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

ABSTRACT

The ability of smart devices' to recognize their owner gains attention with the advent of widespread sensitive usages of these devices such as storing secret and personal information. Unlike the existing techniques, in this paper, we propose a very lightweight single-time user identification technique that can ensure a unique authentication offering a near-to-impossible system to breach by attackers. Here, we have conducted a thorough study over single-time usage data gathered from 33 users. The study reveals some new findings, which in turn, leads us to a novel solution exploiting a new machine learning technique. Our evaluation confirms that the proposed technique operates as good as only 5% false acceptance rate (FAR) and only 6% false rejection rate (FRR). We further evaluate the performance measure through comparing its performance with some traditional machine learning techniques. Finally, we perform a real implementation of this technique as a mobile application to conduct a rigorous study in order to show how this technique works in practical situations. Outcomes of the study demonstrate as low as 2.72% FAR which ensures extremely low false rate.

Keywords

Smartphone, Behavioral biometrics, Security, Mean-SD Clustering.

1. INTRODUCTION

The use of smartphones dramatically changed over the last decade. Now, they are hardly been using for making phone calls, rather being used as a replacement for many electronic gadgets as well as non-electronic substances [1]. Their pervasive applications start including sensitive uses, for example, doing the bank transactions, storing sensitive data, financial activities, emails, etc., and people are frequently entrusting these devices with such secret personal information which are prone to face attackers anytime. Consequently, the issue of identifying the owner of these devices to ensure the absolute security has become a prime concern in recent times.

According to the study presented in [2], even though people are concern about the security issues of smart devices, they are uncomfortable using the available security techniques. The available user identification techniques can be divided into three fundamental categories [3], which are: (i) knowledge-based, (ii) object-based, and (iii) biometric-based. However, besides having their own drawbacks, all of these techniques can be a subject to burglary. Category itypically includes password or PIN system which is inconvenient for users as they require memorizing. Furthermore, they can be stolen as easily as eavesdropping or shoulder surfing [4]. Object-based approaches usually rely on possession of token which, if lost or stolen, imposters will get access to unauthorized data. Though biometric-based techniques rely on the uniqueness of physical/behavioral characteristics of a person, they can be easily mimicked or stolen by an intruder.

Recently, many studies are performed exploring user identification based on usage monitoring. One such study conducted by [5], which takes usage data and continuously performs authentication. Continuous authentications [5–8] keeps authenticating current users thus gives more security against impostors. Such techniques require passwords or fingerprints or face-detections constantly, which makes these techniques not of resource-hungry but also very user-unfriendly (requiring passwords or security questions)[9, 10] or costly (requiring extra devices for face-detection or fingerprint). Therefore, a simple, single-time, easy-to-use, less resource-hungry user identification technique is yet to be discovered.

In this paper, we address this issue by introducing a new type of single time, behavior analytics based user identification technique. Taking into account two human behaviors—how the screen is being touched and how the device is being held—this technique exploits a new machine learning mechanism namely Mean-SD clustering. 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 hard 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 studied 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 2.72% 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 and motivation of our work. Section 3 presents our proposed technique for user identification, which includes working methodology, data capturing, and user identification through clustering. Section 4 elaborates experimental design for evaluation of our proposed technique including experimental platform, developed application for data collection, and demography of the participants. In Section 5, we evaluate experimental data analysis. Section 6 makes comparative analysis over the performance of our proposed technique compared to either available machine learning techniques. Section 7 summarizes experimental findings. Section 8 presents outcomes of user evaluation through an implementation of our proposed technique is a smartphone.Finally Section 9 and 10 concludes the paper with pointing our future work.

