Demographic Group Classification of Smart Device Users

Adel R. Alharbi and Mitchell A. Thornton
Department of Computer Science and Engineering
Southern Methodist University
Dallas, Texas USA 75275–0122
Email: {aalharbi, mitch}@lyle.smu.edu

***Abstract*—Interacting with smart devices is a common experience and is becoming an integral part of daily life for many people. Modern smart devices are equipped with a large variety of environmental and user input sensors. We hypothesize that a fusion of smart device sensor data can provide biometric data that allows for classification of user demographics such as age, gender, and native language. A smart device is instrumented with sensor data collection software and with user demographic classification software. An experiment is devised where data is collected for a sample group of users. The data is analyzed, and two classification algorithms are implemented based on fusion of the different sensors. The classification methods are based upon decision tree and principle component analysis. The results of the experiment indicate that high accuracy is achieved for user demographic classification. Finally, we further discuss the applications and limitations.**

I. Introduction

Interacting with smart device applications is an integral part of many users' daily activities. Smart device users interact with diverse applications for purposes such as texting, phoning, reading email, and playing games. These interactions coupled with the diverse set of sensors present in the device result in unique behavioral patterns for each user. These patterns may be considered for use in biometric-based authentication such as that described in [1]–[4]. Moreover, these patterns can play a significant role in predicting user demographics, allowing for customized delivery of commercial or social networking services. The design challenge is to develop a methodology that can be used to classify a large number of smart device users in a way that is secure. More importantly, it enables companies to deliver services more efficiently.

Recent studies have proposed diverse approaches for the non-intrusive authentication systems in smart devices. Lin et al. [1] categorized authentication approaches into dynamics-based features and behavior-based approaches. The first approach maintains extraction of various statistical features, such as distance, speed, average, velocity, and acceleration. The second approach exploits the statistics of predefined types of behavioral actions using data distribution visualization techniques, such as scatter, box, and histogram plots [1]. Both approaches could authenticate users in a short period of time. However, they depend on the amount of testing data in order to increase the performance and accuracy [1], [3].

Our study contribution uses the behavior-based approach because it is more practical since the data calculation operations are time consuming. The main objective of the approach is to collect the unsupervised behavioral data that is captured during the normal usage of the smart device. The collected data can be transferred over the operator cloud network or simply stored into the device's SD card [4]. The motivation is to investigate a new way of providing various operator services for the right group of users at the right time using the recognition patterns, without breaking any users' privacy and security systems. We hypothesize that a fusion of smart device sensor data can provide biometric data that allows for classification of user demographics such as age, gender, native language, and other categories. The approach does not require any users' personal information or additional hardware materials, and there is zero cost for the operators [5].

The rest of the paper is organized as follows: Section II briefly surveys the related work of behavioral biometrics. Section III describes our data acquisition in the data collection setup, the data collection application. Section IV discusses our unique methodology of proposing a new data model based on the three behavior models. It also investigates them in terms of principal component analysis, decision tree classifier and additional machine learning techniques. Section V presents the experimental results and discussions of the proposed approach. Finally, Section VI discusses the conclusion and future work.

II. Related Work

Researchers studied and discovered the keyboard dynamics recognition patterns. Thornton [6] characterized keyboard dynamics as the characteristics of writing or cadence patterns of arrangements of keystrokes on a keyboard. Thornton also explained that the musical writing patterns of diverse individuals had characteristics that are one of a kind. Maxion et al. [2] also discussed this matter. Their work was based on the data collection from twenty eight participants who typed the same ten-digit numbers using only the right-hand index finger. They proposed an approach that used statistical machine learning classification algorithms, in particular the random forest classifier. They achieved an unweighted False Acceptance Rate (FAR) of 99.97% and False Rejection Rate (FRR) of 1.51%.

Researchers also studied the touch-screen behaviors for the smart device's touch screens. One significant study by Frank et al. [3] investigated whether the studied classifier was able

to consistently authenticate users. They proposed management of thirty behavioral touch features as a framework and then used it as a behavioral pattern. They used weighted k-nearest neighbor (WkNN) and support vector machine (SVM) classifiers. They collected data from forty-one users. They achieved a median Equal Error Rate (EER) of 0% for intra-session authentication and 2% to 3% for inter-session authentication of the enrollment phases. Lin et al. [1] also discussed touch-screen fusions. Their work was the first reported research that adopted the histogram features of touch-screen attributes. They also used only the WkNN classifier, and the collected data was from fifty-five users. They achieved a ERR of 2.9% to 3.6% when the number of touches was sixty.

Lately, studies have focused on the behavioral factors of the holding posture for the human hands in smart devices' applications during normal usage. Nixon et al. [4] proposed a novel mobile user authorization and classification approach based on the recognition user's gesture. They studied the produced data of three volunteers from the three-axis accelerometer and gyroscope of the mobile built-in sensors. Their analysis demonstrated that gesture could offer a solution for device data protection.

In conclusion, although the surveyed works studied behavioral biometrics and machine learning, they were limited to the number of the extracted data fusions of one or two data fusion features. Another limitation was that they only concentrated on the concept of authentication.

III. Data Acquisition

The section mainly focuses on two parts, which are the data collection setup and the data collection application.

A. The Data Collection Setup

The pilot study carried out data collection using an Android device. The target of this data collection was to encourage users to produce navigational keystrokes, touch screen, and device orientation behaviors in a natural way. The data collection application was developed to be User-Interface (UI) friendly and simple, and also to be an enjoyable game-like experience.

1) Selection Criterion: The criterion that was used in the selection process was to divide case subjects into demographic groups. The demographic groups were based on users' shared characteristics from a wide range of attributes.

2) Timing Period: The investigators estimated the overall time that it took a case subject to interact with the application. The result of the overall data collection time was approximately between twenty-five to thirty minutes per case subject.

3) The Data Collection Design Decisions: The data collection was designed to ensure all the volunteers had the same environment set up during the recording sessions [3]. The data collection design is as follows:

- **User adaption to application tasks:** None of the users were aware of any of the data collection application tasks.
- **Data collection application tasks ordering:** The application gave the chance of selecting any tasks at any given time with no order.
- **Device orientation:** Due to the fact that G1-sensors are very sensitive with respect to the earth's gravity, the study required all participants to sit down in a chair.
- **Device screen size vs. readability:** The study chose a device with a large screen (tablet) to ensure that the soft keyboard and the questions could be clearly seen.
- **Application screen interface layout orientation:** The study had to fix the application to be only in vertical screen interface layout mode to ensure that all user input interactions were in the same layout orientation.

4) The Data Collection Captured Device Tool: The study used one device which was ASUS MeMO Pad HD 7 (ME173X) [7]. The research used one device to ensure that there were no bias values in the data collection.

5) Data Collection: The research collected the sample data from twenty-two participants. The users were 73% male and 27% female. Most of the volunteers were not English users (68%), and the rest were English users (32%). Moreover, the study compared two operating systems, specifically Apple and Android. Apple users made up 64% and Android users made up 14%. The rest were familiar with both of the operating systems (22%). Additionally, the data collection had various user nationalities as follows; United States (31.8%), Saudi Arabia (22.7%), China (18.1%), Palestine (9%), Mexico (9%), United Arab Emirates (4.5%), and Pakistan (4.5%). Table I shows the age distribution with a resolution of five years.

TABLE I: Age Distribution.

< 18	19-23	24-29	30-35	36-41	42-47	48- 53
0%	41%	36%	9%	4.5%	4.5%	4.5%

B. The Data Collection Application

The data collection application offered regular usage UI's such as entering routine information, re-entering copied information from other sources, and re-entering complex tokens. Additionally, the application offered reading an article, drawing, zooming in/out on a picture, and playing an oriented game as in Table II.

In Table II, each UI had biometric behavior purposes where more than one behavior biometric was implemented to collect more behavior data per particular UI. For example, in User Information, UI used keystrokes and device orientation behaviors implemented at the same UI. Similarly to the example, most of the UIs implemented the device orientation behavior.

1) Extracting Behavior Features: Any Android applications have the capability to run Android API classes. Part of these APIs can be used to extract the users' behavioral data. The application used Text Watcher API to capture the key that the user pressed on the soft keyboard. The application used Motion Event, and On Touch Listener APIs to capture the case subject finger movements over the touch-screen device. The Sensor Event, Sensor Event Listener, and Sensor Manager

TABLE II: The Data Collection Application UI descriptions.

UI Name	Description
User Information	If the user selected this option, the user information screen appeared. It would ask the user to enter normal information such as name, phone number, email address, city, and zip code.
Quote for today	If the user selected this option, this screen appeared and would ask the user to re-write one fixed quote.
Three Tokens	If the user selected this option, this screen appeared. It would ask the user to retype three fixed complex tokens which included letters, numbers, and punctuations.
SMU Article	If the user selected this option, the SMU Lyle School screen appeared. It would ask the user to read the article and swipe to the right in order to answer five related questions.
Hidden Secret Game	If the user selected this option, this screen appeared with a picture and three questions related to the picture. The user would try to zoom in/out to find the answers to these questions.
Drawing	If the user selected this option, this screen appeared. It would ask the user to draw anything the user would like such as a house, any word, or any signature with the user's finger.
Avoider Game	If the user selected this option, this game screen appeared. The idea of the game was to avoid the Avoider from the black clouds and eat the yellow suns in order to win this game.

were used to obtain the user behavior movements while holding the capture device tool. Moreover, the application used the asynchronous task techniques to gain a better performance of the API to run in the background of the application [8]. How the application extracted the behavior biometrics fusion features is addressed below.

The application captured three key actions as Before, On, or After typing a particular letter. The behavioral keystroke fusion had data features which were key actions ($KS\_Actions$), key code which is the ASCII key code of a letter (Key_Code), and key timestamp in milliseconds (KS_T) [2].

The application also captured the four types of touch actions indicating the finger position on the screen which were Up, Down, Cancel and Move action types, each of the touch actions associated with the touch position of X-coordination and Y-coordination in pixels. In fact, the touch behavior observation also captured the pressure and size of the area covered by the touched finger, which took a range value from zero to one [8]. The conduction of behavioral touch-screen fusion had six data features which were the touch actions ($T\_Actions$), X-coordination ($XCoor$), Y-coordination ($YCoor$), pressure (Pr), size (S), and touch timestamp in milliseconds (T_T) with respect to the number of fingers that touched the device's screen [3].

Finally, the application captured the three-axis accelerometer sensor's data. The study aimed to observe the changing and the impulses that occur in the three dimensions' data which were X (X), Y (Y), Z (Z), and arm movement timestamp in milliseconds (Ori_T). The X axis indicated to the right (horizontal) direction. The Y axis indicated to the up (vertical) direction. The Z axis indicated to the outside direction of the front face of the device's screen [8].

2) Security and Hiding Methods: The data collection application was implemented to provide a simple and unique security and hiding method for volunteers' behavior-collected data. The collected data was secured with a password that was only obtained by the investigators with Android API running in the application's background [8].

3) Recording Techniques: The data collection application opened the chance to capture the behavior data by Android APIs. Moreover, the application stored them into the SQLite database internally, and then the database files could be simply extracted from the device's SD card [9].

IV. Methodology

This section concentrates on using data fusion visualizing techniques and then modeling the Multi-Fusions. Next, the section investigates the relationships between the Multi-Fusions data model features and uses the data transformation solutions. Finally, the methodology uses the power of the principal component analysis with the decision tree classifier.

A. Data Fusion Visualizations

There are many techniques of data visualization; this section uses scatter, box, and histogram plots [10]. The study observed the user genders whether male (red) or female (blue) from the X and Y coordination features as shown in Figure 1.

