IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 26, 2020, accepted July 4, 2020, date of publication July 13, 2020, date of current version July 21, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3008019

A Three-Step Authentication Model for Mobile Phone User Using Keystroke Dynamics

BALJIT SINGH SAINI[1], PARMINDER SINGH[1], (Member, IEEE), ANAND NAYYAR[2,3], (Senior Member, IEEE), NAVDEEP KAUR[4], KAMALJIT SINGH BHATIA[5], (Member, IEEE), SHAKER EL-SAPPAGH[6,7], AND JONG-WAN HU[8,9]

[1]School of Computer Science and Engineering, Lovely Professional University, Phagwara 144411, India
[2]Graduate School, Duy Tan University, Da Nang 550000, Vietnam
[3]Faculty of Information Technology, Duy Tan University, Da Nang 550000, Vietnam
[4]Department of Computer Science and Engineering, Sri Guru Granth Sahib World University (SGGSWU), Fatehgarh Sahib 140407, India
[5]Department of Electronics and Communication Engineering, Govind Ballabh Pant Institute of Engineering & Technology, Ghurdauri, Pauri 246194, India
[6]Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS), Universidade de Santiago de Compostela, 15782 Santiago de Compostela, Spain
[7]Information Systems Department, Faculty of Computer and Artificial Intelligence, Benha University, Banha 13518, Egypt
[8]Department of Civil and Environmental Engineering, Incheon National University, Incheon 22012, South Korea
[9]Incheon Disaster Prevention Research Center, Incheon National University, Incheon 22012, South Korea

Corresponding authors: Jong-Wan Hu (jongp24@inu.ac.kr) and Baljit Singh Saini (baljitsaini28@gmail.com)

This work was supported by a 2020 Incheon National University Research Grant.

ABSTRACT The use of keystroke dynamics for user authentication has evolved over the years and has found its application in mobile phones. But the primary challenge with mobile phones is that they can be used in any position. Thus, it becomes critical to analyze the use of keystroke dynamics using the data collected in various typing positions. This research proposed a three-step authentication model that could be used to authenticate a user who is using the mobile in sitting, walking, and relaxing position. Furthermore, the mobile orientation (portrait and landscape) was considered while taking input from the user. Apart from using traditional keystroke features, accelerometer data were also combined for classification using Random Forest(RF) and K-Nearest Neighbour(KNN) classifiers. The three-step authentication method was able to authenticate a user with an EER of 2.9% for the relaxing landscape position. Finally, the model was optimized using Particle Swarm Optimization (PSO) to reduce the feature set and make the model more practical for mobile phones. Optimization helped to reduce the number of features from 55 to 17 and improved the EER to 2.2%. The research validated that relaxing and walking positions are the best positions to authenticate a user using keystroke dynamics.

INDEX TERMS Three-step authentication, optimization, particle swarm optimization, random forest, particle swarm optimization (PSO).

I. INTRODUCTION

Mobile phones have become part of our daily life. We start our day by looking at our social media/email notifications on the mobile phone. Most of the online transactions that a person does be it online shopping or bill payment happens via mobile phones. Most of the personal user data like passwords, credit card numbers, etc. are saved on mobile phones. Thus, data security becomes one of the prime issues. For example, On 3rd October 2017, Yahoo revealed that in August 2013, the data breach of Yahoo had affected over 3 billion user accounts [1]. Recent studies [2], [3] reveal that users PINs and passwords are stolen by hackers. To prevent data breaches from intruders, digital resources (data, passwords, etc.) are protected by using a process called authentication. The authentication methods can be classified into three broad categories: knowledge-based, token-based, and biometric-based. The three techniques are summarized in Figure 1.

Biometric based authentication is based on "something you are". Biometrics is one of the safest and most reliable ways of authentication as it is the characteristic of the user only. The most popular biometric methods used are finger scan [4], [5], retina scan [6], face scan [7], [8], keystroke dynamics [9], [10], etc. Finger scan is the most popular biometric being used in mobile phones. In recent times face

The associate editor coordinating the review of this manuscript and approving it for publication was Theofanis P. Raptis.

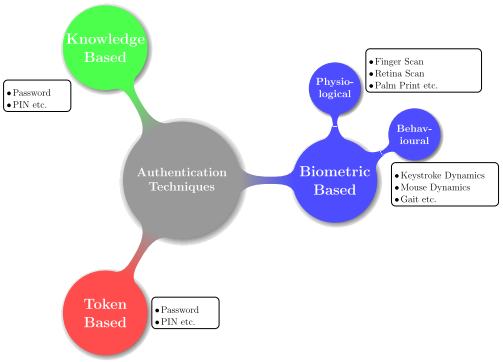


FIGURE 1. Authentication techniques.

recognition in mobile [11], [12] is fast becoming popular. But all these methods come at an increased cost. Keystroke dynamics is a behavioral metric that can be used as an alternate to the physical biometric methods like face and finger can. Keystroke dynamics authenticate a user based on his/her typing pattern. Typing is an integral part of using a mobile phone. Thus, no extra hardware is required which will help lower the cost of the device.

Keystroke dynamics is a behavioral biometric and since our behavior changes under different situations, so will the typing pattern. When we are sitting, we have a complete and static view of the mobile. When we are walking the position of the mobile is not static. Since the walking path may not be regular (smooth), the mobile will shake, and the typing pattern will change. Also while walking the concentration of the user is not entirely towards typing as he is also concerned about not colliding into other walkers. Similarly, while using a mobile device in a relaxed position (like lying on the bed/sofa), the pattern may change due to the change in orientation of the mobile phone or laid-back attitude of the user. Thus, it becomes critical to analyze the typing pattern in these different situations and to come up with a uniform authentication model that can be used for user authentication under any situation.

Majority researchers conduct their study by taking the input from the user in sitting position [13]–[15] only. But some of the recent studies [16], [17] considered taking the input from the user in different positions like sitting, walking, and standing. Roh *et al.* [18] collected the data from users while they were sitting with their hands placed on the table and while they were walking. Shen *et al.* [19] and Lamiche *et al.* [15] used a hand-hold-walk scenario to operate the smartphone to enter the passcode while they were walking. However, based on the literature review done to our best efforts no study so far has analyzed the user typing behavior in the above mentioned three different positions, i.e., sitting, walking, and relaxing. Further, the variation in typing patterns based on the orientation of the phone has also not been considered. With the use of mobile phones, additional features like the angle of holding the mobile (motion feature) can also be considered for user authentication.

This study proposes a novel three-step model of user authentication in three environments, i.e., sitting, walking, and relaxing, and holding the phone in both orientations, i.e., portrait and landscape. In total, this three-step authentication model will be able to authenticate the user in six positions sitting-portrait, sitting-landscape, relaxing-portrait, relaxing-landscape, walking-portrait, walking-landscape. The developed model is used for one-time authentication when the user will login by typing the given password. The model is further optimized using PSO to reduce the number of features required to build the user profile.

Thus, the main contribution of this paper are:

- Development of a three-step model for user authentication applicable in three typing positions - sitting, walking and relaxing.
- Feature subset optimization using PSO.

The scope of the study is limited to analyzing the typing pattern of an individual in three different situations only: sitting, walking, and lying. Also, the user needs to be familiar with the use of mobile devices and especially typing. The mobile device to be used for taking input from the users will be a touch device and not hard keypad based. The input from the user will be a fixed-length password, and each user will work in an uncontrolled environment. Since, a lot of external factors like emotion [20], [21], physical health, fatigue [22] etc. can impact the typing rhythm of the user, hence the user was asked to overcome these factors explicitly on their own.

The organization of the remaining paper is: Section II reviews the baseline research and the latest work done. Section III introduces the methodology used for data collection and feature extraction. Section IV presents the proposed model and discusses the outcomes of the analysis. Section V details the results after optimization. Section VI lists the limitations of the study and section VII summarises the paper and discusses the future scope of the work.

II. LITERATURE REVIEW

Keystroke dynamics is a major area of research due to its low implementation cost as compared to other biometric methods like finger scan and facial recognition which involves costly hardware. Just like any other biometric system the first phase in the keystroke dynamics system is data collection. The input text from the user can be *static* [23] or *dynamic* [15]. Static text is predetermined. But in the dynamic or free text, the user may enter different text during the authentication phase than that entered during the enrolment phase.

Once the input is decided, and the data is collected the next step is to extract features. In keystroke dynamics, the system can record the timings of a key event. Two timings are recorded - the time of key press and time of key release. Based on the difference between these two timings for the same or consecutive keys certain features are extracted. The most commonly used feature is latency [13], [24], [25]. *N-graph* [26] feature is also based on key timings which have been used by researchers. *Key hold time* (the period for which a key is pressed) also known as *dwell time* is another pop-

ular time-based feature used very rigorously in past studies [27], [28]. These features extracted from the keypress events are termed as keystroke or touch features.