2. RELATED WORK AND MOTIVATION

Traditional smartphone security system includes password or pin or pattern lock. Behavioral biometric techniques are also being used increasingly nowadays. All these techniques need memorizing except behavioral biometric technique. In addition to that, those techniques can be guessed or stolen as easily as shoulder surfing [11]. A study conducted on password based system [12] using keyboard layout which can withstand shoulder surfing. However, this system still needs memorizing as well as strongly depend on keyboard layout of a specific language. Though behavioral biometric techniques, for example, fingerprints, cannot be guessed and gives higher security but they can also be stolen or mimicked by imposters. Besides, they often require additional hardware, hence, can be costly. One such approach studied in [13], introduces an authentication system based on simultaneous face and voice recognition. Because of using two biometric characteristics, this system is hard to break. However, it requires too much resource as well as costly, therefore, not suitable for low-resource devices such as mobile devices.

The study in [14] is another example of behavioral biometric based system. It uses an additional identification layer immediately after password/pattern approach. Furthermore, another study [15] classifies users based on the usage of the user's hand movements while holding the device, and the timing of touch-typing when the user enters 4-digit PIN/password. Though both of these approaches gives high security, they are still vulnerable as they use passwords/PIN.

Two other studies [16] and [17] develops systems which enhance the security. [16] implements biometric analysis in combination with pattern recognition and [17] uses multi-touch gestures. However, this mechanism presents difficulties to users as it demands complex gestures.

Studies conducting on continuous authentication [5, 18, 19] performs user identification in the background continuously. Even though these systems give higher security, these are not suitable for deploying in low-resource devices as they are consuming more power and extremely resource-hungry.

In summary, conventional single-time user identification systems exhibit thread to being stolen or mimicked. The initial motivation for our research arose from the need to provide difficult-to-mimic user identification technique for mobile devices. Although continuous identification systems give more security, their strength depends on run-time data capturing and rigorousness of continuous data analyzing, which makes these unsuitable for low-resource devices as they consume 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. Touch-based usage monitoring is free from the hassle of memorization. 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. 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.

3. PROPOSED TECHNIQUE FOR USER IDENTIFICATION

In this section, we describe our proposed technique in details. Our working methodology involves two phases: training

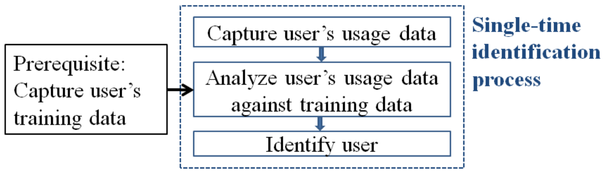


Figure 1: Methodology of our application

phase and user identification phase. The training phase is done only once. Based on the training result, user identification is performed. Two phases are elaborated in this section. First, we present the overview of working methodology along with the steps in training phase. Subsequently, we elaborate the user identification technique in details.

3.1 Working Methodology Overview

Our proposed technique requires training data on which the user identification will perform. The training data is collected from the user and immediately used to train the application. The training results are stored in the device based on which identification task will be performed. The training data will no longer be needed. After training, the system will be ready to identify its valid user. Identification starts by capturing usage data provided by the person who is using the device currently. This usage data then measured against the stored training data and provides a result. Based on this result, user identification is performed. Fig. 1 shows a block diagram of the proposed method.

3.2 Training Phase

Training phase includes two steps—collection training data and generating training result. This training phase occurs only once at the first time starting the application with the data given by the owner of the device. Later the device automatically identifies it's owner based on the result obtained from this training phase. The two steps are elaborated in the following subsections.

3.2.1 *Collecting Data*

Before designing our proposed technique, we studied user behavior based on touch usage. As a result of the study, we identified that we can extract 13 features from any user's touch usage. These 13 features individually do not show any significant characteristic that can specify a user. However, in combination, these features contribute to the unique identification of the user. These 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.

3.2.2 *Training*

As discussed before, each usage data contains 13 features, therefore, each data can be considered as a 13-dimensional vector. In the data collection phase, we collect such data sets

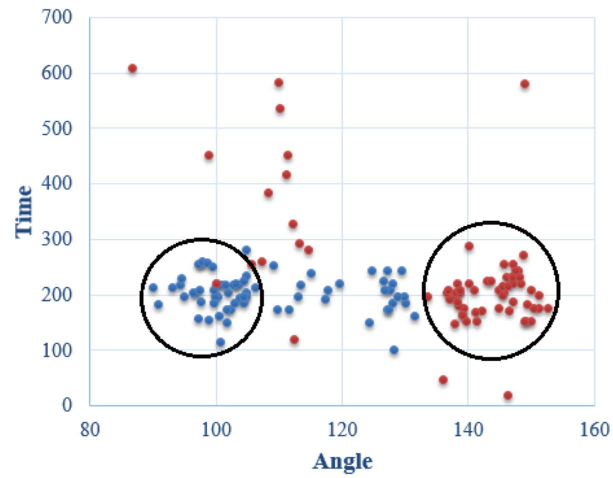


Figure 2: Proposed Mean-SD Clustering technique

from the user for training. Such sets of data for a specific user forms a cluster in 13-dimensional space. Consequently, we can say that in 13-dimensional space, data collected from each person forms separate clusters and therefore, they can be uniquely identified later if we know their cluster information. Our proposed solution identifies the cluster for each user by their cluster centroid, which is the mean of cluster members and cluster size. In the training phase, after collecting training data, the algorithm calculates the cluster information and stores in the device. The cluster size is varied and determined by taking a weighted sum of the average and standard deviation, which can be expressed by Eq. 1 presented as follows:

$$C = \left(M_{\text{avg}} \times W_{\text{avg}}\right) + \left(S_{\text{SD}} \times W_{\text{sd}}\right) \tag{1}$$

where M_{avg} represents average and S_{SD} represents the standard deviation of the cluster in training data. Besides, W_{avg} and W_{sd} are the weight of average and the weight of standard deviation of the cluster members respectively. Fig. 2 demonstrates the impression of such cluster size. Though in our case, the cluster will be in 13-dimensional space, here, we represent in 2D to make it visible in a simple manner. Only two features–time and angle–are considered in this figure. Time is taken along X-axis, and Y-axis represents angles.

3.3 User Identification

User identification starts by collection a single usage data of the current user. Like before, 13 features are extracted from this usage data. This data will indicate a point in the 13-dimensional region which will be used to determine the user. From the training phase, we have the cluster centroid and the cluster size stored in the device, which represents the biometric characteristics of a user. The distance from the new point to the cluster centroid is calculated. The user is identified the owner if the new point falls into the cluster size. Otherwise, the user will be identified as a malicious user. Our proposed clustering technique identifies users based on their clusters which are defined by their cluster centroid

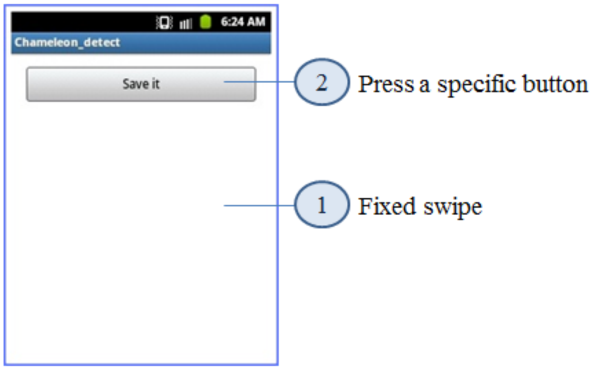
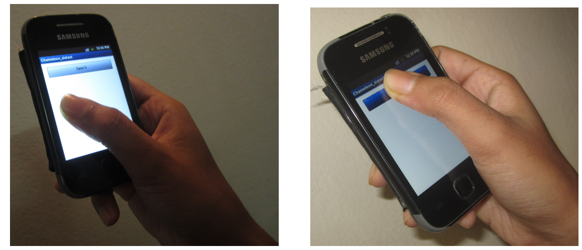


Figure 3: Screenshot and step-wise tasks of our application



(a) Task 1: Fixed swipe over the touchscreen

(b) Task 2: Press a specific button over the touchscreen

Figure 4: Data collection procedure in our experimentation

and size. The cluster size can be varied by the mean and standard deviation of the cluster. For this reason, we name our proposed technique as Mean-SD clustering.