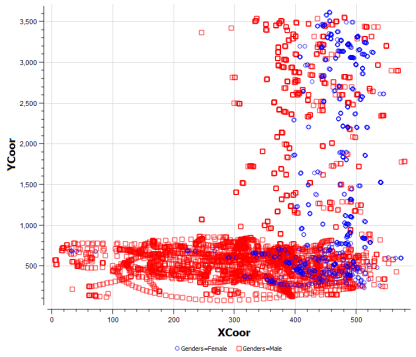

Fig. 1: User genders by X and Y-coordination features.

Upon further investigation of user demographic groups, the study revealed that users of different ages apply different touch pressures. Figure 2 shows the five different user pressure ages presented with boxes, and each box has its mean in a thick line [10].

Figure 2 shows the 25-year-old user gains of about 0.2 of pressure mean value where the 28-year-old user obtained the lowest pressure mean value among all other user ages. Another observation between these users' pressure mean was the difficulty of using the device's application such as the operating system type (Apple vs. Android) or trying to use the application itself.

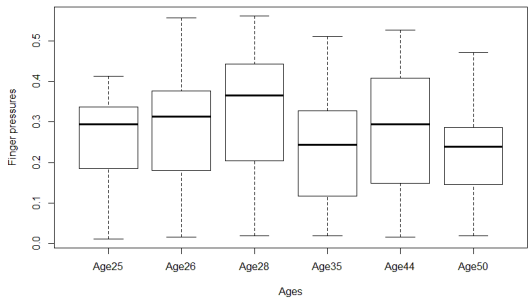

Fig. 2: User ages by pressure feature.

One of the goals was to identify users' languages. Figure 3 shows the keystroke ASCII key code with the frequency, and also shows English users in blue and non-English users in red. The frequency means how often the keystrokes might appear. The figure also displays the statistical probability curve for English keystrokes' key code when it took place between the 0.1 until 0.2 of the frequency values [10].

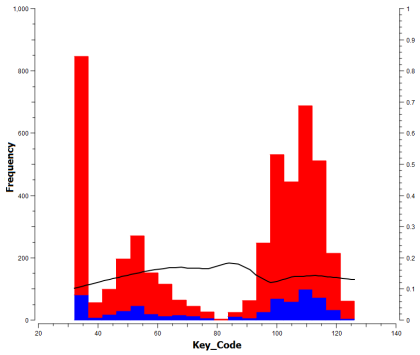

Fig. 3: User languages by keystrokes' key code feature.

B. Multi-Fusion (MF) Model

According to the previous discussion in the extracting behavior features section, the three biometric behaviors were the keystroke, touch screen, and device-orientation behaviors. The observation of the three biometric behaviors' data features leads us to conclude that each can be expressed into three data models. The first data model is the keystroke data model that can be a trajectory data vector encoded as:

$$KS = [KS\_Actions_i \ , Key\_Code_i \ , KS\_T_i] \quad (1)$$

The second data model is the touch-screen data model that can be a trajectory data vector encoded as:

$$T = [T\_Actions_i \ , XCoor_i \ , YCoor_i \ , Pr_i \ , S_i \ , T\_T_i] \quad (2)$$

The third data model is the device-orientation data model that can be a trajectory data vector encoded as:

$$Device\_Ori = [X_i \ , Y_i \ , Z_i \ , Ori\_T_i] \quad (3)$$

The study method was to combine the three data models into the MF data model to generate a recognition pattern based on the MF data features. As a result, the MF model has thirteen features and can be presented as:

$$MF = [KS\_Actions_i \ , Key\_Code_i \ , KS\_T_i \ , T\_Actions_i \ , XCoor_i \ , YCoor_i \ , Pr_i \ , S_i \ , T\_T_i \ , X_i \ , Y_i \ , Z_i \ , Ori\_T_i] \quad (4)$$

The i represents the user's actions while using the smart device. This leads us to conclude that $i \in \{ \ 1 \ , \ ... \ , \ \infty \ \}$.

C. Data Transformation

The research targeted to solve some of the issues with the collected data. There were many solutions that could be applied to the research. This subsection focuses on two proposed solutions.

