Enhanced Free-Text Keystroke Continuous Authentication based on Dynamics of Wrist Motion

Borui Li, Han Sun, Yang Gao
Binghamton University, SUNY
Binghamton, NY 13902
{bli28, hsun28, ygao44}@binghamton.edu

Vir V. Phoha
Syracuse University
Syracuse, NY 13244
vvphoha@syr.edu

Zhanpeng Jin
Binghamton University, SUNY
Binghamton, NY 13902
zjin@binghamton.edu

***Abstract*—Free-text keystroke is a form of behavioral biometrics which has great potential for addressing the security limitations of conventional one-time authentication by continuously monitoring the user's typing behaviors. This paper presents a new, enhanced continuous authentication approach by incorporating the dynamics of both keystrokes and wrist motions. Based upon two sets of features (free-text keystroke latency features and statistical wrist motion patterns extracted from the wrist-worn smartwatches), two one-vs-all Random Forest Ensemble Classifiers (RFECs) are constructed and trained respectively. A Dynamic Trust Model (DTM) is then developed to fuse the two classifiers' decisions and realize non-time-blocked real-time authentication. In the free-text typing experiments involving 25 human subjects, an imposter/intruder can be detected within no more than one sentence (average 56 keystrokes) with an FRR of 1.82% and an FAR of 1.94%. Compared with the scheme relying on only keystroke latency which has an FRR of 4.66%, an FAR of 17.92% and the required number of keystroke of 162, the proposed authentication system shows significant improvements in terms of accuracy, efficiency, and usability.**

I. Introduction

Conventional one-time authentication methods, such as password, fingerprints, and face recognition have dominated the access authentication of computers today, which however, suffer from severe security limitations. For instance, intruders can potentially access the system before its being locked out or logged off. Continuous authentication has proven to be an effective approach for defending against the authentication attacks on the fly. Through uninterrupted monitoring of the user's behavioral interaction with the computer during the work session, continuous authentication can seamlessly validate the user's presence and detect fraudulent behaviors.

In designing a reliable and effective continuous authentication mechanism, there have been several popular biometric methods known as soft biometrics [1], bio-signal biometrics [2], mouse dynamics biometrics [3], and keystroke dynamics biometrics [4]–[8]. Compared with soft biometrics and bio-signals biometrics, behavioral biometrics such as mouse and keystroke dynamics usually do not need extra sensors adhering to the human body and can achieve a very high recognition accuracy. Other merits of keystroke or mouse behavioral biometrics for continuous authentication include, non-invasiveness, simplicity, cost-effectiveness, and resilience to counterfeit attempts. Existing research on keystroke dynamics primarily focus on the static, fixed-text, one-time authentication, as a complementary security mechanism for password login access. Recently, many research efforts have explored the intrinsic characteristic patterns of free-text keystroke dynamics for continuous authentication [9]–[12].

In this paper, we propose and develop an enhanced continuous authentication system by incorporating the dynamics of both keystrokes and wrist motions. With the increasing advances and popularity of wearable devices, such as wristbands and smartwatches, human wrist motion behaviors can be recorded and associated with the keystrokes while typing. Compared with the traditional keystroke-based authentication, our wrist motion enhanced, free-text keystroke approach can provide higher efficiency and accuracy for continuous authentication, without sacrificing user experience or involving extra overhead. Based upon two sets of features (free-text keystroke latency features and statistical wrist motion patterns extracted from the wrist-worn smartwatches), two one-vs-all Random Forest Ensemble Classifiers (RFECs) are constructed and trained respectively. With a Dynamic Trust Model (DTM) similar to the one in [7], the proposed system can achieve a real-time decision making for nearly every keystroke. To the best of our knowledge, this is the first work that incorporates wrist motion behaviors with free-text keystroke dynamics to implement a true, on-the-fly continuous authentication. The specific contributions of this study include:

- Leveraging the one-vs-all RFECs, the proposed system can effectively detect both *imposters* (a subject who belongs to the training database and whose identity is labeled as the attacker) and *intruders* (a subject of other cases who does not belong to the database).
- The wrist motion behavioral patterns captured by the wrist-worn smart devices (e.g., smartwatches) can significantly improve the detection accuracy of imposters and intruders and also reduce the detection latency.
- The new context-aware keystroke latency feature cell generation and selection method can inspect the latency of every single keystroke in the context, boost the authentication accuracy, and solve the latency fluctuation problem (Different digraphs or trigraphs often have different latencies in different words).