3.4 Performance Matrics

In the training phase, we used training data sets to calculate the cluster parameters. We collected training data sets from different users for this purpose. Therefore, we get multiple clusters where each cluster indicates a specific user. Similarly, test data sets are collected in order to evaluate the effectiveness of the training. The performance measure is determined by calculating False Acceptance Rate (FAR) and False Rejection Rate (FRR). The FAR indicates at what percentage the system inaccurately accepts an invalid user as a valid one. Similarly, FRR is the measure of the likelihood that the system will incorrectly reject a valid user. These two measures are the probability indicating how efficiently the system will perform.

4. 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.

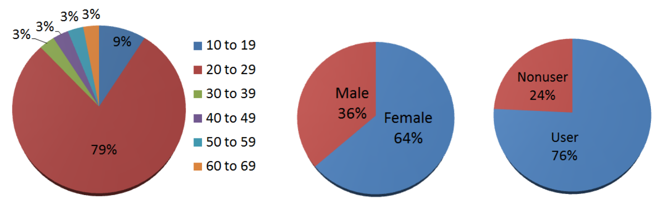
4.1 Experimental Platform

For experiment purpose, we use Samsung Galaxy Young gt-s5360A [20] device to collect data. The device possesses a processing capability of 832 MHz ARMv6 along with a Broadcom VideoCore IV and memory of 384MB RAM. We developed an Android application to get user's touch-based usage data using the device.

4.2 Application for Data Collection

We developed our application in Android 2.3 (Gingerbread) for data collection. Our application monitors and records usage patterns of different users from the perspective of all 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. Fig. 3 shows our application's screenshot and step-wise tasks. Besides, we present a user in action while using the application in Fig. 4.

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



(a) Age range (years) (b) Gender (c) Touchscreen usage

Figure 5: Demography of participants

4.3 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 genders, and different levels of experience in interacting with touchscreen electronic devices. We show the demography of participants in Fig. 5.

Fig. 5a shows that our participants mostly exhibit youth having an age range of 20 to 30 years. We have picked such skewed diversity in terms of age range, as survey studies existing in the literature [21], [22] exhibit a similar skewed diversity in favor of youth for usage of touchscreen based 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. Among them, we have 10 sets of data for training and 5 sets of data for testing. We perform several iterations of the same process adopting the same number of data sets in a random manner.

	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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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
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

Table 1: Findings of unsupervised clustering (each row represents a cluster and each column represents a user)

5. 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.

5.1 Outcomes 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.

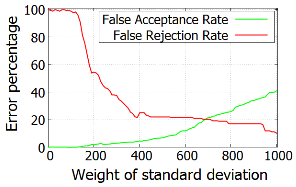
Now, the above findings suggest that there might be a near one-to-one mapping between the users and the clusters of data. Therefore, it might be possible to efficiently cluster the data such that the users could be identified through the clustering. Next, we present outcomes of one such efficient clustering, which we have already elaborated in Section 3.

5.2 User identification using Mean-SD clustering

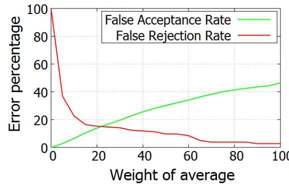
To perform an in-depth investigation of out proposed Mean-SD clustering over all of our collected data, we cluster the collected data from the perspective of both individual features and multiple features. We perform a number of iterations with such clustering tasks using randomly chosen sets of data 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.