Certain features specific to mobile phones are used in collaboration with the touch features for better profile building of the user. The force by which a key is pressed (pressure) and the amount of screen touched by finger (size) were used as features in [14] and [29]. Meng *et al.* [30] used the speed of touch in each direction and the time of every single touch along with keystroke features for user authentication using mobile phones.

The use of keystroke dynamics in mobile phones also opened the use of sensor data along with aforesaid features. The sensor data include the data collected from mobile sensors like accelerometer, gyroscope, and geomagnetic sensors. The features extracted from these sensors are termed as motion features.

Corpus *et al.* [31] compared the use of keystroke features and motion features. They observed that the accuracy of the system improved from 49.44% to 61.11% when both these features were used together as against the use of only keystroke features. Lee *et al.* [28] combined motion data derived from an accelerometer with keystroke data extracted from a six-digit PIN to get an EER of 7.789% using one-class Support Vector Machine (SVM) classifier. This study further using the opposite gender to act as imposter reduces the FAR. A new trimmed mean feature selection method on features extracted from motion and touch data was proposed by Kim *et al.* [32]. They got the best EER of 13.44% when they excluded the lowest 50% rated features. Wu and Chen [33] conducted a study on features based on time, acceleration, pressure, size, and orientation of the mobile device. The input data used was four six-digit PINs. A combination of all the five features resulted in the best result of 99.13% accuracy using SVM. Lamiche *et al.* [15] combined keystroke and gait features derived from accelerometer readings to achieve an accuracy of 99.11% using Multi-Layer Perceptron (MLP) classifier. Giuffrida *et al.* [34] developed a sensor-based authentication system called ''UNAGI''. Gyroscope and accelerometer data was used along with keystroke data generated from fixed password input. They achieved an EER of less than 1%.

Sitová *et al.* [16] used the hand movement, grasp, and orientation features for authenticating a user in two different positions: walking and sitting. In both cases, different values of EER were observed: 10.1% for sitting and 7.2% for walking, which meant that the user typing profile changes with change in position. In another study, [18] used accelerometer and gyroscope data along with other features like key-stamp and tab size for mobile user authentication in walking and sitting position (sitting while putting table on phone). The EER in both cases was 6.39% and 10.81% respectively. Takahashi *et al.* [35] combined motion features along with flick inputs. The user was asked to provide the data in three different positions: sitting inside a car, sitting otherwise, and walking. In another study, Shen *et al.* [19] compared three scenarios: holding the phone in hand, holding the phone in hand while sitting on the table, and holding the phone in hand while walking. The FAR achieved for the three positions was 5.01%, 7.85%, and 10.95% respectively while the FRR achieved was 6.85%, 9.27%, and 13.12% respectively. The input used was a user-chosen password of varying length (8-16 characters). A User authentication framework based on touch and motion features was proposed by Bo *et al.* [36]. The features extracted were pressure, area, position, velocity, and acceleration. Accuracy of around 99% was achieved under a controlled environment.

The study [37] focused on determining the location of the phone before authenticating the user. The study also used accelerometer data to determine the position of the phone (held in a pocket or held in hand) with an accuracy of 80%. Buriro *et al.* [38] proposed a bi-modal biometric authentication solution ''Touchstroke'' based on sensors like magnetometer, gyroscope, and accelerometer. User data was collected in different positions: standing, lying, walking downstairs and upstairs, and simply walking. The best FAR achieved was.02%,.03%,.02%,.03% and.03% for each of the five positions respectively. A study close to our proposed method was done by Crawford and Ahmadzadeh [17]. The authors proposed a two-phased model for user authentication. Gyroscope data was used to determine user position in the first phase with an accuracy of 97.3%, 91.5%, and 92.2% for sitting, standing, and walking positions respectively. In the second phase, the user was authenticated based on sitting, standing, and walking positions with an accuracy of 97.3%, 97.7%, and 97.7% respectively.

After feature extraction and template creation, classification algorithms are used to identify users. Some of the commonly used classification algorithms are RF [39], [40], KNN [19], SVM [41] etc. Optimization of the feature set is a crucial aspect as it enhances system performance. Optimization techniques like Ant Colony Optimization [42], Particle Swarm Optimization [43], and Firefly Algorithm [44] has been successfully used by researchers to enhance their keystroke dynamics systems. Shanmugapriya and Ganapathi [43] achieved an accuracy of 97.02% using PSO and were also able to reduce the training and testing times in comparison against the neural network, while Lee *et al.* [45] were able to reduce the EER to 6% when a subset of features was used as against the use of all features.

A. RELATED TERMINOLOGY

The effectiveness of any biometric system is measured via the use of error metrics namely, False Acceptance Rate (FAR), False Rejection Rate (FRR), and Equal Error Rate (EER).

FAR is the percentage of incorrect acceptances defined in Eq. 1, i.e., how many imposters were accepted by the system considering them as a genuine user.

$$FAR = \frac{No.\ of\ incorrect\ acceptances}{Total\ login\ attempts\ by\ imposter} * 100 \qquad (1)$$

FRR is the percentage of incorrect rejections defined in Eq. 2, i.e., how many times a genuine user got rejected by the system considering him as the imposter user.

$$FRR = \frac{No.\ of\ incorrect\ rejections}{Total\ login\ attempts\ by\ genuine\ user} * 100 \quad (2)$$

To measure FAR and FRR a threshold value is set. This threshold value determines the point after which a genuine user will be considered as an imposter. Having a strict value for threshold will result in genuine users being rejected by the system while having a lenient value for threshold will result in imposters being accepted as a genuine user. Thus, there is a trade-off that gives rise to another error metric called EER. EER is the point at which the FAR and FRR become equal. Lower the value of EER, higher is the system performance. The relationship between the three metrics is defined in Figure 2

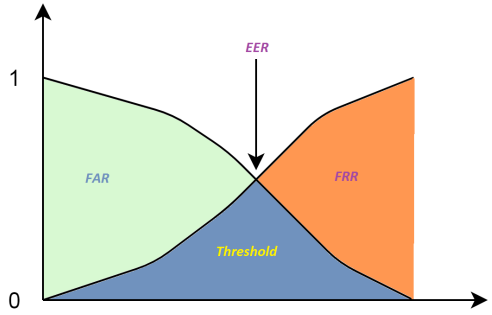


FIGURE 2. Relationship between FAR, FRR and EER.

Accuracy is another common metric used to measure system performance. It simply is the number of correct observations (genuine or imposter) made out of the total attempts. Higher the value of accuracy, the higher is the system performance.

III. THREE-STEP AUTHENTICATION MODEL

The proposed model is depicted in Figure 3. This model proposes to break the entire authentication process into three steps.

- Step 1 identifies the orientation of the device
- Step 2 identifies the user typing position for the identified orientation
- Step 3 creates a classification model for each of the three positions

The orientation of the phone is identified by using the physical sensors of the device which help to measure the physical position of a device. These position sensors comprise of orientation sensors and magnetometers. To identify the user position, the position in question is considered to be a positive class, and the other two positions are taken as a negative class. After this step one of the six possible positions - sitting-portrait, sitting-landscape, relaxed-portrait, relaxed-landscape, walking-portrait, or walking-landscape of the user will be identified. Once the user position is identified the last step is to authenticate the user. During the enrolment phase, a template for each of the six positions will be built and saved in the database. During authentication based on the position identified a new template would be built and matched with the one already stored corresponding to the identified position. If the two matches the user will be accepted as an authentic user else will be rejected as an imposter.

A. DATA COLLECTION

Since no earlier study has collected the input from the user in a relaxing position and also the mobile phone orientation has not been considered, a new data set was developed for this study. A mobile phone application was created for data collection. Data for this research was collected from 40 users. The mobile app was installed on each user's phone. The users were asked to use the app at different times during the day (morning, afternoon, evening, etc.) as per their convenience. This method was adopted so that the data collected is close to a real-world scenario where the user's behavior is uncontrolled. The input text used was ".tie5Roanl" which is considered to be a strong password [23]. Each user gave 30 input in every session. A simple password is not used as they are easy to impersonate [46]. It is also difficult to distinguish between users if simple passwords are used as compared to using a strong password like ".tie5Roanl" which has a mix of characters. The application required the user to enter the password again from the beginning if they made a typing error. The user entered the data in three different postures, i.e., sitting position, relaxing (lying on the bed, sofa, etc.), and while walking. The data from the user was taken in both landscape and portrait mode. 150 data entries were taken for every single position thus resulting in a total of 900 entries from each user. Figure 4 shows the snapshot of the developed application for collecting the input in sitting position and portrait orientation.

For every correct input provided by the user the following information was stored:

- SNO - the serial number that uniquely identifies each entry of the user.
- Name - Name of the user as entered during sign-up. This is required for preparing the data for training and testing purposes.
- E-mail - The email id of the user as entered during sign-up.
- Age - The age of the user as entered during sign-up.
- Hand Type - This tells whether the user is left-handed or right-handed. This information is also gathered during sign-up.
- SID - It is the session identifier. This helps to identify how many different sessions the user used to provide the entire data.
- Date - This value contains the day, month, and year of each session.
- Time - This value contains the start time of each session.
- Body Position - This field tells about the position in which the user was (sitting, relaxed, or walking) when the entry was done.

