One-handed Keystroke Biometric Identification Competition

John V. Monaco[1], Gonzalo Perez[1], Charles C. Tappert[1], Patrick Bours[2], Soumik Mondal[2], Sudalai Rajkumar[3], Aythami Morales[4], Julian Fierrez[4] and Javier Ortega-Garcia[4]

[1]*Pace University, Pleasantville, New York, USA,* {jmonaco,gperez,ctappert}@pace.edu
[2]*Gjøvik University College, Gjøvik, Norway,* {patrick.bours,soumik.mondal}@hig.no
[3]*Tiger Analytics, Chennai, India,* sudalai@tigeranalytics.com
[4]*Universidad Autónoma de Madrid, Madrid, Spain,* {aythami.morales,javier.ortega,julian.fierrez}@uam.es

Abstract

This work presents the results of the One-handed Keystroke Biometric Identification Competition (OhKBIC), an official competition of the 8th IAPR International Conference on Biometrics (ICB). A unique keystroke biometric dataset was collected that includes freely-typed long-text samples from 64 subjects. Samples were collected to simulate normal typing behavior and the severe handicap of only being able to type with one hand. Competition participants designed classification models trained on the normally-typed samples in an attempt to classify an unlabeled dataset that consists of normally-typed and one-handed samples. Participants competed against each other to obtain the highest classification accuracies and submitted classification results through an online system similar to Kaggle. The classification results and top performing strategies are described.

1. Introduction

Keystroke biometric applications have been investigated over the past several decades, attracting both academics and practitioners. There are commercial products available that analyze a sequence of keystrokes for human identification, or provide additional security through password hardening and continuous authentication. It is common to see error rates below 10% for short text authentication [11], and below 1% in long text applications [12]. In terms of continuous authentication, an intruder can accurately be identified in less than 100 keystrokes [4]. While many performance evaluations are derived from normal typing behavior obtained in laboratory or natural settings, there has not been much research to determine how the performance of a keystroke biometric system degrades as a result of a user impairments, such as typing with one hand after having enrolled with a normal both-hands typing sample. Such a scenario might be encountered in production or during a field experiments that impose little or no condition on how the system should be used.

There are many performance-degrading scenarios that may be encountered during deployment of a keystroke biometric system. Variations in typing behavior can occur as a result of distractions, cognitive load, and sickness, to name a few. Consider the scenario in which a user has enrolled with normal two-hand typing and later restricted to typing with only one hand as a result of an injury or multitasking (*e.g.* using a desktop mouse with one hand while typing with the other). A robust keystroke biometric system should be able to handle this situation appropriately, although the correct response of such a system is not known at this point. Should the user be re-enrolled with a one-hand sample or can the user still be identified under this constraint? The results of this competition can help answer these questions.

2. Benchmark dataset

A unique keystroke biometric dataset was collected from three online exams administered to undergraduate students in an introductory computer science course during a semester. Each exam contained five essay questions that required typing a response directly into a web page. Students took the three exams through the Moodle learning platform and their keystrokes were logged by a Javascript event-logging framework [1] and transmitted to a server. For the first exam students were instructed to type normally with both hands, for the second exam with their left hand only, and for the third exam with their right hand only.

The benchmark dataset consists of 64 students who provided at least 500 keystrokes on each exam. Approximately 1/3 of all exam attempts occurred in an electronic classroom on standard desktop computers to ensure instructions were followed when typing with just one hand. The remaining

		Handedness			
		Amb.	Left	Right	Total
Typ. style	HaP	2	1	7	10
	HaP hybrid	1	2	22	25
	Touchtype	0	0	23	23
	Total	3	3	52	58

Table 1: Subject population demographics

students completed the exams individually on their personal computers, which was a mix of desktop and laptop computers. It has been shown that combined laptop and desktop keyboards has an effect on system performance [14]. This variable alone makes the dataset less than ideal, as some students completed the three exams using different model keyboards.

A subset of the data collected from the first exam with normal both-hands typing was designated as a labeled training dataset. A classifier needs to be able to successfully classify unlabeled samples under each condition: normal both-hands typing, left-hand typing, and right-hand typing. The labeled portion of the dataset consisted of 500-keystroke normally-typed samples from 64 subjects. Students also completed a short survey for demographic information, including handedness and typing style. Students identify themselves as ambidextrous, right, or left handed. They were also given a description of several typing styles and instructed to choose the style that closely matched their typing behavior: touch typist (touchtype), hunt-and-peck (H&P), or hybrid of touchtype and hunt-and-peck (H&P hybrid). A summary of the demographic information is shown in Table 1, showing 58 of the 64 students as some did not complete the survey.

