Body-Taps: Authenticating Your Device Through Few Simple Taps

Diksha Shukla[1], Guangcheng Wei[1], Donghua Xue[1], Zhanpeng Jin[2], and Vir V. Phoha[1]
[1]Syracuse University, Syracuse, NY 13244
[2]University at Buffalo, SUNY, Buffalo, NY 14260
{dshukla,gwei100,dxue01}@syr.edu, zjin@buffalo.edu, vvphoha@syr.edu

Abstract

To fulfill the increasing demands on authentication methods on the smart mobile and wearable devices with small form factors and constrained screen displays, we introduce a novel authentication mechanism, Body-Taps, which authenticates a device based on the Tap-Code gestures in the form of hand movements captured through the built-in motion sensors. The Body-Taps require a user to set a Tap-Code as an unlock code for the device by tapping the device on the set anchor points on his or her own body. The target device is authenticated based on two criterion: (1) the user's knowledge of the set Tap-Code, and (2) the Body-Tap gestures measured through the smart device's built-in motion sensors (accelerometer and gyroscope). Our experiments show that the proposed Body-Taps system can achieve an average authentication accuracy over 99.5% on a dataset comprising of 230 Body-Tap samples from 23 subjects, using Random Forest (RF), Neural Network (NNet), and Linear Discriminant Analysis (LDA) classifiers. Our work yields a light-weight, low-cost, and easy-to-use secure authentication system that requires minimal efforts and offers satisfactory usability.

1. Introduction

Smart mobile and wearable devices have seen incredibly dramatic growth during the past ten years. It was projected that approximately 310.4 million wearable devices were sold worldwide in 2017, which encompasses a variety of device types including smartwatches, body-worn cameras and head-mounted displays [22]. Along with this market surge, there have also seen increasing concerns about security and privacy of those personal devices. That is, how to properly protect and secure those devices against accidental and malicious access. Traditional methods of authentication on mobile devices using pins or passwords have a number of shortcomings. For example, pins or passwords can be stolen or leaked; complex pins or passwords are hard to remember and easy passwords are readily guessed, etc. Another major shortcoming is the difficulty of entering pins or passwords on traditional devices for visually challenged people. Recent work using hand and wrist movements for authentication on mobile or wearable devices has shown significant promise and overcome some of the difficulties. However, the authentication accuracy of those approaches are still not up to the mark. To this end, in this study we introduce a new, light-weight, user-friendly authentication technique based on a series of body taps, typically four to six body taps at key locations of the body, such as the shoulders and each side of the waist (see Figure 1). The proposed approach is specifically designed for those wearable and mobile devices with small form factors or constrained screen displays.

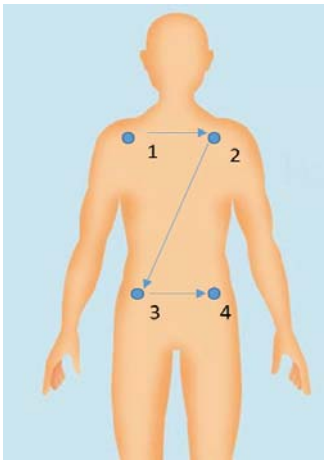


Figure 1: Body-Taps for Tap-Code $1- > 2- > 3- > 4$, which can be any combination of arm and wrist movements (indicated as arrows) between the body anchor Tap Points.

Figure 1 shows an example of wrist movements for entering Tap-Code $1- > 2- > 3- > 4$. This involves a sequence of movements; (1) tap the phone at the left shoulder ($Tap - Point1$), (2) tap the phone on the right shoulder ($Tap - Point2$), (3) tap the phone on the left side of the waist ($Tap - Point3$), and (4) tap the phone on the right

978-1-5386-7180-1/18/$31.00

Table 1: Anchor Points on the Users Body and the Corresponding Tap Code.

Body Anchor Point	Tap Code
Left Shoulder	Tap - $Point1$
Right Shoulder	Tap - $Point2$
Left Waist	Tap - $Point3$
Right Waist	Tap - $Point4$

side of the waist ($Tap - Point4$). Table 1 shows the anchor body points used in our design and the corresponding Tap-Code. Body-Tap gestures captured as the wrist movements by the motion sensors built within the smart devices provide us with a set of distinguishable features, which can be used for verifying and authenticating the identity of the user.

Our method thus provides two distinct modes of authentication: one consisting of a pin through body taps, and the second characterizing the movement patterns between body tap positions on the body captured through the built-in motion sensors (i.e., accelerometer and gyroscope). People may question and argue that the body taps can be observed easily and the accuracy of the motion behavior based authentication would not be sufficient to build a practical system. In response, we would like to point out that, (1) one can use different combinations of body taps, thereby providing different combinations of Body-Codes as the pins; and the purpose of body taps at different locations of the body is to provide a distinct sequence of flight patterns that involves wrist and arm movements horizontally, vertically, and diagonally. The security and vulnerability of pins or passwords per say are not the focus of this paper although they do provide a first layer of defense. (2) The discriminability of the accelerometer and gyroscope based authentication at its worst is comparable to those reported in the literature and we posit that it will provide an enhanced security level because of the anthropometric differences. Even if the height and weight of an impostor is similar to the victim, the arm length and the wrist movement involved in creating the Body-Taps will provide unique signatures. And, (3) our method provides a much safer option for visually impaired people who may traditionally have difficulty of entering pins or passwords or at worst may completely give up the pins or passwords.