WIFS'2017, December, 4-7, 2017, Rennes, France. 978-1-5090-6769-5/17/$31.00 ©2017 IEEE.

The remainder of this paper is organized as follows. Section II describes the state of the art of keystroke dynamics based continuous authentication. Section III presents the processing flow and methodological details of our proposed continuous authentication approach, including feature extraction, classification, Dynamic Trust Model (DTM), and decision fusion and decision making. The experimental setting and results are discussed in Section IV & V. Finally, Section VI concludes this research and foresees the future work.

II. Related Work

Keystroke dynamics, given its unique advantages of low implementation cost, transparency, and non-invasiveness, have recently emerged as a popular approach for continuous authentication. Monaco *et al.* [4] extract the statistical features of key press duration times and digraph transition times and achieves a very high authentication accuracy based on distance measurement (99% on 14 subjects, 96% on 30 subjects). Shimshon *et al.* [5] use a clustering approach to combine the digraphs which have similar fly-time together as one feature. The experiment on 10 true users and 15 imposters can reach a FRR of 0.63% and FAR of 0.41%. A recent study [6] employs 12 algorithms for free-text keystrokes authentication system on different devices with PC keyboard, soft keyboard and touch keyboard, and conducts a comparative study about different algorithms on different keyboards. Ceker *et al.* [9] implement a two-component Gaussian Mixture model to reach a EER of 0.08% for the 30 users dataset. Brain *et al.* [8] investigate whether typing behavior is affected by the cognitive demands of a given task and the demographic features of the typist. Roth *et al.* [11] design and evaluate an authentication system based on keystroke sounds for both static-text and free-text.

Although many prior work have demonstrated very high recognition rates of keystroke dynamics, most existing research on keystroke-based continuous authentication often neglects a very important aspect in evaluating their effectiveness and performance — detection latency. Without loss of generality, we use the number of keystrokes required before the detection of an imposter/intruder as an indicator of detection latency, instead of the individual-specific typing speed. Bours *et al.* [7] implement a dynamic scoring mechanism to present the estimated trust level while the user is typing to achieve real-time decision-making. The keystrokes used for detection of imposters will be different (less keystrokes are needed for the imposters who type in an obviously different way). In our study, we improve this dynamic trust model by integrating the decision fusion to achieve a comprehensive evaluation on both accuracy and detection latency.

Furthermore, as the recent advances of wearable sensors and smart devices (like smartwatches), many studies have explored the use of those wearable devices for authentication and security purposes. For example, researchers have used the motion sensor data from the smartwatches to infer the specific keystroke actions and even what the user is typing [13]–[15]. Some other work [16] focus on using the motion sensor on smartwatches to build a touch-based authentication system. However, few studies have explored the use of wearable sensors or smartwatches to enhance the keystroke-based biometric authentication on computers. As widely reported in literature [17], [18], gait and arm movement patterns recorded by smartwatches have been successfully adopted as an effective authentication approach. In this study, we propose to investigate the user's wrist motion patterns during the typing process and then incorporate these features into the keystroke-based authentication through a strategic decision fusion.

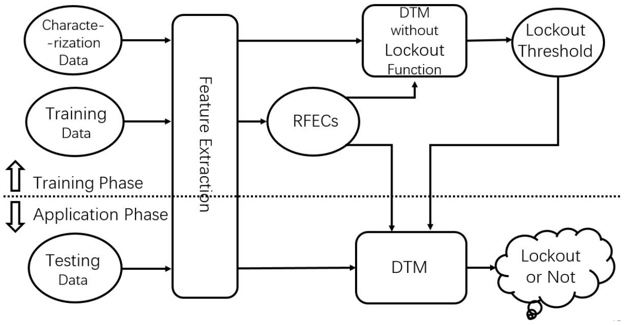

Fig. 1. Data flow diagram of the proposed system.