The unlabeled dataset used for competition evaluation included typing samples from all three scenarios: 203 normal both-hands typing samples, 131 left-hand typing samples, and 137 right-hand typing samples, with 61 students from the training set appearing in the testing set. The number of samples per student in the testing set ranged from 1 to 38. In the training dataset, the subject ID, handedness, typing style, and press and release event timestamps of each key were made available. Timestamps were in millisecond precision and normalized to begin at 0 at the start of each sample. In the testing dataset, key name, press and release event timestamps, typing condition (both, left, right), and a unique sample ID were made available. Timestamps were normalized in a similar way. In cases where a subject provided a large amount of data for one exam, several samples were created by taking 500-keystroke segments separated by at least 50 keystrokes apart.

3. Submission evaluation

Classification results were evaluated based on the proportion of correctly classified samples (recognition accuracy). A sample is correctly classified if the correct subject identity for that sample is given in a submission file. A competition website allowed participants to register, download the training and testing datasets, make submissions, and discuss the competition in an open forum [1]. Submissions were made through an automated system with the leaderboard and results publicly available. The submission format was a CSV file with the header: "sample, user" and rows for each sample classification. The competition began September 1, 2014, and ended October 31, 2014.

Submissions were limited to one per day until the final day of the competition. To avoid overfitting submissions were evaluated on 50% of the unlabeled data until the end of the competition when the evaluation was based on 100% of the unlabeled data. The first place winner was awarded a Futronic FS88 fingerprint scanner.

The leaderboard was calculated as follows. The rank of each competition participant is first determined separately for each condition (both, left, and right hand keystroke input) in the unlabeled dataset using accuracy (ACC). The leaderboard is then determined by taking the sum of the three ranks from each condition. Thus, the best possible score is 3 (first place in each condition) and the worst is $3N$ (last place in each condition), where N is the number of participants in the competition. The leaderboard is designed to select a model that operates well under all three conditions. The highest ranking competition participant on the leaderboard was used to determine the winner, and the best (highest ranking) submission from each participant considered. A benchmark script was provided for participants to quickly parse the data and begin building a classification system. This script used a kNN classifier with a naive set of features that obtained a classification accuracy of less than 5% in each condition.

The competition was designed to be challenging and unique in the area of keystroke biometrics. It represents a realistic scenario that may be encountered in a keystroke biometric system, tackling a problem with no straightforward solution. Well known methods of keystroke biometric authentication have shown considerable degradation due to environment variables. It may be possible to account for one-handed typed samples when only normally-typed samples are known.

4. Competition results

A total of nine participants registered for the competition, and three actively competed against each other for first

[1] http://biometric-competitions.com/mod/competition/view.php?id=7

Rank	Team	Both	Left	Right
1	Gjøvik University College	82.8 ± 2.7	30.5 ± 4.0	40.2 ± 4.2
2	Sudalai Rajkumar S	82.8 ± 2.6	27.5 ± 3.9	32.1 ± 4.0
3	Universidad Autónoma de Madrid	69.5 ± 3.2	16.8 ± 3.3	20.4 ± 3.4
[Baseline]		61.1 ± 3.4	6.2 ± 2.1	9.5 ± 2.5

Table 2: Final competition leaderboard - the top three teams

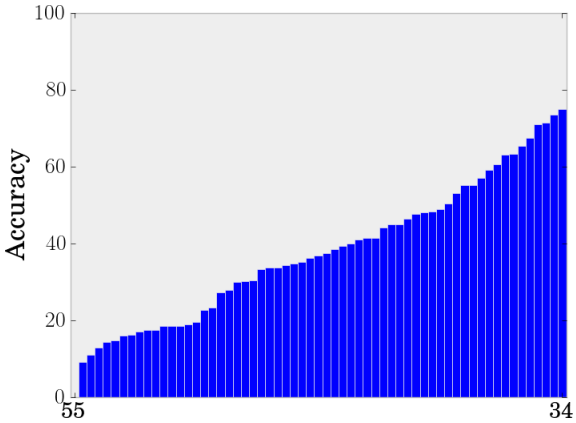
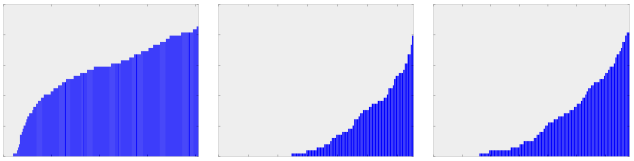