It is argued that, because of the anthropometric variations, such as the geometric dimensions of the body, specifically the torso, arm length, wrist, even the shape of the hand and length of the fingers, of one individual from another, a rich set of features can be captured through the motion sensors (such as the accelerometer and gyroscope) when the subject holds a phone and taps and moves the hand (and wrist) between different parts of the body. Our results show that it is possible to successfully authenticate based on hand movement between body taps.

Moreover, as mentioned above, our method is also of significant use to visually impaired people, because it requires no explicit pin or password entries. In addition this method is particularly suitable for the stand-alone authentication scenarios on smart mobile and wearable devices with small form factors and constrained screen displays.

Contributions: Our work brings forth the following contributions to the field of mobile devices' security.

1. We proposed a new authentication mechanism, Body-Taps, which requires minimal efforts and is suitable for constrained screen devices. Design of the system is such that it only requires a user to tap the phone at key anchor points on his/her body. The system creates a user specific template from the features which capture unique movement of the user's phone movement for each tap and in between two taps. System asks the user to enter the previously set Tap-Code (training phase) for the verification to gain access to the device. If the matching score is higher than a set user specific threshold, access is granted otherwise access is denied.

2. We designed, implemented, and rigorously evaluated the Body-Taps system using an iPhone. In our experiments on a dataset of 230 Body-Taps samples from 23 subjects, our method could achieve an average authentication accuracy over 99.5% using Random Forest (RF), Neural Network (NNet), and Linear Discriminant Analysis (LDA) classifiers.

The rest of the paper is organized as follows. Section 2 presents the related work. The design of the experiments with data collection and analysis are described in Section 3. We present the performance of our authentication in Section 4. Finally, we draw our conclusions and discuss future directions for the research in Section 5.

2. Related Work

The majority of user authentication methods are composed of the characteristics that users possess, e.g., fingerprint, iris [1], palm print [1], gait [12, 23], and context based behavior analysis [14, 27], what users know such as pins, and passwords or a combination of both of the above that includes touch based authentication [2, 11], and speech recognition [1]. These methods have practical limitations as they require extensive computation power and can be very taxing for mobile phone devices. On the contrary, accelerometer and gyroscope based methods do not require excessive computation to authenticate a device.

Researchers have explored a plethora of motion sensors based authentication schemes that focus on improving the usability, performance, and memorability to authenticate a device [10, 16, 17, 21, 24, 26]. The methods based on motion gestures generally follow two different types of implementations: one relies on the analysis of the position of the

mobile device [6, 8, 13, 19], and the other focuses on the position of the user's hand [23]. Prior studies [8, 13, 18, 19] have used the accelerometer data in short sessions from either a customized device or a common Android device.

Kamil Burda [3] presented a new approach to authenticate users based on the way they picked up a smartphone on a table or in their front pockets, an activity performed frequently every day, using the smartphones accelerometer sensor. Conti *et al.* [4] proposed a new biometric measure to authenticate the user of a smartphone: the movement the user performed when answering (or placing) a phone call. Luca *et al.* [5] presented Back of Device Shapes, an authentication method for smartphones that uses the back of the device for input. Feng *et al.* [7] proposed two novel methods, a Statistic Method to intuitively apply the classifier on the statistic features of the data; and a Trajectory Reconstruction Method to reconstruct the Mobile Device Picking-up (MDP) motion trajectories and extract specific identity features from the traces. Kunnathu [15] attempted to build a statistical model to identify a user, based on how the user picked up the phone and how he/she held the phone to the ear. Lu *et al.* [20] proposed a finger-gesture-based authentication method, where the in-air-handwriting of each user was captured by wearable inertial sensors. Lee *et al.* [17] proposed Secure Pick Up (SPU), to authenticate the users, by implicitly observing the way they bend their arms when they pick up a smartphone to interact with the device.

3. Data Collection and Feature Analysis

3.1. Data Collection

With the approval of our university's Institutional Review Board (IRB), we collected experimental data from 28 volunteer participants. The participants were informed that their phone movement patterns would be collected while they entered the chosen Body-Tap-Code by tapping the phone at the set anchor points on their bodies. We designated the anchor points corresponding to a Tap-Code as shown in Figure 1. Table 1 details the body anchor points and corresponding Tap-Code.

All the participants in our data collection study were university students, faculty, or staff. We developed an iOS-based application to record the accelerometer and gyroscope readings. Data recording rate was set to 60 Hz. We collected data in three different sessions; (1) a pre-training session, (2) a training session, and (3) a testing session.

3.1.1 Pre-training Session

We collected a dataset comprising of 5 different users for a pre-training analysis. We asked each participant to create Body-Taps for a set of three fixed Tap-Codes. These Tap-Codes were designed to cover all the possible combinations of the tap points in our study, i.e., all possible combinations of taps 1, 2, 3, and 4. Following are the fixed Tap-Codes that we provided to the participants.

1). $1- > 2- > 3- > 4- > 3- > 2- > 1$
2). $1- > 3- > 1- > 4- > 2- > 4- > 1$
3). $1- > 1- > 2- > 2- > 3- > 3- > 4- > 4$

Every participant created Body-Taps corresponding to the given Tap-Codes, 10 times each. We recorded the accelerometer and gyroscope readings while user created the Body-Taps.

Table 2: Average and Standard Deviation of the Time Taken by the Participants to Enter Body-Tap-Codes.

Tap-Code Swing Type	**Time Taken** (sec)	
	Mean (μ)	**Std** (σ)
$Tap1\ to\ Tap1$ **or** $Tap2\ to\ Tap2$	1.11	0.14
$Tap3\ to\ Tap3$ **or** $Tap4\ to\ Tap4$	1.23	0.08
$Tap1\ to\ Tap2$ **or** $Tap2\ to\ Tap1$	1.40	0.17
$Tap3\ to\ Tap4$ **or** $Tap4\ to\ Tap3$	1.35	0.23
$Tap1\ to\ Tap3$ **or** $Tap2\ to\ Tap4$	1.69	0.27
$Tap3\ to\ Tap1$ **or** $Tap4\ to\ Tap2$	1.93	0.31

3.1.2 Training Session

We recruited 23 different volunteer participants to collect a training dataset. We asked each participant to chose a Tap-Code and create Body-Taps for the chosen Tap-Code (see Figure 1). Although we did not provide any direction for the length of the Tap-Code, majority (61%) of the participants chose a tap code of the length 4. Choice of Tap-Code of length 6 was the second highest majority (35%) in our dataset. The rest 5% participants chose the Tap-Code of other lengths. We believe that the observed Tap-Code lengths in our dataset are consistent with the users' preferences of choosing a pass code for their mobile devices. Please note that many mobile devices on the market use the standard four or six digit unlock codes.

3.1.3 Testing Session

We invited the same 23 participants, 3 to 5 days after their first participations in the training session, and asked them to create Body-Taps for the previously chosen Tap-Code. We again recorded the accelerometer and gyroscope readings as our testing dataset.

Each data collection session was preceded with a practice session with the goal to make participants familiarize with the device and the chosen Tap-Code. We explained the process of entering the chosen Tap-Code (i.e., creating the Body-Taps) and allowed each participant to practice entering Body-Taps for the chosen Tap-Code for 2-3 minutes before the recorded session.

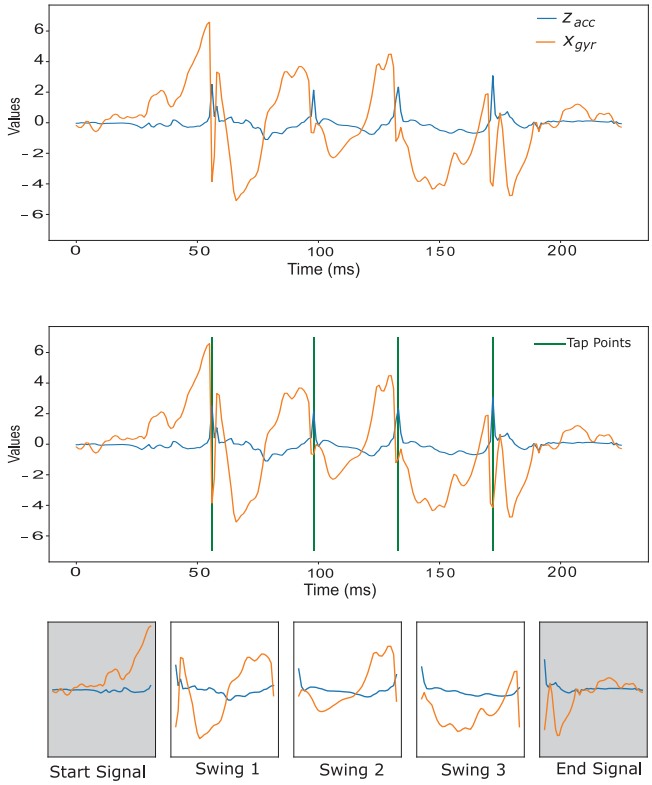


Figure 2: Tap Point Identification. Figure show the accelerometer and gyroscope signal components z_{acc} and x_{xyr} (top), vertical green lines show the selected peaks of the signals, identified as Tap-Points (mid), and different Swings separated into short windows (bottom). Start and End part of the signal (grey shadowed) is discarded for further processing and features extraction steps.

3.2. Data Preprocessing and Feature Analysis

3.2.1 Data Analysis

We collected the three-axis accelerometer ($x_{acc}, y_{acc}, z_{acc}$) and the three-axis gyroscope ($x_{gyr}, y_{gyr}, z_{gyr}$) readings from the phone while user entered the chosen Tap-Code. We computed the magnitude, m_{acc}, for the accelerometer and magnitude m_{gyr} for the gyroscope which we referred as the fourth component of the accelerometer and the gyroscope signals, respectively. m_{acc} is defined as, $\sqrt{x_{acc}^2 + y_{acc}^2 + z_{acc}^2}$ and m_{gyr} is defined as, $\sqrt{x_{gyr}^2 + y_{gyr}^2 + z_{gyr}^2}$.

The recorded signals were very noisy and hence we performed data smoothing. We used simple moving average (SMA) with a window size of 5 points for all four components of the accelerometer and gyroscope signals as a smoothening process.

Algorithm 1: Tap Point Identification Algorithm.

Input: Sensor Signals, $z_{acc}(t)$ and $x_{gyr}(t)$.
Output: $\{t_{TapPoints}[] \mid t_i, i = 1, 2, ..., (N-1)\}$, where N is the length of the Body-Tap-Code.

for *(if at time* t_i*,* $z_{acc}(t_i)$*, and* $x_{gyr}(t_i)$ *exist)* **do**
  if $z_{acc}(t_{i-1}) < z_{acc}(t_i)$ ***AND*** $z_{acc}(t_{i+1}) < z_{acc}(t_i)$ ***AND*** $x_{gyr}(t_{i-1}) < x_{gyr}(t_i)$ ***AND*** $x_{gyr}(t_{i+1}) < x_{gyr}(t_i)$ **then**
    $t_{Candidate}[] \leftarrow t_i$
  else
    //Do Nothing
  end
end
for $t_j \in t_{Candidate}[]$ **do**
  if $(t_{j+1} - t_j) < Threshold$ **then**
    $t_{TapPoints}[] \leftarrow max(t_j, t_{j+1})$
  else
    $t_{TapPoints}[] \leftarrow t_j$
  end
end
return $t_{TapPoints}[]$ //Final Tap Points

3.2.2 Tap Points Identification

We closely observed the phone tapping process by users in our pre-training dataset and noted that a tap consists of the following three action sequences: (1) the phone moves towards the body anchor point, (2) the subject taps the phone on the anchor point, and (3) the phone moves away from the body anchor point. The sequence of these three actions also possesses the unique behavior of movements as observed in our pre-training dataset. The sensor readings show a clear peak at the Tap Point. We also noted that the z_{acc} component of the accelerometer readings and the x_{gyr} component of the gyroscope readings were sufficient to locate the Tap Points accurately, although there were signal peaks at points other than the tap points. To identify the peaks corresponding to a tap, we applied a threshold-based elimination scheme. We observed the mean and standard deviation of the time taken between two consecutive taps and set a separate threshold as an average for each swing type. Table 2 shows the overall mean and standard deviation of the time taken in a swing. Based on the above observation, we developed a Tap Point Identification algorithm, Algorithm 1, to identify the tap points in our dataset. Figure 2 shows an example of the sensor signals, z_{acc}, and x_{gyr} corresponding to a Body-Tap-Code entry. It can be clearly seen in the figure that the sensor readings are at peak around the Tap Points. The vertical lines in green color are marked as the locations of the identified Tap Points.

Table 3: Features extracted from the accelerometer and gyroscope signals for each swing. Number in a cell is the Feature ID corresponding to the respective feature and signal component. Feature rank is given inside (.) next to Feature ID.

Feature Name	Feature Id for Signal Component							
	Accelerometer				Gyroscope			
	x_{acc}	y_{acc}	z_{acc}	m_{acc}	x_{gyr}	y_{gyr}	z_{gyr}	m_{gyr}
Mean	1 (36)	19 (21)	37 (16)	55 (113)	76 (29)	94 (31)	112 (23)	130 (143)
Median	2 (2)	20 (32)	38 (27)	56 (114)	77 (14)	95 (5)	113 (1)	131 (144)
Variance	3 (93)	21 (100)	39 (104)	57 (74)	78 (79)	96 (52)	114 (135)	132 (57)
Standard Deviation	4 (94)	22 (101)	40 (105)	58 (75)	79 (124)	97 (130)	115 (136)	133 (90)
Median Absolute Deviation	5 (95)	23 (67)	41 (106)	59 (76)	80 (125)	98 (131)	116 (137)	134 (145)
Inter-Quartile Range	6 (64)	24 (68)	42 (107)	60 (47)	81 (126)	99 (82)	117 (138)	135 (91)
Power	7 (96)	25 (69)	43 (108)	61 (115)	82 (40)	100 (83)	118 (139)	136 (146)
Energy	8 (97)	26 (102)	44 (109)	62 (116)	83 (127)	101 (132)	119 (140)	137 (147)
Peak to Peak Amplitude	9 (98)	27 (103)	45 (72)	63 (117)	84 (128)	102 (84)	120 (88)	138 (58)
Autocorrelation	10 (65)	28 (45)	46 (110)	64 (77)	85 (80)	103 (85)	121 (141)	139 (148)
Kurtosis	11 (44)	29 (38)	47 (111)	65 (48)	86 (51)	104 (86)	122 (54)	140 (149)
Skewness	12 (24)	30 (4)	48 (28)	66 (118)	87 (8)	105 (41)	123 (55)	141 (42)
Spectral Entropy	13 (25)	31 (46)	49 (112)	67 (119)	88 (81)	106 (133)	124 (89)	142 (150)
Median Frequency	14 (20)	32 (33)	50 (34)	68 (49)	89 (35)	107 (87)	125 (56)	143 (59)
Peak to RMS Ratio	15 (99)	33 (70)	51 (73)	69 (120)	90 (129)	108 (134)	126 (142)	144 (60)
Minimum	16 (66)	34 (3)	52 (6)	70 (78)	91 (30)	109 (17)	127 (11)	145 (61)
Maximum	17 (37)	35 (71)	53 (39)	71 (121)	92 (9)	110 (53)	128 (12)	146 (62)
Number of Peaks	18 (26)	36 (13)	54 (7)	72 (22)	93 (15)	111 (10)	129 (18)	147 (19)