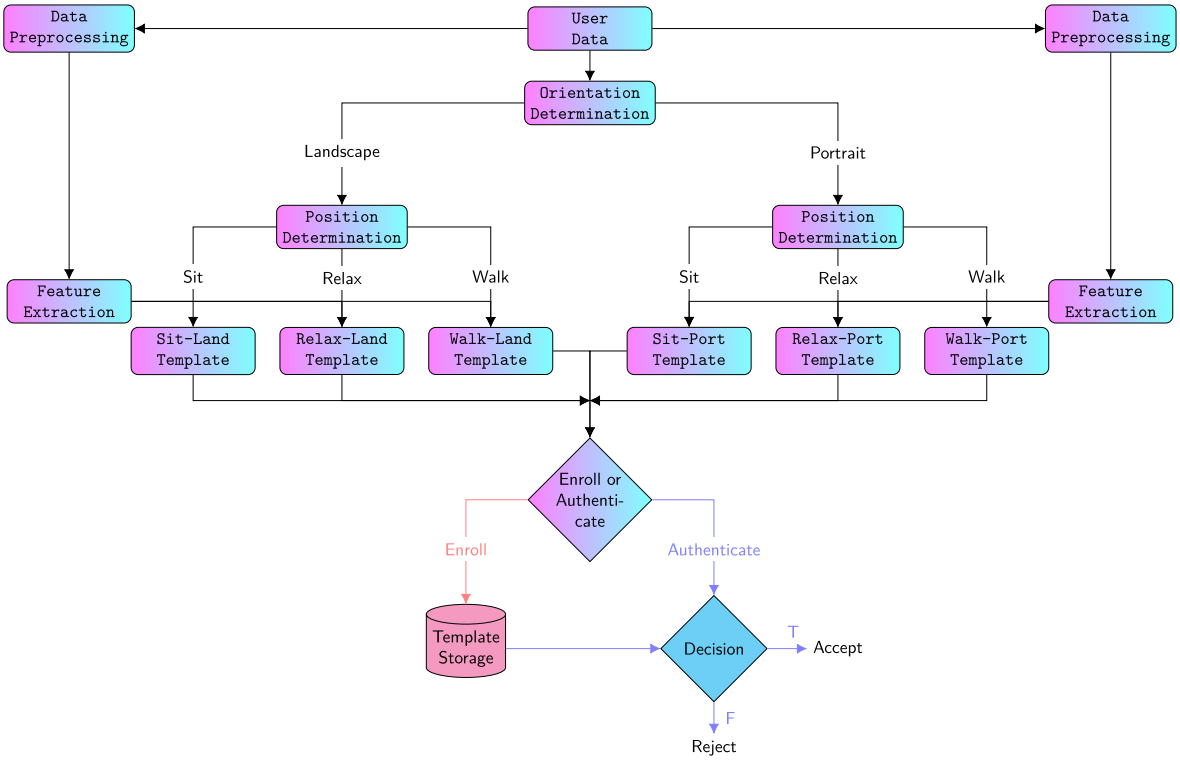


FIGURE 3. Proposed three-step authentication model.

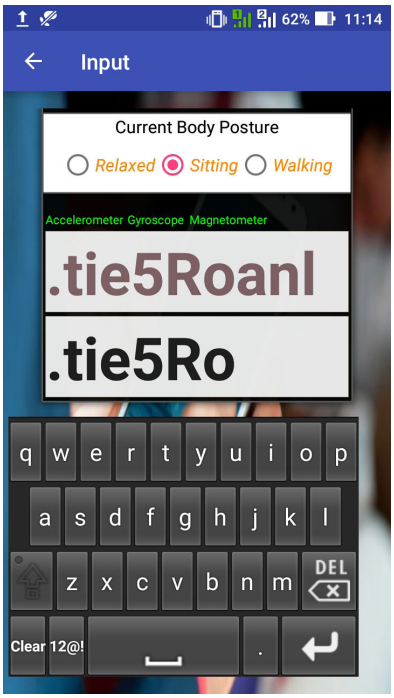


FIGURE 4. Application snapshot in sitting-portrait position.

- Orientation - This field tells the phone orientation, i.e., whether the entry was typed in landscape mode or portrait mode.
- Press time (Ptime) - This is the time (in msec) when a particular key was pressed, e.g. tPtime represents the press time for "t".
- Release time - This is the time (in msec) when a particular key was released, e.g., tRtime represents the keypress time for "t".
- X coordinate value (accelerometer) - This reading consists of a series of values of x co-ordinate which are for the entire duration of a particular input
- Y coordinate value (accelerometer) - This reading consists of a series of values of y co-ordinate which are for the entire duration of a particular input
- Z coordinate value (accelerometer) - This reading consists of a series of values of z co-ordinate which are for the entire duration of a particular input

The participants were proficient mobile users, and hence no training was provided for using the mobile application. All users did not provide 150 inputs for each of the six positions. There were certain other inconsistencies in data like giving the majority of the inputs in one session (only 20-30 inputs per session were required) and having two many outlier values in the data. After accurate screening data of only 36 users were considered for evaluation purposes.

B. FEATURE EXTRACTION

Touch data and motion data were used to extract the features for authentication. Touch data refer to the data collected when a key on the keypad on the touchscreen is touched.

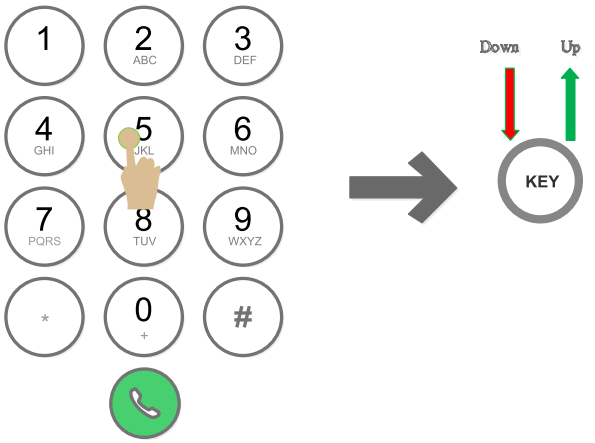


FIGURE 5. Touch data collection.

Figure 5 shows this data collection process. The *Down* means that the finger is touched on the screen and the *Up* means that the finger is lifted off the screen.

The touch data features can be divided into hold time (HT) and latency.

- Hold time: is the duration for which a key is pressed by the user i.e., the time interval between the key Up and the key down event for any key.
- Latency: From two consecutive keys, four data values are generated.
  1) Press-press latency: is the time interval between the key down of the previous key to the key down of the next key.
  2) Press-release latency: is the time interval between the key down of the previous key to the key up of the next key.
  3) Release-press latency: is the time interval between the key up of the previous key to the key down of the next key.
  4) Release-release latency: is the time interval between the key up of the previous key to the key up of the next key.

Let there be two keys pressed: k_i and k_{i+1}, and Dk_i be the key down time for key k_i,Uk_i be the key up time for key k_i, Dk_{i+1} be the key down time for key k_{i+1}, and Uk_{i+1} be the key up time for key k_{i+1}, Then, the above five touch features can be expressed as:

$$Hold\ time(HT) = Uk_i - Dk_i \tag{3}$$

$$Press - press\ latency(PP) = Dk_{i+1} - Dk_i \tag{4}$$

$$Press - release\ latency(PR) = Uk_{i+1} - Dk_i \tag{5}$$

$$Release - press\ latency(RP) = Dk_{i+1} - Uk_i \tag{6}$$

$$Release - release\ latency(RR) = Uk_{i+1} - Uk_i \tag{7}$$

Considering the input ".tie5Roanl" a total of 46 features were extracted. 10 characters in the input result in 10 HT. 9 pairs of characters result in 9 latency of each type, thus, a total of 36 latency.

Motion data refers to data collected due to movement in the phone position while typing. In other words, it refers to the angle of holding the mobile phone. The value of motion data is measured in terms of x, y, and z-axis. As depicted in Figure 6 the right and left direction of mobile is represented by the x-axis, the down and up the direction of mobile is represented by the y-axis and the back and front direction of mobile is represented by z-axis. To record this data accelerometer sensor was used.

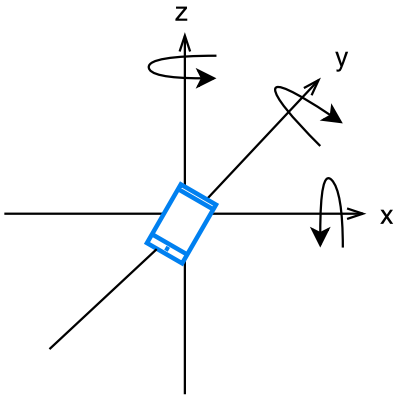


FIGURE 6. Axis configuration.

