Systems
6 New England Executive Park, Burlington MA 01803, USA
e-mail: {Yunbin.deng, Yu.Zhong}@baesystems.com

Editors: Y. Zhong and Y. Deng, *Recent Advances in User Authentication Using Keystroke Dynamics Biometrics*
DOI: 10.15579/gcsr.vol2.ch4, GCSR Vol. 2, pp. 59-70, 2015

[12], which is a very powerful advanced machine leaning method, to the challenging problem of keystroke dynamics biometric. We further take advantage of the rich sensor modalities available for mobile devices and strengthen our keystroke dynamics biometrics using multi-modal typing features.

4.1 Introduction

In the past few years, mobile devices with touch screen features have been used by more and more users. In the year 2013, over 1 billion smartphones had been sold. Smartphone users range from teenagers to presidents, from civilian to military personnel. These devices contain lots of private, personal, and business information, and sometimes are used to remotely access information critical to national security. However, these devices have no physical security protections like the traditional PC and workstation have. As such, the security of these mobile devices poses greater challenge than traditional office equipment. For example, a stolen unlocked smartphone could potentially leak all critical information stored on the phone and some other remotely accessible data.

On the bright side, these mobile devices are equipped with many sensing modalities not available on traditional computing equipment. Sensors suite on a modern device can include: touch screen, accelerometer, gyroscope, magnetometer, camera, finger scanner, microphone, GPS, proximity sensorhear rate sensor, gesture sensor, barometer, etc. Among them, touch screen sensors, accelerometer, and gyroscope can be used to strengthen keystroke dynamics authentication on mobile devices [21, 31]. The touch sensor can sense not only the event of touching, but also the size and pressure of touching. As such, mobile devices have great potential to achieve high security if proper technologies are developed to exploit these rich sensing modalities.

To meet the new challenges for mobile keystroke dynamics biometrics, we present a deep learning approach [12], which is a very powerful advanced machine leaning method. Since its introduction, it has dominated the speech analysis field, drastically improving performance records on many tough problems that have been kept unsolved for years. It is starting to take over other challenging research fields as well, e.g., face recognition [27]. We have previously applied deep learning method to static keystroke dynamics biometrics authentication for desktop computers, where it has demonstrated superior performance compared to the state-of-the-art. In this chapter we investigate the deep learning approach to keystroke dynamics authentication on mobile devices [6], and explore additional sensory data available on mobile devices for augmented keystroke dynamics biometrics. We will evaluate our approach on mobile keystroke dynamics dataset and compare it with the state-of-the-art.

The rest of this chapter is organized as follows. Section 4.2 gives a review of mobile keystroke biometrics. Section 4.3 introduces deep learning approach to keystroke dynamics authentication for mobile devices. Section 4.4 describes user verification experiments and performance of the proposed algorithms on public data set. We draw conclusions and layout future work in Section 4.5.

4.2 Mobile Keystroke Dynamics Authentication Literature

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, in the same way as handwritings or signatures, due to their similar governing neurophysiological mechanisms. Keystroke biometrics has desirable properties due to its low cost, user-friendliness, and non-intrusiveness. Continuous authentication is possible using keystroke dynamics just as a mere consequence of people's use of computers. Keystroke dynamics biometrics has been an active research area for a couple decades [9, 10, 14, 15, 17, 19, 22, 23, 30, 32].