3.3. Feature Extraction

We first divided the smoothed signal into swings based on the tap points identified in the previous step using Algorithm 1. Entering a body tap code of length N creates $N-1$ swings (see Figure 2). For example, a body tap code $1- > 2- > 4- > 3$ (length $N = 4$) would create the following three swings; (1) $1- > 2$, (2) $2- > 4$, and (3) $4- > 3$. In other words, the first swing is resulted from the tap point $Tap1$ to the tap point $Tap2$, the second swing is resulted from the tap point $Tap2$ to the tap point $Tap4$, and the third swing is resulted from the tap point $Tap4$ to the tap point $Tap3$. We discarded the portions of the signal before the first tap point and after the last tap point in a Body-Tap-Code entry. Figure 2 shows a data sample from a user corresponding to a Body-Tap entry and the process of dividing the signal into swings. First, it shows the whole signal for the accelerometer component z_{acc} in red color and for the gyroscope component x_{gyr} in blue color. Next, the vertical lines in green color (in the middle window) indicate the identified tap points. The bottom set of windows show the Tap-Code divided into different swings. Note that the first and last window (shadowed) is part of the Body-Tap entry process but is not part of any swing. We discarded the start and end signal for each Body-Tap entry signal.

We extracted 150 different features from each remaining window (i.e., swing), and assigned a unique Feature ID, F_{id},

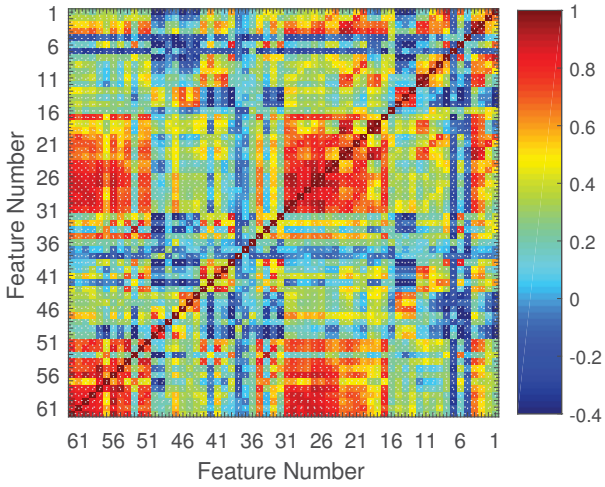


Figure 3: Correlation among the top 63 ranked features, which is referred as Subset $63F$ in the paper.

to each feature. Table 3 lists all the features and the corresponding F_{id}. For example, the feature with $F_{id} = 1$ is the mean of x_{acc}, the feature with $F_{id} = 67$ is the spectral entropy of m_{acc}, and the feature with $F_{id} = 112$ is the mean of z_{gyr}. Feature ID $F_{id} = 73, 74, 75$ correspond to the DTW between x_{acc} and y_{acc}, y_{acc} and z_{acc}, x_{acc} and z_{acc} signals respectively. Also, Feature ID $F_{id} = 148, 149, 150$ correspond to the DTW between x_{gyr} and y_{gyr}, y_{gyr} and z_{gyr}, x_{gyr} and z_{gyr} signals respectively. We refer to the features

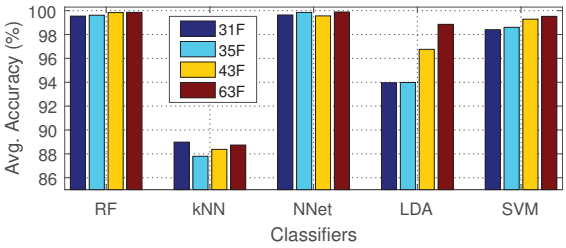


Figure 4: Average Classification Accuracy. $31F$ (blue) represents the average accuracy obtained using the subset of top 31 features, $35F$ (cerulean) using the subset of top 35 features, $43F$ (yellow) using the subset of top 43 features, and $63F$ (red) using the subset of top 63 features.