The motion features are extracted from the X, Y, and Z coordinates of the mobile. For every input, the user holds the mobile phone for quite some time which results in a vector of values for each coordinate. Thus, for doing the analysis the mean, root mean square (RMS), and standard deviation of each directional co-ordinate and for every single input were calculated using the Eq. (8), Eq. (9), and Eq. (10) respectively.

Given n data values for any direction (assume x-axis), the formulas used to drive the motion features are:

- Mean:

$$\mu(x) = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{8}$$

- RMS:

$$RMS(x) = \sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2} \tag{9}$$

- Standard Deviation:

$$SD(x) = \sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i - \mu(x)} \tag{10}$$

In total, every user data resulted in 55 features as shown in Table 1.

TABLE 1. Comprehensive list of features (Touch + Motion).

Touch					Motion		
HT	PP	PR	RP	RR	Mean	RMS	SD
10	9	9	9	9	3	3	3

C. OUTLIER REMOVAL AND NORMALIZATION

After the acquisition, the data were partitioned according to position and orientation. The data of each user was divided into six categories for processing. Every user provided the input data without any supervision. Under such an uncontrolled environment the user sometimes becomes complacent and this can result in inconsistent values in the data. Such inconsistencies in the data can impact the results in a negative manner. Consider a situation where the user while typing the data in a walking-landscape position stumbled across a stone and he paused for a second to regain his composure to type again. Thus, outlier removal becomes very important and this leads to improved data quality [47]. WEKA [48] tool was used for outlier removal. The outlier values were removed from the data using the ''InterQuartileRange'' filter. This filter is used for detecting outliers and extreme values based on interquartile ranges. The outliers are calculated as per Eq. 11 or Eq. 12:

$$Q3 + OF * IQR < x <= Q3 + EVF * IQR \tag{11}$$

or

$$Q1 - EVF * IQR <= x < Q1 - OF * IQR \tag{12}$$

The Extreme values are calculated using the Eq. 13:

$$x > Q3 + EVF * IQR \tag{13}$$

Similarly, for Z-Score normalization was done using ''Standardize'' filter. Removing the outliers left the data entries for each user to be different. In order to have a consistency of data for analysis, for every user, 100 entries were chosen for each position.

The typing pattern of any user can vary with time. Even the user can exhibit a variation in the typing pattern from session to session. There can be various reasons for this change: stress, mood change, state of mind, physical conditions, etc. So, it becomes important to normalize the data. Data is normalized to a range between 0 to 1. Z-Score normalization [49] method was used for normalizing the data. Z-Score method changes all values to a common scale having an average 0 and standard deviation 1.

$$Z = \frac{x - \mu}{\sigma} \tag{14}$$

where, x is the data point, μ is the mean and σ represents the standard deviation

1) DATA TRAINING AND TESTING

The data for analysis for each user was prepared by combining the 100 entries of the genuine user with 50 entries from imposters. Then the data was into a ration of 7:3, where 70% of the data was used for training and 30% was used for testing.

D. DATA CLASSIFICATION

Classification is the technique of assigning a new input to a set of categories, the basis of which is an existing data (training data) whose categorization is already known. It is a supervised learning technique in which an existing training set of correctly identified instances is available [50]. Any algorithm used to classify an input to a category is called a classifier. The classification of data was done using Random Forest and KNN classifiers.

Random Forest is a decision tree-based ensemble learning technique that can be used for regression as well as classification. It creates multiple trees during the training period and gives the mode class as an output during classification. It is a supervised learning method that works by building an ensemble of decision trees. The advantage of using Random Forest is that it does not suffer from overfitting and it can manage missing values. KNN is a non-parametric technique that can also be used for both classification and regression. The input provided is the k closest training samples from the set of all samples and it generates the class membership as an output during classification.

WEKA tool was used for analysis, and both the classifiers were available in WEKA. The default implementation of both these classifiers was used with the value of the parameters as described below:

Parameter values for Random Forest:

- numIterations - 500
- seed - 1
- maxDepth - 0 (means unlimited depth)
- bagSizePercent - 100 (means 100% training data is used)
- batchSize - 100

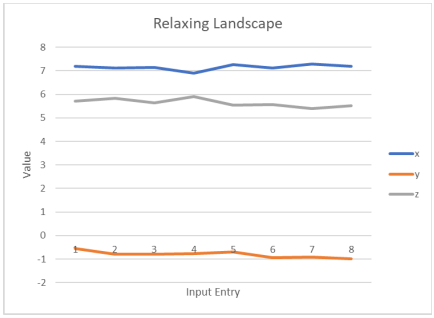
Parameter values for KNN:

- k - 7 (number of neighbours)
- batchSize - 100
- nearestNeighbourSearchAlgorithm - LinearNNSearch
- windowSize - 0

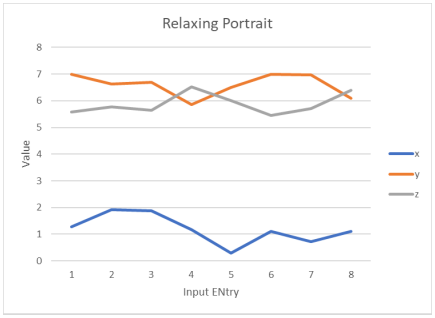
The output from WEKA is used to interpret the values of FAR and FRR. WEKA produces an output file which contains different readings for FAR and FRR at different threshold. Next, this data was fed into an ''R-Language'' script which calculated the EER i.e., the threshold at which FAR is equal to FRR. This was done for each user individually. Since, it's an authentication model which will be deployed in practical on individual user's mobile device so the system will be trained for each user separately and the system can set a different threshold value for that particular user.

IV. RESULT AND ANALYSIS

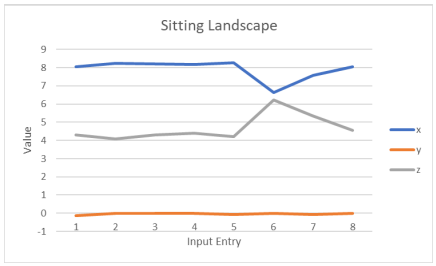
The orientation of the device can be detected by the use of inbuilt sensors in mobile phones. Thus, no separate experimentation was performed for this. The second step of the model is to determine the user typing position. Here, a similar approach as followed by [17] was adopted. For position recognition the data of individual user is considered. The data of one position is compared against each of the other positions. In general, we observe that when we type in portrait and landscape mode the angle at which the mobile is held changes. In a similar manner typing in a different position the angle of holding the mobile is also different.



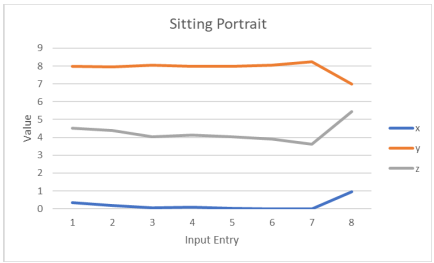
(a) Relaxing-Landscape



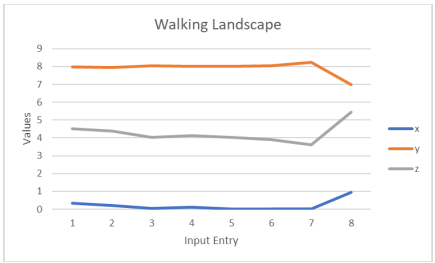
(b) Relaxing-Portrait



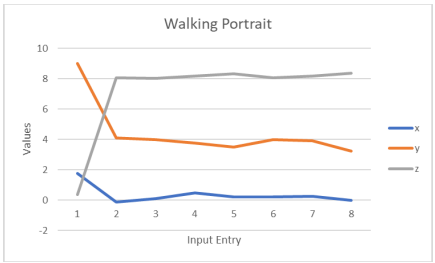
(c) Sitting-Landscape



(d) Sitting-Portrait



(e) Walking-Landscape



(f) Walking-Portrait

FIGURE 7. Plot for motion values of User 17.

TABLE 2. Error rates with motion features for position identification.

		Random Forest			KNN		
User Position	Mobile Orientation	FRR (%)	FAR (%)	EER (%)	FRR (%)	FAR (%)	EER (%)
Relaxing	Portrait	4.5	5.9	5.0	6.5	4.8	5.2
Sitting	Portrait	18.1	10.5	13.3	19.9	16.4	17.4
Walking	Portrait	19.4	4.4	8.7	20.2	6.3	11.8
Relaxing	Landscape	7.5	4.8	5.2	10.2	4.3	6.1
Sitting	Landscape	11.9	6.4	8.1	13.9	10.2	11.4
Walking	Landscape	10.2	6.3	7.2	9.5	7.6	8.1

The graphs in Figure 7 show the x, y, and z-axis reading of accelerometer for one of the users (User 17) in all the six typing positions. Only eight inputs (E1-E8) have been taken so that the graph is readable and depicts the pattern. Figure 7a, 7c and 7e shows that in landscape mode the reading for x, y and z-axis are different for each position. Similarly, in portrait mode the readings are different. The values for one axis may be similar in two positions like relaxing landscape and sitting landscape but the combination of all the three-axis reading is different for each position which makes it easier to identify one position against the other. Also, the graphs show that in the landscape mode the keystroke features form an even profile as compared to portrait mode which leads to better EER rate in relaxing-landscape mode. The graphs in portrait mode have some inconsistent values. So, the most appropriate feature that can help identify the typing position of the user seems to be the motion features. Since the orientation can be automatically detected the data was analyzed by considering portrait and landscape mode separately. Table 2 shows the results for analyzing typing position using motion features.

The results were promising with both the classifiers, although they were slightly better with Random Forest. Table 2 shows the error rates for a position measured against the other positions e.g., EER of 5.0% for the relaxing-portrait position is measured by considering the relaxing position as the positive class and the sitting and walking position as the negative class. The working of the model is such that it stops position identification the moment it identifies one position. So, for a user if the model returns ''relaxing'' position, it will not test for either ''sitting'' or ''walking'' position and will move to the third phase. The error rates in the case of both orientations were better for relaxing position as compared to sitting and walking position. The overall error rates are better for landscape mode as compared to the portrait mode. The FRR rates are a bit higher for sitting and walking positions. This is an issue of a little concern as it can lead to misclassification of the position which in result will lead to choosing the wrong model for authentication and the user may get rejected, in which case he/she has to repeat the authentication process again. Table 3 shows the accuracy of the results.

TABLE 3. Accuracy with motion features for position identification.

		Random Forest	KNN
User Position	Mobile Orientation	Accuracy (%)	Accuracy (%)
Relaxing	Portrait	90.2	89.7
Sitting	Portrait	83.4	78.2
Walking	Portrait	85.6	79.2
Relaxing	Landscape	90.0	88.6
Sitting	Landscape	85.3	83.6
Walking	Landscape	85.6	80.7

The overall accuracy achieved is above 85%. The accuracy of the relaxing position is approximately 90%. Also, the accuracy of the relaxing position is higher for both the classifiers.

Once the user position is determined the next step is to authenticate the user. Now, the comparison of user data for every position is done with the data of other users. So, the user data serves as the positive class and the other users' data serves as the negative class. The data were classified using the training and testing data on Random Forest and KNN classifiers. The data was analyzed based on the 55 features extracted during the feature extraction phase. Table 4 shows the resulting error rates using both classifiers and for each of the positions. The results shown are averaged over all users. The results were highly positive in the case of Random Forest, and the error rates decreased to a great extent. The least value of EER achieved was 2.9% in relaxing-landscape position. The worst EER value produced was 5.7% in the case of sitting-portrait which is almost similar to the best EER value obtained in case of motion features. The best combination for user authentication was relaxing-landscape as this position resulted in the least FAR and FRR of 0.7% and 9.2% respectively. With KNN the EER values were the worst achieved so far with the lowest being 18%. The EER values for all combinations are around 18% except for sitting portrait position for which it was 25.8%.

Table 5 compares the accuracy achieved with both the classifiers using the fusion of keystroke and motion features. As expected the accuracy achieved with the Random Forest was better with 81.22% being the lowest accuracy achieved in sitting-portrait position. Relaxing-landscape position resulted in the best accuracy value of 90.76%. The average accuracy rate achieved with RF was 86.5%. In general, the landscape mode had better accuracy rates in comparison with the portrait mode.

TABLE 5. Accuracy with combination of motion and keystroke features.

		Random Forest	KNN
User Position	Mobile Orientation	Accuracy (%)	Accuracy (%)
Relaxing	Portrait	85.10	74.20
Sitting	Portrait	81.22	65.24
Walking	Portrait	87.45	76.62
Relaxing	Landscape	90.76	76.60
Sitting	Landscape	85.81	70.84
Walking	Landscape	89.34	76.67

We further validated the results by checking if there was significant difference between the EER values for different positions. One-Way ANOVA was used for the same. The null-hypothesis set for the same was:

H_0: There is no significant difference between the EER for different positions. The value of F (5.24) was greater than F-critical (2.25). Hence the null hypothesis was rejected which means that there is significant difference between the EER for different positions. Also, the p-value (.0013) was less than the significance level (.05) which also suggests that there is significant difference between the EER values for different positions. Further post hoc test using Tukey's method showed that EER was significantly different for the following positions:

Relaxing Portrait	- Sitting Portrait
Relaxing Portrait	- Relaxing Landscape
Relaxing Portrait	- Walking Landscape
Sitting Portrait	- Walking Portrait
Sitting Portrait	- Relaxing Landscape
Sitting Portrait	- Walking Landscape
Walking Portrait	- Relaxing Landscape
Walking Portrait	- Walking Landscape
Relaxing Landscape	- Sitting Landscape
Relaxing Landscape	- Waling Landscape
Sitting Landscape	- Walking Landscape

Table 6 shows the comparison between the results obtained in this study and the previous studies. Although a direct comparison is not possible because the dataset, data collection environment, and classifiers used in each research are different. But a generic comparison shows that this research has resulted in better results in terms of EER and FAR. Although the FRR is high, the EER is much better than any other study

TABLE 4. Error rates using both motion and keystroke features.

		Random Forest			KNN		
User Position	Mobile Orientation	FRR (%)	FAR (%)	EER (%)	FRR (%)	FAR (%)	EER (%)
Relaxing	Portrait	14.9	1	4.6	25.7	1.5	18
Sitting	Portrait	18.8	1.2	5.7	34.8	2.3	25.8
Walking	Portrait	12.6	1.1	4.2	23.3	1.7	18.4
Relaxing	Landscape	9.2	0.7	2.9	24.4	1.7	18.4
Sitting	Landscape	14.1	0.9	4.7	29.2	1.7	18.9
Walking	Landscape	10.7	0.7	3	25.7	1.6	18.5

TABLE 6. Comparison of results with previous research.

Study	Feature	User Position	Number of Users	FAR(%)	FRR(%)	EER(%)
[16]	Keystroke, tap and motion	Walk, Sit	100			7.16
[17]	Keystroke and motion	Walk, stand and sit	36	Sit-1.7 Walk-1.4 Stand-1.8	Sit-6.1 Walk-5.6 Stand-5.3	
[19]	Motion	Sit, stand, walk	48	5.01	6.85	
[35]	Keystroke, Flick and motion	Sit, walk, car	20			Sit-2.5 Walk-0.2 In car-0.1
[51]	Keystroke and tap	Sit	28			21.02
This study	Motion, and keystroke	Sitting, Relaxing and Walking	40	RL-0.7 RP-1 SL-0.9 SP-1.2 WL-0.7 WP-1.1	RL-9.2 RP-14.9 SL-14.1 SP-18.8 WL-10.7 WP-12.6	RL-2.9 RP-4.6 SL-4.7 SP-5.7 WL-3 WP-4.2

which is the prime factor to measure the performance of a biometric system.

V. MODEL OPTIMIZATION

The accuracy of the system developed above is encouraging. The model was further optimized using a wrapper-based approach. Optimization is desired to get one of the two outcomes - reducing the error rates or reducing the number of features while keeping the error rates at almost the same value [52], [53]. Reduction in the number of features is a desired output of optimization because the model is developed for mobile phones, which have limited processing power. Hence, a reduction in the number of features will result in a fast performing model. In a wrapper-based approach, the weights to attributes are assigned based on the performance of the attribute measured using a classification model. To measure the performance, an inductive algorithm is deployed which acts as an evaluation module. To determine the accuracy of such a system estimation techniques are employed and the subset of attributes is chosen by the classifier. The working of such a method is shown in Figure 8.

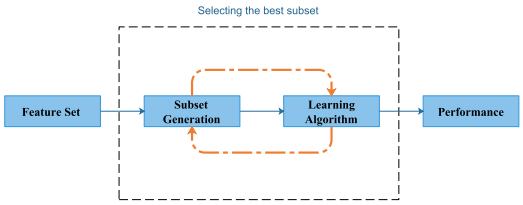


FIGURE 8. Wrapper based optimization model.

PSO was used for optimization and classification was done using Random Forest and KNN classifiers. PSO was preferred over other techniques because [54]

- It has less number of parameters to tune
- Is less sensitivity to the nature of the objective function compared to the conventional mathematical approaches and other heuristic methods
- Less dependent on initial points

PSO works on the principle of simulating bird flocking behavior. The set of random solutions is initialized and then the optimal solution is searched by updating through the next generations [55]''.

In PSO, a ''particle'' refers to a particular solution in the entire solution space. A fitness function is used to calculate the fitness value of every particle. A particle's velocity enables it to search in the problem space. Each particle flies in the problem space by following the current optimum particles.