Clarke and Furnell performed a feasibility study on keystroke based user authentication on mobile phones [4]. Key hold time and error rate (number of pressing the backspace key) were used as features in their study. They achieved 12.8% EER using neural network classifiers on mobile handset with 12-key hardware keyboard [5]. Maiorana et al. [20] also investigated the feasibility of using keystroke dynamics for user verification on mobile phones. They proposed a new statistical classifier which is computational efficient for use in a mobile environment. They assessed the discriminative power of different subsets of keystroke timing features, and obtained an EER of 13.59%. Their study indicates that keystroke dynamics biometrics provides effective authentication for mobile devices, but needs boost in order to facilitate a highly secure authentication scheme. Simple statistical methods were also employed by Campisi et al. [2] for keystroke dynamics biometrics on mobile phones. Their analysis reinforced the suggestion that for mobile devices, a strong secure authentication scheme cannot rely solely on keystroke dynamics; however keystroke dynamics can be a valuable module of a more complex security system. Buchoux and Clark [1] studied various classifiers for keystroke analysis on smart phones, and their study suggested that statistical classifiers are the most effective given the trade-off between computational requirements and authentication accuracy. Zahid et al. investigated keystroke dynamics for mobile phones with numeric keyboards where each key is multiplexed for several characters [28]. They proposed four digraphs customized for these keyboards: horizontal/vertical digraph which is the time to switch between keys horizontally/vertically, and non-adjacent horizontal and vertical digraph which is the time to switch between non-adjacent keys horizontally/vertically. They demonstrated that these features, combined with the conventional key hold time and error correction rate, were capable to capture user characteristics. Using Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) classifiers with these features, they obtained an average error rate of 2% FAR after the verification mode on a dataset containing 25 subjects.

Trojahn and Ortmeier [29] compared keystroke dynamics performance using hardware keyboards and software keyboards on mobile phones during the login process. They found that despite a small performance degradation using virtual keyboard input, it is still feasible for keystroke dynamics biometric authentication. Kambourakis et al. [18] proposed to enhance traditional keystroke dynamics features with speed and distance the finger moved for smartphones with touchscreens. This upgrade resulted in an EER of 26% on a 10-digit PIN and an EER of 13.6% on short passphrases.

As we have discussed that although it provides effective authentication scheme for mobile devices, keystroke dynamics alone may come short of meeting strong security requirements. Fortunately there are many advanced sensors embedded in mobile devices which may be exploited for improved authentication performance. These sensors can either facilitate more comprehensive keystroke characterization for augmented keystroke dynamics biometrics, or provide other biometric modalities to be fused with keystroke dynamics for improved system authentication performance.

Touchpad pressure sensors measuring finger pressure exerted on touchpad during typing events provide straightforward augmentation for keystroke dynamics biometrics for mobile devices with touch screens. Saevanee and Bhatarakosol [24] explored the use of finger pressure on touch pad in addition to keystroke dynamics for user authentication. They found that finger pressure features are more discriminative than the conventional keying time features, and obtained an accuracy of 99% using finger pressure features with the PNN analytical method. Jain et al. [16] also suggested that by fusing touch screen features with conventional keystroke features, superior performance to hardware keyboards could be achieved for touch screen smartphones. Chang et al. [3] proposed a graphical password interface with enlarged virtual keys for improved keystroke dynamics utility and authentication accuracy. They also examined the use of finger pressure features to enhance the authentication scheme. They demonstrated that, by fusing pressure features with keystroke timing features, they reduced the EER of the keystroke dynamics based authentication system from 12.2% to 6.9%, on a dataset containing 100 subjects and 20 imposters. Trojahn et al. [28] investigated combinations of keystroke time features for keystroke dynamics authentication for mobile devices. They also explored additional touch features such as touch pressure and the size of the key touch for enhanced keystroke dynamics. They found that the additional touch features reduced more than 30% of the error of the timing feature based keystroke authentication scheme using a dataset of 152 subjects.

Mobile devices are typically embedded with inertial sensors including accelerometers and gyroscopes which record the motion of the device. These motion characteristics have been exploited to improve the accuracy of keystroke dynamics biometrics for mobile devices. Ho [13] explored the use of accelerometer statistics, key tap size, and key duration features to authenticate mobile device user during the login stage. The study showed that accelerometer statistics performed the best among the three feature types, while fusing the three feature types drastically improved the accuracy when individual feature type was used. Giuffrida et al. [9] used motion measurements from inertial sensors including accelerometer and gyroscopes to substantially boost keystroke dynamics authentication performance on mobile phones. In another study, [30] exploited four features extracted from sensors in touchscreen smartphones to fully characterize the keystroke dynamics: accelerations during key pressing, the touching pressure, touching area, and key hold and inter key time. Experiments conducted using keystroke analysis on 4-digit and 8-digit PINs using a dataset containing more than 80 subjects yielded an EER of 3.65%. Trojahn et al. [28] fused keystroke dynamics biometrics with gait characteristics from gyroscopes for continuous authentication on mobile devices.