Figure 1: Accuracy distribution per student



(a) Both (b) Left (c) Right

Figure 2: Accuracy distribution per sample

		Condition			
		Both	Left	Right	Avg.
Handedness	Ambidextrous	38.9	14.3	1.5	21.2
	Left	41.3	14.6	10.6	25.0
	Right	55.1	17.8	24.4	35.9
	Avg.	53.7	18.3	23.7	35.2

Table 3: Handedness versus typing condition accuracy

place. There were a total of 48 submissions from four participants. Including the benchmark results there were a total of 49 submissions, so that any unlabeled sample could be correctly classified at most 49 times.

The final leaderboard of the competition is shown in Table 2. Baseline results were obtained with a 1-nearest-neighbor classifier using scaled Manhattan distance [2] and 218 commonly used keystroke features [12]. This baseline is different than the baseline obtained by the starter code provided to participants. To calculate the 95% confidence intervals, a bootstrap method was used. The submissions from each team and typing condition were resampled with replacement 10^4 times. The team from Gjøvik University College placed first, with the highest rank in each typing condition and the highest overall classification accuracy. The classification strategies of the three teams are described in Sections 5, 6, and 7.

4.1. Results analysis

First, the classification accuracy for individual students and samples was examined. It is well known that biometric system accuracy depends heavily on the enrolled users [15, 7]. The distribution of classification accuracies for each student is shown in Figure 1 and for each sample in Figure 2. The classification accuracy of each student and each sample was calculated using all submissions from all the competition participants. The student and sample with the highest and lowest classification accuracies are labeled in the figures. One student's samples were never correctly classified by any of the 49 systems.

The distribution of sample classification accuracies differs dramatically between one-handed and normally typed samples. Overall, 91 samples were never correctly classified: 49 typed with the left hand, 32 with the right hand, and 10 with both hands. The sample accuracy (ACC) under each typing condition is shown in Figure 2.

The 49 submissions were further analyzed in order to better understand the effect of several user and environment variables in the identification task. The classification accuracy was broken down by the self-reported handedness and typing style, attributes not verified by an expert. Of the 64 students, six did not provide handedness and typing style information. Of the 58 students who did provide the information, 55 appeared in the unlabeled dataset which consisted of 397 samples: 18 from students identified as ambidextrous, 20 from left handers, and 359 from right handers. And of these 397 samples, 91 were typed with the left hand, 144 with the right hand, and 162 with both hands.

The accuracies for each handedness and condition are shown in Table 3, with the average row and column weighted by the frequencies listed above. Left-handed and ambidextrous students were more easily identified from their left-hand samples, and right-handed students from their right-hand samples. This is understandable since dominant hand samples are more likely to be produced in a consistent keystroke rhythm than non-dominant hand samples.

The classification accuracy and sample frequency for subjects who reported a typing style is shown in Table 4,

Typing style	No. Samples	Accuracy
Hunt-and-peck	91	43.2 ± 0.74
Hunt-and-peck hybrid	144	33.5 ± 0.56
Touchtype	162	30.5 ± 0.52

Table 4: Accuracy versus typing style

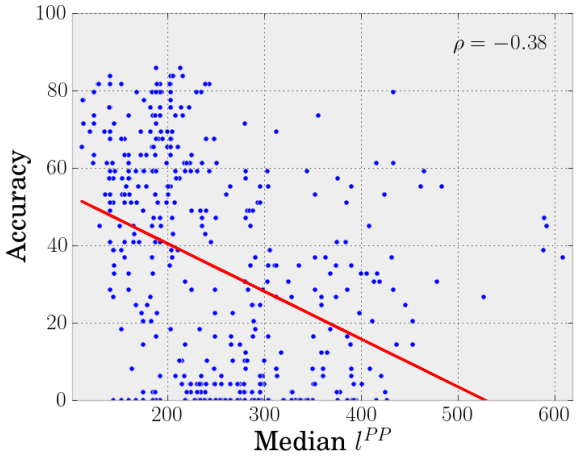


Figure 3: Accuracy versus median keypress latency showing the best-fit linear regression line

with 95% confidence intervals calculated be resampling the submissions for each typing style. The hunt-and-peck typists demonstrated higher performance than the hunt-and-peck hybrid and touch typists.

