Securing Sensitive Information in Smart Mobile Devices through A Difficult-to-Mimic and Single-Time Usage Analytics

ABSTRACT

User identification has become one of the prime concerns in recent times with the advent of widespread usage of smart devices such as smartphone and storing secret and sensitive information in the devices for regular use. The advent demands a light-weight user identification technique that will be very difficult to mimic through different types of attack such as eavesdropping or shoulder surfing. However, devising such a user identification technique is little explored in the literature. To this extent, we attempt to propose a new single time usage analytics based user identification technique. To do so, we collect single-time usage data comprising touch usage, hold pattern, etc., from 33 participants and show that the participants can be mostly individually identified through unsupervised clustering. Being inspired by this finding, we propose a novel machine learning based user identification technique named Mean-SD clustering. We compare performance of our proposed technique against that of several existing machine learning techniques and demonstrate its efficacy through analyzing the collected usage data. Further, we confirm the light-weight nature and high accuracy having the nature of difficult-to-mimic pertinent to the proposed Mean-SD clustering technique through analyzing the collected data and performing user identification over another set of participants by letting them use mobile smart devices running our proposed technique.

Keywords

Smartphone, Behavioral biometrics, Mean-SD Clustering.

1. INTRODUCTION

The use of pervasive electronic devices such as smartphones has been increasing day-by-day. According to the market analysis and forecasting, there will be 2.16 billion smartphones in use globally by 2016 [1]. Nowadays, the smartphones are used not only for making or receiving calls but also to avail special assistances through providing applications (checking email, enjoying personal photos, etc.) and services (mobile payment, smart home [2], etc.). As a consequence, users of the devices have started storing personal and secret sensitive information such as SSN, bank account number etc., in those devices. This makes the issue of identifying valid users utmost important in recent times.

According to the study presented in [3], the task of user identification for electronic devices can be classified in three fundamental approaches: (a) knowledge-based, which typically uses a password or a PIN, (b) object-based, which relies on possession of a token, and (c) biometric-based, which relies on the uniqueness of physical/ behavioral characteristics of a person such as fingerprints, facial features, iris, and voice. A password or PIN can offer security through user identification. However, survey results [4] have already shown that most users agree that using PIN is very inconvenient and they do not have confidence in the protection that the PIN-based approaches offer. Alongside, in case of the PIN gets lost, the approach can easily be breached. Similarly, the major security drawback of an object-based mechanism is that, if the token gets lost or stolen, an imposter can gain unauthorized data access in a device. On the other hand, though biometric-based approaches are known to be more reliable than traditional identification schemes, the security of this system can be weakened in many ways. For example, a biometric template might be stolen by an impostor. Moreover, several research studies [5], [6] have shown that it is possible to create a physical spoof exploiting standard biometric templates.

Currently, most of the conventional approaches for identifying users are based on the solutions offered by smartphones, which usually involve the use of a PIN, a strong password, pattern lock, or in rare cases some sort of external security token devices. In recent years, user identification based on usages monitoring has started to be explored. Most of such solutions are either still password/ PIN-based [7, 8] or demands run-time usages monitoring [9]. Here, the approaches based on password/ PIN are vulnerable to different security threats such as eavesdropping [10]. On the other hand, the run-time monitoring based approaches are generally resource-hungry because of non-stop observation of usages. Moreover, accuracy of these solutions depends on the capability of continuous observations. Consequently, these techniques become cumbersome when applied and do not always provide a satisfactory user experience. Therefore, to efficiently secure information nowadays, devising a simple, easy-to-use, less resource-hungry, and single-time user identification technique exhibits its utmost significance. Additionally, the technique needs to be difficult-to-mimic in case of protecting sensitive information such as SSN or bank account information.

To address this issue, this paper presents a new behavior-based single-time user identification technique, which takes into account two types of human behaviors: how the screen is being touched and how the device is being held. Here, we perform experiments with several machine learning approaches to evaluate the effectiveness of our considered human behaviors. We get high error rate for existing techniques. Therefore, we propose a new identification technique named Mean-SD Clustering to enhance the efficacy of our user identification task demanding limited resource. Our rigorous experimental evaluation confirms that mostly every user has a unique behavior on touching the screen and holding the phone in combination, and consequently the user can be identified through these behavioral biometric. The strongest part of our proposed technique is that it exploits two behavioral metrics that are very difficult to mimic simultaneously. Therefore, it would be extremely difficult for eavesdroppers to get access through breaching our proposed technique, which we confirm through getting zero false acceptance in our experimental evaluation.

Based on our work, we make the following set of contributions in this paper:

- We analyze single-time usage data collected from 33 participants. our analysis through unsupervised clustering reveals that most of the participants get dominated by individual clusters and most of the clusters dominate only one participant. This finding paves a foundation for further investigating the usage data for user identification.

- Next, we propose a novel light-weight machine learning technique called Mean-SD clustering for performing the user identification task.

- We confirm the efficacy of our proposed clustering technique through identifying users from our data set of 33 participants with as low as 5% False Acceptance Rate (FAR) and 6% False Rejection Rate (FRR). We perform necessary parameter tuning to achieve such low false rates. The false rates in combination are mostly lower compared to other available machine learning techniques that incur higher resource overhead.

- Finally, we implement our proposed technique in smartphones and perform user evaluation through the implementation. The user evaluation conducted under diversified situations and conditions demonstrates that the False Acceptance Rate gets to 0% even under attacks such as eavesdropping and shoulder surfing at an expense of higher False Rejection Rate, a less benign aspect compared to FAR.

The remainder of the paper is as follows: Section 2 covers related work. Section 3 explains motivation of our work. Section 4 presents our proposed technique for user identification, which includes working methodology, data capturing, and user identification through clustering. Section 5 elaborates experimental design for evaluation of our proposed technique including experimental platform, developed application for data collection, and demography of the participants. In Section 6, we evaluate experimental data analysis. Section 7 makes comparative analysis over the performance of our proposed technique compared to either available machine learning techniques. Section 8 summarizes experimental findings. Section 9 presents outcomes of user evaluation through an implementation of our proposed technique is a smartphone.Finally Section 10 and 11 concludes the paper with pointing our future work.

2. RELATED WORK

Conventional user identification mechanisms are based on passwords and/or biometric data. The common drawback of password-based identification mechanisms is that they can be guessed or stolen [11]. Consequently, leading towards easy exploitation of the mechanisms by malicious users, password or PIN does not remain enough to safeguard electronic devices and data access through them. One of the main reasons behind such happening is shoulder surfing or eavesdropping over physical environment [10].

Biometric-based identification is known to be more reliable than the traditional password or PIN-based schemes. According to the study presented in [12], there are two types of biometric-based identification approaches namely physiological and behavioral biometrics. Among such approaches, the study in [13] focuses on fingerprint-face features. Another study [14] exploits iris recognition. Further, the studies in [15], [16] combine multiple biometric features to implement a person identification system. However, deploying these approaches in limited-resource devices such as mobile devices are very difficult and costly.