The process starts by initializing a set of random particles commonly known as solutions. With each next-generation two values are generated: a local best value ($\hat{x}_i(t)$), which is the optimum value of particle and a global best value ($g(t)$) which is the best value obtained by any of the particles in the solution space.

The velocity of any particular particle at time t is calculated using the Eq. 15

$$v_i(t+1) = wv_i(t) + c_1 r_1[\hat{x}_i(t)] + c_2 r_2[g(t) - x_i(t)] \quad (15)$$

whereas a particle's position is calculated using the Eq. 16

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (16)$$

where:

i	- index of particle
w	- inertial co-efficient
c_1, c_2	- acceleration co-efficient
r_1, r_2	- random values
$v_i(t)$	- particle velocity
$x_i(t)$	- particle position
$\hat{x}_i(t)$	- best position
$g(t)$	- swarm's best solution

The analysis parameters used for PSO were:

- r1, r2 = 1
- Particle count = 40
- c1, c2 = 1
- Iterations = 100
- $v_i(t) = 0.001$

TABLE 7. Error rates after optimization using PSO.

		Random Forest			KNN		
User Position	Mobile Orientation	FRR (%)	FAR (%)	EER (%)	FRR (%)	FAR (%)	EER (%)
Relaxing	Portrait	9.8	0.7	3.3	23.7	1.5	18.2
Sitting	Portrait	15.6	1	4.5	24.2	1.6	18.2
Walking	Portrait	11.3	1	3.7	21.7	1.7	17.5
Relaxing	Landscape	7.4	0.5	2.2	12.4	0.9	4
Sitting	Landscape	12.4	0.8	4.1	25.7	1.5	18.3
Walking	Landscape	8.6	0.5	2.2	21.6	1.4	17.4

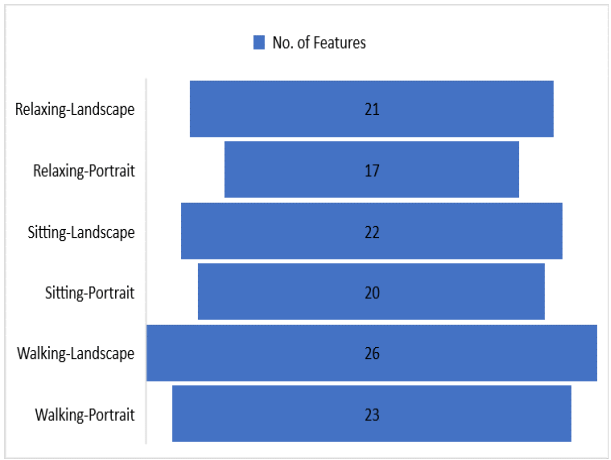
TABLE 8. Accuracy after optimization using PSO.

		Accuracy(%)	
User Position	Mobile Orientation	Random Forest	KNN
Relaxing	Portrait	90.16	76.34
Sitting	Portrait	84.30	75.79
Walking	Portrait	88.65	78.30
Relaxing	Landscape	92.58	87.60
Sitting	Landscape	87.62	74.20
Walking	Landscape	91.39	78.38

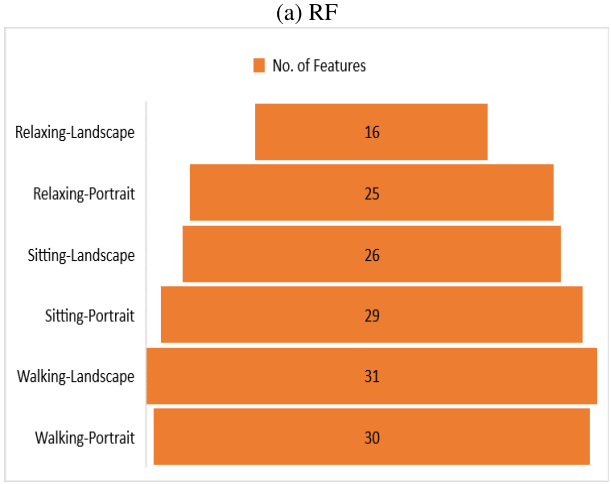
The data was split into two parts: one for feature selection and the other for training and testing. The total split was 30:50:20 i.e., 30% for feature selection, 50% for training and 20% for testing. Table 7 shows the results obtained using PSO on the proposed model. The results show that the Random Forest classifier outperformed KNN for all situations. There was a huge difference in the FRR and EER values. The FAR values were also better with RF as compared to KNN. The lowest FRR value achieved with RF was 7.4% in relaxing-landscape while it was 12.4% in relaxing-landscape in the case of KNN. The lowest EER value achieved with RF was 2.2% in relaxing-landscape mode while it was 4% in the relaxing-landscape mode in the case of KNN. The best FAR value achieved with RF was 0.5% in relaxing-landscape and walking-landscape modes while it was 0.9% in the relaxing-landscape mode in the case of KNN. The FAR values indicate that in all the positions only in 1 out of 100 attempts the system will fail to recognize an imposter. The EER rate values range from 2.3% to 4.5% for different positions. The FRR is on the higher side which in turn effects the EER. Another observation is that the error rates are low in landscape mode as compared to portrait mode. Error rates for *relaxing* position are least as compared to others. Overall, *relaxing-landscape* and *walking-landscape* position have the lowest EER of 2.2%.

Table 8 shows the accuracy of the model after optimization. Relaxing-landscape and sitting-portrait positions had the best and worst accuracy 92.58% and 84.30% respectively while in general, the model accuracy was 89.11%.

Model optimization was done with the goal of reducing the features used for analysis. After applying PSO with RF and KNN the resultant number of features was reduced. It can be observed in Figure 9 that the final feature set was different for each position. The total number of features for each position vary. As compared to the original 55 features the reduction



(a) RF



(b) KNN

FIGURE 9. Number of features selected using PSO.

in the number of features is significant. The feature set varies from a minimum of 17 features in the relaxing-portrait position to a maximum of 26 features in the walking-landscape position. Another notable fact was that the motion features were present in all the feature-subsets derived for each position separately. Also, five HT time features derived from keys "e", "l", "o", "t", and "5" were also part of the feature subset.

Figure 10 compares the error rates before and after optimization (the 'O' after the error rates means after optimization) for the Random Forest classifier. All the error rates

FIGURE 10. Comparison of error rates before and after optimization.

improved after optimization. The improvement in the case of FRR was significantly higher as compared to improvement in the case of FAR and EER. For example, there was an improvement of almost 5% in the FRR value for relaxing-portrait before and after optimization. Similarly, the accuracy of the model also increased. For specific positions also the accuracy improved. For both orientations using a relaxing position, the accuracy crossed 90%. The same was the case for walking-landscape position. The results prove that optimization not only reduces the number of final features in the feature set but also increases the system accuracy.

VI. LIMITATIONS

This study presented a three-step model for user authentication in three positions. These positions were selected on intuition. There can be other typing positions like standing. Even while getting data in these positions, the user data might have been biased. Like some users reported that they concentrated entirely on typing and avoided rush areas while entering the input in walking position. Lastly, the study did not put any restriction on the hand posture being used; participants could use either one thumb or both thumbs or index finger as per their convenience.

VII. CONCLUSION

This study presented the results of a study aimed at using keystroke dynamics for authenticating a mobile user in different positions. A three-step approach was proposed where the first step was to depict the orientation of the phone, the second step to determine the user position using accelerometer data and the last step was to authenticate the user. It was found that determining the position of the user before authentication improved the error rates as compared to the previous studies. The orientation in which the mobile is held also impacts the typing pattern of the user. The FAR achieved was less than 1% which satisfies the European standards for the access control system (EN 50133-1) [56]. The model was further optimized using PSO which reduced the feature set by at least 50%. The error rates were least for relaxing and walking position making them desired positions to use while authenticating a user using keystroke dynamics.

As part of future work, the accuracy of the system can be measured by considering the emotional state of the user especially the mood of the user as the user is relaxed while lying on the bed as compared to when he/she is walking where the mind is a little cautious about not to collide when an object. Further, the model can also be improved by lowering the FRR which in turn will lower the EER value. Finally, the robustness of the model against authentication attacks can also be tested.

ACKNOWLEDGMENT

The authors would like to thank all the users who participated in the study.

REFERENCES

[1] (2017). *Yahoo Says all of Its 3BN Accounts Were Affected by 2013 Hacking*. [Online]. Available: https://www.theguardian.com/technology/2017/oct/03/yahoo-says-all-of-its-3bn-accounts-were-affected-by-2013-hacking

[2] S. Zhong, H. Zhong, X. Huang, P. Yang, J. Shi, L. Xie, and K. Wang, "Connecting human to cyber-world: Security and privacy issues in mobile crowdsourcing networks," in *Security and Privacy for Next-Generation Wireless Networks*. Cham, Switzerland: Springer, 2019, pp. 65–100.

[3] M. Mehrnezhad, E. Toreini, S. F. Shahandashti, and F. Hao, "Stealing PINs via mobile sensors: Actual risk versus user perception," *Int. J. Inf. Secur.*, vol. 17, no. 3, pp. 291–313, Jun. 2018.