The typing speed of a student may be indicative of classification accuracy. We hypothesized that faster typists may type more consistently and therefore be easier to classify as a result of fewer spurious key latencies and durations. Let the duration of a keystroke and latencies between keystrokes be defined as

$$\text{Duration: } d_i = t_i^r - t_i^p$$

$$\text{RP-latency: } l_i^{RP} = t_{i+1}^p - t_i^r$$

$$\text{PP-latency: } l_i^{PP} = t_{i+1}^p - t_i^p$$

$$\text{RR-latency: } l_i^{RR} = t_{i+1}^r - t_i^r$$

$$\text{PR-latency: } l_i^{PR} = t_{i+1}^r - t_i^p$$

As an estimate of typing speed, we consider the median latency time between consecutive key presses, l_i^{PP}. In Figure 3, the mean accuracy of each sample is plotted against the median key press latency. With a Pearson correlation coefficient of $\rho = -0.38$, the typing speed of a subject may only be a weak indication of the difficulty in identifying that subject's samples.

In the following sections, we describe the classification stategies taken by the top three competing teams.

5. First place strategy

5.1. Feature Definition

From the available raw timing data, we extracted duration and latency information. Duration is defined as the time elapsed between pressing down a key and releasing that same key. Latency can be defined the time difference between releasing one key and pressing down the next one (RP latency) [2].

5.2. Classification

In our study, we have used pairwise coupling [9] by using 2 regression model and one prediction model in a multi-classifier architecture [10]. For regression models, we have applied Artificial Neural Network (ANN) and Counter-Propagation Artificial Neural Network (CPANN), while a Support Vector Machine (SVM) is used for the prediction model. The score vector for each pair coupling is $(f_1, f_2, f_3) = (Score_{ann}, Score_{svm}, Score_{cpann})$. Initial analysis showed that training accuracy of the classifiers was reduced when combining latency and duration features, compared to only using duration features. Therefore, we decided to use only duration features for our analysis.

For pairwise coupling, we used a bottom up tree structure, as shown in Figure 4. In this particular example, the pairs are created in increasing order, but in the actual analysis we have selected these pairs randomly. For each coupling we have calculated the total classification score (S_i) based on the weighted fusion of the individual classifier score mentioned above:

$$S_i = \frac{1}{\sum_{j=1}^{3} w_j} \sum_{j=1}^{3} w_j f_j$$

We selected the user from the pair which has the maximum value of S_i for the next level of the tree. By using different weights we have obtained different results, as discussed in the following section.

5.3. Results

We have tried different weights for the above fusion techniques and, for the "Both Hand" category the recognition rate ranged from 58.1% to 74.4%. For the "Left Hand" category the recognition rate ranged from 16.8% to 26.7% and for the "Right Hand" category the recognition rate ranged from 21.9% to 35.0%.

Next, we tried only using two classifiers with different weights based on the training accuracy of a specific pair (*i.e.* removing the classifier with the least accuracy for that specific pair). The results obtained from this analysis are slightly better. For the "Both Hand" category the recognition rate ranged from 55.7% to 82.8%, for the "Left Hand" category the recognition rate ranged from 19.9% to

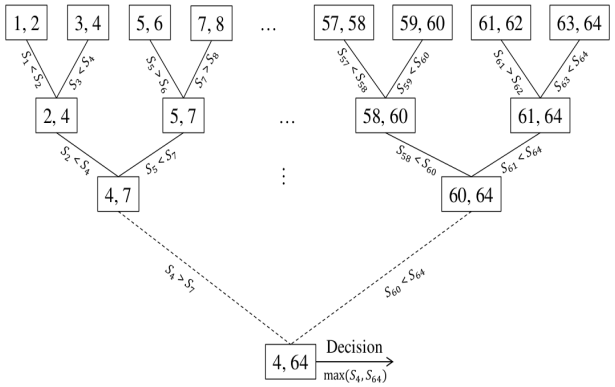


Figure 4: Tree structure based pairwise coupling.

30.5% and for the "Right Hand" category the recognition rate ranged from 24.8% to 34.3%.