So far, several research studies have directly or indirectly exploited behavioral biometrics in the realm of user identification. The underlying mechanisms of such techniques can be categorized into two groups: run-time identification and single-time identification. An example of run-time identification is TIPS [17] touch-based identity protection service, which achieves 90% accuracy in run-time user identification. The underlying process of TIPS unobtrusively identifies users in the background through continuously analyzing touchscreen gestures in the context of a running application.

The study presented in [18] also exploits users' touches. It authenticates users silently and transparently through exploiting the dynamics mined from users' touch behavior biometrics and the micro-movement of the device caused by users' screen-touch actions. More recently, another study investigates multiple aspects of usage in combination to identify a user [9]. It collects different attributes of user's usages attributes such as touch positions, dwell time, scroll positions, typing speeds, etc., to re-authenticate the user through run-time investigation. However, this type of study also needs continuous monitoring of several usage attributes demanding high resource overhead.

Another research study [19] investigates combining biometric analysis with pattern recognition to enhance the security. It introduces a two-factor identification mechanism by augmenting biometric analysis with graphical passwords. It captures users' finger-in-dot time, which is the time in milliseconds from the moment the participant's finger touches a dot to the moment the finger is dragged outside the dot area, along with finger-in-between-dots time, which represents the speed at which the finger moves from one dot to the next. It utilizes these values in accordance with graphical passwords, which exhibit no difference from being shoulder surfed or eavesdropped as such happens in the case of textual passwords. Additionally, a similar research study [7] investigates the feasibility of authenticating users based on users' typing of 10-digit number on a touchpad.

Researchers also suggest the use of biometric trait as a second verification factor when authenticating a user of a smartphone. A recent research study in this regard [8] investigates a system to use jointly with the login passphrase to make a decision if the person that entered the passphrase is truly the legitimate user of the device. Such approaches many-a-times offer a complicated experience for general users.

Another research study [20] develops a user identification system using multi-touch gestures. Here, a user performs the gesture using all five fingers at once, and biometrics are drawn from the hand's geometry as well as the dynamics of the gesture itself. This mechanism exhibits difficulties to users as it demands complex gestures.

Nonetheless, another research study [21] demonstrates that it is possible to identify a user immediately through identifying the pattern of a password with an implicit identification layer. Here, the touchscreen data of smartphones along with password pattern performs the task of distinguishing between the rightful user and an attacker. Furthermore, recently a bimodal biometric identification solution named Touchstroke [22] has been proposed. It classifies users based on the use of the user's hand movements while holding the device, and the timing of touch-typing when the user enters a 4-digit PIN/password. Both of these approaches utilize passwords for their operations and thus remain prone to experiencing common vulnerabilities of using passwords.

In summary, existing identification methods utilize either run-time or single-time approaches. Here, strength of the run-time identification methods depends on the capability of run-time data capturing as well as rigorousness of continuous data analyzing. Consequently, most of such approaches are resource-hungry. On the other hand, conventional single-time identification methods rely on the typical PIN/password or pattern recognition for higher accuracy.

As a remedy for this situation, a simple, easy-to-use, less resource-hungry, and single-time user identification technique is required to be devised. This is exactly where the contribution of our work presented in this paper lies.

3. MOTIVATION OF OUR WORK

The initial motivation for our research arose from the need to provide difficult-to-mimic user identification technique for mobile devices. Technique exploiting continuous tracking of user's usages make mobile devices slow through consuming significant CPU and memory. Therefore, the motivation of our work is to develop single-time user identification technique that would require tracking of user's usage only for one time. We chose to explore the use of user's touch- and holding orientation based usage monitoring for identification since they usually have the benefits of not involving additional physical gadgets and not demanding the user's attention for a long period of time. Also, typing strong passwords on a touchscreen are more likely to exhibit repetitive errors and providing effective lock patterns requires memorization of complex patterns. Touch-based usage monitoring is free from this kind of hassle. Therefore, the inspiration of our work is to develop single-time user identification technique through taking the advantage of user's touch-based usage monitoring. Alongside, another easily-capturable usage metric is holding orientation of a smartphone, which exhibits a good potential to vary user-to-user in a subtle way. Therefore, the holding orientation is also another candidate of being considered for capturing effective and distinguishable single-time usage metric. To the best of our knowledge, combining the touch-based usage and holding orientations are yet to be investigated in the literature for user identification. Therefore, in this paper, we attempt to perform the investigation.

4. PROPOSED TECHNIQUE FOR USER IDENTIFICATION

This section describes our proposed technique for single-time user identification in details. Here, first, we present the overall working methodology. Subsequently, we elaborate two main tasks that are needed to be performed, data capturing from users and identifying the user based on the collected data.

4.1 Working Methodology

Our proposed technique requires a set of training data pertinent for the original user's usage as a prerequisite for its operation. The training data is taken and then immediately used to train the application. The training results are then stored in the device. The data collection and training is done only once. The identification starts by capturing the usage data of a user under investigation. This usage data is then checked against the training result to decide on whether the captured data is obtained from the user who provided the training data or not. Based on this decision, the identification task gets completed.

The two basic tasks of our proposed technique are capturing data from a user and identifying the user based on the captured data. In the next subsections, we present both these tasks in more detail.

4.2 Data Capture

We have extracted 13 features during the experimental stage. Extracted features are touch coordinates (start and end coordinates of swipe), finger pressures (the force applied) over the start and end finger positions, velocity over the swipe, hold-time (the duration of interval between the starting and ending of the swipe), tilt angle, and rotation matrix while pressing the button.

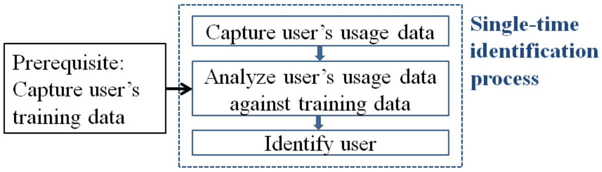

Prerequisite: Capture user's training data
Capture user's usage data
Analyze user's usage data against training data
Identify user
Single-time identification process

Figure 1: Methodology of our application

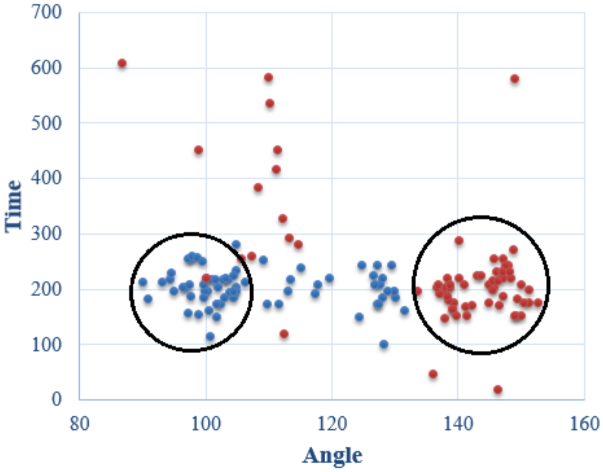


Figure 2: Proposed Mean-SD Clustering technique

We collected 15 observations from each user. Among them, 10 observations were used as training set and rest 5 observations as the test set. The identification procedure can be divided into two phases; training phase and testing phase. The training set is used to calculate the cluster mean in the training phase. The trained and test data were compared in testing phase and thus, we calculated False Acceptance Rate (FAR) and False Rejection Rate (FRR).

