Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 5677978, 15 pages
https://doi.org/10.1155/2021/5677978

Research Article

Multimodal Continuous User Authentication on Mobile Devices via Interaction Patterns

Xiaomei Zhang, Pengming Zhang, and Haomin Hu

School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai 201620, China

Correspondence should be addressed to Xiaomei Zhang; xmzhang@sues.edu.cn

Received 21 May 2021; