# ITSME: Multi-modal and Unobtrusive Behavioural User Authentication for Smartphones

Attaullah Buriro[1], Bruno Crispo[2,1], Filippo Del Frari[1], Jeffrey Klardie[3], and Konrad Wrona[4]

[1] Department of Information Engineering and Computer Science, University of Trento, Italy,
attaullah.buriro@unitn.it, bruno.crispo@unitn.it, filippo.delFrari@unitn.it
[2] DistrNet, KULeuven, Belgium,
bruno.crispo@cs.kuleuven.be
[3] Vrije Universiteit Amsterdam, The Netherlands
[4] NATO Communications and Information Agency, The Hague, Netherlands,
konrad.wrona@ncia.nato.int

**Abstract.** In this paper, we propose a new multi-modal behavioural biometric that uses features collected while the user slide-unlocks the smartphone to answer a call. In particular, we use the slide swipe, the arm movement in bringing the phone close to the ear and voice recognition to implement our behaviour biometric. We implemented the method on a real phone and we present a controlled user study among 26 participants in multiple scenario's to evaluate our prototype. We show that for each tested modality the Bayesian network classifier outperforms other classifiers (Random Forest algorithm and Sequential Minimal Optimization). The multimodal system using slide and pickup features improved the unimodal result by a factor two, with a FAR of 11.01% and a FRR of 4.12%. The final HTER was 7.57%.

**Keywords:** Smartphone, Behavioral Biometrics, Sensors, Transparent Authentication

## 1 Introduction

The last decade mobile handheld devices have gone through a major evolution; smaller yet more powerful processors, better batteries and improved hardware such as gps, wifi and connectivity chips are all developments that enabled this progression. Most relevant examples are the Android platform launched by Google in 2008 [1] and the iPhone smartphone by Apple in 2007 [2]. Since their introduction these devices have overtaken most competitors and together captured a very dominant market share: in 2014 Android and iOS combined account for 96.3% of the smartphone operating system market [3].

Continuing hardware improvements combined with extensive user research have resulted in highly capable smartphones that provide users with rich communication capabilities. While this improves users' lives on one hand, it brings very serious security and privacy threats to the user on the other. A typical modern smartphone allows the user to do mobile banking, have full control over her email, continuously keep track of her location and many ways to indulge in social communications through popular apps such as Facebook, Whatsapp, Instagram and Twitter.

All these apps store privacy sensitive data about the user, which often become easily accessible once access to the phone is obtained. Unwanted access gained after a phone is lost, or even temporal access when not paying attention for a short period of time could have serious consequences.

Authentication techniques have traditionally been based on something a user knows (password, PIN), something she owns (keys, badges) or a combination of those two (ATM card + PIN). Certain properties make these insecure; passwords and pins are easily forgotten, but also easily guessed [4]. Keys and badges can be lost, or duplicated. Besides, requiring smartphone's users to carry an extra device for the sole purpose of authentication is not realistic. Recent updates in both Android and iOS include such biometric authentication; face-unlock on Android [5], and fingerprint unlock on iOS [6].

Biometric authentication is the process of verifying ones identity based on biometric features. The study and development of biometric authentication solutions have come a long way since it's first mention by Bertillon in 1870s [25]. Most popular features are physiological and behavioural features. Physiological characteristics are based on features of the body, e.g. fingerprints, hand geometry, iris or retina scans. Behavioural characteristics are based on behaviour, e.g. keystrokes, gait, signature placement and voice. Other biometrics use chemical features (based on events that happen in a persons body, measured by e.g. odour or temperature) and cognitive features (based on brain responses to specific stimuli, e.g. odour or sound).

Initial biometrics used information from a single source. These so-called unimodal systems had to deal with a range of problems like noisy data, spoof attacks and unacceptable high error-rates. Some of these issues can be addressed by combining multiple sources of information [7]. Due to the presence of multiple (mostly) independent features, the performance is expected to increase [8].

Using biometrics authentication for smartphone users faces two important challenges. First, users may use the phone in different situations and context (i.e. while walking, sit on a chair, standing up, in the dark, etc.). Thus any realistic solution should accomodate the possibility that data acquisition may fail or that a particular feature might be temporarily unavailable. Second, the solution must require as small effort as possible to users. Studies suggest that usability issues are a major driver of users' adoption decisions [9]. A recent study [33] reports that 70% users do not use any PIN/passwords to protect mobile phones because these are more annoying to users compared to other telephony related problems such as lack of coverage or low voice quality.

To partially address these challenges this paper presents a novel multimodal biometric system for smartphone users authentication. The system uses slide-unlock features, pickup movements and voice features while placing or answering a call. Being multi-modal the solution aim at robustness, such that users can still be authenticated even if some of the modalities fail.

To address the problem of usability, our authentication scheme requires zero effort to users. To the best of our knowledge this is the first authentication solution for smartphone that is completely unobtrusive. Users are not required to perform any action for the sole purpose of authentication. In fact, entering a password or PIN is more noticeable. Last but not least, our system can be implemented on most of the smartphones available on the market today.