5.2.1 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



(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

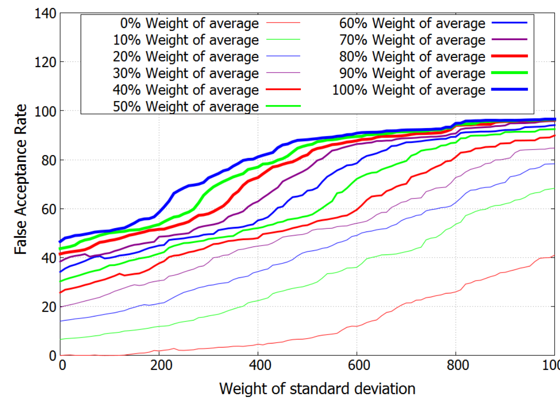
average fixed to 0 is shown in Fig. 6a. Similarly, the effect of changing only the weight of average keeping the weight of standard deviation fixed to 0 is shown in Fig. 6b. Both Fig. 6a and Fig. 6b are pertinent for the feature of rotation about the Z-axis while pressing the button. Note that, here we exploit Eq. 1 already presented in Section 3.

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.

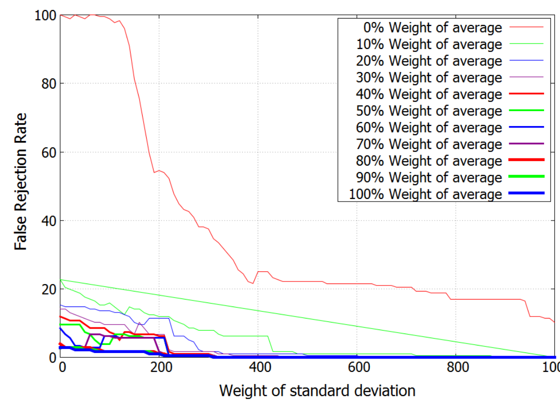
Now, after analyzing the impacts of different possible values of the weights, we can find that the best possible accuracy of the result obtained in the case of the feature of rotation about the Z-axis while pressing the button with the cluster size of ($AVG \times 5\% + SD \times 370\%$). Fig. 7c presents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

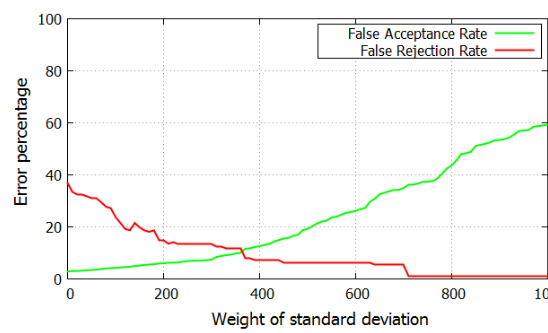
Note that, even though FAR and FRR vary for different features, all the values of FAR and FRR are highly significant. FAR and FRR for only one feature exhibit significant values rendering it not a feasible solution. Now, after reaching this extent, we apply the clustering technique to a different combination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



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

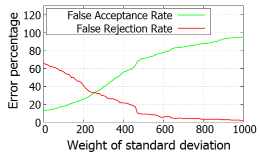


(b) Effect of changing the weight of average and the weight standard deviation on False Rejection Rate

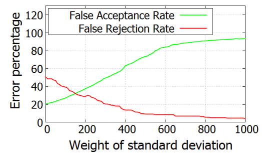


(c) False Acceptance Rate and False Rejection Rate for the feature providing the best-possible outcome

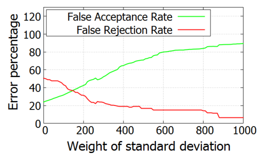
Figure 7: Effect of changing the weight of average and the weight of standard deviation on False Acceptance Rate and False Rejection Rate for the feature providing the best-possible outcomes, which is rotation about the Z-axis while pressing the button in our developed application



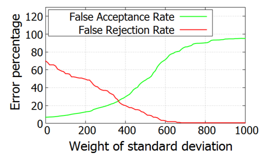
(a) X coordinate of the starting point used in swiping



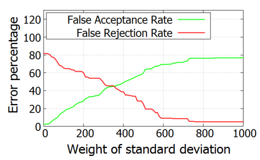
(b) Y coordinate of the starting point used in swiping