III. Methods

Figure 1 illustrates the data flow diagram of our proposed keystroke continuous authentication system, including two RFECs and a DTM. During the training phase, through the feature extraction, the *training data* are fed into two separate RFECs for learning digraph/trigraph latency data and wrist motion data. In the meantime, part of the genuine users' data and the imposters' data, which are not included in either training or testing dataset, are taken as the *characterization data*. With the well-trained RFECs and the characterization data, the DTM without lockout function can characterize the lockout threshold for the genuine user and the weights of these two RFECs for further decision fusion. In the application phase, with the two well-trained RFECs and the characterized lockout threshold, the DTM can fuse the decisions from the two RFECs and make a final decision of lockout or not on the *testing data*. It is worthy to note that characterization data will be solely used for generating the classifiers' weights and threshold of genuine users, instead of being used for any application or evaluation purpose. The details of those components in this framework will be discussed in the following.

A. Context-Aware Keystroke Latency Features

The digraph and trigraph latency (i.e., the time elapsed between the two or three keys) have been proven to be the most effective features in keystroke biometrics [4]. However, depending on the specific word where the digraph or trigraph is extracted, the latency can vary significantly [19]. Many existing solutions to address the latency fluctuation issue in free-text keystrokes rely on the post statistical analysis of a large amount of keystrokes [4], [5], [9]. This solution, however, is ineffective in representing the context-aware variations of digraph/trigraph latency and prohibitive for on-the-fly continuous authentication using the real-time keystroke data.

TABLE I
EXAMPLE OF DIGRAPH/TRIGRAPH FEATURE CELL

TH	HE	IN	...	AND	ION
53 ms	101 ms	78 ms	...	201 ms	230 ms
72 ms	101 ms	78 ms	...	201 ms	230 ms

In this study, we propose to alleviate the context-aware latency fluctuation problem by automatically constructing and selecting a set of representative features. Specifically, instead of using the raw latency data of any single digraph or trigraph, we define a feature cell as the latencies of a entire set of bigraphs/trigraphs and create a new feature cell when a new keystroke alters one of the latency value. In this way, the context-aware relationship between the latency of a specific bigraph/trigraph and the latency of the rest bigraphs/trigraphs can be largely retained. Then we utilize the feature importance estimation function by RFEC to choose the optimized feature set for each individual genuine user. Through experiments, it is shown that the feature selection can filter out many noisy features to mitigate the influence of latency fluctuations.

a) Generation of Feature Cells: Prior research [10] used a complete latency feature table, i.e., a 26-by-26 matrix in which each cell corresponds to an English digraph and may contain multiple latency values due to the typing variations. However, each cell is obtained through isolated statistical analysis, that is, without the knowledge of the specific word and context where the bigraph is located. In this study, we propose to use the latency of 50 most frequently used digraphs [TH HE IN ER AN RE ON AT EN ND TI ES OR TE OF ED IS IT AL AR ST TO NT NG SE HA AS OU IO LE VE CO ME DE HI RI RO IC NE EA RA CE LI CH LL BE MA SI OM UR] and 10 most frequently trigraphs [THE AND ING ION TIO ENT ATI FOR HER TER] in the English language [20], as a *feature cluster*. As illustrated in Table I, only if one digraph or trigraph in the feature cell has a new latency (resulted from a new keystroke), a new feature cell will be formed in which all other latency values remain unchanged. For example, if the user types the two keys "T" and "H" sequentially with a latency of 72ms, the TH digraph is updated and a new feature cell is generated. In this way, every individual keystroke will no longer be analyzed in an isolated manner. Instead, it will be collectively inspected along with the latency of those most recent and relevant bigraphs/trigraphs in the context (i.e., those latency values remain in the feature cell). Moreover, this mechanism allows the system to verify the likelihood of every single keystroke in a continuous manner, without sacrificing the generalization performance.

b) Optimization of Feature Cells: Because our feature cells only contain the raw latency of digraphs and trigraphs, the classifier can have more flexibility to extract and capture the characters for different genuine users. To further find the optimal digraphs and trigraphs for different users, we utilize the filtering and retraining model similar to the gene selection in [21]. In this filtering and retraining model, the RFEC is iteratively trained. In each iteration, according to the estimation of significance, part of the less important feature cells are discarded and then a new RFEC is trained with the updated feature set. Through the testing using the characterization data, the RFEC which performs the best and the corresponding feature set are kept as the characterized feature set for this user.