The rest of this paper is organized as follows: Section 2 discusses related work, Section 3 describes the background knowledge. Section 4 presents the solution and the validation methology. Section 5 describes how we configured parameters in the models we used. Section 6 and Section 7 present and discuss the results of our approach. The paper is concluded in Section 8.

## 2 Related Work

This section reports related work that specifically take mobile devices into consideration. A wider survey of biometric authentication in general can be found in [10] and [11].

### 2.1 Unimodal Systems

In [12], Frank et al. consider touch operations for continuous authentication where a single type of operations are used (strokes or slides). An Equal Error Rate (EER) of 13% has been reported for one single stroke, and 2% to 3% for 11 subsequent strokes. In [13], a user is authenticated not only on the password pattern they input, but also the way they perform that input. A lab study and a long-term study provide evidence that it is possible to distinguish users and to improve the security of password patterns and even simple screen unlocks. The accuracy rate of the simple unlock is 57% at best (two-finger vertical unlock), while the accuracy of the password patterns is around 77%. In [14], Angulo et al. explored the same approach for improving password-patterns with biometrics. Using a Random Forest classifier an EER of approximately 10.4% is achieved.
Sae-Bae et al. [15] present a multi-touch gesture-based authentication technique. A classifier that uses pattern recognition techniques classifies movements characteristics of the center of the palm and fingertips. An average EER of 10% with single gestures was achieved, with improvements up to 5% EER when combining multiple gestures in a sequence.
In [16] Derawi et al. authenticate users based on gait recognition using accelerometers available in any modern mobile device. Using a low end phone (the Google G1 phone containing the AK8976A embedded accelerometer sensor) an EER of 20% is reached.

Tao et al. [17] implement a fast face detection and registration method based on a Viola-Jones detector [18]. A face-authentication method based on subspace metrics is developed. Experiments using a standard mobile camera showed that the method is effective with an EER of 1.2%.

### 2.2 Multimodal Systems

In [19], Saevanee et al. used SMS texting activities and messages in a multimodal authentication system. Keystroke dynamics and linguistic profiling was used to discriminate users with error rates of 20%, 20% and 22%, respectively. A fusion of these three led to an overall EER of 8%.
Buriro et al. [20] presented a sensor-enhanced touchstroke based smartphone authentication. Their study makes use of two human behaviors, i.e., how a person holds her phone and how she types her 4-digit *free text* PIN. Using Bayesian classifier and Random Rorest classifier, they achieved 1% EER.
Aronowitz et al. [21] introduced a new biometric modality called chirography which is based on user's writing on multi-touch screens using their fingers. By fusing this with face and voice features, an EER of 0.1% is reached in an office environment, and 0.5% in noisy environments.
In [22] Ferrer et al. introduced a multimodal biometric identification system that is based on the combination of geometrical, palm and fingerprint features of the users' hand.
In [23] a multimodal authentication approach is presented by Kim et al., using teeth and voice data acquired using mobile devices. The individual matching scores obtained from these biometric traits are combined using a weighted-summation operation. An EER of 2.13% was reported.
In [24], McCool et al. introduced a fully automatic bi-modal face and speaker system. A Nokia N900 was used during tests and EER results of 13.3% and 11.9% for female and male trials respectively have been reported for the fused score. This is a 25% performance improvement for the female trials, and 35% improvement for male trials.

## 3 Background

In this section we explain the technology and building blocks we used to build our solution.

### 3.1 Considered Sensors

We considered three built-in smartphone sensors, namely, accelerometer, orientation and gyroscope. The way in which each of these sensors work is explained below:

The *acceleration* ($acc_n$) is the acceleration applied to the device, including the force of gravity, measured on three axis' x, y and z. Android's sensor API uses a standard three-axis coordinate system. This system is defined relative to the

device's screen when it is held upright as shown in Figure 1 (a). The acceleration that is applied to a device $A_d$ is calculated using the forces (including gravity $g$) that are applied to the sensor $F_s$ itself using the following equation:

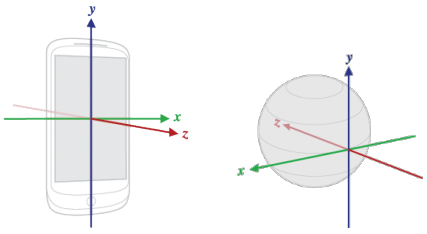
$$A_d = -g \sum \frac{F_s}{mass} \tag{1}$$

The *gyroscope* ($gyro_n$) measures the rate of rotation in radians per second ($rad/s$) around all axis'. The same coordinate system as described above is used.

The *orientation* ($rot_n$) is the rotation around the x- (pitch), y- (roll) and z-axis (azimuth) in radians ($rad$). Note that the orientation uses a different coordinate system than the accelerometer and the gyroscope[5]

- X is defined as the vector product $Y \cdot Z$ (it's tangential to the ground at the device's current location and roughly points West).
- Y is tangential to the ground at the device's current location and points towards the magnetic North Pole.
- Z points towards the center of the Earth and is perpendicular to the ground.

See Figure 1 for a graphical representation.




(a) (b)

Fig. 1: (a) Coordinate system relative to the device. [Source: Android SensorEvent] (b) Coordinate system used in Android's orientation sensor. [Source: Android SensorManager]

### 3.2 Considered Classifiers

Classification is a way of comparing an unknown query input sample with the stored templates. The Classifier/Matcher is the main component of any biometric system. The goal of the classifiers is to classify a variable $y = x_0$ called the class variable, given a set of attribute variables $\mathrm{X} = \{x_1 \ldots x_n\}$. The classifier

[5] http://developer.android.com/reference/android/hardware/SensorManager.html

$c : \mathrm{x} \rightarrow y$ is a function that maps a data instance x to a class value of $y$. The classifier itself it learned from a dataset D, consisting of samples over $(\mathrm{x}, y)$. Relating this to our scenario, the attribute variables x are the features that we extract from the touch events, motion sensors and microphone; they are the slide, pickup and voice samples. The class variable $y$ is either *target* (meaning the instance belongs to the class learned during training) or *unknown* (meaning the instance does not appear to belong to the previously learned class).

We performed verification with three different classifiers, i.e., One-class BayesNET (BN) classifier, One-class Random Forest (RF) and One-class Sequential Minimal Optimization (SMO)-a Weka version of support vector machine (SVM). We chose these classifiers because they were shown to be very effecient in previous behavioral-based work [20,26]

We imported Weka library in our project and implemented our prototypes on smartphone.

During the training phase we only have training data available for a single instance class; the genuine user (the *target* class). At prediction time new instances with unknown class labels will have to be classified as either the *target* class or *unknown*. To handle this type of learning problem, typically called one-class classification, we wrap each classifier in a one-class classifier[6].

### 3.3 Performance Metric

Based on the binary outcome of this function (accept or reject), two types of errors can occur; false rejections and false acceptances. A false rejection occurs when a legitimate user is rejected access from the system and a false accept occurs when a imposter is granted access to the system. The errors are measured in the so-called False Rejection Rate (FRR) and False Acceptance Rate (FAR). These rates are calculated as follows:

$$FAR(\Delta) = \frac{FA(\Delta)}{nI} \tag{2}$$

$$FRR(\Delta) = \frac{FR(\Delta)}{nG} \tag{3}$$

Given a specific threshold $\Delta$, the FAR is defined as the number of false acceptances ($FA$) divided by the number of imposters $nI$ and the FRR is defined as the number of false rejections ($FR$) divided by the number of genuine users $nG$.

To evaluate the interaction of these error rates the Weighted Error Rate (WER) is used. The WER shows the combined error rate of both FAR and FRR with a weight $\alpha$ assigned to each. If the false accepts are considered worse than false rejects (focus on security), a weight $> 0.5$ should be used. If false rejects

[6] http://weka.sourceforge.net/packageMetaData/oneClassClassifier/

are worse than false accepts (focus on usability), than a weight $< 0.5$ is more appropriate. A special error rate is the EER where both errors have the same weight (i.e. $\alpha = 0.5$). The WER is defined [27] as follows:

$$WER(\alpha, \Delta) = \alpha FAR(\Delta) + (1 - \alpha) FRR(\Delta) \tag{4}$$

Given a specific weight $\alpha$, the goal is to find the optimal threshold $\Delta_\alpha^*$ for which the WER is as low as possible. This function can be defined as:

$$\Delta_\alpha^* = \underset{\Delta}{\operatorname{argmin}} \, |\alpha FAR(\Delta) + (1 - \alpha) FRR(\Delta)| \tag{5}$$

In our opinion the usability of an authentication system is of paramount importance for its adoptability. Therefore, in our system, we consider a false reject worse than a false accept, and we'll use $\alpha = 0.4$ in our evaluations.

As proposed by Poh et al. in [27], the final evaluation looks at the performance of the system after deciding on the weight $\alpha$ and the optimal threshold $\Delta_\alpha^*$. This is measured by the so called Half Total Error Rate (HTER), which is calculated as follows:

$$HTER(\Delta_\alpha^*) = \frac{FAR(\Delta_\alpha^*) + FRR(\Delta_\alpha^*)}{2} \tag{6}$$

The lower the HTER, the better the system performs given the chosen weight $\alpha$.

## 4 Our Solution

In [28] Conti et al. introduce a new biometric measure to authenticate smartphone users; the movement a user performs when answering (or placing) a phone call. Several experiments with a prototype in a controlled environment have shown that the method is effective and that the performance is comparable to that of other transparent authentication methods, like face or voice recognition. These experiments also highlighted an issue with the data acquisition process, due to the variability in determining the end of the arm movement. To address this issue without compromising the unobtrusive nature of the initial idea we extended the solution as follows.

When placing or answering a phone call, three common steps have to be taken: 1) the user must unlock her phone, 2) bring it to her ear and 3) speak into the microphone. Our multimodal authentication solution uses features from all three steps to determine whether or not the current user is genuine, or if she is an imposter.

The complete system consists of four parts: slide movement recognition, pickup movement recognition, voice recognition and fusion. The data features are described in this section, while the next section describes the actual classification framework including fusion.

### 4.1 Setup

We conducted a controlled user study to test our mechanism in terms of performance and robustness. We recruited 26 participants of which 16 were male, and 3 operated their phone using their left hand. All of them were familiar with the slide-to-unlock pattern. Ages of our volunteers were ranging from 14 to 55. 2 participants were 14-19, 12 were 20-29, 7 were 30-39, 1 was 40-49 and 4 were 50 or older.

We created an Android application that targets SDK version 4.4 (*Kitkat*) and minimally requires version 4.0.3 (*Ice Cream Sandwich*). We implemented both the training phase and the classification phase using Weka 3.7 on android smartphone. The training module allows the user to anonymously record slide movements, pickup movements and voice samples which are sent to a central server. The classification module was implemented as a proof of concept and to analyze the performance on mobile phones.

A central server running on the Amazon cloud platform collected the training features in a database. A local running Java application (using Java 1.7) using the same classification module as implemented in Android was then used to test the robustness of the system. We used a Google Nexus 4 device by LG running Android 5.1 during the study. This device has a 4.7 inch screen, a Qualcomm APQ8064 Snapdragon 1.5GHz Quad-core processor, 2GB RAM, an accelerometer, gyroscope and proximity meter.

In each session, we first explained the purpose of the study to the participant and asked them if we could use their data anonymously, and noted their age and gender. After that we moved to the actual trials. Each user was required to collect at least 20 slide samples, 20 pickup samples and 10 voice samples. Samples that where distorted in any way could be removed by the user.

For the slide and pickup movements we instructed the participant to first do five movements while sitting or standing still and after that five while walking around. Then the user was asked to open a news app and read the fifteenth headline, which required the user to count while scrolling to the headlines. This usually confused users, and many had to recount from the top because they tried to wrap their head around the purpose of this task, and lost count. The goal of this distraction task was to minimize the learning effect that can occur when doing the same movement many times in quick succession. After the user read the article, she was again requested to record five movements while sitting, and five movements while walking.

### 4.2 Data Collection

We use the default Android slide lock as depicted in Figure 2. The center knob can be dragged towards any direction. When the user drags the knob and then release it at least as far as the circular boundary (slightly visible in the right image in Figure 2), the phone will be unlocked. If the knob is released before reaching the boundary, the phone stays locked.
During the training phase a pickup event starts when the user clicks the start

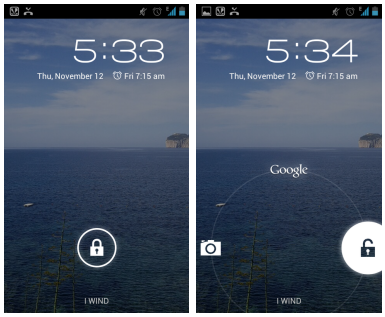



Fig. 2: Android slide lock. On the left the default state, on the right the state when a user drags the knob towards the circular boundary.

button, and ends automatically when the phone is at the user's ear (detected by the proximity detector). When used in combination with the other two modalities (e.g. during authentication), the sample starts when the slide unlock ends, and also finishes when the phone reaches the user's ear.
The Android system continuously delivers SensorEvents [7] to an event listener. As we use three sensor (accelerometer, gyroscope, orientation sensor), a delivered event can be produced by one of the sensors. Every time we receive a new event for any of the sensors, we extract the x, y and z values, and store them.

For the voice sample recording we requested the user to simply speak into the microphone as if they were answering a phone call, but to make sure to use a relatively lengthy sentence to fill the 2.5 seconds of recording. Most users used a sentence similar to *Hello, this is John Doe. Who am I speaking to?*. An audio sample is recorded for 2500 ms at a sample rate of 8 kHz using 16 bits per sample with one channel. The resulting pulse-code modulation (PCM) data is stored in a temporary WAV file on the device.

### 4.3 Feature Extraction

**Slide** A slide sample starts when the user touches the knob for the first time, and ends when the knob is released (e.g. the user stops the touch event). One slide is a path encoded as a sequence of vectors $(t_n, x_n, y_n, p_n, s_n)$. Only complete samples (samples that would unlock the phone in the original non-biometric implementation) are considered, others are simply discarded.

During the slide event the features in Table 1 are recorded at a average sampling rate of 150 Hz. From the given MotionEvent we extract multiple features. The *time offset* $(t_n)$ indicates the offset since the start of the touch event in milliseconds.

The *x- and y-position* $(x_n, y_n)$ are measured in pixels and indicate the exact position of the knob (controlled by the users touch) on the screen. Over time

[7] http://developer.android.com/reference/android/hardware/SensorEvent.html

Table 1: Slide features

| Feature | Unit |
|---|---|
| Time offset | $ms$ |
| X-position | $px$ |
| Y-position | $px$ |
| Pressure | Normalized value between 0 and 1 |
| Size | Normalized value between 0 and 1 |

these coordinates create a path from the initial position of the knob towards the boundary of the circle, indicating exactly how the user moved the knob.

The *touch pressure* ($p_n$) of the touch event indicates the approximate pressure applied to the surface of the screen. The value is normalized to a range from 0 (no pressure at all) to 1 (normal pressure), but values higher than 1 may be generated depending on the calibration of the input device.

The *size* ($s_n$) is a scaled value of the approximate size of the area of the screen being touched. The actual value in pixels corresponding to the touch is normalized with the device specific range of values and scaled to a value between 0 and 1.

**Pickup** During the pickup event the features in Table 2 are extracted at a average sampling rate of 190 Hz. The *time offset* ($t_n$) indicates the offset since the start of the pickup event in milliseconds. One pickup movement is encoded as a sequence of vectors $(acc_n^x, acc_n^y, acc_n^z, gyro_n^x, gyro_n^y, gyro_n^z, rot_n^x, rot_n^y, rot_n^z, t_n)$.

Table 2: Pickup features

| Features | | | | Units |
|---|---|---|---|---|
| 1-3 | X-acceleration | Y-acceleration | Z-acceleration | $^m/_{s^2}$ |
| 4-6 | X-gyroscope | Y-gyroscope | Z-gyroscope | $^{rad}/_s$ |
| 7-9 | X-orientation | Y-Orientation | Z-Orientation | $rad$ |
| 10 | Time offset | | | $ms$ |

**Voice** Using the recorded voice sample, we calculate the Mel-frequency cepstral coefficients (MFCCs) [29] and store them in a feature vector. MFCCs have been very popular in the realm of speech recognition due to its ability to represent the speech amplitude spectrum in a compact form [30]. Creating MFCCs is done by 1) converting the waveform to frames, 2) take the discrete Fourier transform, 3) take the Log of the amplitude spectrum, 4) Mel-scaling and smoothing and 5)

applying discrete cosine transform. The MFCC features are then used as data instances that we use to create models for our classifiers

### 4.4 Data Fusion

In our multimodal mechanism we use multiple biometric traits (slide movement, pickup movement and voice) which need to be fused to output one single decision: accept or reject. We fused these modalities at match-score level. However, because each modality performed differently, we give each modality a weight, based on it's unimodal performance.

Consider three modalities $x$, $y$ and $z$, having an error rate ($er$) of 0.1, 0.2 and 0.3 respectively. Obviously, modality $x$ is much better than $y$ and $z$, and should therefore have a higher weight. For each classifier $c$ we can calculate a success index. The success index indicates how much the classifier contributes to the sum $1 - er(c)$ for each classifier $c$.

$$index(c) = 1 - \frac{er(c)}{\sum_{i=1}^{n} er(i)} \tag{7}$$

The eventual weight can then be calculated using:

$$weight(c) = \frac{index(c)}{\sum_{i=1}^{n} index(i)} \tag{8}$$

Filling in the values for three modalities $x$, $y$ and $z$, they would get weights of 0.42, 0.33 and 0.25 respectively. Better modalities get get higher weights.

### 4.5 Decision Making

To measure the performance of the classifiers we use the cross-validation method. The dataset is randomized and then split into $k$ folds of equal size. In each iteration, one fold is used for testing, and the other $k - 1$ folds are used for training the classifier. We use $k = n$, meaning we apply leave-one-out cross-validation. The test results are averaged over all folds, which give the cross-validation estimate of the accuracy. This method is useful because we are dealing with small datasets and the idea is to test each sample. Using cross-validation we utilize the greatest amount of training data from the dataset.

When evaluating the performance of a biometric system, multiple criteria should be considered [27]. Biometric authentication systems make decisions based on the following decision function:

$$f(x) = \begin{cases} accept, & \text{if } c(I, \mathrm{x}) \geq \Delta \\ reject, & \text{otherwise} \end{cases} \tag{9}$$

where $c(I, f)$ is the output of the underlying classifier $c$ that indicates how certain it is that the claimed identity $I$ is correct based on the given dataset (features) x. The threshold $\Delta$ defines when an identity claim is accepted or rejected. Access to the system is accepted if the score is greater than or equal to the threshold, and rejected otherwise.

## 5 Parameters

Before we can show any results, we first need to identify the exact data and models under test. During the research we did many extensive tests to find the optimal setup. These tests led us to the best performing combination of parameters. The actual performance of the best classifier will be discussed and evaluated in the next section.

The tests have been carried out on a random subset (length: 10) of the participants in the user test. For each configuration considered we calculated the equal error rate (EER) based on all samples of the genuine user, and 10 random samples per other (non-genuine) user.

### 5.1 Parameters

For each modality we use all attributes, and do a grid search to find the best performing set of parameters. We also record the average model generation time so we can filter out configurations that would take too long on mobile phones.

Table 3: Best classifier per modality.

| | Slide | | Pickup | | Voice | |
|---|---|---|---|---|---|---|
| Classifier | Comp. Time | EER | Comp. Time | EER | Comp. Time | EER |
| BayesNET | 64 | 0.1242 | 762 | 0.2045 | 205 | 0.2681 |
| Random Forest | 4453 | 0.1434 | 13988 | 0.2083 | 4402 | 0.2452 |
| SMO | 8433 | 0.1864 | ~144000 | - | 548 | 0.2709 |

Table 3 gives for each modality an overview of the best performing classifiers. The parameter tests show that the Bayesian network classifiers yield the best results overall. Only with the voice modality the Random Forest classifier yields slightly better results. However, the Bayesian network is much faster.

From this point on when talking about the classifier, we mean the Bayesian classifier, with its parameters configured as shown in Table 4.

## 6 Results

The results we present here are based on the user data we collected during the controlled users tests, fed into the classifiers with their parameters configured as

Table 4: Parameter configuration per modality

| Modality | Naive Bayes | Markov blanket | Max parents | Score type | Alpha |
|---|---|---|---|---|---|
| Slide | T | T | 5 | Entropy | 0.25 |
| Pickup | T | F | 3 | Bayes | 0 |
| Voice | F | F | 5 | Entropy | 0 |

described in Section 5. For each user this gives us a certainty number (higher means more similar to the reference model) for both genuine and impostor samples.
It is important to note that when testing a classifier for user $u$, we use all samples from all other users to do our impostor tests. By doing so, we have much more impostor samples than genuine samples, leaving the FRR much more sensitive to deviations than the FAR.

Given the data from the user we can find the optimal threshold $\Delta_\alpha^*$. The optimal threshold is the threshold for which the Weighted Error Rate (WER) is at its minimum (see Equation 5).

### 6.1 Unimodal Systems

**Slide** Given $\alpha = 0.4$, we found that the optimal threshold $\Delta_\alpha^* = 49$. Re-running the tests with this threshold gives us a FAR of 22.28% and a FRR of 4.84%.

The HTER (defined in Equation 6) can now easily be computed:

$$HTER(49) = \frac{22.28\% + 4.84\%}{2} = \textit{13.56\%} \tag{10}$$

**Pickup** The optimal threshold $\Delta_\alpha^* = 42$. Running the tests with this threshold gives us a FAR of 26.69% and a FRR of 6.19%.

$$HTER(42) = \frac{26.69\% + 6.19\%}{2} = \textit{16.44\%} \tag{11}$$

**Voice** The optimal threshold $\Delta_\alpha^* = 85$. Running the tests with this threshold gives us a FAR of 63.92% and a FRR of 12.69%.

$$HTER(85) = \frac{63.92\% + 12.69\%}{2} = \textit{38.30\%} \tag{12}$$

It is evident that slide and pickup modalities are better than voice modality. Still, we are using it here to show how the use of multimodal biometric authentication can improve a unimodal authentication system.

### 6.2 Multimodal Systems

**Slide+Pickup Modalities** As described in Section 4.4 we use a match-score level fusion method, using weights for each classifier output. We calculate the weight using Equation 8. In the previous subsection we have seen that the slide and pickup classifiers have a HTER of 13.56% and 16.44% respectively. Filling in the equation this gives us a weight of 0.55 for slide and 0.45 for pickup.

The optimal threshold $\Delta_\alpha^* = 55$. Re-running the tests with this threshold gives us a FAR of 11.01% and a FRR of 4.12%. Calculating the HTER gives us:

$$HTER(55) = \frac{11.01\% + 4.12\%}{2} = \textit{7.57\%} \tag{13}$$

Comparing the slide and pickup modalities individually with this multimodal system, we can see that the latter performs almost twice as good.

**Slide+Pickup+Voice Modalities** We have seen that the slide, pickup and voice classifiers have HTERs of 13.56%, 16.44% and 38.30% respectively. Using Equation 8 this results in weights 0.40 (slide), 0.38 (pickup) and 0.22 (voice).

The optimal threshold $\Delta_\alpha^* = 62$. Running the tests with this threshold gives us a FAR of 10.72% and a FRR of 3.93%. Calculating the HTER gives us:

$$HTER(62) = \frac{10.28\% + 3.93\%}{2} = \textit{7.33\%} \tag{14}$$

A quick comparison shows that adding voice modality to the multimodal system using slide and pickup features does not improve the results significantly but still better (HTER 7.33% vs HTER 7.57%).