FAR is the probability that the system incorrectly identifies an invalid user, due to incorrectly matching the input touch-based usage data with training data. The FAR is normally expressed as a percentage, following the FAR definition this is the percentage of invalid inputs which are incorrectly accepted.

Similarly, FRR is the probability that the system incorrectly rejects access to a valid user, due to failing to match the input touch-based usage data with training data. The FRR is normally expressed as a percentage, following the FRR definition this is the percentage of valid inputs which are incorrectly rejected.

4.3 User Identification

As stated before, each training data of a user consists of 13 feature values. Therefore, we can consider each training data as a 13-dimensional vector. A training set of data for each user forms a specific cluster in 13 dimensions which can be used to identify that specific user. The training phase determines the clusters by setting its two parameters: cluster centroid, which is the cluster's mean and cluster size. We named this technique Mean-SD Clustering. The data sets mean values are assigned to cluster's means. The cluster size can be varied with the average and the standard deviation of cluster data. More specially, we set the cluster size by taking a weighted sum of the average and standard deviation. Consequently, we determine the cluster size (C) using the following equation:

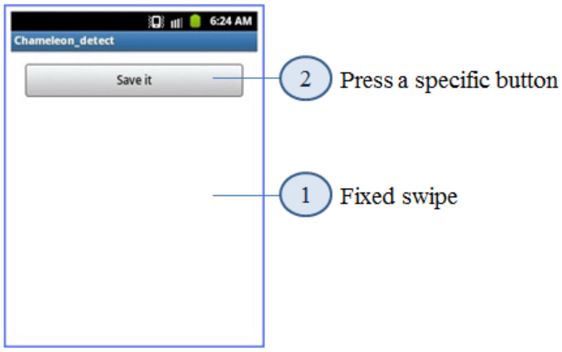


Figure 3: Screenshot and step-wise tasks of our application

$$C = \left(AVG \times W_{\mathrm{avg}}\right) + \left(SD \times W_{\mathrm{sd}}\right) \qquad (1)$$

where AVG and SD represent the average and standard deviation of the training data respectively. Besides, W_{avg} is the weight of average and W_{sd} is the weight of standard deviation of training data.

The concept of cluster size is shown in Fig. 2. Here, we represent our clustering technique in 2D for simplicity purpose. In this figure, we consider only two features time and angle. We take time in Y-axis and angle in X-axis. Each cluster represents a user.

In the experimental stage, we investigate the cluster size by varying the weight of AVG, SD or both. Identification of a user is performed by comparing the data captured with stored clusters parameters. If the data falls into any of the cluster generated before by the Mean-SD clustering, it belongs to the user whose cluster it is. Otherwise, the user is identified as an intruder.

5. EXPERIMENTAL DESIGN FOR EVALUATION OF OUR PROPOSED TECHNIQUE

In order to demonstrate the accuracy of our proposed technique, we perform a set of experiments. We present the experimental setup, demography of collected data, and experimental results in this section.

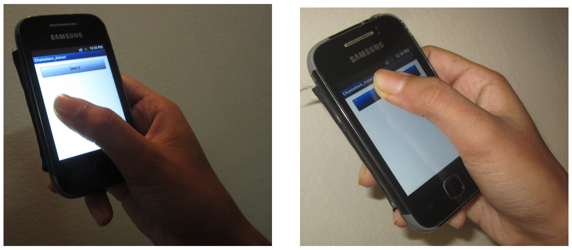
5.1 Experimental Platform

To perform our experiment, we adopt a touch-based device for data collection. Here, we choose Samsung Galaxy Young gt-s5360A. The device possesses a processing capability of 832 MHz ARMv6 along with a Broadcom VideoCore IV and memory of 384MB RAM [23]. We developed an Android application to get user's touch-based usage data using the device.

5.2 Application for Data Collection

We developed our application in Android 2.3 (Gingerbread) for our data collection. Our application monitors and records usage patterns of different users from the perspective of the 13 features. In our data collection process, each user is requested to do two specific simple operations- a fixed swipe on the touchscreen and then press a specific button. Our application's screenshot and step-wise tasks are shown in Fig. 3. In Fig. 4a and Fig. 4b, we present a user in action while using the application.

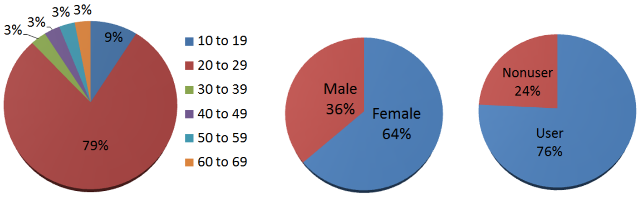
In our experiment, we have collected data from several users following the same process. We present a demography of the participants next.



(a) Fixed swipe over the touchscreen

(b) Press a specific button over the touchscreen

Figure 4: Data collection procedure of our proposed technique



(a) Age range (years)

(b) Gender

(c) Touchscreen usage

Figure 5: Demography of participants

5.3 Demography of the Participants

We have collected data from 33 participants for our experiment. The participants cover different age groups (from 11 to 67 years old), different cultural and educational backgrounds, and having different levels of experience interacting with touchscreen electronic devices. We further present the demography of participants in Fig. 5.

Fig. 5a shows that our participants mostly exhibits youth having an age range of 20 to 30 years. We have picked such skewed diversity in terms of age range, as survey study [24], [25] exhibit a similarly skewed diversity in favor of youth for usage of touchscreen electronic devices. Besides, Fig. 5b presents that we have covered a significant number of both male and female in our experiment. Finally, Fig. 5c shows that our experiment covers non-users of touchscreen electronic devices in accordance with users of touchscreen electronic devices. We have collected data from all of the participants for 15 times.

According to [24] and [25], smartphone users disproportionately tend to be 18 to 34 years old which is represented in our demography fig. 5a also.