Keystroke dynamics biometrics for continuous mobile authentication has been investigated. Feng et al. [7] investigated mobile authentication for both the login and

post login stages. They adopted text independent keystroke features, comprising of keystroke time, tactile pressure from the capacitive touchscreen, with and without haptic feedback. Decision tree, random forest, and Bayes Net classification methods were used. Performance analysis on a dataset of 40 subjects indicated that adopting pressure information improved authentication accuracy, and typing with haptics feedback boost the performance as well. Gascon et al. [8] performed a study on continuous authentication of mobile device users using typing motion behavior on a software keyboard. In addition to the typical keystroke time features, they also utilized data from the accelerometer, gyroscope, and orientation sensor to characterize the motion signature of the typing behavior. These features were collected for predefined short text from 315 subjects. A 2376 dimensional feature vector encodes the statistics and shape of motion measurements in both spatial and frequency domains, was extracted to represent the typing motion behavior. SVM was then used to classify these high dimensional feature vectors.

4.3 Advance in Mobile Keystroke Authentication Algorithm

Although many purely discriminative model approaches exist, such as ANNs and SVMs, models trained on a large amount of background users, without access to the real imposter's data at training time, they do not guarantee good generalization performance to the unforeseen imposters. Recently, DNN was proposed in the machine learning community as a generative-discriminative hybrid approach [12]. The unsupervised generative training step grant the model with good generalization capabilities to unforeseen test data, while the discriminate fine tune step endow the model with super classification accuracy. It has achieved better performance than the ANNs and SVMs approaches in many well defined tasks, including hand-writing digits recognition, speech and language modelling, and object recognition. Previously, we applied DNN approach to the traditional PC based keystroke authentication using keystroke timing features. Here we apply the DNN modeling approach to the problem of mobile soft keystroke authentication, using the timing, taping, and inertial sensor features. The following subsections detail the basic theory of this new approach.

4.3.1 Deep Neural Networks (DNN)

The deep neural networks are hierarchical probabilistic model that are composed of input layer, hidden layers, and output layer. The input layer typically has the same dimension as the input feature vector to be modelled. The units in the hidden layers are called hidden variables. The hidden variables typically have binary values and are called the feature detectors. These hidden layers can be trained one layer at a time in a purely unsupervised fashion, with the output of lower level layer serve as input to the higher level layer. The idea is to build a hierarchical generative model, so that each higher level layer captures more complex non-linear features in the data. These pretrained two-layer generative models are then collapsed into a single multi-layer model and serve as an initialized ANN for further discriminative parameter fine tuning. The

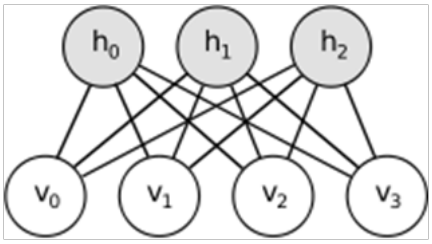

Figure 4.1: Restricted Boltzmann Machine.

dimension of the output layer typically equals the number of classes the classifier tries to resolve.

The pre-training of a generative model is important for the generalization capability of the final model. It also facilitates the fine tuning of the ANN, i.e. providing a good start point. It is well known that ANN is sensitive to the model parameter initialization and can easily fall to local optimal. The DNN pre-train not only avoids the random initialization of ANN parameters, but also significantly speeds up the ANN training process.

