Recognizing Your Touch: Towards Strengthening Mobile Device Authentication via Touch Dynamics Integration

Recognizing Your Touch: Towards Strengthening Mobile Device Authentication via Touch Dynamics Integration

Pin Shen Teh, Ning Zhang
School of Computer Science
University of Manchester
Oxford Rd, Manchester, M13 9PL, UK
{tehp, nzhang}@cs.man.ac.uk

Andrew Beng Jin Teoh
College of Engineering
Yonsei University
Seoul 120-749, Republic of Korea
bjteoh@yonsei.ac.kr

Ke Chen
School of Computer Science
University of Manchester
Oxford Rd, Manchester, M13 9PL, UK
ke.chen@manchester.ac.uk

ABSTRACT

Mobile devices have become an integral part of our routine activities. Some of the activities involve the storage or access of sensitive data (e.g. on-line banking, paperless prescription services, etc.). These mobile electronic services (e-Services) typically require a method to securely identify and authenticate a claimed identity. Currently, e-Services typically use a knowledge-based authentication method by demonstrating the knowledge of a secret (e.g. password), but it is vulnerable to a number of security attacks, e.g. shoulder spoofing and brute force attacks. To thwart the attacks and to make the authentication method more secure, this paper describes our efforts in investigating the benefits of integrating touch dynamics biometrics, into a PIN-based authentication method. It reports the collection of a comprehensive reference dataset from 150 subjects, the extraction of feature data from the dataset, and the classifications and the use of the feature data to identify a user. Experimental results show that, even when the PIN is exposed, 9 out of 10 impersonation attempts can be successfully detected.

Categories and Subject Descriptors

D.4.6 [**Operating System and Protection**]: Access controls, authentication; H.5.2 [**Information Interfaces and Presentation**]: Input devices and strategies

General Terms

Security, Human Factors, Verification.

Keywords

Touch Dynamics, Mobile Authentication, Keystroke Dynamics, Benchmark Dataset, Behavioral Biometrics.

1. INTRODUCTION

The use of mobile devices in handling our daily activities or carrying out daily tasks is becoming very popular. In the last decade, mobile devices have outgrown its initial usage as a form of voice and text communications to that of more advanced applications, such as web browsing, m-commerce and e-mail communications. This is particularly the case with the rapid growth and widespread use of smartphones and digital tablets. The processing capability of these devices has advanced up to the point that most digital activities that can be accomplished on workstations or laptops can also be performed on these portable devices. Routine activities, such as personal and corporate e-mail communications, on-line banking transactions, accessing paperless prescriptions services, route navigation, etc. can be carried out ubiquitously with these devices.

The increased usage and reliance of these devices also imply that they increasingly handle, manage and process private and sensitive data. Therefore, more stringent security services (i.e. security measures) should be embedded in mobile devices. One of these services is user authentication, i.e. how to securely verify a claimed identity. Authentication is the first-line of defense in any computer system or device as it is a pre-requisite for several other security services such as authorization and accountability. In a mobile device context, authentication is typically achieved via a knowledge-based authentication method, and with this method, a user proves their identity by demonstrating the knowledge of a secret. This secret could be a PIN (personal identification number), a password, a shared secret (which is similar to password, but with higher entropy) or a private key corresponding to a public key certified in a digital certificate. The use of secrets is vulnerable to a number of security attacks, e.g. theft of a mobile device, shoulder spoofing and brute force attacks. To thwart the attacks or to make the attacks harder to succeed, we have been working on integrating biometric-based with knowledge-based authentication methods.

Prior to the emergence of touch dynamics, keystroke dynamics (the interaction between human input and physical keyboard) on either workstation [3, 25] or a web-based environment [6, 31] have been the major topic of research. A comprehensive review of keystroke dynamics [33] reveals that, since 2007, there has been growing efforts on examining the possibility of applying the concept of keystroke dynamics in mobile platforms. Earlier work was mainly focused on mobile devices with physical keypads [4, 7, 13, 21] and early generation smartphones [35], but recently, the work has been geared towards touchscreen devices [29, 27, 8].

Touch dynamics refers to the process of measuring and assessing human touch rhythm on mobile devices, such as digital tablets, smartphones, or touchscreen panels. When a human interacts with a mobile device, a digital signature is generated. The signatures generated from interactions by different individuals are believed to be rich in discriminative properties, which are fairly unique to each individual and hold potential as personal identifiers.

This technology can be integrated with existing knowledge-based authentication to form a so-called multi-factor authentication mechanism, which strengthens the security of mobile device. A touch dynamics based authentication method can be implemented by employing existing sensors embedded in a mobile device (without the need for any additional hardware), making the implementation comparatively cheaper than other biometrics systems. The availability of higher resolution sensors in recent mobile devices provide added opportunities to the development of touch dynamics biometrics by allowing the extraction of more discriminative feature data types.

Mobile devices usually operate in an on-the-go fashion, so their lighting and background noises may change continuously. In this regard, the acquisition of touch dynamics biometrics feature is less affected by these factors than iris (e.g. undesirable in low light condition) or voice biometrics (e.g. susceptible to background noise). Apart from that, it requires little intervention by the user as it is an integral part of a user's mobile input activities. For these reasons, a touch dynamics biometrics system may be more acceptable by the general public than other biometrics systems.

As touchscreen devices only came along not long ago, there are still limited benchmark datasets publically available; so far, we are only able to find three such datasets, but only one of the datasets uses PIN based input and none is conducted on a widescreen digital tablet (to be discussed in Section 6.1). The creation and collection of live data is a time and resource consuming process, and this may be the reason for the lack of open datasets [10]. However, the research, design and performance evaluation of touch dynamic biometrics systems require the availability of such benchmark datasets.

This paper reports our effort on the creation and use of a touch dynamics dataset to investigate the benefits of integrating touch dynamics with a PIN-based authentication method. It reports the collection of a comprehensive reference dataset consisted of two sets of input PINs collected from 150 subjects, the extractions of feature data from the dataset, and these feature data are timing, finger touch size and pressure feature data. Thirdly, it uses three light-weight algorithms to classifier these feature data types and used them to identify a user. Experimental results show that, with the integration of touch dynamics biometrics, even when the PIN is exposed, 9 out of 10 impersonation attempts can be successfully detected.

The structure of this paper is as follows. Next section explains experimental setup and the methods and procedures used in the data acquisition phase. Section 3 describes the properties of the dataset. Section 4 describes potential feature data that can be extracted from the dataset and, for proof-of-concept, we illustrate how the captured feature data may be used for authentication in the mobile context (Section 5). Section 6 compares our dataset with other public datasets and critical analyses related work in the context. Finally, Section 7 concludes the paper and outlines our future work.

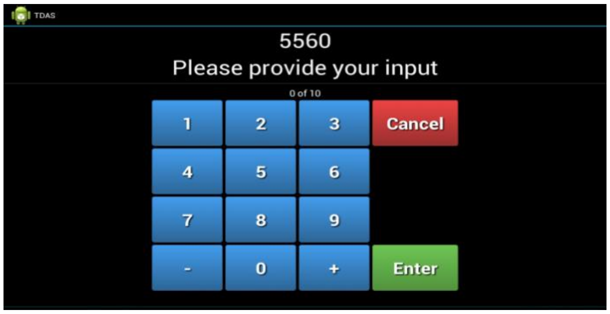
2. EXPERIMENT SETUP

Experiments should be conducted with well-defined protocols and procedures, as, in this case, we can minimize external factors from inflicting noise into the data collected [15]. In this section, we discuss how data collection device, environment and intervals are determined.

2.1 Data Collection Device

As mentioned above, there are different types of mobile devices, e.g. featured phones, smartphones, digital tablets and laptops. Most research works conducted in touch dynamics employed various mobile phones. In comparison with mobile phones, digital tablets have a relatively larger screen resolution, which means that a higher subject input variation, and therefore a better feature discrimination, can be captured [27]. For this reason, we have chosen to use a digital tablet as our data collection device. The device is a commercial off-the-shelf Samsung Galaxy Tab 10.1 (GT-P7510) digital tablet. It has a 10.1" widescreen, and is powered by 1GHz dual-core processor and equipped with a 1-GB RAM. The entire data collection process was performed using this tablet. The justification for using a predefined device, rather than subject specific devices, was to remove uncontrolled variables such as subject preferences, program compatibility and functionality differences. In this way, the results obtained from the experiment can better reflect the discriminative power of touch dynamics feature data and the classification algorithm used.

The device runs under Android 4.0.4 (Ice Cream Sandwich) and a data collection tool that was developed using Java and Android API Level 15. Majority of research works on touch dynamics are carried out on the Android platform. This is because Android is an open source mobile operating system, which gives application developers a greater control and customization for their application developments, and researchers a cheaper option to conduct their experiments. Figure 1 shows a screen capture of the data collection tool.



5560
Please provide your input
0 of 10

Figure 1. Sample data collection screen

2.2 Data Collection Environments

There are three main environments in which data may be collected: (i) where ever subjects are while they carry out their activities as usual; (ii) under a controlled laboratory environment in a fixed location; or (iii) in multiple fixed locations. The first option is expensive, as due to the ubiquitous nature of mobile devices, subjects are likely to be on-the-move and following the subjects while collecting the data may not be convenient and could be costly. The second option is least costly in terms of setting up and running the experiments, but the data collected may not give a true reflection of real world scenarios. To balance costs/feasibility with real-life situations, we have chosen to use the third option, i.e. we let subjects to choose their preferred locations where their natural touch dynamics are extracted. The locations used included offices, homes, inside cars, classrooms, café, and public areas.

2.3 Collection Method

Data should be collected when subjects are in a stable state, i.e. after they are familiar with the device input facility and the data collection procedure, as, otherwise, data collected may not properly capture subjects' input features. Improperly captured data may increase false positive and false negative rates when they are used to authenticate the subjects. According to [24], input patterns, styles or speeds can vary and stabilize over time. To ensure data are collected after the patterns, styles and speeds are stabilized, and to reduce the effect of subjects' unfamiliarity with the input facility and procedure, one of the two approaches may be used. The first one is to divide a data collection session into several sub-sessions that are separated by selected time frame, e.g. 4 sub-sessions each with 1 week apart, and the data collected in multiple sub-sessions are cumulatively merged into a single set. This approach provides a good level of accuracy, but may suffer from a higher dropout rate [19], as we could not expect participants be obliged with multiple session time commitment especially on a voluntary basis. As a result, the sample size of the dataset may be reduced. The second approach is to collect data in a single session, but subjects are asked to familiarize with the input facility and procedure as many times as necessary before collecting their data. This approach is commonly used in experiments reported in the literature [5, 16, 18]. An entire data collection process takes an average of 15 to 20 minutes (in addition to time taken to familiarize the input device and procedure). In our experiments, we have taken approach two described above.

3. DATASET

This section describes the dataset in detail. It provides justifications in terms of how subjects are selected, what data have been collected and how it is stored and represented.

3.1 Subject Size

The subject size refers to the number of subjects from whom data are collected. Typically a subject size of greater than 100 subjects is regarded as a large subject size [33]. Using a larger subject size can provide more data to verify the scalability of a chosen classifier as mentioned in [7]. Most of the relevant works in the touch dynamics domain were carried out with a subject size smaller than this value. Only a handful published works [9, 29, 34] uses a subject size greater than 100 subjects. However, in the latter group of works, the subjects involved were restricted to a certain population and the datasets were not made publically available for evaluation. To overcome these restrictions, we, at the time of this writing, have collected touch dynamics data from 150 subjects. The dataset is shared in 3 different packages each consisted of an incremental size of 50 subjects. In this way, any researcher who may be interested in using the dataset has the option to conduct comparisons between different subject sizes within the same subject grouping.