1) Missing Values: The MF data model combined fusions from the three data models (1), (2), and (3). However, each data model had different instance numbers that produced missing values in the combining process. The solution has to replace the missing values with zeroes to fix the issues that might occur when calculating the max, mean, and min during the classification process [10], [11].

2) Noise Values: The X, Y, and Z features had noise values because the acceleration measured by the G1's accelerometer was not equal to gravity. The solution has to use the re-center normalization technique [10]. The re-center operated as a standard z-score transformation. The variables' mean value was subtracted from each value. Then, each variable was divided by the standard deviation. As a result, the produced variable features had an average of zero and a standard deviation of one [11].

D. Multi-Fusion Relationships

The research visualized the MF feature relationships by using the correlation matrix. Figure 4 shows the hidden feature relationships with each MF attributes by coloring the correlation relationships [10]. The color becomes dark blue as higher correlation exists between two data features, and is red where there is less correlation. Moreover, the size of the circle represents whether the feature has large or small correlation with other features [11].

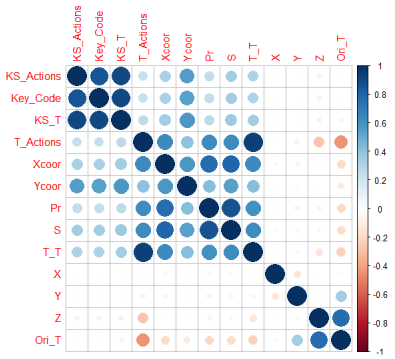

Fig. 4: Correlation matrix for the MF data features.

In Figure 4, it is very clear to see in the diagonal the standard deviation for the features with the value of one. On the other hand, in the opt-in diagonal are the correlations of the features. For example, the value 0.57 appears in the column of X-coordination with the row of Y-coordination and similarly reappears in the row of Y-coordination with the column of X-coordination. In fact, this means these are positively correlated which means having the same circle size and color (blue).

E. Principal Component Analysis (PCA)

PCA as a concept is an unsupervised method, and it is useful for identifying underlying hidden fusions rather than analyzing the discovered data. In the previous investigation, the research showed that the MF had hidden relationships that were produced in multi-dimensions. By implementing the PCA, the study was able to exploit and reduce the MF dimensions into a new set of principal components which could be used in representation, evaluation and more importantly classification [10], [12].

PCA discovers the useful components and ranks them in top-down order. It also gives the chance to select the useful components by separating them from useless components using the component variances. This selection process is an important step to take because having these less-ranked components will affect the learning curves and the classification process [12]. As a result, the study had approximately thirteen new principal components. At this point, the study considered taking the top ten principal components and omited the last three components since they had a rate of change of zero [11]. The study also considered that the proportion of variances depended on the size of the collected data set. The collected data set will be changeable because of collecting more users' behavioral data. Future studies may add or subtract one or two components to or from the ten principal components, depending on classification performance.

F. Classification Framework

The research chose decision tree classifier J48, also known as C4.5, since it served the goal of deciding which user belonged to the right demographic groups [10], [11]. J48 is a pruned tree that is based on a top-down strategy, a recursive divide and conquer strategy. It spreads feature values and categorizes them into leaves by selecting the feature to split on at the root node. Then, it creates a division for each possible feature value and splits the variables into subdivisions, one for each division that extends from the root node. In fact, it repeats the execution recursively for each division by selecting a feature at each node. Moreover, it uses the only variable that arrives at that division to make the selection [13]. Figure 5 shows an example of the proposed classifier using the PCA values for the genders' demographic group.

V. Experimental Results and Discussions

Every participant who used our smart device generated MF actions that contained the three biometric behaviors as discussed previously. Each participant conducted between 11,000 to 18,500 MF actions. As a result, the total number of the collected data was 315,470 MF actions. The study aimed to use the data transformation solutions and the PCA to produce the ten PCA components. Moreover, the settings of the decision tree classifier that were employed in the experiment were as default [13].

