Alternative Approaches to Smartphone User Continuous Authentication

Peter Story
Gordon College
Wenham, MA, USA
peter.story@gordon.edu

Wendy Rummerfield
University of Redlands
Redlands, CA, USA
wendy.rummer13@yahoo.com

Marcelo Colomé
Federal University of Santa Maria
Santa Maria, Rio Grande do Sul, Brazil
marcelocolome@gmail.com

Paolo Gasti
New York Institute of Technology
Manhattan, NY, USA
pgasti@nyit.edu

Kiran Balagani
New York Institute of Technology
Old Westbury, NY, USA
kbalagan@nyit.edu

***Abstract*—Smartphone continuous authentication seeks to add a layer of defense beyond existing entry-point authentication systems. Most pre-existing research focuses on data gathered from the touch screen itself, such as the characteristics of tapping and swiping. However, our focus is on features from different sources of data, including data recorded by the smartphone accelerometers and built-in microphone. Our three approaches were:**

1) **Using phone movement as a biometric.**
2) **Detecting heartbeat through movement, as a means of liveliness detection.**
3) **Using touchscreen interaction sounds as a biometric.**

I. Introduction and Related Work

Smartphones are increasingly ubiquitous today, and have the ability to hold personal sensitive information; however, these devices are susceptible to loss, theft, or intrusion. A range of technologies exists to mitigate those risks and guard against attacks.

Attacks are defined as attempts to view private information such as emails, pictures, etc. without triggering a detection system [6]. The most widely deployed security measures against attacks are *personal identification numbers* (PINs) and *gesture-based entry-point authentication*, both of which require user attention for entry. As stated in Frank et al. [2], users tend to set weak passcodes and long timeouts, or completely disable this functionality because of the perceived inconvenience of these systems, leaving the device vulnerable.

An alternative authentication technique employs touch-based features to actively verify the user on a continuous basis. Frank et al. [2] used vertical and horizontal swipe data, in both portrait and landscape mode, from the touchscreen to distinguish between the primary user and an attacker. With *Touchalytics* as a starting point, Serwadda et al. [7] compared 10 different algorithms, the first attempt to pinpoint the best algorithm to distinguish between authorized users and attackers. This is an active area of research, and authors favor different verification algorithms: Frank et al. [2] used k-nearest neighbors (kNN) and support vector machine (SVM) Song et al. [8] found that the Gaussian mixture model (GMM) is better than SVM for abnormality detection while Serwadda et al. [7] favored Logistic Regression, SVM, and Random Forests as the best user authentication approach.

Another fertile area of research is touch-feature selection, or deciding which features to include in the model, based on which are most helpful in discriminating between users and attackers. Song et al. [8] used the Fisher feature selection technique, which favors low within-class-variance and high between-class-variance to improve discriminative capabilities, while Govindarajan et al. [3] favors unsupervised feature selection without class information, making it more scalable than supervised methods. Nevertheless, there is not a universally agreed on feature set or feature selection technique across different domains.

In addition to the questions of verification algorithm choice and feature selection, there are also research gaps in the area of security. Govindarajan et al. [3] addressed the problem of securing the user's template when outsourcing touch data. Their technique successfully enabled the secure computation of exact Manhattan and Euclidean distances between the template and new input, in order to efficiently verify users. Another gap has been the assumption of zero-effort attacks, in which the attacker does not attempt to mimic the user. To illustrate the weakness of this assumption, Serwadda et al. [6] demonstrated a successful attack using a Lego robot to generate gestures based on behavioral biometric patterns representative of the general population, creating increases in Equal Error Rates (EERs) between 339%

and 1004%. This was one of the first demonstrations against touch-based continuous authentication, and our focus will be designing measures to counter such an attack.

II. Contribution

Our goal is to design and test new biometric features focusing on those that can distinguish between humans and robotic attackers. These features may be used in an authentication scheme with two layers: the first discriminates between the user and other humans, and the second layer determines whether a robot is providing input to the phone. Such an authentication system would thwart a robotic attack like that described in [6].

A. Phone Response to Tapping

The first group of features relate to the motion of smartphones during touch screen interactions. When a user executes a touch or swipe, they first move the phone towards their finger. Next, after the finger comes in contact with the screen, the user's touch pushes the phone away from his or her body. And finally, lifting the finger from the phone, the hand holding the phone overcompensates, pushing the phone toward the user.