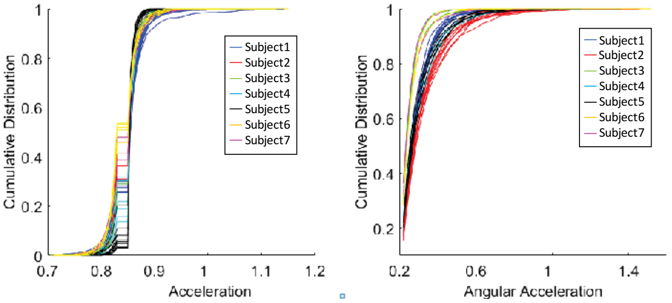

Fig. 2. Cumulative distribution of acceleration and angle acceleration

B. Wrist Motion Features

In addition to the well-acknowledged difference in keystroke latency dynamics, human typing behaviors may also involve the difference in wrist motion patterns. It has been demonstrated that the statistical distributions of acceleration and angular acceleration features captured by a smartwatch when the user is performing gesture-based operations on smartphones are individually unique [16]. Given the observation that every person has their own habit and preference of moving their hands when they are transiting from one key to another, it is thus argued that the individual uniqueness also exists when typing on computers. This assumption can be proved, as shown in Fig. 2, via the cumulative distributions of the acceleration and angular acceleration in the 2-minute free-text typing sessions by 7 subjects (the experimental setup remains the same as the standard protocol described in Section IV). In Figure 2, each color represents one subject and each curve represents one session (each subject took at least three independent sessions). Apparently, most of the curves in the same color are closely clustered, and the curves with different color are distinguishably separated.

We use the accelerometer and gyroscope available in smartwatches to capture the acceleration and angular acceleration data. For feature extraction, we segment the signals into a series of time sliding windows of size 500 and the moving step is set as 50 for continuous monitoring purpose. We extract the probability distribution features of acceleration and angular acceleration from each sliding window, according to their amplitudes computed as $m = \sqrt{x^2 + y^2 + z^2}$. Then two separate filters are designed to remove the noises resulting from some non-typing activities, such as the fast movements while reaching the mouse. The distribution features obtained from the two sensors on the tth window is annotated as:

$$dis_{acc}(t) = pdf_{acc}(t), dis_{gyr}(t) = pdf_{gyr}(t) \quad (1)$$

$$Auth(t) = [dis_{acc}(t), dis_{gyr}(t)] \quad (2)$$

During the data collection process, the subjects are instructed to wear two smartwatches on both hands. The acquired sensory data from both hands are trained and tested

TABLE II
THE KEYSTROKE VERIFICATION PERFORMANCE BASED ON SMARTWATCH MOTION SENSOR ON LEFT AND RIGHT HANDS

Genuine User Index	Left Hand			Right Hand		
	FRR	FAR	OOB Error	FRR	FAR	OOB Error
Subject1	1.43%	11.38%	0.058	**0.49%**	**4.39%**	**0.037**
Subject2	**1.5%**	**3.24%**	**0.049**	0.5%	31.03%	0.097
Subject3	2.67%	16.69%	0.079	**0.34%**	**8.57%**	**0.032**
Subject4	1.96%	18.83%	0.059	**1%**	**0.24%**	**0.028**
Subject5	**4.42%**	**16.37%**	**0.046**	1.21%	28.81%	0.083
Subject6	1.85%	9.41%	0.057	**0.65%**	**7.02%**	**0.030**
Subject7	4.86%	21.41%	0.088	**0.02%**	**14.08%**	**0.006**

separately. Through experiment, we found that the better performance is always obtained from the hand with a smaller Out-Of-Bag (OOB) error as Table II (the definition of OOB will be introduced in section III-C). Thus a more suitable hand to wear smartwatch for different individuals can be chosen according to the corresponding OOB. In the following study, all users will always wear the smartwatch on the preferred hand when they are typing.

C. Random Forest Ensemble Classification

Due to the high variance of keystroke dynamics and the large feature dimension, traditional classifiers are vulnerable to overfitting. Thus, it is necessary to seek a classifier which can handle the high variance data, be resistant to overfitting, and automatically select the most important and suitable features. In this study, the Random Forest Ensemble Classifier (RFEC) is proposed which is an aggregation of all base classifiers $\{h(X, \Theta_k), k = 1, 2, ..., K\}$. Θ_k is the parameter set for each individual decision tree, and K represents the number of trees. The construction process of the RFECs and the principle of feature importance estimation by RFEC are as follows:

1) K subsets are randomly extracted from the original training data set by the Bootstrap algorithm, based upon which K decision trees are trained. For each extraction, the subset which is not chosen is named as the "Out-Of-Bag" (OOB) data (OOB_k);
2) For each decision tree, if there are N features, each time M features are selected ($M \leq N$). The decision tree keeps choosing and splitting out the "most significant" features ($F \in M$) until it is fully grown;
3) Aggregation at the decision level is realized through the majority voting of these K decision trees with their corresponding weights w_k ($w_k \propto \frac{1}{E(OOB_k)}$, in which E is the error estimation function);
4) The feature importance estimation is calculated by permuting values of one feature each time across OOB data and measuring how worse the MSE of RFEC predictions becomes after the permutation.