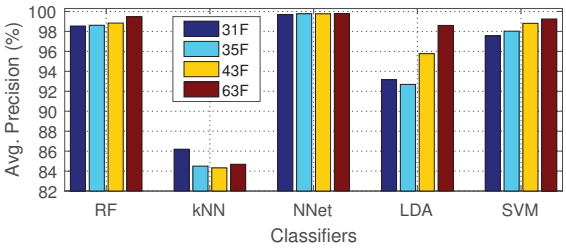


Figure 5: Average Precision Score

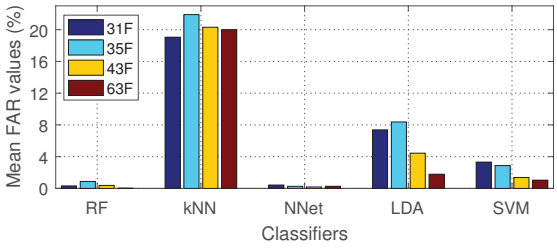


Figure 6: False Accept Rate (FAR) values

with their F_{id} in the rest of the paper.

3.4. Feature Ranking and Selection

We analyzed the extracted features and carried out a feature ranking and selection step so as to run our analysis with a compact but informative feature set. This step is also important to optimize the resource consumption in the target device to execute our authentication system on those devices as they often have limited resources.

We used the correlation (see Figure 3) based attribute evaluator to rank the 150 features in Table 3 and systematically select: (1) Feature Subset One ($F92$) – the top 92 features, (2) Feature Subset Two ($F63$) – the top 63 features, (3) Feature Subset Three ($F43$) – the top 43 features, (4) Feature Subset Four ($F35$) – the top 35 features, and (5) Feature Subset Five ($F31$) – the top 31 features. The method heuristically assigned a high score to a feature attribute subset which had high correlation with the class, and had low correlation with each other. Rank of

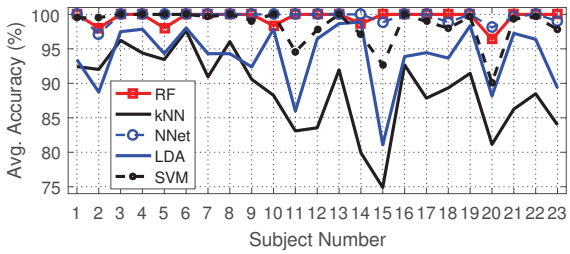


Figure 7: User Wise Accuracies obtained using feature subset with top 31 features $31F$.

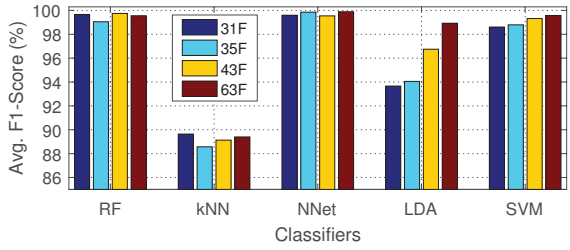


Figure 8: Average F1-Scores

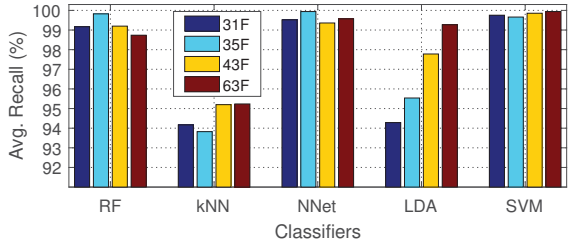


Figure 9: Average Recall Score

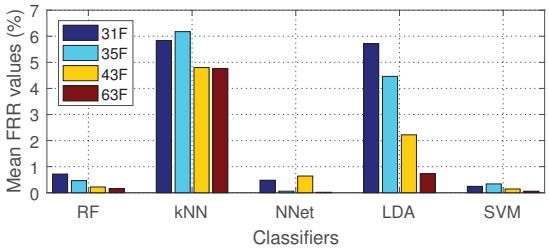


Figure 10: False Reject Rate (FRR) values

each feature is given in Table 3 inside brackets (.) next to their Feature IDs. For example, the feature with $F_{id} = 54$ was ranked 7, and the feature with $F_{id} = 125$ was ranked 56 by our feature ranking algorithm. DTW features with $F_{id} = 73, 74, 75, 148, 149, 150$ were ranked 50, 122, 123, 43, 63, 92 respectively.

4. Performance Evaluation

We used various classifier implementations from Weka [9] and R[25] to test the performance of our proposed user authentication model. We computed the classification accuracy while training the classifiers on our training samples

from the training dataset and supplied a test set from the testing dataset[1]. Figure 4 summarizes the overall classification accuracy obtained by our system using the Random Forest (RF), k-Nearest Neighbor (k-NN)[2], Neural Network (NNet), Linear Discriminant Analysis (LDA), and Support Vector Machine (SVM) classifiers. The figure shows the effects of the selected feature subsets on the performance of classifiers. Note that the RF, and NNet based authentication system has the best overall user classification accuracy in our dataset. Also, we observed that there is no significant change in the classification accuracy with a reduced feature set.