With pilot data, we demonstrate that the motion of the smartphone during touchscreen interaction is sufficiently distinctive to discriminate between users.

B. Liveliness Detection Through Heart Rate

The next group of features involves the small changes in acceleration of the smartphone that could be representative of a person's heartbeat. As a means of liveliness detection, heart rate has promise as biometric to authenticate against a robotic attack. If heart rate can be detected from the smartphone's motion sensors, it could be difficult for the most advanced prosthetics to replicate wrist and finger micro-motions. Additionally, the average person's resting heart rate ranges from 60 to 100 beats per minute and differs between sexes, ages, and depends on activity level.

C. Acoustic Signature of Touch Interactions

The last group of features focuses on the sound generated when a user interacts with a smartphone. Sounds are generated when a user's finger touches the screen, and we anticipate that these sounds will be useful for discriminating between users. These features will be captured by the built-in microphone present in a regular smartphone. The contact between the user and the screen will be divided into different categories, such as touch and swipe movements, that will generate acoustic signatures for each user.

III. Approach

A. Phone Response to Tapping

When people interact with a smartphone touch screen, their touches cause the phone to move. These motions can be detected and recorded by the phone accelerometer and magnetometer. From recorded motion information, we can calculate each user's usage profile, and future interactions with the device can be compared to this profile to verify the user's identity.

B. Liveliness Detection Through Heart Rate

In order to to extract heart rate using motion, data analysis was performed on information collected from the accelerometer as the subject interacts with the typing application. During testing, pilot data was collected from the user during touches and stored in a database file. Using SQLite, accelerometer information for each coordinate direction were further examined in Matlab using a specialized program called *CaptureBPM*, written to measure beats per minute (BPM).

After removing outliers using the interquartile range (IQR) method, *CaptureBPM* attempts to identify a person's heartbeat by searching for peaks in an acceleration signal above thresholds. Peaks are defined as points that have a greater value than two neighboring points. Further a threshold is the lower limit in which the detected peaks must be above to be counted as a beat. Since limits vary between coordinate directions, there will be significant guess and check involved in finding the optimal threshold.

Furthermore, additional features will be calculated in order to create an identity vector which will act as a template for the user. This vector will include characteristics such as mean, standard deviation, peaks counted, and the percentage of peaks above the threshold. Once created, these vectors will be compared against other users to determine the robustness of heart rate as a biometric for continuous authentication.

C. Acoustic Signature of Touch Interactions

When a user types on a smartphone software keyboard, sound is produced that can be recorded using the built in microphone. We anticipate that these sounds are distinctive to users, and might be useful in an authentication system for discriminating between users. We designed a framework that extracts features from the sounds produced during touch events, constructs user profiles, and compares those profiles.

IV. Experiments

A. Phone Response to Tapping

Our experiment consisted of five lab members and one of the authors using an app designed to record phone motion during typing. Each user participated in three sessions of approximately 5-10 minutes each. In each session, the user typed responses to five simple questions. Each response required that at least 100 characters be typed. The experiments were conducted in a quiet, distraction-free environment. Participants sat on a stool, away from any tables to avoid the effect that leaning against the arms of a chair or the surface of a table might have on motion. In each session, the users were asked a different set of five questions, to avoid any effect of users being familiar with the questions.

B. *Liveliness Detection Through Heart Rate*

Pilot data was collected from our team as well as lab members. Using our application, each user was asked to type in answers to two simple questions with a minimum of 50 characters while: 1) the phone is flat on the table, 2) the device is being held while the subject is sitting (not resting arms on any surface) , and 3) while the subject is standing away from a wall holding the smartphone. Users did not have a time limit during testing.

To confirm results from the program, participants were asked to use the commercially available *Instant Heart Rate* Android application to monitor his/ her heart rate before and after producing text on our application [4]. These values will provide a target range for BPM during usage in order to substantiate the Matlab program's results. *Instant Heart Rate* is regarded as "the most accurate Heart Rate Monitor app for any smartphone and it does not need any external hardware" [4]. This Android app employs the camera and flash to compute a person's heartbeat by searching for changes of light in the finger. During pulses, blood rushes through the finger artery, which causes changes in brightness that can be detected with the smartphone camera [5]. It is important to note that the Android *Instant Heart Rate* application is not a candidate for continuous authentication because it requires the user to maintain the position of the hand during interaction with the device which would most likely not be natural to the user.

C. *Acoustic Signature of Touch Interactions*

Pilot data was collected among our group, with two sessions for each member, totaling six data collections. This pilot data collection was made using an application developed for the Android operating system, designed for storing sensors values in the database. The application was modified in order to solve some issues and also to support sound recording. Audio was recorded in the 3gp audio format, because it is supported by the Android OS. Later, the audio files were converted to WAV audio format, in order to provide a common file format that is well-supported by professional audio software.

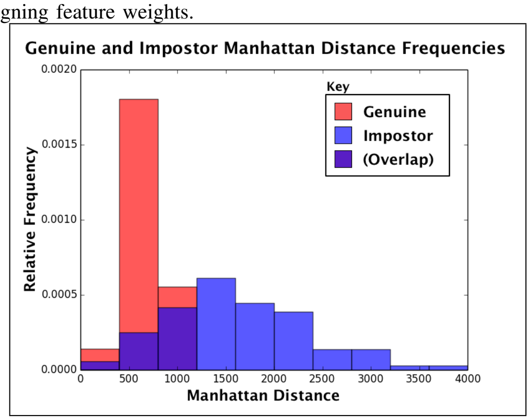
V. Results

A. *Phone Response to Tapping*

User profiles were composed of features of the motion immediately preceding and following each letter typed on the keyboard. First, the features were computed for each letter typed. Next, the trimmed mean of the features were divided by the trimmed standard deviation, and this standardized value was included in the user's profile vector.

We evaluated the features with cross-validation under a zero-effort attack model. The genuine user case was evaluated by forming user profiles from two of a user's sessions, and calculating the distance between this vector and the vector of the remaining session, for all permutations. The impostor case was evaluated by forming user profiles from all three of a user's sessions, and calculating the distance between all sessions from the other users. These comparisons resulted in two sets of distances, one set from the genuine user comparisons, and the other set from the impostor comparisons. For each distance, the comparison can be either accepted or rejected, based on whether the distance is below or above a chosen threshold.

Fig. 1. Manhattan distances from genuine and impostor groups, after assigning feature weights.



As visible in Figure 1, the distributions of the genuine user and impostor distances appear to differ. Our numerical results also confirmed the viability of these features for the purpose of distinguishing between users.

We found the Manhattan verifier to consistently offer better performance than the Euclidean verifier. We were able to reduce our EERs by assigning different weights to the features. Specifically, assigning less weight to orientation improved performance and more weight to both acceleration in the x-axis and to the acceleration magnitude.

B. *Liveliness Detection Through Heart Rate*

As expected, the Matlab program was not able to detect a heart beat when the phone was lying on the table.

This is because the average distance between points in the accelerometer data was less than .1m/s^2 and too small to detect in the program.

Table I summarizes the results of two users when the phone was in their hands while sitting. A computed heart rate is considered accurate if the detected BPM was between the range found using *Instant Heart Rate* and deemed incorrect if not inside the span. BPM was detected with 100% accuracy during this session; however, it is important to keep in mind the small sample size.

Heart rates measured before and after a single sitting trial ranged from one to six beats per minute. The variability of heart rates during testing could create difficulty for the program in identifying a single rate. In consequence, the heartbeats computed in Matlab gravitated towards one of the ranges found using the *Instant Heart Rate* application. Nevertheless, measured heart rate ranges varied between subjects and have discriminative capabilities.

As seen in the table below under ”Ratio of Peaks Above Threshold”, the ratio of beats counted over the total number of beats is similar for each user in a specific situation. These ratios are consistent for each user in a specific context and may be distinctive between individuals. The percentages in different scenarios could be one of the features applied in the creation of a template for each user.

TABLE I
SITTING RESULTS

User	Start Heart Rate	End Heart Rate	Computed Heart Rate	Ratio of Peaks Above Threshold
1X	74	80	75.736	1.2622
1Y	74	80	76.763	1.2793
1Z	74	80	75.026	1.2504
2X	72	66	69.674	1.1612
2Y	72	66	69.721	1.1620
2Z	72	66	69.605	1.1600

In the next set of samples shown in Table II, the subject was asked to interact with the typing application while standing up. This testing will be used to determine the algorithm’s effectiveness in other controlled situations. *CaptureBPM* correctly computed a heart rate while standing with an accuracy of 93.33%. Reduction in precision compared to the sitting trials could result from the increased amount of physical freedom. While sitting in a chair, bodily movement is limited to the torso and upper body. When standing, corporeal motion is extending to the legs, which can bend or bounce and sway back and forth causing extraneous movement.