4.3.1.1 Pre-Train of DNN

The first step to build a DNN is to perform a layer-wise unsupervised training of Restricted Boltzmann Machines (RBMs). A RBM is one type of Markov random field that has two layers, a visible layer and a hidden layer. The units in the visible layers (v), are connected to all units in the hidden layer (h) with associated weights $\mathbf{W}$. Note there is no connection within each layer. A simple RBM with four input units and three hidden units is illustrated in Fig.(4.1).

The units in the visual layer can be real value, integer, or binary, depending on the type of input data to be modelled. The hidden units are typically binary stochastic variables, i.e, $h \in \{0, 1\}$. The Gaussian RBM is chosen for the first layer of RBM to model the real values of the keystroke features. The value of input and hidden variable, $\{v, h\}$, defines the state of the machine and the energy of the state, E, is defined as

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

where $\theta = \{W, a, b, \sigma\}$ are parameters specifying the RBM [25]. 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 bias and variance parameters for the input layer. The value of F specifies the capacity of the model. It depends on the data complexity and amount of data available to train the parameters. It is task specific and typically not a very sensitive parameter.

The binary output of the first layer Gaussian RBM further serves as input for higher level RBMs to capture more complex non-linear structure embedded in the data. This

process is also known as automatic feature engineering. Higher level RBMs in the hierarchical generative are all defined as binary RBMs, i.e., both the visible and hidden layers contains only binary units. Their energy functions are defined as

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

The RBMs are stochastic and the joint distribution of visible and hidden units is defined by

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

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

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

The likelihood of the training data is then specified as

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

However, the exact layer-wise maximum likelihood training of the RBM is intractable (except some trivial cases with very small value of D and F) as the computation takes time that is exponential to the dimension of D and F. An approximate solution is provided by a technique known as "*Contrastive Divergence*" [11].

4.3.1.2 Fine Tuning of DNN

The output of the unsupervised pre-train step is decks of RBMs, which can be stacked together and be added with a final classification layer to form an initialized ANN. This is conceptually illustrated in Fig.(4.2), where two RBMs are collapsed by sharing the middle units and a final layer (with weight W3) is added to perform keystroke classification. The parameters of the final classification layer can be trained the same way as training a typical ANN with back-propagation algorithm. Like training a typical ANN, there are many tricks to make the training faster while at the same time keeping the training process converge and avoiding overfitting.

The DNN discriminative fine tuning solves a non-linear optimization problem. The cost function can be minimum classification error or reconstruction error depending on whether it solves a classification or code/compression problem. A popular method to solve this problem is stochastic gradient descent. For large training data set, it is more efficient to compute the derivatives on a small, random mini-batch of training cases. In addition, the learning rate parameters, which can vary during the training process, need to be chosen carefully to balance training speed and avoid divergence.