Fully growing each decision tree allows RFEC to be capable of processing high dimensional features. The majority voting with w_k aggregation method makes RFEC resistant to overfitting and effective in handling high variance data. With the importance estimation of the random forest, a feature optimization strategy is presented (see Section III-A).

D. Decision Fusion and Dynamic Trust Model

With the goal of combining the classification results from the two RFECs, a Dynamic Trust Model (DTM) [7] is adopted and modified using the decision fusion [22] to improve the efficiency and stability of intrusion detection in the proposed authentication system. According to the *FAR* and *FRR* of the two RFECs on the characterization data from both the genuine user and the imposters, the weights a_i for the N classifiers are calculated as follows (where H_i is the estimation hypothesis made by the ith classifier, 1 represents the genuine user while 0 represents the attacker):

$$a_i = \begin{cases} log\frac{1-FRR_i}{FAR_i}, & \text{if } H_i = 1 \\ log\frac{1-FAR_i}{FRR_i}, & \text{if } H_i = 0 \end{cases} \tag{3}$$

The parameters u_i is defined under different hypothesis:

$$u_i = \begin{cases} 1, & \text{if } H_i = 1 \\ -1, & \text{if } H_i = 0 \end{cases} \tag{4}$$

With the parameters u_i and weight a_i, the final fused decision $f(u_1, ...u_n)$ from N classifiers can be made as follow:

$$f(u_1, ...u_n) = \begin{cases} 1, & \text{if } \sum_{i=1}^{n} a_i u_i > 0 \\ -1, & \text{otherwise} \end{cases} \tag{5}$$

The algorithmic details of DTM is presented in Algorithm 1. The trust score's range is from 0 to 100. During the training phase, with the fused decisions on the current genuine user's characterization data, through the DTM with lockout function (i.e., lines 8-14 in Algorithm 1), the minimum score is taken as the lockout threshold for the genuine user. In the application phase, the trust score of the system is initially set as 100. When a new sample of both the keystroke latency and the wrist motion pattern arrives, the decision fusion is performed to determine if this sample belongs to the genuine user or the imposters/intruders, and accordingly the trust score is incremented or decremented by one. If the trust score drops below the lockout threshold, the current user will be locked out immediately, and the number of keystrokes used for this detection is recorded for further performance evaluation. The system is capable of evaluating the validity of the current user nearly for every keystroke sample.

Algorithm 1: Dynamic Trust Model

```
Input: $S$ represents the Trust Score, $T$ represents the Lockout
       Threshold, $E_f$ is the estimation result from the decision fusion
       of two RFECs
Output: $Invalid\_User\_Detection$ ($True$ or $False$)
1  Set the initial trust score $S = 100$;
2  if $S > 0$ and $E_f$ is invalid user then
3      $S - -$;
4  else if $S < 100$ and $E_f$ is true user then
5      $S + +$;
6  else if $S > T$ then
7      $Invalid\_User\_Detection \leftarrow False$;
8      Go back to Step 2 with the updated $E_f$ by RFEC based on new
         wrist motion and latency data;
9  else
10     $Invalid\_User\_Detection \leftarrow True$;
11     System lock out, the number of keystrokes for the detection is
         recorded;
12 end
```