6. EXPERIMENTAL DATA ANALYSIS

Before implementing our proposed solution, we conduct a study to analyze the behavior of the experimental data for unsupervised learning. The objective of performing unsupervised clustering is to investigate whether there lies any user-cluster mapping for the extracted features. Then we perform a thorough analysis of our proposed solution and investigate, if after clustering a set of training data pertinent to a single user, we can identify the user as the original one and identify other users as aliens based on newly-input test data from all the users. We present our analytical outcomes along with how far we could reach the objectives in both cases in the following subsections.

6.1 Outcomes of Unsupervised Clustering

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
4	0	1	0	0	0	1	0	0	0	0	0	0	1	4	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	1	0	3	2	0	0
7	1	0	0	0	0	0	0	0	4	1	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	1	0	0	0	2	2	0	0	0	1	0	0	1	0	0	0	0	0	1	3	1	0	0	0	0	0	0	0	0	0	0
9	0	0	1	0	0	0	5	1	0	0	4	4	0	0	0	0	0	0	7	0	0	1	3	4	0	0	0	1	0	0	0	0	0
10	0	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	0	0	6	0	0	1	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	1	0	0	2	0	1	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	2	0	0	0	0	0	0	1	0	0	6	0	0	0	0	0	0	1	2	0	0	0	0	0	0	0	0	0	0
16	4	4	5	7	0	6	0	4	0	0	2	0	5	0	0	0	6	3	0	0	0	2	0	0	0	0	0	2	0	0	0	0	2
17	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	2	0	0	0	0	0	0	0	1	0	0	0	0	0	0	2	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	7	0	0	0	0
23	0	2	0	0	0	0	0	0	0	0	0	0	0	3	0	0	1	0	0	0	0	0	0	2	3	2	0	0	0	2	4	1	5
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	2	0	0	0	3	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
26	0	0	0	0	0	0	0	0	1	4	0	0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0

Table 1: Findings of unsupervised clustering

In unsupervised clustering, we simply attempt for clustering all the available data collected from all the users without telling the owner of the data. After applying the unsupervised clustering on our collected data, we find 26 clusters. Note that, the number of clusters is close to the number of participants, i.e. 33. Findings of unsupervised clustering is shown in Table. 1. Here, each row represents a cluster and each column represents a user.

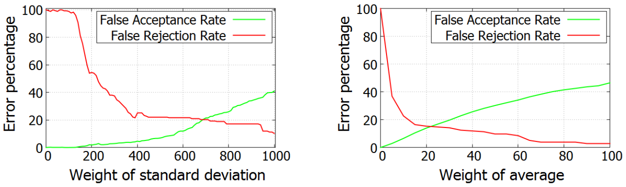
Analyzing the data presented in Table. 1, we can find that most of the rows in the table, i.e., clusters obtained by unsupervised clustering, is covered by a few columns, i.e., by a few of the participants. Besides, most of the participants cover only a few clusters, as most of the columns cover only a few rows. Here, both the blue and red shaded cells of the table exhibits the covering.

If we investigate a bit more, we can find that most of the clusters are dominated by a very small number of users, which is very close to 1 in most of the cases. Alongside, most of the users dominate only a very small number of clusters, which is again very close to 1 in most of the cases. We present the dominances in the table using red shaded cells.

Now, the above findings suggest that there might be a near one-to-one mapping between the users and the clusters of data. Therefore, it might be possible to efficiently cluster the data such that the users could be identified through the clustering. Next, we present outcomes of one such efficient clustering, which we have already elaborated in Section 4.

6.2 User identification using Mean-SD clustering

To perform an in-depth investigation of out proposed Mean-SD clustering over all of our collected data, we cluster the collected data from the perspective of both individual features and multiple features. We perform a number of iterations with such clustering tasks using randomly chosen sets of data for both individual features and all features. We perform these clustering tasks to lead towards efficient user identification. Moreover, we investigate the effect of allowing mismatch among the features in our identification process. We describe all the identification processes below.



(a) Effect of changing only the weight of standard deviation (keeping the weight of average to 0)

(b) Effect of changing only the weight of average (keeping the weight of standard deviation to 0)

Figure 6: Effect of changing either the weight of standard deviation or the weight of average on False Acceptance Rate and False Rejection Rate for the feature of rotation about the Z-axis while pressing the button in our developed application

6.2.1 *Single Feature Identification*

At first, we attempt to identify users based on each single feature. To do so, we vary the cluster size depending on the weights of standard deviation and average. We analyze the effect of both the weights through changing only one while keeping the other constant. The effect of changing only the weight of standard deviation keeping the weight of average fixed to 0 is shown in Fig. 6a. Similarly, the effect of changing only the weight of average keeping the weight of standard deviation fixed to 0 is shown in Fig. 6b. Both Fig. 6a and Fig. 6b are pertinent for the feature of rotation about the Z-axis while pressing the button. Note that, here we exploit Eq. 1 already presented in Section 4.

To further investigate the impact of changing weights of average and standard deviation, we separately analyze their impacts on False Acceptance Rate (FAR) and False Rejection Rate (FRR). Fig. 7 portrays the individual impacts. Here, Fig. 7a shows the effect of changing the weights on False Acceptance Rate for a single feature. Here, we consider the feature of rotation about the Z-axis while pressing the button in our developed application. This figure demonstrates that with an increase in the cluster size, the False Acceptance Rate increases rapidly. Here, note that we can increase the cluster size through increasing either of the weights individually or both the weights simultaneously. Similarly, Fig. 7b shows the effect of changing the weights on False Rejection Rate for a single feature. Here, in contrast to the previous case, the False Rejection Rate decreases with an increase in the cluster size.

Now, after analyzing the impacts of different possible values of the weights, we can find that the best possible accuracy of the result obtained in the case of the feature of rotation about the Z-axis while pressing the button with the cluster size of $(AVG \times 5\% + SD \times 370\%)$. Fig. 7c presents this best-possible results. Here, the lowest value of FRR is 8% and the lowest value of FAR is 11%.

Following the same process, we compute FAR and FRR for each of the features. Fig. 8 shows the effects of changing weights of average and standard deviation on False Acceptance Rate and False Rejection Rate for each individual feature.

Note that, even though FAR and FRR vary for different features, all the values of FAR and FRR are highly significant. FAR and FRR for only one feature exhibit significant values rendering it not a feasible solution. Now, after reaching this extent, we apply the clustering technique to a different combination of features. Therefore, next, we present outcomes of user identification based on multiple features.

6.2.2 *Multiple Features Identification*

In addition to attempting to identify a user based on a single feature, we also attempt for the same identification task based on matching multiple features. The purpose behind such attempt is two-folded. Firstly, to improve the accuracy of identification. Secondly, to make the system more secure, as it is comparatively much more difficult for a malicious user to simultaneously mimic more than one behavior of the original user.