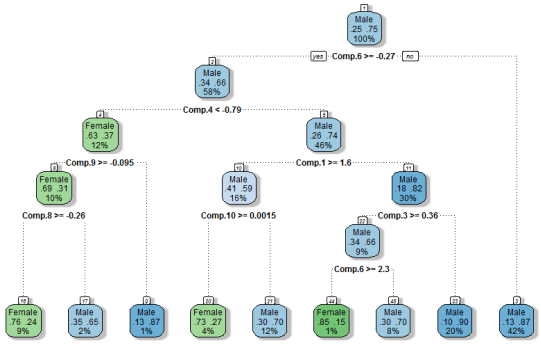

Fig. 5: Example of decision tree classifier by the PCA.

A. Experimental Results

To validate the proposed approach, the research aimed to randomize completely the produced data and use the various training percentage splits starting with a full training set until 10%. Each training percentage split was examined in fivefold cross-validation with five iteration repetitions to have more reliable results. The experiment noted the users' demographic groups as D1: Genders, D2: Ages, D3: Languages, D4: Operating System Types, D5: Nationalities.

Table III shows the accuracy rates in various training percentage splits with the corresponding five demographic groups.

TABLE III: Accuracy rates.

Train %	**D1**	**D2**	**D3**	**D4**	**D5**
Full Set	96.64%	93.15%	97.79%	96.45%	95.39%
90%	95.24%	90.77%	97.17%	94.9%	93.56%
80%	94.88%	90.2%	97.04%	94.6%	93.21%
70%	94.55%	89.63%	96.88%	94.27%	92.67%
60%	**94.22%**	**88.89%**	**96.68%**	**93.82%**	**92.03%**
50%	93.69%	88.18%	96.41%	93.23%	91.43%
40%	93.11%	87.06%	96.17%	92.54%	90.55%
30%	92.33%	85.66%	95.8%	91.68%	89.39%
20%	91.21%	83.51%	95.16%	90.18%	87.74%
10%	89.23%	79.72%	94.22%	88.02%	84.9%

Table III shows the full training set has the highest accuracy among other various training percentage splits. However, having the full training set in the smart device application is not practical. The experiment had to come with different training percentage splits to be stored in the cloud or the application SD card [9].

The experiment extended to examine using the biometric measurements [1]–[3] which were the FAR and FRR in order to get the perfect training percentage split. Tables IV and V show the FAR and FRR rates.

TABLE IV: FAR rates.

Train %	D1	D2	D3	D4	D5
Full Set	7%	1%	4%	3%	1%
90%	11%	1%	6%	5%	2%
80%	11%	1%	6%	5%	2%
70%	12%	1%	6%	5%	2%
60%	**13%**	**1%**	**7%**	**5%**	**2%**
50%	14%	1%	7%	6%	3%
40%	16%	2%	7%	6%	3%
30%	17%	2%	8%	7%	3%
20%	20%	2%	10%	8%	4%
10%	24%	2%	12%	10%	5%

TABLE V: FRR rates.

Train %	D1	D2	D3	D4	D5
Full Set	2%	7%	1%	3%	6%
90%	3%	8%	2%	4%	8%
80%	3%	9%	2%	4%	9%
70%	3%	9%	2%	4%	10%
60%	**3%**	**9%**	**2%**	**4%**	**11%**
50%	4%	10%	2%	5%	11%
40%	4%	11%	3%	5%	12%
30%	5%	11%	3%	6%	14%
20%	5%	13%	3%	6%	16%
10%	6%	16%	4%	8%	19%

Tables IV and V show the rates of FAR and FRR increase as the training percentage splits decrease where the EER occurs [1], [3]. The perfect training percentage splits should have both the lowest FAR and FRR rates. Our experiment results concluded that 60% or above of the training percentage splits will have very excellent promised results based on the lowest FAR and FRR rates. Moreover, the accuracy rates of 60% of the training percentage split are as follows; D1 (94.22%), D2 (88.89%), D3 (96.68%), D4 (93.82%), and D5 (92.03%).