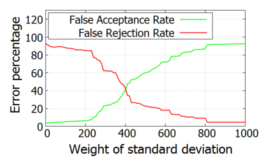
(c) X coordinate of the ending point used in swiping



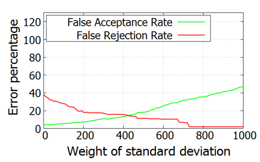
(d) Y coordinate of the ending point used in swiping



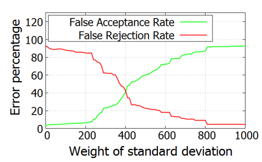
(e) Pressure on screen of the starting point used in swiping



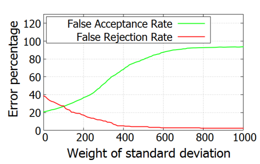
(f) Pressure on screen of the ending point used in swiping



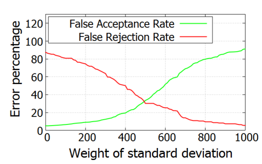
(g) Tilt angle while pressing the button



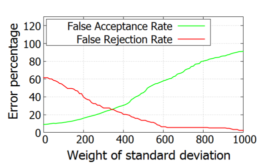
(h) Rotation about the x-axis while pressing the button



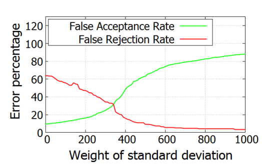
(i) Rotation about the y-axis while pressing the button



(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

user identification based on multiple features.

5.2.2 *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.

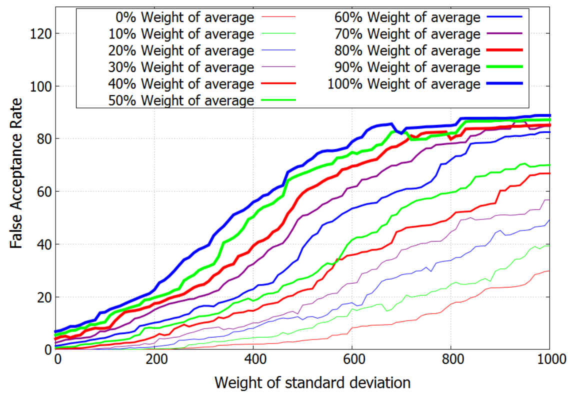
The best result obtained while considering all features in combination is for the cluster size of $(AVG \times 5\% + SD \times 530\%)$. Fig. 9c presents variations FAR and FRR pertinent for this cluster size. Here, the best possible outcome provides only 5% FAR and 6% FRR.

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.

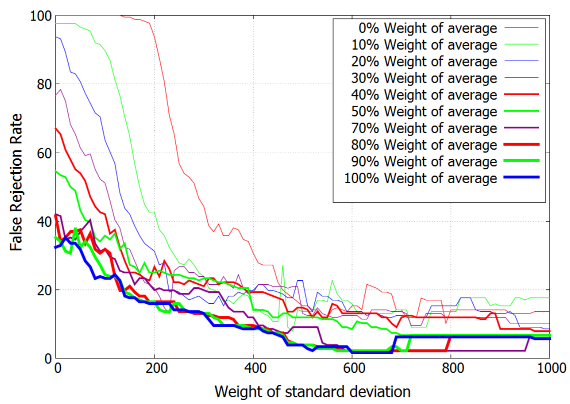
Even though we get two opposing trends in FAR and FRR while considering the features in isolation and while considering all the features in combination, we can achieve the best possible outcome in the case of considering all features in combination. Fig. 7c and Fig. 9c validates the phenomena of achieving the better outcome through considering all the features in combination.