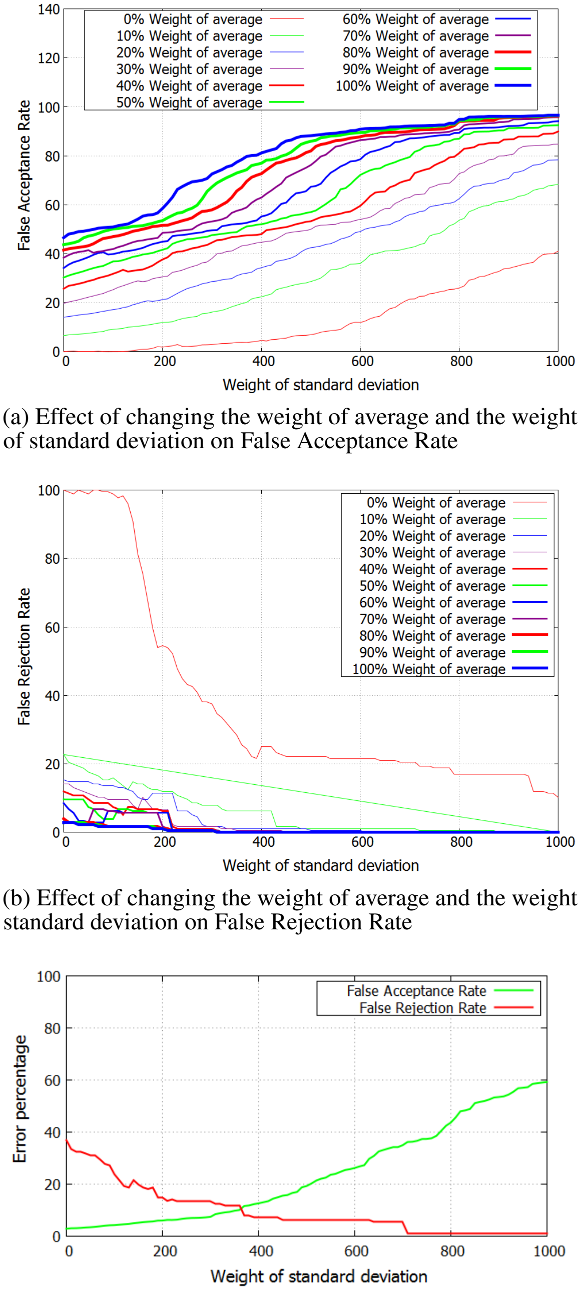
In our analysis based on multiple features matching, we utilize the same weight of average for all the features as well as the same weight of standard deviation for all the features. Similar to our previous analysis, here, we independently vary both the weights.

We present the effect of changing the weights of average and standard deviation on False Acceptance Rate for all features is clarified in Fig. 9a. Here, the finding is similar to that we have already found for a single feature. The False Acceptance Rate increases with an increase in the cluster size. Similarly, the False Rejection Rate decreases with an increase in the cluster size. Fig. 9b shows this result.

The best result obtained while considering all features in combination is for the cluster size of $(AVG \times 5\% + SD \times 530\%)$. Fig. 9c presents variations FAR and FRR pertinent for this cluster size. Here, the best possible outcome provides only 5% FAR and 6% FRR.

Now, if we compare the results pertinent for considering only a single feature and the results pertinent for considering all features in combination, we can find some interesting observations. After analyzing Fig. 7a and Fig. 9a, we find that the FAR decreases if we consider all features in combination compared to the case of considering only a single feature. In the case of FRR, the scenario can get changed. Here, FRR may decrease if we consider only a single feature compared to the case of considering all features in combination. Fig. 7b and Fig. 9b present such a case of getting decreased FRR through considering only a single feature.

Even though we get two opposing trends in FAR and FRR while considering the features in isolation and while considering all the features in combination, we can achieve the best possible outcome

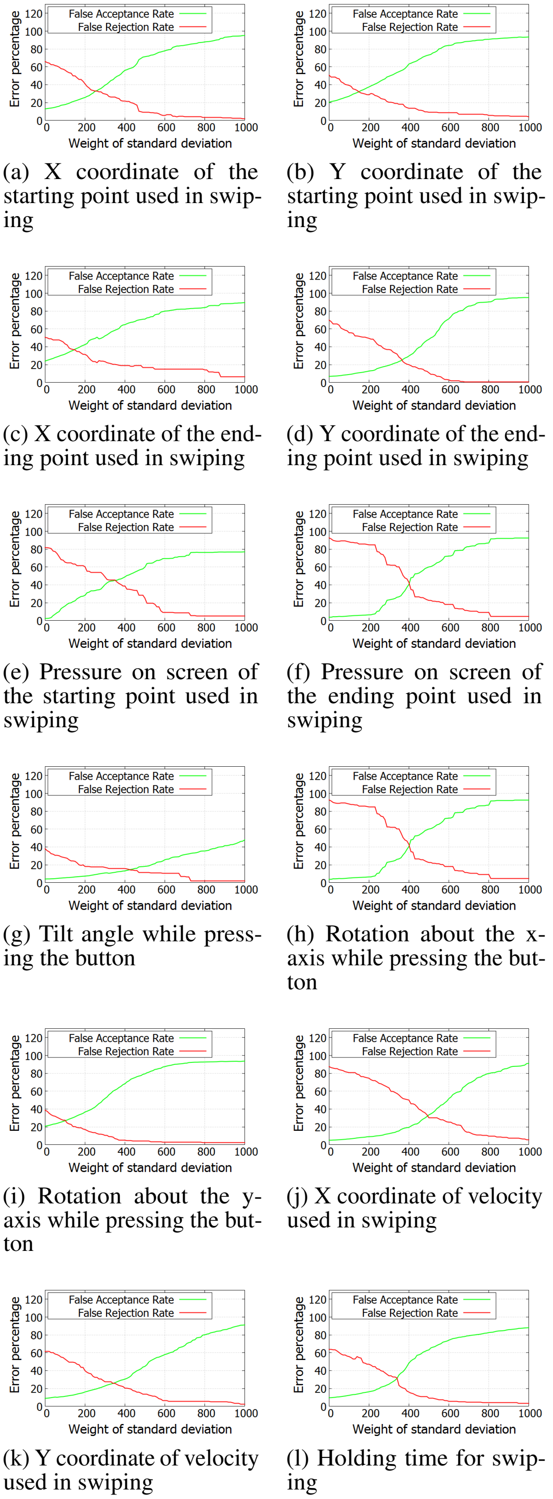


(a) Effect of changing the weight of average and the weight of standard deviation on False Acceptance Rate

(b) Effect of changing the weight of average and the weight standard deviation on False Rejection Rate

(c) False Acceptance Rate and False Rejection Rate for the feature providing the best-possible outcome

Figure 7: Effect of changing the weight of average and the weight of standard deviation on False Acceptance Rate and False Rejection Rate for the feature providing the best-possible outcomes, which is rotation about the Z-axis while pressing the button in our developed application