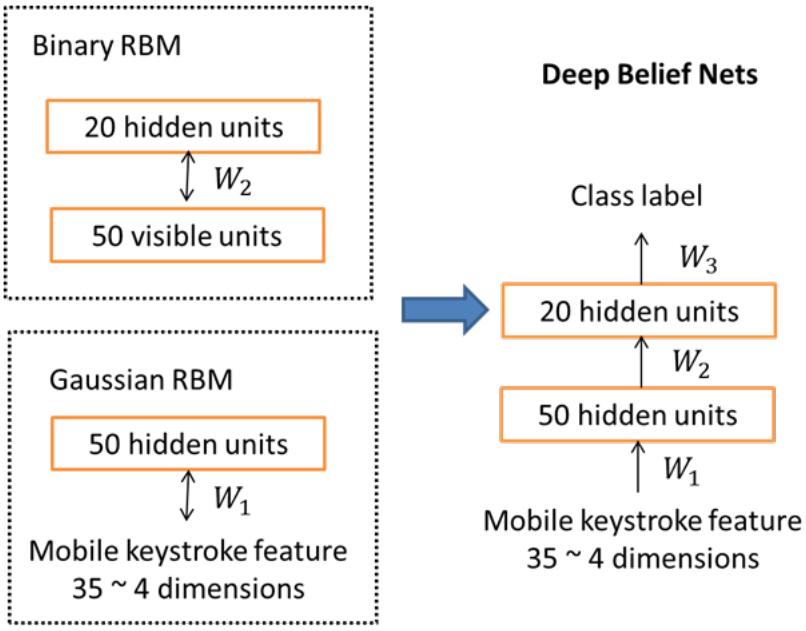

Figure 4.2: Train DNN for keystroke dynamics authentication in two steps. *Left:* unsupervised training of RBMs, *Right:* Convert RBMs into DNN.

To minimize the risk of model overfitting to the training data, many techniques are found to be useful, including 1) the simplest and most effective method is to have as much training data as possible; 2) Choose a model size with the right capacity; 3) train many models on different subset of training data and apply model averaging; 4) using cross-validation data set to monitor cross validation data error rate and stop the training when the validation performance stop to improve; 5) injection noise to the weight parameter during the training to enhance robustness; 6) Normalize the input feature to have zero mean and maybe unit stand deviation; 7) Apply threshold to the parameter to avoid some parameter getting to big; 8) introduces regularization term to the optimizations function; 9) Apply drop out technique to randomly disable some units during the training process [26].

4.4 Experiments

This section presents a study on a public available data set on mobile keystroke authentication. The dataset was provided by Stanford University and was developed for mobile devices to include timing and additional touching and accelerometer features. Performance of various advanced user authentication algorithms will be compared on this data set.

4.4.1 Stanford TapDynamics mobile keystroke dataset [13]

The Stanford TapDynamics dataset recruited 55 subjects to input one out of five randomly assigned PIN code on an Android phone. The recorded data includes the duration of each key tap, the latency between each key tap, the size of each key tap, and all accelerometer readings over the course of a login attempt. Each login consists of five key taps. A total of 1704 data samples were recorded. The author preprocessed the data and made the features publicly available. The feature set includes for each sample: five features for duration, four features for latency, five features for tapping size, and accelerometer features. Twenty-one accelerometer features are generated per training example by computing various statistics over all accelerometer readings in a login attempt. Specifically, the computed statistics are the mean, min, max, variance, first quartile, second quartile, and third quartile for the x,y, and z components over all accelerometer readings in a training example. Overall, each data sample consists of thirty five features.

4.4.2 Apply Deep Learning to mobile keystroke authentication

We apply deep neural network, the best performing keystroke algorithm based on evaluation on the CMU keystroke data, to this mobile keystroke authentication task. To compare it with other techniques published on the same dataset, the experimental setup is kept the same as in [13]:

1. For each PIN code, all data for users with the PIN are selected to form a subset 'D'.
    a) For each user in 'D', the first 15 sample are used as positive training set, and the remaining 15 samples are used for positive training set.
    b) The first 15 samples from all other users in 'D' became negative training set, and the remaining 15 samples are used for negative training set.
    c) For each user in 'D', training and test a classifier and compute average FAR and FRR among all users in 'D' to get FAR and FRR for each PIN.
2. We compute the average FAR and FRR over five PINs to obtain FAR and FRR for each classifier.

The Stanford study [13] compared four classifiers, including Manhattan distance, Random forest, Gaussian discriminant analysis, and SVMs with linear kernel, among which the SVM classifier performed the best on the dataset. In addition, the impact of each sensor modality was studied by cumulatively removing sensor features and comparing the FAR and FRR using the SVM methods. Table 4.1 summarizes the sensor impact results based on the best SVM classifier. The baseline system used all the sensors and the combined feature dimension is 35. This system achieved FAR and FRR at 4.4% and 5.3%, respectively. When only the keystroke timing features are used, i.e, with only 4 dimension timing feature, the FAR and FRR are dropped to 28.4% and 17.4%, respectively.

We have applied the deep neural network (DNN) to the same problem and performed the impact of sensor modality experiments as well. The input features were first subject

Table 4.1: The impact of sensor modality using the SVM and DNN classifiers [13].