When analyzing the test data of a user who is typing with only one hand, we adjusted our analysis slightly. Assume the test data is typed with the right hand; then typing keys on the right hand side of the keyboard will be typed in a normal manner, while keys on left hand side of the keyboard might be typed using only 1 finger of the right hand (most likely right index finger in this case). This means that typing characteristics change for keys on the other side of the keyboard. Instead of considering all typed text in the test data, we evaluated the performance based only on the keys in the test data that are on the right hand side of the keyboard. This strategy worked only for "Right Hand" category, where the recognition rate improved from 34.3% to 40.2% for a particular fusion weight. But, on the other hand, the recognition rate for "Left Hand" category reduced from 29.0% to 22.1%.

6. Second place strategy

6.1. Feature definition

The features we extracted from the raw data include durations and key press latencies for each individual key. Latency features include press to press (PP) and release to press (RP) latencies [3]. These features are extracted directly from the given raw data. Keys were also grouped together based on their position on the keyboard such as left, right, top, middle and bottom. The groups are made such that left key group consists of those keys which are present on the left side of the keyboard and so on. Key press duration and latency features are also extracted for each of these groups.

6.2. Classification

Each individual user had 500 keystrokes in the training dataset. We have sampled those 500 keystrokes into development and validation sample to test the accuracy of the model being built. Models are built on the training sample and validated on the validation sample to get the best parameters for the model.

Approach	Both	Left	Right
All key features	82.8	20.6	27.0
Left and right side features	82.8	27.5	32.1

Table 5: Second place strategy classification accuracy

We have used Random Forests [5] for our classification. Best parameters for the model are obtained through minimizing the classification error on the validation sample. We have also tried using other classification techniques like logistic regression and SVM, but Random Forests gave better training accuracy when compared to others.

6.3. Results

The results suggest that instead of using the features of all the keys, using only those features related to keys present on the left side of the key board yielded better results for "Left-Hand" category. Same is the case for "Right-hand" category as well. So we have used only those features for "Left-hand" and "Right-hand" classification.

7. Third place stategy

7.1. Feature definition

Our strategy is focused on matching features related with small strings of two and three keys (also called digraphs and trigraphs respectively [8]) in a similar way as fixed passwords. For each of the digraphs we create a feature vector composed of: duration (time interval between press and release of the same key, also known as dwell or hold time) and RP latency. In case of the trigraphs, we add the time interval between the press and release of alternate keystrokes.

7.2. Classification

For the matching of the feature vectors we proposed a multi-algorithm approach based on two algorithms [13]:

Algorithm A - Normalized Distance: This algorithm computes the distance between the feature vectors in the unlabeled and training datasets. The distance between a feature vector $v = [v_1, v_2, \ldots, v_N]$ from the unlabeled dataset and his training samples is measured as:

$$d_i = \frac{1}{N} \sum_{k=1}^{N} \frac{1}{\sigma_k} \left(|v_k - \mu_k| \right) \qquad \forall i \in 1, \ldots, M$$

where $\sigma = [\sigma_1, \sigma_2, \ldots, \sigma_N]$ and $\mu = [\mu_1, \mu_2, \ldots, \mu_N]$ are the standard deviation and mean of the training samples, N is the number of features (3 for digraphs and 6 for trigraphs)

Approach	Both	Left	Right
Algorithm A	50.7	12.2	16.7
Algorithm B	50.7	8.4	16.7
Combined	69.4	16.8	20.4

Table 6: Third place strategy classification accuracy

and *M* is the number of digraphs and trigraphs shared by the unlabeled and training datasets. The 80 lowest values of d (40 digraphs and 40 trigraphs) obtained for each user are combined by the mean rule (applying a correction factor of 1.5 to the digraphs) to obtain a final score S_A for each user.

Algorithm B SVM Classifier: The keystroke model of Algorithm B has been built using a Least Squares Support Vector Machine (LS-SVM). Least Squares Support Vector Machines are reformulations to standard SVMs, which lead to solutions of the indefinite linear systems generated within them. The meta-parameters of the LS-SVM models are optimized using the keystroke dataset included in the BiosecurID multimodal database [6]. This training procedure is used to generate a LS-SVM model per user, using their own training samples as positive vectors and training samples of the other 63 users as negative vectors. To identify an unlabeled sample, the LS-SVMs of all the users generate a score for each trigraph shared between the unlabeled and training datasets. The 40 highest scores of each user are combined by the mean rule to obtain a final score S_B for each user.