[4] T. N. Tan and H. Lee, "High-secure fingerprint authentication system using Ring-Lwe cryptography," *IEEE Access*, vol. 7, pp. 23379–23387, 2019.

[5] X. Cheng, A. Pitziolis, and A. Lasebae, "Implementing fingerprint recognition on one-time password device to enhance user authentication," in *Proc. Int. Symp. Cyberspace Saf. Secur.* Cham, Switzerland: Springer, 2019, pp. 448–461.

[6] K. Okokpujie, E. Noma-Osaghae, O. Okesola, O. Omoruyi, C. Okereke, S. John, and I. P. Okokpujie, "Integration of iris biometrics in automated teller machines for enhanced user authentication," in *Proc. Int. Conf. Inf. Sci. Appl.* Singapore: Springer, 2018, pp. 219–228.

[7] X. Wang, H. Xue, X. Liu, and Q. Pei, "A privacy-preserving edge computation-based face verification system for user authentication," *IEEE Access*, vol. 7, pp. 14186–14197, 2019.

[8] D. Banerjee and K. Yu, "3D face authentication software test automation," *IEEE Access*, vol. 8, pp. 46546–46558, 2020.

[9] A. Mhenni, E. Cherrier, C. Rosenberger, and N. E. Ben Amara, "Double serial adaptation mechanism for keystroke dynamics authentication based on a single password," *Comput. Secur.*, vol. 83, pp. 151–166, Jun. 2019.

[10] A. M. Gedikli and M. O. Efe, "A simple authentication method with multilayer feedforward neural network using keystroke dynamics," in *Proc. Mediterranean Conf. Pattern Recognit. Artif. Intell.* Cham, Switzerland: Springer, 2019, pp. 9–23.

[11] P. Perera and V. M. Patel, "Face-based multiple user active authentication on mobile devices," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 5, pp. 1240–1250, May 2019.

[12] P. Wang, W.-H. Lin, B.-H. Wu, K.-M. Chao, and C.-C. Lo, "A cross-age face recognition approach using fog computing architecture for user authentication on mobile devices," in *Proc. IEEE 15th Int. Conf. E-Bus. Eng. (ICEBE)*, Oct. 2018, pp. 86–93.

[13] P. Campisi, E. Maiorana, M. Lo Bosco, and A. Neri, "User authentication using keystroke dynamics for cellular phones," *IET Signal Process.*, vol. 3, no. 4, pp. 333–341, Jul. 2009.

[14] M. Antal, L. Z. Szabó, and I. László, "Keystroke dynamics on Android platform," *Procedia Technol.*, vol. 19, pp. 820–826, Apr. 2015.

[15] I. Lamiche, G. Bin, Y. Jing, Z. Yu, and A. Hadid, "A continuous smartphone authentication method based on gait patterns and keystroke dynamics," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 11, pp. 4417–4430, Nov. 2019.

[16] Z. Sitová, J. Šeděnka, Q. Yang, G. Peng, G. Zhou, P. Gasti, and K. S. Balagani, "HMOG: New behavioral biometric features for continuous authentication of smartphone users," *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 5, pp. 877–892, May 2016.

[17] H. Crawford and E. Ahmadzadeh, "Authentication on the go: Assessing the effect of movement on mobile device keystroke dynamics," in *Proc. 13th Symp. Usable Privacy Secur.*, 2017, pp. 163–173.

[18] J.-H. Roh, S.-H. Lee, and S. Kim, "Keystroke dynamics for authentication in smartphone," in *Proc. Int. Conf. Inf. Commun. Technol. Converg. (ICTC)*, Oct. 2016, pp. 1155–1159.

[19] C. Shen, T. Yu, S. Yuan, Y. Li, and X. Guan, "Performance analysis of motion-sensor behavior for user authentication on smartphones," *Sensors*, vol. 16, no. 3, p. 345, Mar. 2016.

[20] A. F. M. N. H. Nahin, J. M. Alam, H. Mahmud, and K. Hasan, "Identifying emotion by keystroke dynamics and text pattern analysis," *Behav. Inf. Technol.*, vol. 33, no. 9, pp. 987–996, Sep. 2014.

[21] I. Tereikovskyi, L. Tereikovska, O. Korystin, S. Mussiraliyeva, and A. Sambetbayeva, "User keystroke authentication and recognition of emotions based on convolutional neural network," in *Proc. Int. Conf. Artif. Intell., Med. Eng., Educ.* Cham, Switzerland: Springer, 2019, pp. 283–292.

[22] S. P. Banerjee and D. Woodard, "Biometric authentication and identification using keystroke dynamics: A survey," *J. Pattern Recognit. Res.*, vol. 7, no. 1, pp. 116–139, 2012.

[23] K. S. Killourhy and R. A. Maxion, "Comparing anomaly-detection algorithms for keystroke dynamics," in *Proc. IEEE/IFIP Int. Conf. Dependable Syst. Netw.*, Jun. 2009, pp. 125–134.

[24] K. S. Balagani, V. V. Phoha, A. Ray, and S. Phoha, "On the discriminability of keystroke feature vectors used in fixed text keystroke authentication," *Pattern Recognit. Lett.*, vol. 32, no. 7, pp. 1070–1080, May 2011.

[25] S. Mondal and P. Bours, "Person identification by keystroke dynamics using pairwise user coupling," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 6, pp. 1319–1329, Jun. 2017.

[26] F. Bergadano, D. Gunetti, and C. Picardi, "User authentication through keystroke dynamics," *ACM Trans. Inf. Syst. Secur.*, vol. 5, no. 4, pp. 367–397, 2002.

[27] S.-S. Hwang, H.-J. Lee, and S. Cho, "Improving authentication accuracy using artificial rhythms and cues for keystroke dynamics-based authentication," *Expert Syst. Appl.*, vol. 36, no. 7, pp. 10649–10656, 2009.

[28] H. Lee, J. Y. Hwang, D. I. Kim, S. Lee, S.-H. Lee, and J. S. Shin, "Understanding keystroke dynamics for smartphone users authentication and keystroke dynamics on smartphones built-in motion sensors," *Secur. Commun. Netw.*, vol. 2018, pp. 1–10, Mar. 2018.

[29] C.-J. Tasia, T.-Y. Chang, P.-C. Cheng, and J.-H. Lin, "Two novel biometric features in keystroke dynamics authentication systems for touch screen devices," *Secur. Commun. Netw.*, vol. 7, no. 4, pp. 750–758, Apr. 2014.

[30] Y. Meng, D. S. Wong, R. Schlegel, and L.-F. Kwok, "Touch gestures based biometric authentication scheme for touchscreen mobile phones," in *Proc. Int. Conf. Inf. Secur. Cryptol.* Berlin, Germany: Springer, 2012, pp. 331–350.

[31] K. R. Corpus, R. J. D. Gonzales, A. S. Morada, and L. A. Vea, "Mobile user identification through authentication using keystroke dynamics and accelerometer biometrics," in *Proc. Int. Workshop Mobile Softw. Eng. Syst. (MOBILESoft)*, 2016, pp. 11–12.

[32] D. I. Kim, S. Lee, and J. S. Shin, "A new feature scoring method in keystroke dynamics-based user authentications," *IEEE Access*, vol. 8, pp. 27901–27914, 2020.

[33] J. Wu and Z. Chen, "An implicit identity authentication system considering changes of gesture based on keystroke behaviors," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 6, Jun. 2015, Art. no. 470274.

[34] C. Giuffrida, K. Majdanik, M. Conti, and H. Bos, "I sensed it was you: Authenticating mobile users with sensor-enhanced keystroke dynamics," in *Proc. Int. Conf. Detection Intrusions Malware, Vulnerability Assessment*. Cham, Switzerland: Springer, 2014, pp. 92–111.

[35] H. Takahashi, K. Ogura, B. B. Bista, and T. Takata, "A user authentication scheme using keystrokes for smartphones while moving," in *Proc. Int. Symp. Inf. Theory Appl. (ISITA)*, Oct./Nov. 2016, pp. 310–314.

[36] C. Bo, L. Zhang, X.-Y. Li, Q. Huang, and Y. Wang, "SilentSense: Silent user identification via touch and movement behavioral biometrics," in *Proc. 19th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2013, pp. 187–190.

[37] A. Primo, V. V. Phoha, R. Kumar, and A. Serwadda, "Context-aware active authentication using smartphone accelerometer measurements," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2014, pp. 98–105.

[38] A. Buriro, B. Crispo, F. Del Frari, and K. Wrona, "Touchstroke: Smartphone user authentication based on touch-typing biometrics," in *Proc. Int. Conf. Image Anal. Process.* Cham, Switzerland: Springer, 2015, pp. 27–34.

[39] A. Kołakowska, "Usefulness of keystroke dynamics features in user authentication and emotion recognition," in *Human-Computer Systems Interaction*. Cham, Switzerland: Springer, 2018, pp. 42–52.