3.2 Subject Demography

To reflect real world situations as much as possible, the demography of the subjects taking part in data collection should be as diverse as possible. In other words, people from different age groups, of different genders and with different device usage frequencies (this indirectly correlates with device familiarity) should be represented as much as possible. Unfortunately, more often than not, subjects recruited for experiments published in literature were confined to people within a research institute or academia. [17] is the only piece of work in literature we are able to locate, which involved the use of subjects from diverse population. Inspired by this work, we have made the best effort to reach out to the general public within our available resources. Table 1 summarized the demography of the subjects involved in our dataset.

Table 1. Subject demography of our dataset

Properties		Details
Subjects		150
Population (groups)	Academia	18
	Public	132
Age (years)	<20	28
	20-40	66
	>40	56
Daily Usage Frequency	Rare	49
	Average	32
	Often	69
Gender	Male	45
	Female	105
Hand Preference	Left-hand	14
	Right-hand	136

3.3 Input Type

PIN input has been the most widely used authentication method for mobile devices, so we first focused on a 4-digit numerical input ("5560"), and then a 16-digit numerical input ("1379666624680852"). The use of two different PIN lengths allows experimental evaluations of the effects of different input string lengths. These two predefined numbers were carefully chosen with the following key positioning combination strategies.

- Apart: keys are separated by at least one key apart.
- Repetition: reoccurrence of identical key.
- Adjacent: keys located diagonally to each other.
- Sequence: keys situated horizontally or vertically to each other.

These positioning strategies were used to spread the variety of input strings. A graphical illustration of the approach is depicted in Figure 2.

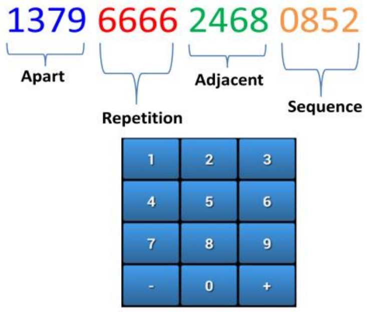


Figure 2. Four different key positioning strategies

Predefining a universal input string across every subject offers a significant advantage of increasing the total number of impersonation samples available for our testing phase (Section 5.2) without need for collecting additional data.

3.4 Sample Size

More complex data classification algorithms, such as neural networks and support vector machines, usually require the use of a large training sample size to achieve a significant performance. On the other hand, simpler algorithms such as k-nearest neighbor may work well on small sample input [13]. It is impractical to expect a large number of repeated inputs from subjects during the enrolment stage. Therefore, in this data collection process, subjects were only required to repeat each input string for 10 consecutive times, resulting in 20 samples per subject (10 for short digit samples and 10 for long digit samples, respectively). In terms of error handling, any input mistake made by a subject was automatically discarded and the subject was prompted to repeat that particular input sample instance, which is a common practice as explained in the literature [4, 20, 26].

3.5 Raw Feature Representation

A number of application programming interfaces (APIs) have been used to capture the subjects' feature data. In detail, each single screen touch event (finger touching down or lifting up from the touchscreen) is detected by the onTouchListener API. The timestamps of each key press and release were logged by invoking the nanoTime() API. This API returns the most precise timer available on the device's system (in nanoseconds). Usually, a human's tapping speed is much lower than this pace, this value can be normalized to the desired resolution upon feature extraction. We also use the API functions under the MotionEvent class, getSize() and getPressure(), to retrieve the values of finger circumference and pressure, respectively, when a subject touches the screen. These functions return a normalized decimal value between 0 and 1. However, we have noticed that getPressure() always return a value of 1.0. We have tried to resolve this issue but no success. However, as some other devices also encounter the same problem, we anticipate that this problem will be resolved by the mobile operating system's provider in their subsequent API version update. Each completed touch event on a key generates two timestamps (t_{press} and $t_{release}$), a finger touch size (ps) and a pressure value (pv). These events play a major part in the feature extraction and template generation process. For each repeated input sample (r) of raw touch dynamics data, feature data and the particular key press (k_{press}) and key release ($k_{release}$) were recorded and stored in a separate file for each subject using the format shown below:

$$\{r\},\{k_{press}\}\{t_{press}\}\{k_{release}\}\{t_{release}\}\{ps\}\{pv\}$$

4. METHODOLOGY

4.1 Feature Extraction

Two types of features are captured. These are timing data and finger touch size (ps), both are captured during subject interactions with the input keys. The acquisition of ps is straight forward, obtained directly from the return value of an Android API function without further customization. However, for timing data acquisition some manipulation to the touch event timestamps is required. The timing data extracted can be further divided into two categories: (1) Dwell Time (DT), i.e. the time duration for the touch action of the same key (also known as interval, press or hold time); (2) Flight Time (FT), i.e. the time interval between the touch actions on two successive keys (also known as latency). As shown in Figure 3, there are four variants of FT.

It is interesting to point out that, according to [30], there is a possibility that FT_1 may have a negative value. This case occurs whenever a subject presses the next key before releasing the previous one. This case occur when using a computer keyboard, but it is unlikely to occur on touchscreen inputs due to the physical and geometrical size of on-screen keys. It is also much less likely for a subject to use multiple fingers simultaneously when providing their inputs. As a result, the chances of pressing the next key before releasing the previous one is significantly reduced or in some cases eliminated.

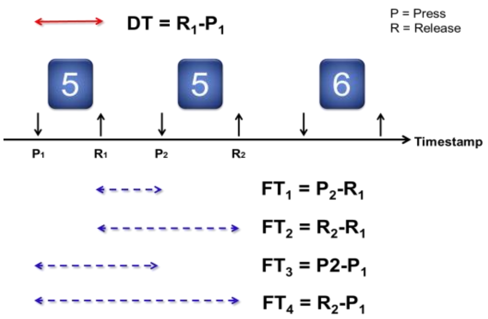


Figure 3. Types of timing feature data extracted

4.2 Template Generation

Template generation is a process by which a subject's touch feature samples are combined and transformed into a compact yet representative structure. A subject's template should uniquely capture the subject's touch feature. For each subject, we use six feature data types, and, for each feature data type, a template data is generated. This template data comprise of two items, a mean (μ) value and a standard deviation (σ) value. The equations below show how the template data for a feature data type, DT, is calculated. For example, given a training sample set of n number of DT, the template data for DT is calculated as:

$$\mu = \frac{1}{n} \times \sum_{i=1}^{n} DT_i$$

$$\sigma = \sqrt{\frac{1}{n} \times \left(\sum_{i=1}^{n} DT_i^2 - \frac{\left(\sum_{i=1}^{n} DT_i\right)^2}{n} \right)}$$

The same procedure and computation are applicable to other feature data types.

4.3 Classifier

We use three matching functions to, respectively, compute and compare the likeliness of a test sample against a reference template feature. The likeliness, which is measured in terms of a similarity score (s), is computed by feeding the test sample value (τ) of a feature vector element of position (i) and the value of μ and σ from the reference template into each function, i.e. $s_i = f(\tau_i, \mu_i, \sigma_i)$. The three matching functions used are Gaussian Estimation (GE), Z-Score (ZS), and Standard Deviation Drift (SD), as given below:

$$f_{GE}(\tau, \mu, \sigma) = e^{-\frac{(\tau-\mu)^2}{2\sigma^2}}$$

$$f_{ZS}(\tau, \mu, \sigma) = \frac{|\tau - \mu|}{\sigma}$$

$$f_{SD}(\tau, \mu, \sigma) = e^{-\frac{|\tau-\mu|}{\sigma}}$$

We calculate a similarity score for each individual element within the intended feature vector. Then we compare these scores against an empirical threshold (φ) to make a partial decision (D_i) for each feature data element of position (i) in that feature vector, i.e.:

$$D_i = \begin{cases} 0, & s_i \le \varphi \\ 1, & s_i > \varphi \end{cases}$$

Here, 1 and 0, respectively, indicate acceptance and rejection. A final decision is then made to determine if a test sample belongs to the reference template, and this is done by using the formula:

$$D_{final} = \begin{cases} accept, & \frac{\sum_{i=1}^{n} D_i}{n} \geq 0.5 \\ reject, & \frac{\sum_{i=1}^{n} D_i}{n} < 0.5 \end{cases}$$

where n refers to the total elements in the feature vector considered, and D_{final} is the final acceptance or rejection decision of a given test sample.

5. PERFORMANCE ANALYSIS

5.1 Evaluation Criteria

Two main metrics are used to measure the accuracy level of a biometrics authentication system. These are the False Rejection Rate (FRR) and False Acceptance Rate (FAR). FRR is the percentage ratio of the number of legitimate subjects who are falsely rejected against the total number of legitimate subjects. FAR is the percentage ratio of the number of illegitimate subjects who are falsely accepted against the total number of illegitimate subjects. There is another performance metric, called Equal Error Rate (EER), which is derived from FRR and FAR. EER is obtained by plotting a graph for each of FRR and FAR against a matching threshold and the interception point between the two graphs is the EER value. Typically lower values of FRR and FAR will lead to lower value of EER, in turn, indicates a better accuracy performance of a biometrics authentication method. EER is commonly used to measure and compare the accuracies of different biometrics systems.

5.2 Training and Performance Testing Setup

For each subject recruited, 2 set of 10 input samples were collected; one with 4-digit input and the other with 16-digit input. For each input length category, 7 out of 10 were used for training (i.e. for subject's template generation) while the remaining 3 for testing. To reduce intra-session variability effect, the 7 samples are selected randomly from the 10-sample set. Samples used for training are not reused for testing.

In the FAR test, a subject's template was compared against all the other subjects' testing samples. This process was reiterated for all the subjects' templates. As there were a total of 150 subjects recruited and each subject has 3 testing samples, the total number of illegitimate attempts conducted was $150 \times (150 - 1) \times 3 = 67{,}050$.

In the FRR test, a subject's template was compared against the subject's own testing samples. As there were a total of 150 subjects and each subject has 3 testing samples, the total number of legitimate attempts was $150 \times 3 = 450$.

5.3 Results Discussion

5.3.1 *Feature Data Types*

Experiments have been conducted to investigate the accuracy performances (measured in terms of EER) of different feature data types. As shown in Figure 4, the ps feature outperforms all timing related feature data types. This may be due to the fact that ps could capture more properties from a subjects touch pattern. For example, the amount of force used, finger arrangement, touch angle and finger thickness. The mixture of these properties establishes a distinctive pattern and was found to be greatly unique among different subjects.

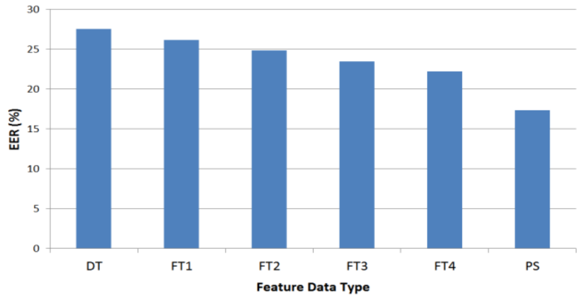


Figure 4. Accuracy performances of different feature data types

As for timing related feature data types, the accuracy performance of any variant of FT is slightly better than DT. This implies that the time taken for subject's finger to traverse from one key to another has more discriminative power than how long a key is hold down. Throughout the data collection process, we observed that there existed different input key combinations between different subjects even when the input strings entered are identical. For instance, 55_60, 5_56_0, or 556_0, where the underscore symbol represents minor pause or delay between successive key pairs. The information inhabits within the natural short pauses between different groupings of input keys may have increased the uniqueness of FT.

Although ps proved to be the best feature data type in terms of EER, its EER performance of 17.31% is still rather unsatisfactory. To improve this performance, we have combined ps with different combinations of timing feature data types. So, in a given authentication instance, multiple feature data types are used, and the final decision is made by combining individual decisions made on each chosen feature data type using the AND voting rule. As a result, the accuracy performance has been markedly improved. As shown by the results in Table 2, the more feature data types used, the lower the EER value, the better the accuracy performance. The lowest EER value is achieved when all six feature data types are used. In this case, the EER value is 8.50%, which is more than doubled the accuracy performance when ps is used alone.

Table 2. Accuracy performance comparison between combinations of feature data type

Feature	FAR	FRR	EER
DT	52.39	2.67	27.53
FT_1	44.37	8.00	26.18
FT_2	40.97	8.67	24.82
FT_3	40.97	6.00	23.48
FT_4	37.71	6.67	22.19
PS	16.61	18.00	17.31
FT_4, PS	16.69	5.33	11.01
FT_3, FT_4, PS	15.84	5.33	10.59
FT_2, FT_3, FT_4, PS	15.39	6.00	10.69
$FT_1, FT_2, FT_3, FT_4, PS$	14.76	6.00	10.38
$DT, FT_1, FT_2, FT_3, FT_4, PS$	**8.99**	**8.00**	**8.50**

5.3.2 *Input String Lengths*

Input string lengths may also affect the accuracy performance of a biometrics authentication system. To investigate the effect, we have calculated the EER values using the three matching functions and two sets of digits with respective lengths of 4 and 16 digits. The 4-digit set represents a short input string, while the 16-digit set represents a long input string. These results are plotted in Figure 5. As can be seen from the figure, a longer input string leads to a lower EER value, which indicates a better accuracy performance. This may be explained as follows. When the input strings length increases, feature data samples within each input string, and the number of different chunking combinations (breaking up longer inputs into smaller subsets for easier memorization [23]) also increase, and so is the ability to better capture a subject's touch pattern. In addition, the number of illegitimate feature data samples required to match that of a legitimate reference template also increases. Therefore, the longer the input strings the better accuracy performance one could achieve from a biometrics authentication system.

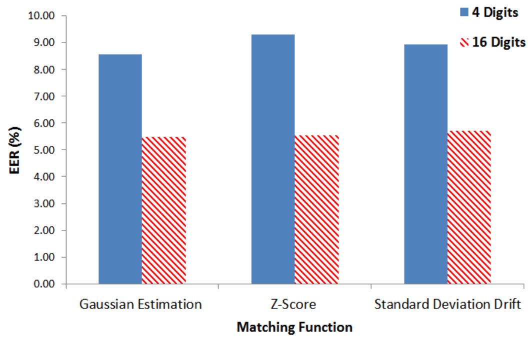


Figure 5. The effects of the input string lengths on the EER values

5.3.3 Subject Size

We have also investigated the effects of subject sizes on the EER values. This experiment is based on three matching functions, GE, ZS, and SD. Two set of datasets are investigated; one has 50 subjects and the other 150 subjects. For each subject set, we used two input digits lengths, 4-digit and 16-digit. The experimental results are shown in Table 3. From the table, we can observe that when the subject size of 50 is used, the average EER value decreases slightly from 7.93 to 6.81 as the input string length increases from 4-digits to 16-digits. However, when the subject size is 150, this average value decreases considerably from 8.92 to 5.59 as the input string length increases from 4-digits to 16-digits. This inconsistency is likely to be caused by the lower discriminative ability of shorter input string length (discussed in the previous section). Ideally, an input string tested on larger subject size should achieve lower or at least comparable error value than when tested on smaller subject size. As shown by the results in the table, when the input string length is 16-digits, the differences in the EER values produced by different matching functions remains fairly constant when the subject size goes up from 50 to 150. This indicates that the 16-digits input string can provide us with more consistent experimental results.

Table 3. EER values vs subject sizes

Classifier	50 Subjects		150 Subjects	
	4-Digit	16-Digit	4-Digit	16-Digit
GE	7.71	6.27	**8.55**	**5.49**
ZS	8.51	6.70	9.30	5.54
SD	7.57	7.45	8.92	5.74
Average EER	7.93	6.81	8.92	5.59

5.3.4 Classifier Performance

The results in Table 4 also indicate the accuracy performance of the three matching functions. The EER values produced by all three functions are comparable. When the subject size is 150 and the input string length is 16-digits, the differences in the EER values are less than 0.25%. Gaussian Estimator (GE) produces the lowest EER value, i.e. 8.55% when the input string is 4 digits long and 5.49% when the input string is 16 digits long, both under the case where 150 subjects were used. This says that, even if the input string is known to the impersonator, 9 out of 10 impersonation attempts can be successfully identified. As the input string length increases, the success rate in identifying impersonation attempts also increases. These results are encouraging. They indicate the potential of using touch dynamics with knowledge-based authentication to heighten the security of user authentication in mobile device/service access. In addition, touch dynamics biometrics is cost-effective, as it does not require the use of additional hardware, and usable, as it is already part of the mobile device interface. So, it can be an attractive building block for a more effective authentication solution not only in a physical, but also a virtual environment.

Table 4. Performance between classifiers on different input lengths

Classifier	4 Digits			16 Digits		
	FAR	FRR	EER	FAR	FRR	EER
GE	12.21	4.89	**8.55**	9.43	1.56	**5.49**
ZS	15.27	3.33	9.30	8.64	2.44	5.54
SD	8.95	8.89	8.92	10.36	1.11	5.74

5.3.5 With and Without Touch Dynamics

One potential application area of touch dynamics biometrics is to integrate it into an existing authentication system to extend it into a so-called multi-factor authentication system. In such a system, an impersonator, to successfully sneak through the authentication verification process, would have to produce an acceptable touch pattern, in addition to possessing the right login credential. Assume that a two-factor authentication system is used; one factor is PIN-based authentication and the other factor is touch dynamics authentication. As shown in Table 5, in the case where the PIN is exposed, the chances for an impersonator to be successfully authenticated is drastically reduced from 100% (if only PIN is used) to 9.43% (if both PIN and touch dynamics are used). However, the weakness of using the touch dynamics based authentication system is that there is a 1.56% increase in the chances of rejecting a legitimate subject incorrectly.

Table 5. The comparison between FAR and FRR with the presence of touch dynamics

Feature	FAR	FRR
PIN	100	0
PIN + Touch Dynamics	**9.43**	1.56

6. RELATED WORK

6.1 Public Dataset

The data collection process conducted by [1] involved 51 subjects inputting a fixed password "rhu.university" on a virtual keyboard of a window touchscreen phone (Nokia Lumia 920). Subjects were required to attend 3 different sessions with an average of 5 days between each of them. However, the actual data collection did not commence until the second session as the first was used as a practice session. A total of 15 samples were collected from each subject divided between the second and third session. Only the timing feature was captured in this dataset, whereas our dataset also captures finger touch size and pressure feature.

Table 6. Comparison of public datasets

Dataset	Subject	Population	Sample	Input	Feature	Setting	Platform
[1]	51	Restricted	15	"rhu.university"	T	Confined	Phone
[2]	42	Restricted	51	".tie5Roanl"	T,S,P	Confined	Phone
[32]	100	Restricted	5	6 to 8 digits	T,S,P	Confined	Phone
This Paper	150	Diversified	10	"5560", "1379666624680852"	T,S,P	Flexible	Tablet

Another related effort on collecting and sharing datasets publically was made by [2]. This work differs from ours in a number of ways. Firstly, the number of subjects involved is more than 3 times smaller than the number involved in our case. Also the entire subject populations were students, which is different from our case where the population consists of both members of the university and general public. In addition, different from our case where all the data were collected via the use of the same type of device, data collection in [2] was done via the use of two types of devices. 37 subjects provided their input on a Nexus 7 while the remaining 5 via the use of a LG Optimus L7 P700 smartphone. The paper did not explain if two different device types would have any performance implications. To allow subjects' sample data be used in EER estimations, the input string was predefined (".tie5Roanl"), which is also the case in our data collection. Also, in this work, the touch events captured include not only the input string but also *shift key* (toggle between lower and uppercase characters) and *keyboard switch key* (toggle between characters and numerical keys). These secondary key events may capture valuable and distinctive information about a subject. Inspired by this idea, in addition to capturing touch events on digit input, we have also recorded the *Enter key* event (pressed upon completion of a PIN input) in our dataset. Also in this related work, most of the subjects provided their passwords for 30 times each on 2 isolated sessions in a period of two weeks (duration in between was unknown). However, some invalid inputs were removed, so the dataset were unified to only 51 input samples per subject (instead of 60 from both sessions).

[32] has reported a collection of dataset based on numerical inputs. In this data collection process, the device used was an early generation smartphone with a physical keypad running on Android 2.0.1 (Éclair) API level 6, which was released in December 2009. By contrast, we adopted a more recent high resolution digital tablet with a later version of a mobile operating system. Subjects were only required to provide 2 samples per session and 5 sessions were used with an interval of at least 1 week apart for each session to eliminate intra-session typing variations. Different from our case where data were acquired in a nonrestrictive environment, their data were acquired in a rather confined environment (i.e. in a classroom). Subjects may feel uncomfortable in a confined environment and may cause subjects to provide their input unnaturally or inconsistently. This inconsistency may have negative impact to accuracy performance. Although the paper did not explicitly give the detail of the subject population, according to the age distribution data (i.e. a bias of up to 85% of the total subjects has the age of 25 or younger), we could infer that the subject population were likely to be centered on university students. This is different from our case, where our dataset were collected from a diversified population and with different age groups and backgrounds. Also in [32], subjects were allowed to freely choose a PIN, and most of the chosen PINs have a length of 4 to 8 digits long. However, the actual PIN selections by each subject were not recorded in the shared dataset. Also, in this dataset, raw finger touch size and pressure data have been recorded, and the timing feature was only recorded in a post processed format (duration and latency). In other words, raw timing values were not recorded. This missing information may hinder the usability of the dataset in a wider context. Finally, different from the usual practice, test samples collected for the FRR test were collected separately from those for FAR test. 10 subjects are randomly chosen to act as impersonators. These impersonators were given the PINs of every other subjects and were asked to impersonate the subjects by providing 5 samples for each subject. This way may significantly reduce the number of impersonation test samples as compared to reusing the samples used for FAR test for FRR test. An overview of presently available public datasets is summarized in Table 6, where T, S and P indicate timing, finger touch size and pressure values, respectively.

6.2 Performance Investigation

Another stream of related work on touch dynamics biometrics is to investigate and compare its accuracy performance [22, 28, 32, 36]. The paper [28] reported their experimental work on testing the viability of identifying subjects based on numerical input string. In this experiment, only 10 subjects were recruited and each subject was asked to input a predefine PIN ("1593") on a HTC Nexus-One smartphone. To investigate if an impersonator could imitate another subject's touch pattern in the event if the subject's PIN is known to the impersonator, the author designed a visualization tool to facilitate a separate set of attackers to imitate a genuine subject's input pattern. Even by deliberately exposing the PIN, timing and pressure feature information via the visualization tool to the attackers, the authors were still able to archive an FAR of 16%. Though some interesting results were obtained from this experiment, the number of subjects used was too small to draw any conclusive remark.