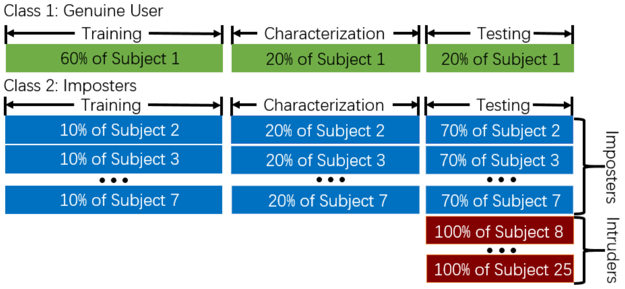

Fig. 3. Data Structure

IV. Experimental Setting

A total of 25 human subjects in age of 23-33 years old (5 female and 20 male) were recruited to participate in the experiments. 7 subjects (Subjects 1-7) were taken as the genuine user and imposters, and the rest 18 subjects (Subjects 8-25) were the intruders. Each time, one of 7 subjects was randomly designated as the genuine user, and the rest 6 subjects were designated as the imposters. For the genuine user and imposters, the data collected from each subject contained 9,030 keystrokes on average from five sessions during a period of two weeks. For each intruder, an average of 1,784 keystrokes were collected from one session. During the training, 60% of data from the genuine user and 10% of data from each imposter were fed into the classifier. 20% of data from the genuine user and imposters were used to characterize the classifiers' weights and the lockout threshold. We then used the rest 20% of the genuine user's data, 70% of the imposters' data and the entire data from each intruder to evaluate system's performance. The Internal Review Board of Binghamton University approved the experimental protocol.

Figure 3 describes the specific composition of the training dataset, characterization dataset, and testing dataset. The different roles of the subjects are defined as follows:

- ***Genuine User***: The authorized person who can access the system.
- ***Imposters***: The people who attempt to illegally access the system and whose data has been partially known by the classifier.
- ***Intruders***: The people who attempt to illegally access the system but whose data is entirely unknown by the classifier. Intruder data is solely used for testing purposes.

In the experiments, subjects were instructed to type a transcript randomly chosen by the subjects themselves. Correction of typing mistakes is allowed but not required. Subjects were required to choose a different transcript for different sessions. We used the computers with the identical configuration (Intel i7-6700 CPU @ 3.4GHz, 16GB RAM) and the same type of keyboards (Dell SK-8175 USB Keyboard 104 Key Standard QWERTY Layout). The subjects were seated on the same type of office chairs with a fixed height of 40 cm during the entire experimental sessions. The keystroke latency data (in milliseconds) was collected through a custom, webpage-style software programmed in HTML, PHP and JavaScript. The smartwatches used in experiments are Sony SWR50 1.6-Inch Display SmartWatch 3 running Android Wear. The accelerometer and gyroscope recordings (at a frequency of 50 Hz) were collected through a custom program programmed in ActionScript 3.0. During the experimental sessions, all the subjects were required to wear the smartwatches in the traditional position on top of the wrist.

TABLE III
Performance Based on Keystroke Latency

Genuine Users	FRR	FAR			Avg # of Keystrokes for Detection of Attacks		
		Imposters	Intruders	All	Imposters	Intruders	All
Sub1	1.7%	18.4%	34.8%	22.6%	60	102	79
Sub2	0.8%	21.9%	43.7%	35.3%	77	158	94
Sub3	3.9%	7.2%	36.9%	15.5%	70	113	79
Sub4	0.7%	8.4%	11.9%	9.6%	37	60	43
Sub5	8.9%	23.3%	29.5%	26.3%	333	400	376
Sub6	8.9%	5.2%	7.5%	6.6%	284	293	286
Sub7	1.9%	3.7%	5.1%	4.1%	33	44	39
Ave	4.6%	12.7%	28.8%	17.9%	128	195	162

TABLE IV
Performance Based on Keystroke Latency and Wrist Motions

Genuine Users	FRR	FAR			Avg # of Keystrokes for Detection of Attacks		
		Imposters	Intruders	All	Imposters	Intruders	All
Sub1	1.5%	0.8%	1.4%	1.1%	61	72	67
Sub2	1.9%	0.9%	1.6%	1.2%	33	43	38
Sub3	3.1%	0.0%	1.7%	0.8%	46	52	49
Sub4	1.0%	0.1%	1.5%	0.6%	23	30	28
Sub5	9.1%	2.7%	4.4%	4.0%	116	122	120
Sub6	0.6%	3.2%	8.4%	7.3%	45	54	52
Sub7	2.2%	0.0%	1.2%	0.8%	33	42	38
Ave	1.8%	1.0%	2.4%	1.9%	51	60	56

V. Results

Three performance indicators are used in the evaluation, including the False Acceptance Rate (FAR), the False Rejection Rate (FRR), and the average number of keystrokes required for detection of each unauthorized access. Specifically, in this study, FRR refers to the percentage ratio between the keystroke test samples of the genuine user that are falsely classified and labeled as the 'attacker' against the total number of keystroke samples from the genuine user. FAR is defined as the percentage ratio between the keystroke test samples of the imposters/intruders that are falsely classified and accepted as the 'true user' against the total number of keystroke samples from the imposters and intruders accessing the system. Based on the implemented DTM which locks out any unauthorized user when the trust score drops below the lockout threshold, the number of keystrokes required for detection of this unauthorized access is recorded as an indicator of the efficiency of the proposed approach. It is obvious that less number of required keystrokes will result in a much faster and earlier detection of unauthorized access to the system.