(a) X coordinate of the starting point used in swiping

(b) Y coordinate of the starting point used in swiping

(c) X coordinate of the ending point used in swiping

(d) Y coordinate of the ending point used in swiping

(e) Pressure on screen of the starting point used in swiping

(f) Pressure on screen of the ending point used in swiping

(g) Tilt angle while pressing the button

(h) Rotation about the x-axis while pressing the button

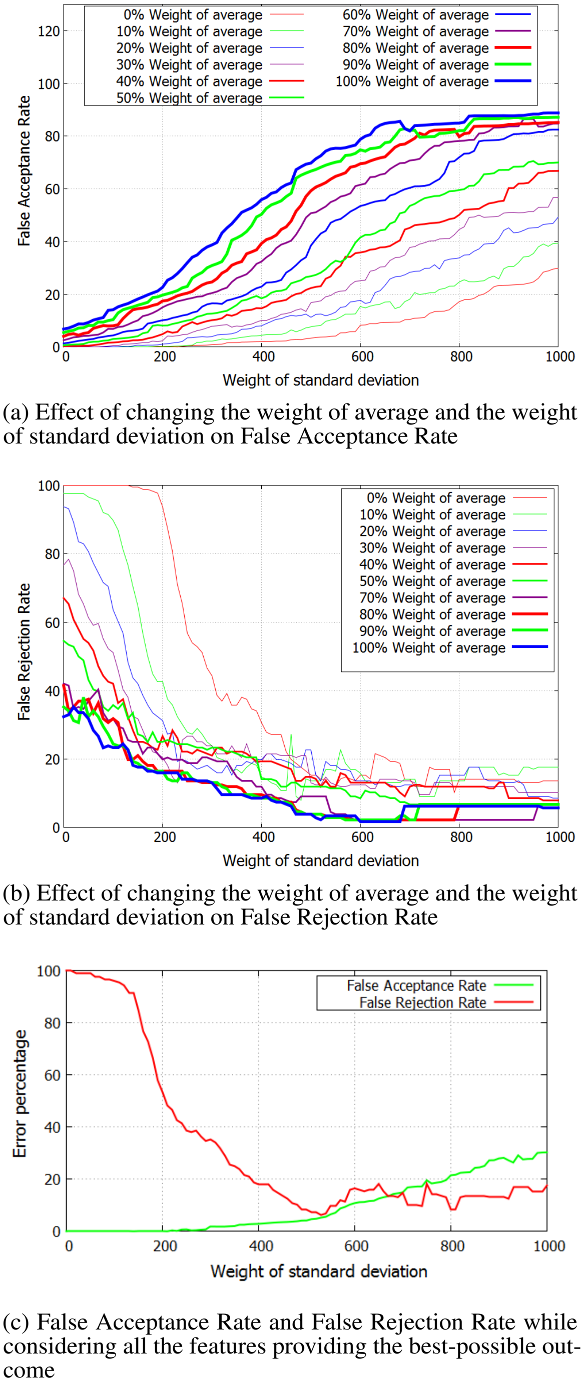
(i) Rotation about the y-axis while pressing the button

(j) X coordinate of velocity used in swiping

(k) Y coordinate of velocity used in swiping

(l) Holding time for swiping

Figure 8: Effect of changing the weight of average and the weight of standard deviation on False Acceptance Rate and False Rejection Rate while considering single feature



(a) Effect of changing the weight of average and the weight of standard deviation on False Acceptance Rate

(b) Effect of changing the weight of average and the weight of standard deviation on False Rejection Rate

(c) False Acceptance Rate and False Rejection Rate while considering all the features providing the best-possible outcome

Figure 9: Effect of changing the weight of average and the weight of standard deviation on False Acceptance Rate and False Rejection Rate while considering all the features providing the best-possible outcomes

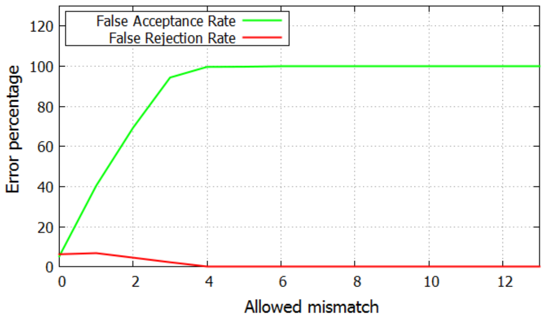


Figure 10: Effect of allowing mismatch while considering all features

in the case of considering all features in combination. Fig. 7c and Fig. 9c validates the phenomena of achieving the better outcome through considering all the features in combination.

6.2.3 *Allowing Mismatch over the Attributes*

Another important observation is related to allowing mismatch. We observed the result allowing mismatch on the best possible result for considering all features. Accuracy decreases while allowing mismatch of features. With the increase in allowed mismatch, the FAR increases instantaneously from zero to 100% and the FRR decreases to zero. In fig. 10, the effect of allowing mismatch of features (0-12) is shown.

7. COMPARATIVE ANALYSIS OF CLASSICAL MACHINE LEARNING ALGORITHMS WITH MEAN-SD

We analyze the outcome of collected experimental data with several classical machine learning algorithms. The objective behind performing this analysis is to perform a comparative evaluation of the performance of our proposed solution against the classical machine learning algorithms.

We choose three classical machine learning algorithms name by k-NN [26], decision tree [27], and a multiclass perceptron algorithm: Kessler's construction [28] considering their wide acceptability in the literature. We present the findings of these machine learning algorithms in Table 2.

We find that k-NN algorithm provides lower false rates compared to Decision tree and Kessler's construction algorithms. More specifically, the FRR is significantly higher for Decision tree and Kessler's construction compared to k-NN. Here, the FRR gets decreased for k-NN with decrease in the value of k. However, with decrease in the value of k, the FAR gets increased.

Table 2 also presents the running time needed to identify a user using the machine learning algorithms and also our proposed Mean-SD clustering technique. We can see that k-NN needs more execution time than other approaches. This is because for each test sample, k-NN checks against all the train samples and finds the best k matches. So, if we have M training samples each with dimension d, then k-NN algorithm needs O(Md) time to detect the class of a test sample. The time increases with the increase of train dataset. Where the other approaches only need a maximum of O(d) time. There is another significant drawback of using k-NN which is, it consumes a large memory as it stores the whole training dataset. Therefore, though k-NN has lower FAR and lower FRR, the large memory usage and longer execution time make it unsuitable for practical implementation. Decision tree algorithm also needs a large memory to store the whole tree. Also, the FRR is too high in this case. Kessler's construction has the highest FRR among these approaches. Comparing with these classical machine learning algorithms, our proposed Mean-SD clustering technique provides the best result with 5% FAR, 6% FRR and the lowest execution time and memory usage.