5.2.3 *Allowing Mismatch over the Attributes*

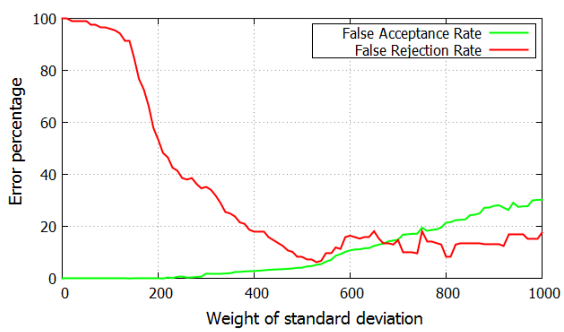
Another important observation is related to allowing mismatch. We observed the result allowing mismatch on the best possible result for considering all features. Accuracy decreases while allowing mismatch of features. With the increase in allowed mismatch, the FAR increases instantaneously from zero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



(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

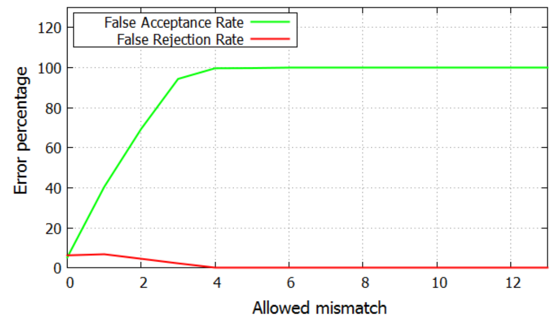


Figure 10: Effect of allowing mismatch while considering all features

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

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 [23], decision tree [24], and a multiclass perceptron algorithm: Kessler's construction [25] 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

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

result with 5% FAR, 6% FRR and the lowest execution time and memory usage.

7. 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.

8. USER EVALUATION

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 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.

9. 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. Here, we consider the usage of a user as a kind of signature identification of that particular user. In our study presented in this paper, we have explored such identification based on single feature. In-depth analysis of multiple features in combination is yet to be done. This could add some more

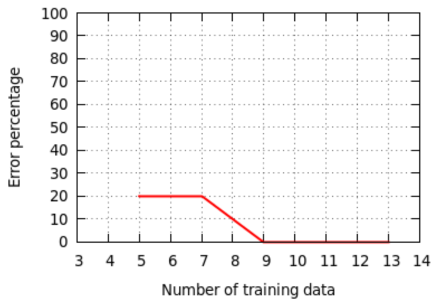


Figure 11: Performance measure for varying number of training data

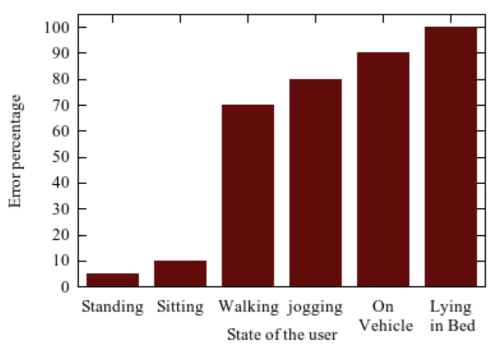


Figure 12: Performance measure for varying state of user

directions to improve performance of the user identification task, which we left as our future work. Besides, we intend to work at the kernel level of a device to enhance performance of identifying a user through following our proposed approach. This could unlock a vast area of observing a user's usages and make the identification system more robust. Additionally, in our system, we used Mean-SD clustering where the 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.

10. 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.

References

[1] Shane Richmond. Smartphones hardly used for calls. *The Telegraph*, 29, 2012.

[2] Ildar Muslukhov, Yazan Boshmaf, Cynthia Kuo, Jonathan Lester, and Konstantin Beznosov. Understanding users' requirements for data protection in smartphones. In *Data Engineering Workshops (ICDEW), 2012 IEEE 28th International Conference on*, pages 228–235. IEEE, 2012.

[3] DM Hutton. Biometrics: Identity verification in a networked world. *Kybernetes*, 2013.

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

[5] Tao Feng, Ziyi Liu, Kyeong-An Kwon, Weidong Shi, Bogdan Carbunar, Yifei Jiang, and Nhung Nguyen. Continuous mobile authentication using touchscreen gestures. In *Homeland Security (HST), 2012 IEEE Conference on Technologies for*, pages 451–456. IEEE, 2012.