The accuracy performance of touch dynamics applied on 4-digit PIN was also investigated by the authors in [22]. They extracted data with regard to timing, finger touch size and pressure by using build-in touchscreen sensor, and in addition, they also extracted linear and angular acceleration as feature vectors using accelerometers and gyroscopes sensors. As the size of data collected from these two sensors is large, they applied preprocessing technique to reduce the size of data. As a result, the computation resource needed for classification reduces. However, the dataset in this experiment was collected in a quite constrained setting, where subjects had to hold the mobile phone in a fixed position. By using on Euclidean distances based classifier, they obtained a performance of 20% EER on a 4-digit PIN input. The performance comparisons among individual features were not given.

The experiment reported in [32] involved a larger number of subjects with the use of input PINs ranging from 4 to 8 digits long. An EER of 8.4% was obtained by using simple statistical classifier. In addition, the authors studied the time required to perform the classification and verification of different PIN lengths and feature combinations; both consumed an average of 12ms. This experimental discovery is useful for potential deployment of touch dynamics biometrics on power limited mobile devices. By far, the most competitive performance was achieved by [36]. The work employed a statistical one-class learning classifier and obtained an average EER of 3.65% tested on a set of given PIN input combinations. An EER of 6.96% and 7.34% were obtained on PIN numbers of “1111” and “5555”, respectively. More work may be necessary to see if the performance is scalable with a larger subject size.

There were also experiments [8, 11, 12, 14] carried out on character-based passwords. However, as the scope of our work in this paper is on numerical PIN inputs, we will not discuss character-based experiments any further. A summary of the comparison between our work and the related works is given in Table 7.

Table 7. Comparison to existing work with PIN input

Paper	Length	Subjects	Device	EER
[28]	4	10	HTC Nexus-One	15.2
[36]	4-8	80	Samsung Galaxy Nexus	3.65
[22]	4	80	-	20
[32]	4	100	Motorola Milestone	8.4
This Paper	4	150	Samsung Galaxy Tab 10.1	8.55
	16			5.49

7. CONCLUSION AND FUTURE WORK

Touch dynamics based authentication may provide us with a number of benefits, such as it is an inherent feature of a majority of mobile devices already in use and it is readily deployable as an additional authentication factor to strengthen e-authentication assurance levels. This paper has investigated the feasibility and benefits of adopting a touch dynamics based authentication method or integrate it with the PIN based authentication method. To evaluate the effectiveness of this integrated approach, a proper dataset is required. We first reported a comprehensive dataset, with the intention of also serving further research on various issues in this context, such as further investigation and comparison of classification methods or the potential use of touch dynamics for authentication purposes. The dataset is available at https://goo.gl/sNACU8. We then implemented and applied three light-weighted matching functions to the dataset to study its accuracy performance. These matching functions imposes lower computational complexity, offers faster authentication speed and incurs less battery consumption, which could be a desirable additional authentication facility for mobile devices. We also showed that accuracy performance can be increased by combining different feature data types.

There are a number of issues that requires further study. These include conducting an even larger subject study to validate the scalability of the proposed methods, and enriching the feature vectors with other feature data types (e.g. touch position or touch motion feature). In addition, what would be the most appropriate method to tackle subject touch pattern changes over time, and evaluating the proposed methods in terms of other evaluation metrics (e.g. actual computational time on mobile device).

8. ACKNOWLEDGMENTS

Our sincere thanks go to the people who have contributed their time and effort to the creation of this dataset, and also to the reviewers for their valuable time and insightful comments.

9. REFERENCES

[1] El-Abed, M., Dafer, M. and Khayat, R. El 2014. RHU Keystroke: A mobile-based benchmark for keystroke dynamics systems. *2014 International Carnahan Conference on Security Technology (ICCST)* (Oct. 2014), 1–4.

[2] Antal, M. and Szabó, L.Z. 2014. Keystroke Dynamics on Android Platform. *Proceedings of the 8th International Conference Interdisciplinarity in Engineering* (Romania, 2014), 131–136.

[3] Bleha, S., Slivinsky, C. and Hussien, B. 1990. Computer-access security systems using keystroke dynamics. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*. 12, 12 (1990), 1217–1222.

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

[5] Chong, M.K., Marsden, G. and Gellersen, H. 2010. GesturePIN: Using Discrete Gestures for Associating Mobile Devices. *Proceedings of the 12th International Conference on Human Computer Interaction with Mobile Devices and Services* (New York, NY, USA, 2010), 261–264.

[6] Cho, S., Han, C., Han, D.H. and Kim, H.-I. 2000. Web-Based Keystroke Dynamics Identity Verification Using Neural Network. *Journal of Organizational Computing and Electronic Commerce*. 10, 4 (2000), 295–307.

[7] Clarke, N.L. and Furnell, S.M. 2007. Authenticating mobile phone users using keystroke analysis. *International Journal of Information Security*. 6, 1 (2007), 1–14.

[8] Dhage, S., Kundra, P., Kanchan, A. and Kap, P. 2015. Mobile authentication using keystroke dynamics. *2015 International Conference on Communication, Information Computing Technology (ICCICT)* (Jan. 2015), 1–5.

[9] Gascon, H., Uellenbeck, S., Wolf, C. and Rieck, K. 2014. Continuous authentication on mobile devices by analysis of typing motion behavior. *Lecture Notes in Informatics* (2014), 1–12.

[10] Giot, R., El-Abed, M. and Rosenberger, C. 2009. GREYC keystroke: A benchmark for keystroke dynamics biometric systems. *Biometrics: Theory, Applications, and Systems, 2009. BTAS ’09. IEEE 3rd International Conference on* (Sep. 2009), 1 –6.

[11] Giuffrida, C., Majdanik, K., Conti, M. and Bos, H. 2014. I Sensed It Was You: Authenticating Mobile Users with Sensor-Enhanced Keystroke Dynamics. *Detection of Intrusions and Malware, and Vulnerability Assessment*. S. Dietrich, ed. Springer International Publishing. 92–111.

[12] Huang, X., Lund, G. and Sapeluk, A. 2012. Development of a Typing Behaviour Recognition Mechanism on Android. *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)* (Jun. 2012), 1342–1347.