Though there is much work needed to improve the heart rate calculation algorithm, this proof of concept shows much promise for BPM monitoring from accelerometer readings. The heart rate calculated with *CaptureBPM* showed a good amount of accuracy for the sitting and standing scenarios.

TABLE II
STANDING RESULTS

User	Start Heart Rate	End Heart Rate	Computed Heart Rate	Ratio of Peaks Above Threshold
1X	65	61	62.534	1.0422
1Y	65	61	62.434	1.0422
1Z	65	61	63.525	1.0587
2X	75	83	80.000	1.3333
2Y	75	83	79.482	1.3247
2Z	75	83	80.981	1.3496

Currently, the Matlab heartbeat extraction program involves considerable guesswork in defining the optimal threshold. However, calculated ratios of number of data points above the threshold show promise as a means to uniquely measure the current operator’s BPM.

C. *Acoustic Signature of Touch Interactions*

The audio file generated during data collection was broken down into small pieces, containing just the sound that happens when the user is touching the screen. The moment when a touch takes place, previously stored in the database using the data collection application, are used to determine the correct time that the audio file should be trimmed. After getting this time in milliseconds, we determined a range in milliseconds for the start and end time of the audio file cut. We determined a 12ms range that happens between 153ms and 165ms after the time that the touch on the screen is captured.

These small audio files were processed in jAudio [1] generating values representing the magnitude and power measurements for each touch sound and then values resulted from the process were combined generating a sound signature for each user.

The next step was to build arrays containing the minimum, median, mean, maximum, and standard deviation values for each of feature values previously extracted from the small audio files, producing a total of ten features, five for amplitude and five for power values. For each user, this process was made three times, producing an array for the first session, one array for the second session and another array combining values from both sessions.

The last step was to calculate the Manhattan and Euclidean distances from these vectors, in order to validate our framework for audio features extraction. We first compared user’s first session and second session generating an array with this values. We called this array as Control values. Then we generated another array contained values from the comparison among different users, and we name this as a Attack vector.

Values for this process can be visualized in the following picture, and in this particular case, are to show the framework outcome.

Fig. 2. Manhattan distances from genuine and impostor groups.

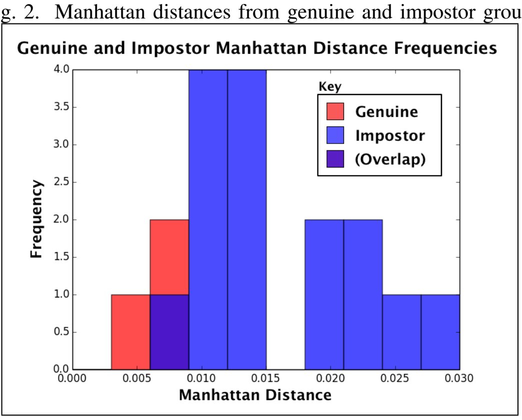
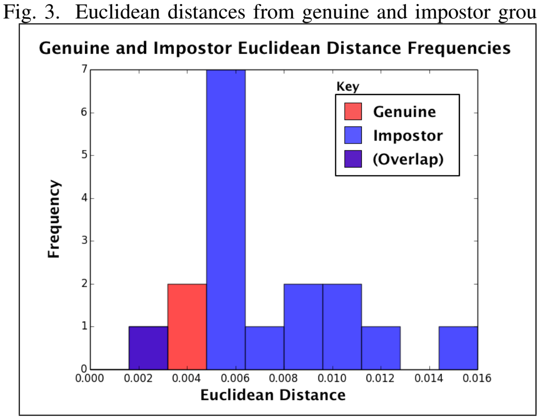


Fig. 3. Euclidean distances from genuine and impostor groups.



VI. Future Work

A. Phone Response to Tapping

We anticipate that performance and real-life applicability could be further improved by generating separate vectors for different contexts. For example, a user's movements might differ significantly from when they are sitting or standing, or whether they are typing on the left or right-hand side of the keyboard. Performance might also be improved with further work on weighting features or by using different verifiers.