[6] Chao Shen, Zhongmin Cai, and Xiaohong Guan. Continuous authentication for mouse dynamics: A pattern-growth approach. In *IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2012)*, pages 1–12. IEEE, 2012.

[7] Mario Frank, Ralf Biedert, Eugene Ma, Ivan Martinovic, and Dawn Song. Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication. *IEEE transactions on information forensics and security*, 8(1):136–148, 2013.

[8] Shamir Ahmed, AS M Rizvi, Rifat Sabbir Mansur, Md Rafatul Amin, and ABM Alim Al Islam. User identification through usage analysis of electronic devices. In *Networking Systems and Security (NSysS), 2015 International Conference on*, pages 1–6. IEEE, 2015.

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

[10] Attaullah Buriro, Bruno Crispo, Filippo Del Frari, and Konrad Wrona. Touchstroke: smartphone user authentication based on touch-typing biometrics. In *International Conference on Image Analysis and Processing*, pages 27–34. Springer, 2015.

[11] Xinwen Zhang, Jean-Pierre Seifert, and Onur Aciicmez. Design and implementation of efficient integrity protection for open mobile platforms. *IEEE Transactions on Mobile Computing*, 13(1):188–201, 2014.

[12] Florian Schaub, Ruben Deyhle, and Michael Weber. Password entry usability and shoulder surfing susceptibility on different smartphone platforms. In *Proceedings of the 11th international conference on mobile and ubiquitous multimedia*, page 13. ACM, 2012.

[13] Sébastien Marcel, Chris McCool, Cosmin Atanasoaei, Flavio Tarsetti, Jan Pesan, Pavel Matejka, Jan Cernocky, Mika Helistekangas, and Markus Turtinen. Mobio: mobile biometric face and speaker authentication. Technical report, Idiap, 2010.

[14] Georgios Kambourakis, Dimitrios Damopoulos, Dimitrios Papamartzivanos, and Emmanouil Pavlidakis. Introducing touchstroke: keystroke-based authentication system for smartphones. *Security and Communication Networks*, 2014.

[15] Alexander De Luca, Alina Hang, Frederik Brudy, Christian Lindner, and Heinrich Hussmann. Touch me once and i know it's you!: implicit authentication based on touch screen patterns. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 987–996. ACM, 2012.

[16] Nalini K. Ratha, Jonathan H. Connell, and Ruud M. Bolle. Enhancing security and privacy in biometrics-based authentication systems. *IBM systems Journal*, 40 (3):614–634, 2001.

[17] Napa Sae-Bae, Kowsar Ahmed, Katherine Isbister, and Nasir Memon. Biometric-rich gestures: a novel approach to authentication on multi-touch devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 977–986. ACM, 2012.

[18] Tao Feng, Jun Yang, Zhixian Yan, Emmanuel Munguia Tapia, and Weidong Shi. Tips: Context-aware implicit user identification using touch screen in uncontrolled environments. In *Proceedings of the 15^{th} Workshop on Mobile Computing Systems and Applications*, page 9. ACM, 2014.

[19] Cheng Bo, Lan Zhang, Xiang-Yang Li, Qiuyuan Huang, and Yu Wang. Silentsense: silent user identification via touch and movement behavioral biometrics. In *Proceedings of the 19^{th} annual international conference on Mobile computing & networking*, pages 187–190. ACM, 2013.

[20] Rizki Abriansyah. Samsung galaxy y gt-s5360 review. 2012.

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

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

[23] Leif E Peterson. K-nearest neighbor. *Scholarpedia*, 4 (2):1883, 2009.

[24] S Rasoul Safavian and David Landgrebe. A survey of decision tree classifier methodology. 1990.

[25] David B Sher and Davin Milun. Generating edge detectors from a training ensemble. In *Optical Engineering and Photonics in Aerospace Sensing*, pages 165–176. International Society for Optics and Photonics, 1993.