Algorithm	Parameter	FAR	FRR	Run-time
k-NN	k=1	1%	9%	$1.1e^{-4}$s
	k=3	1%	10%	$1.5e^{-4}$s
	k=5	0%	11%	$1.6e^{-4}$s
Decision tree		0%	16%	$1.94e^{-6}$s
Kessler		2%	41%	$2.04e^{-6}$s
Mean-SD		5%	6%	$1.6e^{-6}$s

Table 2: Comparative analysis of classical machine learning algorithms and our proposed technique

8. EXPERIMENTAL FINDINGS

This section describes the findings of our experiment. According to our experimental procedure, we get the following findings:

- Our experimental result of several Machine Learning algorithms is shown in Table. 2. We find higher FRR, though the FAR is a bit low.

- The experimental result of our proposed Mean-SD clustering demonstrates that the False Acceptance Rate increases rapidly with an increase in the cluster size and the False Rejection Rate decreases rapidly with an increase in the cluster size. These results are shown in Fig. 7a and Fig. 7b. The best possible accuracy of the result is obtained in the case of the feature of rotation about the Z-axis while pressing the button with the cluster size of $(AVG \times 5\% + SD \times 370\%)$. Fig. 7c presents this best-possible results experiencing FRR is 8% and FAR is 11%.

- Our experiment discovers that relying on single feature is comparatively less secure as it is easy to mimic single feature as well as data may vary time to time for an individual user. Fig. 8 shows the low accuracy while considering a single feature.

- Our experimental result of considering all features is shown in Fig. 9c. The best result obtained while considering all features in combination is for the cluster size of $(AVG \times 5\% + SD \times 530\%)$. Here, the best possible outcome provides only 5% FAR and 6% FRR.

- The experimental result of allowing mismatch on the best possible result for considering all features is shown in Fig. 10. Accuracy decreases while allowing mismatch of features.

9. IMPLEMENTATION

To evaluate the actual performance of our proposed technique, we have developed an Android application, which detects a user according to the *Mean-SD Clustering* algorithm as already discussed in this paper. For a specific user, the app initially takes training dataset from the user for a different number of times. This training dataset is used to distinguish the user and other intruders. To our implementation, we set the tolerance level to two different values - 4% tolerance percentage of average value and 430% tolerance

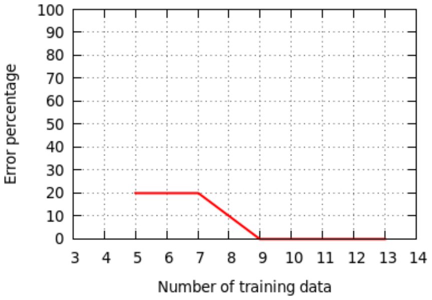


Figure 11: Performance measure for varying number of training data

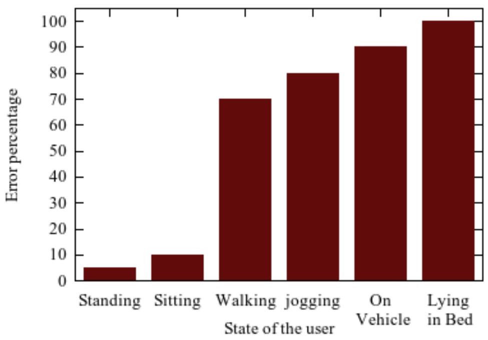


Figure 12: Performance measure for varying state of user

percentage of standard deviation. We set these values following our earlier findings.

We used the following devices:

Samsung GALAXY Tab 10.1 LTE SC-01D: This device has Android version 3.2, dual-core 1.5 GHz CPU, and 1 GB RAM. Sensors for touch detection, finger movement velocity detection, and tilt angle detection were present in the device.

HTC Desire 626: This device has Android version 4.4.4, quad-core 1.2 GHz CPU, and 1 GB RAM. Sensors for touch detection and finger movement velocity detection were present in the device.

We let 13 users use our developed system. 11 of them were from the age range $20-29$ and 2 of them from the age range $30-39$. Besides, 12 of them were regular smartphone users. Alongside, 6 participants were male and 7 were female. 234 tests were performed in total on two devices and for different states of the user. Analyzing outcome of our implementation we have found False Acceptance Rate to be 0.00% and False Rejection Rate to be 31.78%. It is worth mentioning that the False Acceptance rate retains the value 0.00% even for making an attempt through shoulder surfing. This ensures that the solution has a very high precision which is the prime motivation for this research. While testing on device with less number of sensors, we have found higher False Acceptance Rate.

We also have measured the accuracy rate for different training dataset which is shown in Fig. 11. It is found that by taking a minimum of 9 sets of data from the users in the training set, our system exhibits the maximum performance. Moreover, we have measured the accuracy while trying to enter the system in different states of the user. The result is shown in Fig. 12. It is found that the stationary states exhibit good performance except the state of lying in the bed as this state changes the angular position of the device drastically.

Nowadays, the smart phone devices are used for storing various secure and important data of the user such as SSN or bank account information. For protecting such sensitive information where security break is not acceptable at all, our proposed technique is best suited as it has 0% False Acceptance Rate even when trying to mimic through shoulder surfing. False Rejection Rate indicates that, for every 3 attempts by the user, only one is falsely rejected. To ensure the security of such sensitive information, this FRR is acceptable as it gives the highest precision.

10. FUTURE WORK

The main purpose of our system is to identify a valid user of an electronic device and distinguish his/her from any other imposter. The usage of a user is a kind of a signature identification of that particular user. In our experiment, we collected 15 observations from each user in which 10 observations were used as training set and rest 5 as the test set. However, if we would collect more observations' data, the database would get enriched, and hence identify the user in a more precious level. Thus, any imposter could also be detected and the system could prepare itself for a possible security breach. We intend to work at the kernel level of a device to enhance the performance of identifying a user through following this approach. This would unlock a vast area of observing a user's usage, and make the system more robust. Besides, in our system, we used Mean-SD clustering where true cluster size varies with the weights of average and standard deviation. We plan to explore other alternatives to calculate the cluster size in this regard. issues in future to identify the correct user under different circumstances. Besides, in our experimental work, we developed our application on an Android device. In future, we plan to investigate similar mechanism in other devices. Additionally, we plan to tune our system in the implemented device such that the extent of FRR gets reduced.

Our implementation shows that our proposed technique has higher false rejection rate for moving state and lying on bed state of the user. We plan to work on improving the performance for these two states in the future.

11. CONCLUSION

As the usage of technology is increasing day-by-day, users often face the necessity of protecting their confidential and sensitive information from others. The first step for doing so is to identify valid and invalid users. Analyzing usage of electronic devices can facilitate such identification task. However, to the best of our knowledge, state-of-the-art technologies in this regard have focused on this important aspect of usage monitoring through either in a run-time manner requiring high system overhead and resources or through incorporating password/PIN that exhibits significant vulnerability under different types of security threats such as eavesdropping, shoulder surfing, etc.