There are two sets of experiments in this study. The first experiment set is to evaluate the performance of the free-text keystroke continuous authentication approach based on solely digraphs and trigraphs latency features. As shown in Table III, leveraging our novel context-aware keystroke latency feature cell generation and feature selection, the proposed approach can achieve a rather low FRR level for genuine users ($<5\%$) and an acceptable FAR level for imposters ($\sim10\%$, whose data was partially learned by the classifier). However, due to the uncertainty of the intruders' data (which was not introduced to and learned by the classifier), unsurprisingly the system has a relatively higher FAR for intruders (∼30%). The number of keystrokes required for detection of unauthorized accesses ranges from 39 to 376, with an average of 162. This performance is comparable to the results reported in the literature [7]. Nevertheless, our approach demands much less training samples, by taking advantage of the proposed context-aware keystroke latency feature and RFEC's capability on dealing with high-dimensional, high-variance data.

The second experiment set is to evaluate the performance of the enhanced free-text keystroke continuous authentication approach based on both keystroke latency and wrist motion behaviors. According to Table IV, it is shown that the proposed approach can significantly improve the performance of continuous authentication from all three aspects: average FRRs for genuine users decrease to a level below 2%; average FARs for both imposters and intruders decrease to a level below 3%; and the average number of keystrokes required for detection of attacks are only around 56. The results indicate that, the wrist motion behaviors during typing contain individual-specific characteristics which can help verify the user's identity when combined with the keystroke dynamics. It is worthy to mention that, because our method relies on the statistical distribution of wrist motions for feature extraction, a minimum of 500 data samples from the smartwatch (window size) are necessary at the beginning of the authentication process in order to obtain the wrist motion patterns. That is, under a frequency of 50 HZ, around 10 seconds of typing activities which correspond to 20∼25 keystrokes at a normal typing speed need to be recorded at the beginning of each authentication process. After that, leveraging the sliding window and context-aware keystroke latency feature extraction and feature selection strategies, our proposed system can achieve real-time analysis of keystroke dynamics and wrist motion behaviors, resulting in the on-the-fly continuous authentication.

VI. Conclusions

Keystroke dynamics have been extensively investigated as an effective behavioral biometric approach, especially for increasing significant continuous authentication. Most of existing research focus on the fixed-text keystroke authentication, which however, significantly limits its applicability and user acceptance in real-world scenarios. In this study, we propose a novel enhanced free-text keystroke continuous authentication approach based on both keystroke latency patterns and wrist motion behaviors captured by wrist-worn smartwatches. A new feature cell generation and optimization strategy is also proposed to maximize the context-aware verification capability for every single bigraph/trigraph and minimize the influence of latency fluctuations. The experimental results show that, the proposed enhanced keystroke authentication framework can significantly improve the accuracy and efficiency of detection of unauthorized access, while ensuring the continuous monitoring of keystroke dynamics on the fly.

Acknowledgment

This work was supported in part by the National Science Foundation under grants SaTC #1422417, #1527795, and #1564046.

References

[1] A. Dantcheva, P. Elia, and A. Ross, "What else does your biometric data reveal? A survey on soft biometrics," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 3, pp. 441–467, 2016.

[2] M. V. Ruiz-Blondet, Z. Jin, and S. Laszlo, "CEREBRE: A novel method for very high accuracy event-related potential biometric identification," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 7, pp. 1618–1629, 2016.

[3] N. Zheng, A. Paloski, and H. Wang, "An efficient user verification system using angle-based mouse movement biometrics," *ACM Trans. Inf. Syst. Security*, vol. 18, no. 3, p. 11, 2016.

[4] J. Monaco, N. Bakelman, S. Cha, and C. Tappert, "Developing a keystroke biometric system for continual authentication of computer users," in *Proc. European Intell. Security Inform. Conf.*, 2012, pp. 210–216.