The app should be improved in several ways before future studies are conducted. First, the keyboard on the app is smaller than the standard Android keyboard, and this frustrated users.The keyboard should be made as similar to the stock Android keyboard as possible, to avoid user frustration. Improving the keyboard might also result in higher-quality data. Second, we recommend collecting other forms of smartphone interaction, in addition to typing. Different patterns of motion might be associated with clicking links or icons, and this deserves further investigation in order to increase the accuracy and availability of the biometric.

B. Liveliness Detection Through Heart Rate

In order to more effectively verify the heart beats detected using *CaptureBPM*, a entirely new experiment would be performed. First, the data set would include a larger and more diverse set of participants who have varying levels of experience with smartphones. Secondly, the subjects' heart rates would be monitored during the trial with an electrocardiogram (ECG). After extracting a heart rate using the Matlab program, the beats found from the ECG would be compared to the peaks detected in *CaptureBPM*. This will better account for variations of the heart beat during usage of the typing application.

Regarding the peak detection program, the most important aspect to work on is the discovery of a universal method to determine a threshold. Once this can be calculated, heart rate monitoring through the device's accelerometer could be very effective and useful to everyday smartphone users.

Also, if the Matlab program proves successful, the authentication ability of the biometric will be tested against a robotic attack as well as a zero-effort human attack. Much work has already been done on the strength of heart rate as a biometric, however, not in the area of continuous authentication. If this feature proves useful for user verification, heart rate could be a new biometric for continuous authentication that would defend against a robotic attack.

C. Acoustic Signature of Touch Interactions

Because of the small number of subjects used in this research we cannot affirm that audio features extraction can be used to create unique signatures for each user, although more research in this area still to be done, and can be facilitated by the framework for audio features extraction presented in this paper.

VII. Conclusion

Our preliminary experiments showed two of our modalities to have promising performance. The motion surrounding touch strokes was used to discriminate between lab members with a high-degree of accuracy. Heart rate detection through accelerometer readings shows promise as a biometric due to precision with preliminary testing. If further testing confirms the high performance of these modalities, authentication systems built on them could be tested against robotic attacks. We also designed a framework for biometric audio feature extraction, which we validated with real subjects using Euclidean and Manhattan verifiers. This framework should be useful to future research.

VIII. Acknowledgments

- Wendy Rummerfield and Peter Story were funded by National Science Foundation Grant No. 1263283

- Marcelo Colomé was sponsored by CAPES through the Brazilian Scientific Mobility Program (BSMP)
- NYIT hosted the summer REU program, and Dr. Panero and Dr. Dong lead the program
- Dr. Gasti and Dr. Balagani gave guidance to our group throughout the program
- Kathia F. Torres Aviles, a student in last summer's REU who developed the app which we modified and used for data collection

REFERENCES

[1] jaudio 1.0 on sourceforge. http://jaudio.sourceforge.net, May 2013.

[2] M. Frank, R. Biedert, E. Ma, I. Martinovic, and D. 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, January 2013.

[3] S. Govindarajan, P. Gasti, and K. S. Balagani. Secure privacy-preserving protocols for outsourcing continuous authentication of smartphone users with touch data. *Biometrics: Theory, Applications and Systems (BTAS), 2013 IEEE Sixth International Conference on*.

[4] Azumio Inc. Instant heart rate app on play store. https://play.google.com/store/apps/details?id=si.modula.android.instantheartrate, March 2012.

[5] P. Pelegris, K. Banitsas, T. Orbach, and K. Marias. A novel method to detect heart beat rate using a mobile phone. *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, pages 5488,5491, September 2010.

[6] A. Serwadda and V. V. Phoha. When kids toys breach mobile phone security. *2013 ACM SIGSAC conference on Computer & communications security (CCS '13)*, 2013.

[7] A. Serwadda, V. V. Phoha, and Z. Wang. Which verifiers work?: A benchmark evaluation of touchbased authentication algorithms. *Biometrics: Theory, Applications and Systems (BTAS), 2013 IEEE Sixth International Conference on*, pages 1,8, September 2013.

[8] Y. Song, M. B. Salem, S. Hershkop, and S.J. Stolfo. System level user behavior biometrics using fisher features and gaussian mixture models. *Security and Privacy Workshops (SPW), 2013 IEEE*, pages 52,59, May 2013.