## 7 Discussion

The results show that the slide modality is better than the pickup modality. The main cause for this observation is that the pickup classifier is much more sensitive to the kind of activity the user performs while unlocking her phone. Because the rotation, gyroscope and acceleration of the device are the main features of the modality, the user's activity while unlocking has major influence on the classifier outcome: walking, running, standing in a crowded bus; they all have different impact on the motion sensors of the device.

The slide modality on the other hand does not use motion sensors but rather uses touchscreen. Touchscreen determines finger position, pressure and size on a screen which are much less influenced by external factors, making the modality more robust in a range of different scenario's.

When combining the slide and pickup modalities, we can see that the FAR improves significantly.

The voice modality is obviously not good enough (based on our experiments) and may not be deployed in real world because of higher error rates - FAR of 63.92% and FRR of 12.69%. The reason(s) for worst voice results may be due to the low quality of the open source library and by the fact we applied only basic clustering mechanisms. Still, the fusion of all three modalities yielded better results.

System like ours are suitable for risk-based authentication scenarios (e.g. mobile banking applications), where security may need to be traded for availability dynamically and adaptivelly.

## 8 Conclusion and Future Work

In this paper we proposed a new multimodal biometric system for smartphone user authentication that focusses on usability. The system uses features collected during a slide-unlock movement on a smartphone. We use finger position, pressure, size and time offset to generate a model and classify future slide movements. We shown how fusion of unimodal systems to multimodal ones using slide, pickup and voice modalities can significantly improve performance.

We have applied three different classifiers, i.e., BN, RF and SVM. BN classifier outperformed the other classifiers in terms of error rates and computation time.

From the three unimodal traits we tested (slide, pickup and voice); the slide modality performed best with a FAR of 22.28% and a FRR of 4.84%, resulting in a HTER of 13.56%. The pickup modality performed slightly worse, with an FAR and FRR of 26.69% and 6.19% respectively, and an HTER of 16.44%. However, with their fusion, we were able to achieve much improved performance (by a factor of two). A FAR of 11.01% and a FRR of 4.12% were reached, resulting in a HTER of 7.57%.

The voice based model performed much worse as the open source library we used was simply not good enough. However, we have shown the potential improvement of a multimodal system using slide, pickup and voice modalities.

This research can be extended in multiple directions. To validate the results presented here a larger user study should be conducted. The impact situations, context and environment may have on this type of biometrics need to be investigated further.

## 9 Acknowledgement

Authors would like to thank all the volunteers, who participated in this experiment for their valuable feedback and comments .

This work has been partially supported by the TENACE PRIN Project (n. 20103P34XC) funded by the Italian MIUR and EIT Digital MobileShield project.

## References


1. Dan Morrill, Announcing the Android 1.0 SDK, release 1, Google, 2008, http://android-developers.blogspot.in/2008/09/announcing-android-10-sdk-release-1.html, accessed (20-04-2015)
2. Peter Cohen, Macworld Expo Keynote Live Update, PCWorld, 2007, http://www.macworld.com/article/1054764/liveupdate.html, accessed (20-04-2015)
3. , IDC: Android and iOS Squeeze the Competition, Swelling to 96.3% of the Smartphone Operating System Market for Both 4Q14 and CY14, According to IDC, IDC, http://www.idc.com/getdoc.jsp?containerId=prUS25450615, (2015), accessed (20-04-2015).
4. Wood, Helen M: The use of passwords for controlled access to computer resources, US Department of Commerce, National Bureau of Standards, Vol-500, number-9, (1977)
5. Google: Ice Cream Sandwich, http://developer.android.com/about/versions/android-4.0-highlights.html, (2011), accessed (20-04-2015).
6. Chris Velazco: Apples Touch ID Is A 500ppi Fingerprint Sensor Built Into The iPhone 5S Home Button, http://techcrunch.com/2013/09/10/apples-touch-id-a-500ppi-fingerprint-sensor, TechCrunch (2013), accessed (20-04-2015).
7. Ross, Arun and Jain, Anil: Information fusion in biometrics, Pattern recognition letters, Elsevier, Vol-24, number-13, 2115–2125 (2003)
8. Kuncheva, Ludmila I and Whitaker, Christopher J and Shipp, Catherine A and Duin, Robert PW: Is independence good for combining classifiers?, In proceedings of 15th International Conference on Pattern Recognition, IEEE, Vol-2, 168–171, (2000)
9. Bhagavatula, Chandrasekhar and Ur, Blase and Iacovino, Kevin and Kywe, Su Mon and Cranor, Lorrie Faith and Savvides, Marios: Biometric Authentication on iPhone and Android: Usability, Perceptions, and Influences on Adoption, (2015)
10. Jain, Anil K and Flynn, Patrick and Ross, Arun A: Handbook of biometrics, (2007)
11. Yampolskiy, Roman V and Govindaraju, Venu: Behavioural biometrics: a survey and classification, International Journal of Biometrics, Inderscience, Vol-1, number-1, 81–113, (2008)
12. Frank, Mario and Biedert, Ralf and Ma, Eugene and Martinovic, Ivan and Song, Dawn: Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication, IEEE Transactions on Information Forensics and Security, IEEE, Vol-8, number-1, 136–148, (2013)
13. De Luca, Alexander and Hang, Alina and Brudy, Frederik and Lindner, Christian and Hussmann, Heinrich: 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, ACM, Vol-8, number-1, 987–996, (2012)
14. Angulo, Julio and Wästlund, Erik: Exploring touch-screen biometrics for user identification on smart phones, Privacy and Identity Management for Life, Springer, 130–143, (2012)
15. Sae-Bae, Napa and Ahmed, Kowsar and Isbister, Katherine and Memon, Nasir: Biometric-rich gestures: a novel approach to authentication on multi-touch devices, In proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, 977–986, (2012)
16. Derawi, Mohammad Omar and Nickel, Claudia and Bours, Patrick and Busch, Christoph: Unobtrusive user-authentication on mobile phones using biometric gait