We also measured F-score, Precision, and recall for each of the system variations. Figure 8 shows the average F-Score obtained by the system with each feature subset and each classifier used. Figures 5 and 9 show the corresponding precision and recall values obtained by the system.

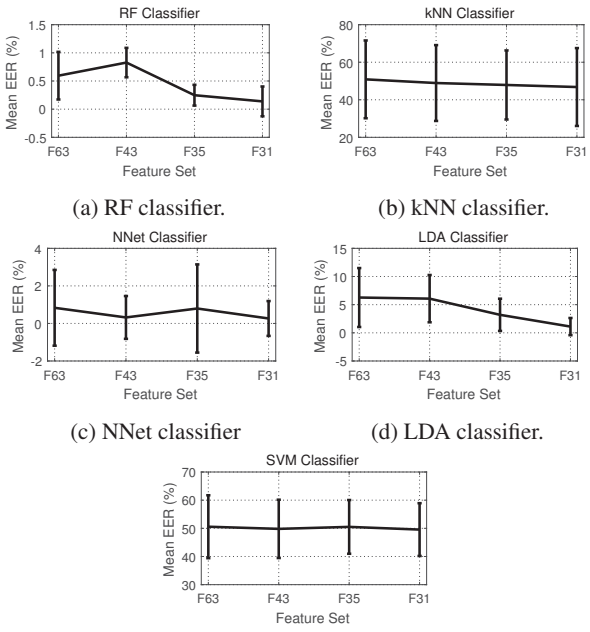
4.1. User-Wise Performance

We evaluated the system performance for each individual user in our dataset. We trained the classifiers using the genuine user samples from the training dataset while randomly selected the impostor samples from other users in the training dataset. For the testing samples, we used the similar approach and created the testing samples for each user. Figure 7 shows the user-wise accuracy obtained based on our dataset of 23 users. Note that for most of the users, the average accuracy obtained is higher than 99%. For more than 50% of the users, we obtained the 100% authentication accuracy using the RF, NNet, and SVM based system. The users #15 and #20 have relatively low average authentication accuracy with 75% and 82% respectively using LDA and kNN based systems. Also, the discussed users did not perform well with all 5 tested classifiers. We believe that these users might not be suitable for our authentication system well and can be removed from the system under failure to enroll policy.

4.2. Error Rates Evaluation

We evaluated our authentication system using the metrics of False Accept Rates (FARs), False Reject Rates (FRRs), and Equal Error Rates (EERs). Figure 6 shows the overall FARs for each classifier based system on each feature subset. Also, Figure 10 shows the corresponding FRRs of the system. Note that the average error rates are under 1% for RF, and NNet based systems which are best performing classifiers in our experiments.

Figure 11 show the mean EER and standard deviation in EER value using all five classifiers- RF, kNN, NNet, LDA, and SVM in our experiments. Note that the mean EER value lies between 0.13% to 0.8% for RF and NNet classifiers. kNN and SVM did not perform well in our experiments (see Figure 11(b) and (e)).



(a) RF classifier. (b) kNN classifier.

(c) NNet classifier (d) LDA classifier.

(e) SVM classifier

Figure 11: Mean EER and standard deviation in mean EER values using RF, kNN, NNet, LDA, and SVM classifiers.

5. Conclusion and Future Work

In this paper, we have introduced a motion sensor-based mobile device authentication model. The method leverages the unique phone movements corresponding to a user chosen Tap-Code. The method requires minimal efforts and provides security to the devices and is robust against traditional pin or password stealing attacks. Based on the results obtained from 230 Tap-Code samples of 23 different users, we have demonstrated that our model can achieve a high accuracy with minimal efforts required. Our model is of significant use for visually impaired users as well as for constrained screen devices where it is very challenging to enter the traditional pins and passwords on the screen. With the increasing use of these devices and nature of usage for sensitive transactions, our findings bring the area a step closer to ensure the security of these devices.

6. Acknowledgement

This work was supported in part by the National Science Foundation (NSF) under grants SaTC-1422417, SaTC-1527795, SaTC-1564046.

[1]The training and test set both comprise of an average of $\sim$ 200 balanced genuine and impostor samples

[2]The value of k was empirically set to 10 to get optimum classification accuracy in our experiments.

References

[1] M. Boatwright and X. Luo. What do we know about biometrics authentication? In *Proceedings of the 4th Annual Conference on Information Security Curriculum Development*, page 31. ACM, 2007.

[2] W. F. Bond and A. A. EA. Touch-based static authentication using a virtual grid. In *Proceedings of the 3rd ACM Workshop on Information Hiding and Multimedia Security*, pages 129–134. ACM, 2015.

[3] K. Burda. Authenticating users based on how they pick up smartphones. In *IIT. SRC 2016: 12th Student Research Conf. in Informatics and Information Technologies*, page 8, 2016.

[4] M. Conti, I. Zachia-Zlatea, and B. Crispo. 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 (ASIACCS)*, pages 249–259, New York, NY, USA, 2011. ACM.