[5] T. Shimshon, R. Moskovitch, L. Rokach, and Y. Elovici, "Clustering di-graphs for continuously verifying users according to their typing patterns," in *Proc. IEEE 26th Convention Electr. Electron. Engineers in Israel*, 2010, pp. 445–449.

[6] P. Kang and S. Cho, "Keystroke dynamics-based user authentication using long and free text strings from various input devices," *Information Sciences*, vol. 308, pp. 72–93, 2015.

[7] P. Bours, "Continuous keystroke dynamics: A different perspective towards biometric evaluation," *Information Security Technical Report*, vol. 17, no. 1, pp. 36–43, 2012.

[8] D. G. Brizan, A. Goodkind, P. Koch, K. Balagani, V. V. Phoha, and A. Rosenberg, "Utilizing linguistically enhanced keystroke dynamics to predict typist cognition and demographics," *Int'l J. Human-Computer Studies*, vol. 82, pp. 57–68, 2015.

[9] H. Ceker and S. Upadhyaya, "Enhanced recognition of keystroke dynamics using Gaussian mixture models," in *Proc. IEEE Military Communications Conf. (MILCOM)*, Oct 2015, pp. 1305–1310.

[10] M. S. Hossain, K. S. Balagani, and V. V. Phoha, "New impostor score based rejection methods for continuous keystroke verification with weak templates," in *Proc. IEEE Int'l Conf. BTAS.* IEEE, 2012, pp. 251–258.

[11] J. Roth, X. Liu, A. Ross, and D. Metaxas, "Investigating the discriminative power of keystroke sound," *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 2, pp. 333–345, 2015.

[12] J. Huang, D. Hou, S. Schuckers, and S. Upadhyaya, "Effects of text filtering on authentication performance of keystroke biometrics," in *Proc. IEEE Int'l Workshop Inf. Forensics and Security (WIFS)*, 2016, pp. 1–6.

[13] A. Sarkisyan, R. Debbiny, and A. Nahapetian, "Wristsnoop: Smartphone PINs prediction using smartwatch motion sensors," in *Proc. IEEE Int'l Workshop Inf. Forensics and Security (WIFS)*, Nov 2015, pp. 1–6.

[14] X. Liu, Z. Zhou, W. Diao, Z. Li, and K. Zhang, "When good becomes evil: Keystroke inference with smartwatch," in *Proc. 22nd ACM SIGSAC Conf. Computer and Commun. Security (CCS)*, 2015, pp. 1273–1285.

[15] C. Wang, X. Guo, Y. Wang, Y. Chen, and B. Liu, "Friend or foe?: Your wearable devices reveal your personal pin," in *Proc. 11th ACM Asia Conf. Comput. Commun. Security (ASIACCS)*, 2016, pp. 189–200.

[16] W.-H. Lee and R. Lee, "Implicit sensor-based authentication of smartphone users with smartwatch," in *Proc. ACM Workshop on Hardware and Archit. Support for Security and Privacy (HASP)*, 2016, pp. 1–8.

[17] A. H. Johnston and G. M. Weiss, "Smartwatch-based biometric gait recognition," in *Proc. IEEE Int'l Conf. BTAS*, Sept 2015, pp. 1–6.

[18] G. Cola, M. Avvenuti, F. Musso, and A. Vecchio, "Gait-based Authentication using a wrist-worn Device," in *MobiQuitous*, 2016, pp. 208–217.

[19] C. Terzuolo and P. Viviani, "Determinants and characteristics of motor patterns used for typing," *Neurosci.*, vol. 5, no. 6, pp. 1085–1103, 1980.

[20] M. S. Mayzner and M. E. Tresselt, "Tables of single-letter and digram frequency counts for various word-length and letter-position combinations." *Psychonomic Monograph Supplements*, 1965.

[21] R. Díaz-Uriarte and S. A. De Andres, "Gene selection and classification of microarray data using random forest," *BMC Bioinformatics*, vol. 7, no. 1, p. 3, 2006.

[22] L. Fridman, A. Stolerman, S. Acharya, P. Brennan, P. Juola, R. Greenstadt, and M. Kam, "Multi-modal decision fusion for continuous authentication," *Comput. & Elect. Eng.*, vol. 41, pp. 142–156, 2015.