Feature Removed (cumulative)	Used Feature Dimension	SVM FAR	SVM FRR	DNN EER
Baseline (no feature removed)	35	4.4%	5.3%	2.8%
Accelerometer statistics	14	11.7%	12.6%	3.7%
Key Tap Sizes	9	17.8%	14.7%	4.0%
Key Tap Duration	4	28.4%	17.4%	5.0%

to mean and variance normalization. The DNN has real input layer with the same dimension as input feature vectors. The middle binary layers have used dimension of 50 and 20 for all experiments. To avoid the tuning of FAR and FRR, we report the EER (equal error rate) of the DNN approach here.

As one can see, the DNN approach performs drastically better than the previously reported best performing SVM classifier for this mobile keystroke authentication task. Again, the best performance is achieved via the fusion of all three sensor modalities. However, when more sensor modalities are excluded, the performance of DNN method drops much more gracefully. The performance achieved with timing only feature based on DNN is comparable to SVM classifier when all sensors are utilized.

4.5 Discussion and future work

Advanced mobile technology provides a rich sensor environment to cope with the new security challenges unforeseen by traditional office computing devices. Although physiological biometrics, such as fingerprint, provides a highly accurate one time authentication, it was shown to be easy to spoof. Mobile keystroke dynamics enables a non-intrusive continuous authentication modality which has the potential to provide a highly accurate and secure solution to overcome mobile identity challenges. Pilot studies have shown that key taping pressure and accelerometer sensory data provide additional useful signatures for mobile user authentication. Our study have shown that recent advance in deep machine learning significantly improves mobile user authentication accuracy. It is worth to note that current studies are often based on a small pool of subjects and the testing scenarios are still quite simplified. Future work in mobile keystroke authentication study should investigate its effectiveness on more complex and realistic use cases and testing on a larger subject pool. In addition, other sensor modalities, such as gyroscope and face image, provided by mobile device should be considered for a comprehensive authentication solution for mobile security.

References

[1] A. Buchoux and N.L. Clarke. Deployment of keystroke analysis on a smartphone. In *Australian Information Security Management Conference*, 2008. p.48.

[2] P. Campisi, E. Maiorana, M. Lo Bosco, and A. Neri. User authentication using keystroke dynamics for cellular phones. *IET Signal Processing*, 3(4):333–341, 2009.

[3] T. Chang, C. Tsai, and J. Lin. A graphical-based password keystroke dynamic authentication system for touch screen handheld mobile devices. *Journal of Systems and Software*, 85(5):1157–1165, 2012.

[4] N.L. Clarke and S.M. Furnell. Authenticating mobile phone users using keystroke analysis. *International Journal of Information Security*, 6(1):1–14, 2006.

[5] N.L. Clarke, S.M. Furnell, B.M. Lines, and P.L. Reynolds. Keystroke dynamics on a mobile handset: a feasibility study. *Information Management & Computer Security*, 11(4):161–166, 2003.

[6] Y. Deng and Y. Zhong. Keystroke dynamics user authentication based on Gaussian mixture model and deep belief nets. *ISRN Signal Processing*, 2013, 2013. Article ID 565183, 7 pages.

[7] T. Feng, X. Zhao, B. Carbunar, and W. Shi. Continuous mobile authentication using virtual key typing biometrics. In *12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pages 1547–1552, 2013.

[8] H. Gascon, S. Uellenbeck, C. Wolf, and K. Rieck. Continuous authentication on mobile devices by analysis of typing motion behavior. In *GI Conference Sicherheit (Sicherheit, Schutz und Verlasslichkeit)*, 2014.

[9] C. Giuffrida, K. Majdanik, M. Conti, and H. Bos. I sensed it was you: authenticating mobile users with sensor-enhanced keystroke dynamics. In *Detection of Intrusions and Malware, and Vulnerability Assessment (DIMVA)*, volume 8550 of *Lecture Notes in Computer Science*, pages 92–111, 2014.