Hence, in this paper, we propose a single-time user identification technique utilizing touch-based and holding orientation based usage monitoring. Here, we apply various existing machine learning approaches to carry out our user identification task. Those approaches offer relatively low-accuracy having significant resource usage, which indicates the necessity of a light-weight and high-accuracy approach. Therefore, we propose a novel clustering technique named Mean-SD clustering to perform our user identification task with high accuracy incurring low resource overhead.

We perform a set of rigorous experimental evaluation to validate the efficacy of our proposed user identification technique. The experimental results indicate that our proposed technique is highly accurate in user identification. Analyzing collected data from 33 users, we find that our technique can identify users with only 5% False Acceptance Rate and 6% False Rejection Rate. Here, we exploit our proposed light-weight clustering technique to confirm its implementation to be easy-to-implement and less resource hungry. The exploitation demonstrates that our proposed technique can be implemented in any off-the-shelf smartphone without the need of any additional hardware. We confirm the potency of our proposed technique to be implemented through developing its real implementation in an Android device. We demonstrate the efficacy of our proposed technique through letting 13 users use the implemented device. Usage of the users reveal that our proposed technique cannot be breached by intruders, i.e., FAR remains to 0%, even after making attempts through eavesdropping and shoulder surfing. Therefore, we envision that our proposed technique will offer a pervasive solution for user identification to mass users for touch-based electronic smart devices.

12. REFERENCES

[1] Number of smartphone users* worldwide from 2012 to 2018 (in billions), Retrieved 30 August, 2015 from http://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/

[2] T. Vu, A. Baid, S. Gao, M. Gruteser, R. Howard, J. Lindqvist, P. Spasojevic, and J. Walling, Distinguishing users with capacitive touch communication, in ACM MobiCom, pp. 197-208, 2012.

[3] S. Nanavati, M. Thieme, and R. Nanavati, Biometrics identity verification in a networked world, John Wiley & Sons, 2002.

[4] N.L. Clarke and S.M. Furnell, Authentication of users on mobile telephones - A survey of attitudes and practices, Computers & Security, Vol. 24, pp. 519-527, 2005.

[5] A. Adler, Vulnerabilities in biometric encryption systems, Audio- and Video-Based Biometric Person Authentication, Vol. 3546, pp. 1611-3349, 2005.

[6] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni, Fingerprint Image Reconstruction from Standard Templates, Pattern Analysis and Machine Intelligence. Vol. 29(7), pp. 1489-1503, 2007.

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

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

[9] S. Ahmed, A.S.M. Rizvi, R.S. Mansur, M.R. Amin, and A.B.M.A. Al Islam, User identification through usage analysis of electronic devices, International Conference on Networking Systems and Security (NSysS), pp. 1-6, 2015.

[10] Google Glass Snoopers Can Steal Your Passcode With a Glance, Retrieved on 30 August, 2015 from http://www.wired.com/2014/06/google-glass-snoopers-can-steal-your-passcode-with-a-glance/

[11] X. Zhang, J. Seifert, and O. Aciicmez, Design and implementation of efficient integrity protection for open mobile platforms, Mobile Computing, IEEE Transactions, Vol. 13(1), pp. 188-201, 2014.

[12] H.M. Wood, The use of passwords for controlled access to remote computer systems and services, AFIPS '77 Proceedings of the June 13-16, 1977, national computer conference, pp. 27-33, 1977.

[13] Y. Sutcu, Q. Li, and N. Memon, Secure Biometric Templates from Fingerprint-Face Features, IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007), pp. 1-6, 2007.

[14] M. Qi, Y. Lu, J. Li, X. Li, and J. Kong, User-Specific Iris Authentication Based on Feature Selection, International Conference on Computer Science and Software Engineering, Vol. 1, pp. 1040-1043, 2008.

[15] J. Bigun, J. Fierrez-Aguilar, J. Ortega-Garcia, and J. Gonzalez-Rodriguez, Combining Biometric Evidence for Person Authentication, Advanced Studies in Biometrics, Vol. 3161, pp. 1-18, 2005.

[16] R. Brunelli, and D. Falavigna, Person identification using multiple cues, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 17(10), pp. 955-966, 1995.

[17] T. Feng, J. Yang, Z. Yan, EM. Tapia and W. Shi, TIPS: Context-Aware Implicit User Identification using Touch Screen in Uncontrolled Environments, HotMobile '14 Proceedings of the 15th Workshop on Mobile Computing Systems and Applications, 2014.

[18] C. Bo, L. Zhang, and X. Li, Silentsense: Silent user identification via dynamics of touch and movement behavioral biometrics, MobiCom '13 Proceedings of the 19th annual international conference on Mobile computing & networking, pp. 187-190, 2013.

[19] J. Angulo, E. WÃď'stlund, Exploring Touch-Screen Biometrics for User Identification on Smart Phones, Privacy and Identity Management for Life, Vol. 375, pp. 130-143, 2012.

[20] N. Sae-Bae, K. Ahmed, K. Isbister, and N. Memon, Biometric-rich gestures: a novel approach to authentication on multi-touch devices, Proceeding CHI '12 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 977-986, 2012.

[21] A. De Luca, A. Hang, F. Brudy, C. Lindner, and H. Hussmann, Touch me once and I know it's you! Implicit Authentication based on Touch Screen Patterns, CHI '12 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 987-996, 2012.

[22] A. Buriro, B. Crispo, F. Del Frari, and K. Wrona, Touchstroke: Smartphone User Authentication Based on Touch-Typing Biometrics, New Trends in Image Analysis and Processing – ICIAP 2015 Workshops, Vol. 9281, pp. 27-34, 2015.

[23] Samsung Galaxy Young gt-s5360, Android 2.3 (Gingerbread), Retrieved 19 September, 2015 from https://en.wikipedia.org/wiki/Samsung_Galaxy_Y

[24] Mobile Demographics, Retrieved on 19 September, 2015 from http://ipcarrier.blogspot.com/2009/11/surprising-smartphone-statistics.html

[25] Smartphone User Growth Statistics and Trends, Retrieved 19 September, 2015 from http://brandongaille.com/smartphone-user-growth-statistics-and-trends/

[26] L.E. Peterson, K-nearest neighbor. Scholarpedia, 4(2), p.1883, 2009.

[27] A survey of decision tree classifier methodology, Retrieved on 19 September, 2015 from http://ntrs.nasa.gov/search.jsp?R=19910014493/

[28] D. B. Sher and D. Milun. "Generating edge detectors from a training ensemble." In Optical Engineering and Photonics in Aerospace Sensing, pp. 165-176. International Society for Optics and Photonics, 1993.