[40] F. Alshanketi, I. Traoré, A. Kanan, and A. Awad, "Adaptive mobile keystroke dynamic authentication using ensemble classification methods," in *Proc. Int. Conf. Intell., Secure, and Dependable Syst. Distrib. Cloud Environ.* Cham, Switzerland: Springer, 2018, pp. 38–49.

[41] I. Tsimperidis, S. Rostami, and V. Katos, "Age detection through keystroke dynamics from user authentication failures," *Int. J. Digit. Crime Forensics*, vol. 9, no. 1, pp. 1–16, Jan. 2017.

[42] M. Karnan, M. Akila, and A. Kalamani, "Feature subset selection in keystroke dynamics using ant colony optimization," *J. Eng. Technol. Res.*, vol. 1, no. 5, pp, pp. 072–080, 2009.

[43] D. Shanmugapriya and P. Ganapathi, "A wrapper-based classification approach for personal identification through keystroke dynamics using soft computing techniques," in *Developing Next-Generation Countermeasures for Homeland Security Threat Prevention*. Hershey, PA, USA: IGI Global, 2017, pp. 330–353.

[44] A. Muthuramalingam, J. Gnanamanickam, and R. Muhammad, "Optimum feature selection using firefly algorithm for keystroke dynamics," in *Proc. Int. Conf. Intell. Syst. Design Appl.* Cham, Switzerland: Springer, 2017, pp. 399–406.

[45] S.-H. Lee, J.-H. Roh, S. Kim, and S.-H. Jin, "Feature subset for improving accuracy of keystroke dynamics on mobile environment," *J. Inf. Process. Syst.*, vol. 14, no. 2, pp. 523–538, 2018.

[46] W. G. De Ru and J. H. P. Eloff, "Enhanced password authentication through fuzzy logic," *IEEE Expert*, vol. 12, no. 6, pp. 38–45, Nov. 1997.

[47] N. Sainis, D. Srivastava, and R. Singh, "Feature classification and outlier detection to increased accuracy in intrusion detection system," *Int. J. Appl. Eng. Res.*, vol. 13, no. 10, pp. 7249–7255, 2018.

[48] *Weka the Workbench for Machine Learning*. Accessed: Dec. 2018. [Online]. Available: https://www.cs.waikato.ac.nz/ml/weka/

[49] E. Sujatha and A. C. Nil, "Multimodal biometric authentication algorithm at score level fusion using hybrid optimization," *Wireless Commun. Technol.*, vol. 2, no. 1, pp. 1–12, 2018.

[50] E. Alpaydin, *Introduction to Machine Learning*. Cambridge, MA, USA: MIT Press, 2014.

[51] D. Buschek, A. De Luca, and F. Alt, "Improving accuracy, applicability and usability of keystroke biometrics on mobile touchscreen devices," in *Proc. 33rd Annu. ACM Conf. Hum. Factors Comput. Syst. (CHI)*, 2015, pp. 1393–1402.

[52] F. Jiménez, C. Martínez, E. Marzano, J. T. Palma, G. Sánchez, and G. Sciavicco, "Multiobjective evolutionary feature selection for fuzzy classification," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 5, pp. 1085–1099, May 2019.
[53] R. Zhang, F. Nie, Y. Wang, and X. Li, "Unsupervised feature selection via adaptive multimeasure fusion," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 9, pp. 2886–2892, Sep. 2019.
[54] M. Elbes, S. Alzubi, T. Kanan, A. Al-Fuqaha, and B. Hawashin, "A survey on particle swarm optimization with emphasis on engineering and network applications," *Evol. Intell.*, vol. 12, pp. 113–129, Feb. 2019.
[55] J. Kennedy, "Particle swarm optimization," in *Encyclopedia of Machine Learning*. New York, NY, USA: Springer, 2011, pp. 760–766.
[56] T. Novak and A. Gerstinger, "Safety- and security-critical services in building automation and control systems," *IEEE Trans. Ind. Electron.*, vol. 57, no. 11, pp. 3614–3621, Nov. 2010.

BALJIT SINGH SAINI received the master's degree in technology from Guru Nanak Dev University, Amritsar, and the Ph.D. degree in keystroke dynamics from Sri Guru Granth Sahib World University (SGGSWU), Fatehgarh Sahib, in 2019. He is currently working as an Associate Professor with the School of Computer Science and Engineering, Lovely Professional University, Phagwara. He has a total teaching experience of ten years and research experience of five years. His areas of interest include biometrics, user authentication, and operating systems.

PARMINDER SINGH (Member, IEEE) received the M.Tech. degree in computer engineering from Punjab Technical University, India, and the Ph.D. degree from Lovely Professional University, India, in 2019. He is currently working as an Associate Professor with the School of Computer Science and Engineering, Lovely Professional University. He has more than 30 articles in SCI/SCIE, Scopus indexed journals, conferences, and book chapters. His research interests include machine learning, deep learning, blockchain, cloud/fog/edge computing, network security, and Web services. He is an Active Member of IEEE. He has been the Session Chair and an Advisory Member for various international conferences.

ANAND NAYYAR (Senior Member, IEEE) received the Ph.D. degree in computer science (wireless sensor networks) from Desh Bhagat University, in 2017. He is currently working with the Graduate School, Duy Tan University, Da Nang, Vietnam. He is a Certified Professional with more than 75 Professional certificates from CISCO, Microsoft, Oracle, Google, Beingcert, EXIN, GAQM, Cyberoam, and many more. He has published more than 300 research articles in various national and international conferences, and international journals (Scopus/SCI/SCIE/SSCI Indexed). He has authored/coauthored cum Edited 30 books of *Computer Science*. He has two patents to his name in the area of Internet of Things and Speech Processing. He was associated with more than 400 international conferences as a Programme Committee/Advisory Board/Review Board member. He is also working in the area of wireless sensor networks, MANETS, swarm intelligence, cloud computing, the Internet of Things, blockchain, machine learning, deep learning, cyber security, network simulation, and wireless communications. He is a member of more than 50 associations as a Senior Member and a Life Member and also acting as an ACM Distinguished Speaker. He awarded more than 25 awards for Teaching and Research—Young Scientist, Best Scientist, the Young Researcher Award, the Outstanding Researcher Award, and the Indo-International Emerging Star Award (to name a few). He delivered more than 200 Talks in various esteemed institutions and universities on several aspects of Computer Science. He is acting as an Editor-in-Chief of IGI-Global journal- *International Journal of Smart Vehicles and Smart Transportation* (IJSVST).

NAVDEEP KAUR is currently pursuing the Ph.D. degree in distributed databases from IIT Roorkee. She is currently serving as a Professor with the Department of Computer Science, Sri Guru Granth Sahib World University (SGGSWU), Fatehgarh Sahib. She has published over 100 articles in various journals and proceedings. She has teaching experience of more than 15 years. Her research interests are information security, mobile computing, cloud computing, and software engineering.

KAMALJIT SINGH BHATIA (Member, IEEE) is currently pursuing the Ph.D. degree in optical-OFDM and wireless communication from a reputed university of India. He is also serving as an Associate Professor with the Department of Electronics and Communication Engineering, Govind Ballabh Pant Institute of Engineering and Technology, Ghurdauri, Pauri, India. He is also a Researcher and a Prolific Author. There are about more than 80 research articles and four books of international level into his credit. He has guided 29 research students at M.Tech. level and 04 at the Ph.D. level are in process. He has about 13 years of experience in teaching and research.

SHAKER EL-SAPPAGH received the bachelor's degree in computer science from the Information Systems Department, Faculty of Computers and Information, Cairo University, Egypt, in 1997, the master's degree from Cairo University, in 2007, and the Ph.D. degrees in computer science from the Information Systems Department, Faculty of Computers and Information, Mansura University, Mansura, Egypt, in 2015. In 2003, he joined the Department of Information Systems, Faculty of Computers and Information, Minia University, Egypt, as a Teaching Assistant. Since June 2016, he has been an Assistant Professor with the Department of Information Systems, Faculty of Computers and Information, Benha University. He is currently a Postdoctoral Fellow with the Centro Singular de InvestigaciónenTecnoloxíasIntelixentes, Universidade de Santiago de Compostela, Santiago, Spain. He has publications in clinical decision support systems and semantic intelligence. His current research interests include machine learning, medical informatics, (fuzzy) ontology engineering, distributed and hybrid clinical decision support systems, semantic data modelling, fuzzy expert systems, and cloud computing. He is a reviewer in many journals, and he is very interested in the diseases' diagnoses and treatment researches.

JONG-WAN HU received the M.S. degree from the G.W.W. School of Mechanical Engineering and the School of Civil and Environmental Engineering, respectively, Georgia Institute of Technology. His Ph.D. is from the Georgia Institute of Technology. He is currently an Assistant Professor with the University of Incheon. His research interests are in the area of computational solid mechanics, composite materials, and plasticity modeling.