[10] D. Gunetti and C. Picardi. Keystroke analysis of free text. *ACM Transactions on Information and System Security*, 8(3):312–347, 2005.

[11] G.E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002.

[12] G.E. Hinton, S. Osindero, and Y.W. Teh. A fast learning algorithm for deep neural networks. *Neural Computation*, 18(7):1527–1554, 2006.

[13] G. Ho. Tapdynamics: strengthening user authentication on mobile phones with keystroke dynamics. Technical report, Stanford University, 2014.

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

[15] A. K. Jain, A. Ross, and S. Prabhakar. An introduction to biometric recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(1):4–20, 2004.

[16] L. Jain, J.V. Monaco, M.J. Coakley, and C.C. Tappert. Passcode keystroke biometric performance on smartphone touchscreens is superior to that on hardware keyboards. *International Journal of Research in Computer Applications & Information Technology*, 2(4):29–33, 2014.

[17] R. Joyce and G. Gupta. Identity authentication based on keystroke latencies. *Communications of the ACM*, 33(2):168–176, 1990.

[18] G. Kambourakis, D. Damopoulos, D. Papamartzivanos, and E. Pavlidakis. Introducing touchstroke: keystroke-based authentication system for smartphones. *Security and Communication Networks*, 2014. in press.

[19] J. Leggett, G. Williams, M. Usinick, and M. Longnecker. Dynamic identity verification via keystroke characteristics. *International Journal of Man-Machine Stud-*

ies, 35(6):859–870, 1991.

[20] E. Maiorana, P. Campisi, N. Gonzalez-Carballo, and A. Neri. Keystroke dynamics authentication for mobile phones. In *ACM Symposium on Applied Computing (SAC)*, pages 21–26, 2011.

[21] Y. Meng, S.W. Duncan, and S. Roman. Touch gestures based biometric authentication scheme for touchscreen mobile phones. In *Information Security and Cryptology*, volume 7763, pages 331–350, 2013.

[22] F. Monrose and A.D. Rubin. Keystroke dynamics as a biometric for authentication. *Future Generation Computing Systems*, 16(4):351–359, 2000.

[23] A. Peacock, X. Ke, and M. Wilkerson. Typing patterns: a key to user identification. *IEEE Security and Privacy*, 2(5):40–47, 2004.

[24] H. Saevanee and P. Bhatarakosol. User authentication using combination of behavioral biometrics over the touchpad acting like touch screen of mobile device. In *International Conference on Computer and Electrical Engineering (ICCEE)*, pages 82–86, 2008.

[25] R. Salakhutdinov. *Learning Deep Generative Models*. PhD thesis, University of Toronto, 2009.

[26] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[27] Y. Taigman, M. Yang, M.A. Ranzato, and L. Wolf. Deepface: closing the gap to human-level performance in face verification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1701–1708, 2014.

[28] M. Trojahn, F. Arndt, and F. Ortmeier. Authentication with keystroke dynamics on touchscreen keypads-effect of different N-Graph combinations. In *3rd International Conference on Mobile Services, Resources, and Users (MOBILITY)*, pages 114–119, 2013.

[29] M. Trojahn and F. Ortmeier. Biometric authentication through a virtual keyboard for smartphones. *International Journal of Computer Science & Information Technology*, 4(5):1–12, 2012.

[30] S. Zahid, M. Shahzad, S.A. Khayam, and M. Farooq. Keystroke-based user identification on smart phones. In *12th International Symposium RAID*, volume 5758 of *Lecture Notes in Computer Science*, pages 224–243, 2009.

[31] N. Zheng, K. Bai, H. Huang, and H. Wang. You are how you touch: user verification on smartphones via tapping behaviour. Technical report, College of William & Mary, 2012.

[32] Y. Zhong, Y. Deng, and A.K. Jain. Keystroke dynamics for user authentication. In *IEEE Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 117–123, 2012.