[13] Hwang, S.S., Lee, H.J. and Cho, S. 2009. Improving authentication accuracy using artificial rhythms and cues for keystroke dynamics-based authentication. *Expert Systems with Applications*. 36, 7 (2009), 10649–10656.

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

[15] Karnan, M. and Krishnaraj, N. 2010. Bio password - Keystroke dynamic approach to secure mobile devices. *2010 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)* (Dec. 2010), 1–4.

[16] Kim, D., Dunphy, P., Briggs, P., Hook, J., Nicholson, J.W., Nicholson, J. and Olivier, P. 2010. Multi-touch authentication on tabletops. *Proceedings of the 28th international conference on Human factors in computing systems* (New York, NY, USA, 2010), 1093–1102.

[17] Kolly, S.M., Wattenhofer, R. and Welten, S. 2012. A Personal Touch: Recognizing Users Based on Touch Screen Behavior. *Proceedings of the Third International Workshop on Sensing Applications on Mobile Phones* (New York, NY, USA, 2012), 1:1–1:5.

[18] Loy, C.C., Lai, W.K. and Lim, C.P. 2007. Keystroke Patterns Classification Using the ARTMAP-FD Neural Network. *Intelligent Information Hiding and Multimedia Signal Processing, 2007. IIHMSP 2007. Third International Conference on* (Nov. 2007), 61 –64.

[19] De Luca, A., Hang, A., Brudy, F., Lindner, C. and Hussmann, H. 2012. Touch Me Once and I Know It's You!: Implicit Authentication Based on Touch Screen Patterns. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2012), 987–996.

[20] Maiorana, E., Campisi, P., González-Carballo, N. and Neri, A. 2011. Keystroke dynamics authentication for mobile phones. *Proceedings of the 2011 ACM Symposium on Applied Computing* (New York, NY, USA, 2011), 21–26.

[21] McLoughlin, I.V. and Naidu, N. 2009. Keypress biometrics for user validation in mobile consumer devices. *Consumer Electronics, 2009. ISCE '09. IEEE 13th International Symposium on* (2009), 280–284.

[22] de Mendizabal-Vazquez, I., de Santos-Sierra, D., Guerra-Casanova, J. and Sanchez-Avila, C. 2014. Supervised classification methods applied to keystroke dynamics through mobile devices. *2014 International Carnahan Conference on Security Technology (ICCST)* (Oct. 2014), 1–6.

[23] Ngugi, B., Kahn, B.K. and Tremaine, M. 2011. Typing Biometrics: Impact of Human Learning on Performance Quality. *J.Data and Information Quality*. 2, 2 (2011), 11:1–11:21.

[24] Ngugi, B., Tremaine, M. and Tarasewich, P. 2011. Biometric keypads: Improving accuracy through optimal PIN selection. *Decision Support Systems*. 50, 4 (2011), 769–776.

[25] Obaidat, M.S. 1995. A verification methodology for computer systems users. *Proceedings of the 1995 ACM symposium on Applied computing* (1995), 258–262.

[26] Robinson, J.A., Liang, V.M., Chambers, J.A.M. and MacKenzie, C.L. 1998. Computer user verification using login string keystroke dynamics. *Ieee Transactions on Systems Man and Cybernetics Part a-Systems and Humans*. 28, 2 (1998), 236–241.

[27] Saravanan, P., Clarke, S., Chau, D.H. (Polo) and Zha, H. 2014. LatentGesture: Active User Authentication Through Background Touch Analysis. *Proceedings of the Second International Symposium of Chinese CHI* (New York, NY, USA, 2014), 110–113.

[28] Sen, S. and Muralidharan, K. 2014. Putting "pressure" on mobile authentication. *2014 Seventh International Conference on Mobile Computing and Ubiquitous Networking (ICMU)* (Jan. 2014), 56–61.

[29] Serwadda, A., Phoha, V.V. and Wang, Z. 2013. Which verifiers work?: A benchmark evaluation of touch-based authentication algorithms. *2013 IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS)* (Sep. 2013), 1–8.

[30] Sheng, Y., Phoha, V.V. and Rovnyak, S.M. 2005. A parallel decision tree-based method for user authentication-based on keystroke patterns. *Ieee Transactions on Systems Man and Cybernetics Part B-Cybernetics*. 35, 4 (2005), 826–833.

[31] Stewart, J.C., Monaco, J.V., Cha, S.-H. and Tappert, C.C. 2011. An investigation of keystroke and stylometry traits for authenticating online test takers. *Biometrics (IJCB), 2011 International Joint Conference on* (2011), 1–7.

[32] Tasia, C.-J., Chang, T.-Y., Cheng, P.-C. and Lin, J.-H. 2014. Two novel biometric features in keystroke dynamics authentication systems for touch screen devices. *Security and Communication Networks*. 7, 4 (Apr. 2014), 750–758.

[33] Teh, P.S., Teoh, A.B.J. and Yue, S. 2013. A Survey of Keystroke Dynamics Biometrics. *The Scientific World Journal*. 2013, (Nov. 2013), e408280.

[34] Trojahn, M., Arndt, F. and Ortmeier, F. 2013. Authentication with Keystroke Dynamics on Touchscreen Keypads - Effect of different N-Graph Combinations. (Nov. 2013), 114–119.

[35] Zahid, S., Shahzad, M., Khayam, S.A. and Farooq, M. 2009. Keystroke-Based User Identification on Smart Phones. *Proceedings of the 12th International Symposium on Recent Advances in Intrusion Detection* (Berlin, Heidelberg, 2009), 224–243.

[36] Zheng, N., Bai, K., Huang, H. and Wang, H. 2014. You Are How You Touch: User Verification on Smartphones via Tapping Behaviors. *2014 IEEE 22nd International Conference on Network Protocols (ICNP)* (Oct. 2014), 221–232.