[5] A. De Luca, E. Von Zezschwitz, N. D. H. Nguyen, M.-E. Maurer, E. Rubegni, M. P. Scipioni, and M. Langheinrich. Back-of-device authentication on smartphones. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2389–2398. ACM, 2013.

[6] M. Ehatisham-Haq, M. A. Azam, J. Loo, K. Shuang, S. Islam, U. Naeem, and Y. Amin. Authentication of smartphone users based on activity recognition and mobile sensing. *Sensors*, 17(9):1–31, 2017.

[7] T. Feng, X. Zhao, and W. Shi. Investigating mobile device picking-up motion as a novel biometric modality. In *2013 IEEE Sixth International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 1–6, Sept 2013.

[8] R. D. Findling, M. Muaaz, D. Hintze, and R. Mayrhofer. Shakeunlock: Securely unlock mobile devices by shaking them together. In *Proceedings of the 12th International Conference on Advances in Mobile Computing and Multimedia*, pages 165–174. ACM, 2014.

[9] E. Frank, M. A. Hall, and I. H. Witten. *Data Mining: Practical Machine Learning Tools and Techniques*, chapter The WEKA Workbench. Morgan Kaufmann, 4th edition, 2016.

[10] T. V. Goethem, W. Scheepers, D. Preuveneers, and W. Joosen. Accelerometer-based device fingerprinting for multi-factor mobile authentication. In *Proceedings of the International Symposium on Engineering Secure Software and Systems (ESSoS)*, pages 106–121, 2016.

[11] N. Z. Gong, M. Payer, R. Moazzezi, and M. Frank. Forgery-resistant touch-based authentication on mobile devices. In *Proceedings of the 11th ACM on Asia Conf. on Computer and Communications Security*, pages 499–510. ACM, 2016.

[12] C. C. Ho, C. Eswaran, K.-W. Ng, and J.-Y. Leow. An unobtrusive android person verification using accelerometer based gait. In *Proceedings of the 10th International Conference on Advances in Mobile Computing & Multimedia*, pages 271–274. ACM, 2012.

[13] F. Hong, M. Wei, S. You, Y. Feng, and Z. Guo. Waving authentication: your smartphone authenticate you on motion gesture. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*, pages 263–266. ACM, 2015.

[14] S. Kentros, Y. Albayram, and A. Bamis. Towards macroscopic human behavior based authentication for mobile transactions. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 641–642. ACM, 2012.

[15] N. Kunnathu. Biometric user authentication on smartphone accelerometer sensor data. *Proceedings of Student-Faculty Research Day, CSIS, Pace University*, 2015.

[16] W.-H. Lee and R. B. Lee. Sensor-based implicit authentication of smartphone users. In *Proceedings of 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pages 309–320. IEEE, 2017.

[17] W.-H. Lee, X. Liu, Y. Shen, H. Jin, and R. B. Lee. Secure pick up: Implicit authentication when you start using the smartphone. In *Proceedings of the 22nd ACM on Symposium on Access Control Models and Technologies (SACMAT)*, pages 67–78. ACM, 2017.

[18] B. Li, Q. Gui, H. B. Ali, H. Li, and Z. Jin. A wearable sit-to-stand detection system based on angle tracking and lower limb EMG. In *Proceedings of the IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, pages 1–6. IEEE, 2016.

[19] J. Liu, L. Zhong, J. Wickramasuriya, and V. Vasudevan. User evaluation of lightweight user authentication with a single tri-axis accelerometer. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*, page 15. ACM, 2009.

[20] D. Lu, K. Xu, and D. Huang. A data driven in-air-handwriting biometric authentication system. In *Proceedings of IEEE International Joint Conference on Biometrics (IJCB)*, pages 531–537. IEEE, 2017.

[21] R. Mayrhofer and H. Gellersen. Shake well before use: Authentication based on accelerometer data. In *Proceedings of the International Conference on Pervasive Computing*, pages 144–161, 2007.

[22] A. McIntyre, N. Ingelbrecht, and B. Blau. Forecast: Wearable Electronic Devices, Worldwide, 2017. Technical Report G00323691, Gartner, Inc., August 2017.

[23] M. Muaaz and R. Mayrhofer. An analysis of different approaches to gait recognition using cell phone based accelerometers. In *Proceedings of International Conference on Advances in Mobile Computing & Multimedia*, page 293. ACM, 2013.

[24] A. Primo, V. V. Phoha, R. Kumar, and A. Serwadda. Context-aware active authentication using smartphone accelerometer measurements. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 98–105. IEEE, 2014.

[25] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2013. ISBN 3-900051-07-0.

[26] H. Wang, D. Lymberopoulos, and J. Liu. Sensor-based user authentication. In *Proceedings of the European Conference on Wireless Sensor Networks (EWSN)*, pages 168–185, 2015.

[27] A. Wójtowicz and K. Joachimiak. Model for adaptable context-based biometric authentication for mobile devices. *Personal and Ubiquitous Computing*, 20(2):195–207, 2016.