B. Discussion

1) Applications: The research focused on classifying users into demographic groups, and it was an extended investigation for smart device authentication systems. In fact, the approach was an excellent complementary mechanism for any user's classification based systems to improve the security and efficiency of the smart device's applications [5].

2) Limitations: Based on the study, there were several limitations. One of the limitations was Android APIs' restriction where some of them could be applied to some smart devices and not to others. The solution to this limitation may be determined and resolved in the future release of the new Android platform [8]. Another limitation was the regular mimic user behaviors issue [1]. The research did not address this issue and assumed that all users acted naturally and did not know any of our experiment application tasks. However, what would happen if the user mimicked the behavior and pretended to be someone else? What would happen if the user knew the application tasks already?

VI. Conclusion and Future Work

A. Conclusion

The investigation studied and modeled the three most common behavior biometrics and combined them into one model as the Multi-Fusions model. Later, the work used the classification users in the Multi-Fusions model by visualizing fusion features and finding their internal relationships. The work had to implement the data transformation and also the PCA to reduce the multi-dimensions. It used the decision tree classifier to classify the produced PCA components. The results of the experiment indicated that high accuracy was achieved for user demographic classification. Finally, we further discussed the applications and limitations.

B. Future Work

There are several significant potential improvements in the investigation. One of the improvements is to implement clustering, more classifiers and feature extraction algorithms. In fact, these concepts will open the opportunity to improve the methodology of the investigation [10]. Another potential improvement is with the data collection. As the data collection increases, the investigation will gain more validation results and possibly discover new users' demographic groups. Furthermore, data collection investigators may consider collecting data samples from multiple smart devices to capture distinct data.

Acknowledgment

This work was approved by SMU Research Compliance at Southern Methodist University in Dallas, TX.

References

[1] C.-C. Lin, C.-C. Chang, and D. Liang, "A novel non-intrusive user authentication method based on touchscreen of smartphones," in *Biometrics and Security Technologies (ISBAST), 2013 International Symposium on*. IEEE, 2013, pp. 212–216.

[2] R. Maxion, K. S. Killourhy *et al.*, "Keystroke biometrics with number-pad input," in *Dependable Systems and Networks (DSN), 2010 IEEE/IFIP International Conference on*. IEEE, 2010, pp. 201–210.

[3] 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," *Information Forensics and Security, IEEE Transactions on*, vol. 8, no. 1, pp. 136–148, 2013.

[4] K. W. Nixon, X. Chen, Z.-H. Mao, Y. Chen, and K. Li, "Mobile user classification and authorization based on gesture usage recognition," in *Design Automation Conference (ASP-DAC), 2013 18th Asia and South Pacific*. IEEE, 2013, pp. 384–389.

[5] J. D. Allen, J. J. Howard, and M. A. Thornton, "Method for subject classification using a pattern recognition input device," Oct. 22 2011, uS Patent App. 13/279,279.

[6] M. A. Thornton, "Keyboard dynamics," in *Encyclopedia of Cryptography and Security*. Springer, 2011, pp. 688–691.

[7] ASUS, "Asus memo pad 7 (me176c)," (Date last accessed 19-March-2015). [Online]. Available: http://www.asus.com/Tablets/ASUS_MeMO_Pad_7_ME176C

[8] R. Meier, *Professional Android 4 application development*. John Wiley & Sons, 2012.

[9] M. Owens and G. Allen, *SQLite*. Springer, 2010.

[10] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*. John Wiley & Sons, 2012.

[11] G. J. Williams, "Rattle: a data mining gui for r," *The R Journal*, vol. 1, no. 2, pp. 45–55, 2009.

[12] N. Panahi, M. G. Shayesteh, S. Mihandoost, and B. Z. Varghahan, "Recognition of different datasets using pca, lda, and various classifiers," in *Application of Information and Communication Technologies (AICT), 2011 5th International Conference on*. IEEE, 2011, pp. 1–5.

[13] I. H. Witten and E. Frank, *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann, 2005.