recognition, Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), IEEE, 306–311, (2010)
17. Tao, Qian and Veldhuis, R: Biometric authentication for a mobile personal device, 3rd Annual International Conference on Mobile and Ubiquitous Systems-Workshops, IEEE, 1–3, (2006)
18. Viola, Paul and Jones, Michael: Robust real-time object detection, International Journal of Computer Vision, vol-4, 34–47, (2001)
19. Saevanee, Hataichanok and Clarke, Nathan L and Furnell, Steven M: Multi-modal behavioural biometric authentication for mobile devices, Information Security and Privacy Research, vol-4, 465–474, (2012)
20. Buriro, A., Crispo, B., Del Frari, F., & Wrona, K: Touchstroke: Smartphone User Authentication Based on Touch-Typing Biometrics, In New Trends in Image Analysis and Processing (ICIAP 15) Workshops, Springer International Publishing, 27-34, (2015)
21. Aronowitz, Hagai and Li, Min and Toledo-Ronen, Orith and Harary, Sivan and Geva, Amir and Ben-David, Shay and Rendel, Asaf and Hoory, Ron and Ratha, Nalini and Pankanti, Sharath and others: Multi-modal biometrics for mobile authentication, IEEE International Joint Conference on Biometrics (IJCB), IEEE, 1–8, (2014)
22. Ferrer, Miguel A and Morales, Aythami and Travieso, Carlos M and Alonso, Jesws B: Low cost multimodal biometric identification system based on hand geometry, palm and finger print texture, 41st Annual IEEE International Carnahan Conference on Security Technology, IEEE, 52–58, (2007)
23. Kim, Dong-Ju and Hong, Kwang-Seok: Multimodal biometric authentication using teeth image and voice in mobile environment, IEEE Transactions on Consumer Electronics, IEEE, vol-54, number-4, 1790–1797, (2008)
24. McCool, Christopher and Marcel, Sebastien and Hadid, Abdenour and Pietikainen, Matti and Matejka, Pavel and Cernocky, Jan and Poh, Norman and Kittler, Josef and Larcher, Anthony and Levy, Christophe and others: Bi-modal person recognition on a mobile phone: using mobile phone data, IEEE International Conference on Multimedia and Expo Workshops (ICMEW), IEEE, 635–640, (2012)
25. Bertillon, Alphonse: Signaletic instructions including the theory and practice of anthropometrical identification, Werner Company, (2008)
26. Sitova, Zdenka, Jaroslav Sedenka, Qing Yang, Ge Peng, Gang Zhou, Paolo Gasti, and Kiran Balagani: HMOG: A New Biometric Modality for Continuous Authentication of Smartphone Users, arXiv preprint arXiv:1501.01199 (2015).
27. Poh, Norman and Bengio, Samy: Database, protocols and tools for evaluating score-level fusion algorithms in biometric authentication, A review, Pattern Recognition, Elsevier, vol-39, number-2, 223–233, (2006)
28. Conti, Mauro and Zachia-Zlatea, Irina and Crispo, Bruno: Mind how you answer me!: transparently authenticating the user of a smartphone when answering or placing a call, In proceedings of the 6th ACM Symposium on Information, Computer and Communications Security, Science and Technology, ACM, 249–259, (2011)
29. Davis, Steven and Mermelstein, Paul: Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences, IEEE Transactions on Acoustics, Speech and Signal Processing, IEEE, vol-28, number-4, 357–366, (1980)
30. Logan, Beth and others: Mel Frequency Cepstral Coefficients for Music Modeling, ISMIR, (2000)
31. Hsu, Chih-Wei and Chang, Chih-Chung and Lin, Chih-Jen and others: A practical guide to support vector classification, 2003

32. Oshiro, Thais Mayumi and Perez, Pedro Santoro and Baranauskas, José Augusto: How many trees in a random forest?, Springer, 154-168, (2003)
33. Survey says 70% password dont protect mobiles: download free Mobile Toolkit, http://nakedsecurity.sophos.com/2011/08/09/free-sophos-mobilesecurity-toolkit, 2014