Combined Approach: the scores are normalized by the min/max technique, which transforms the range of the classification scores to [0-1]. The combination of the scores is based on a weighted sum given by:

$$S_{fusion} = w\,(1 - S_A) + (1 - w)\,S_B$$

where $\{S_A, S_B\}$ are the normalized scores (note that S_A and S_B are dissimilarity and similarity scores respectively) and w the weighting factor obtained through the performances achieved with the training dataset as $w = 1 - EER_A / (EER_A + EER_B)$

7.3. Results

Table 6 shows the performances achieved for all three scenarios and the three different proposed approaches:

The results suggest the complementarity of both algorithms when they are combined at score level. However, the improvement is clearer for the "Both Hand" category in which the accuracy ranged from 50.7% to 69.4%. For the "Left Hand" and "Right Hand" categories the improvements are moderate and range from 12% to 16% and from 16% to 20% respectively.

8. Summary

As expected, classification accuracy degrades considerably for keystroke samples that are typed with only one hand after having trained a model on both hands. A common theme among the competition participants to deal with one-handed samples was to divide the keyboard into left and right components, placing more weight on the side corresponding to the non-obstructed hand. Motivation for this approach is the assumption that typing behavior of the non-obstructed hand will be more natural on the corresponding side of the keyboard. This approach seemed to work well, as seen by the strategy of the first and second place teams.

The one-handed typed samples also seem to be generally easier to classify when the obstructed hand is non-dominant for right-handed subjects (*i.e.* right-handed subjects typing with only their right hand). It seems as though this may be true in general, as classification accuracy for left-handed subjects was higher when typing left-handed samples, although the small number of left-handed subjects calls for future work to validate this finding with greater significance.

Besides handedness, typing speed is also weak predictor of keystroke biometric system performance. The correlation between median PP latency and classification accuracy is too weak to be useful. Perhaps there are other behavioral characteristics that indicate system performance. This is also an area of keystroke dynamics that warrants more research.

Acknowledgments

The authors would like to acknowledge the support from the National Science Foundation under Grant No. 1241585. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation or the US government.

References

[1] Moodle bioauth plugin. (Accessed August 2014).

[2] L. C. Araujo, L. H. Sucupira Jr, M. G. Lizarraga, L. L. Ling, and J. B. T. Yabu-Uti. User authentication through typing biometrics features. *IEEE Transactions on Signal Processing*, 53(2):851–855, 2005.

[3] S. P. Banerjee and D. L. Woodard. Biometric authentication and identification using keystroke dynamics: A survey. *Journal of Pattern Recognition Research*, 7:116–139, 2012.

[4] P. Bours. Continuous keystroke dynamics: A different perspective towards biometric evaluation. *Information Security Technical Report*, 17(1):36–43, 2012.

[5] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[6] J. Fierrez, J. Galbally, J. Ortega-Garcia, et al. Biosecurid: a multimodal biometric database. *Pattern Analysis and Applications*, 13(2):235–246, 2010.

[7] J. Fierrez-Aguilar, J. Ortega-Garcia, and J. Gonzalez-Rodriguez. Target dependent score normalization techniques and their application to signature verification. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 35(3):418–425, 2005.

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

[9] T. Hastie and R. Tibshirani. Classification by pairwise coupling. *The Annals of Statistics*, 26(2):451–471, 04 1998.

[10] T. K. Ho, J. Hull, and S. Srihari. Decision combination in multiple classifier systems. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(1):66–75, 1994.

[11] K. S. Killourhy and R. A. Maxion. Comparing anomaly-detection algorithms for keystroke dynamics. In *Dependable Systems & Networks, 2009. DSN'09. IEEE/IFIP International Conference on*, pages 125–134. IEEE, 2009.

[12] J. V. Monaco, N. Bakelman, S.-H. Cha, and C. C. Tappert. Recent advances in the development of a long-text-input keystroke biometric authentication system for arbitrary text input. In *Intelligence and Security Informatics Conference (EISIC), 2013 European*, pages 60–66. IEEE, 2013.

[13] A. Morales, J. Fierrez, and J. Ortega-Garcia. Towards predicting good users for biometric recognition based on keystroke dynamics. In *International Workshop on Soft Biometrics*, pages 1–14, 2014.

[14] M. Villani, C. Tappert, G. Ngo, J. Simone, H. S. Fort, and S.-H. Cha. Keystroke biometric recognition studies on long-text input under ideal and application-oriented conditions. In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on*, pages 39–39. IEEE, 2006.

[15] N. Yager and T. Dunstone. The biometric menagerie. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(2):220–230, 2010.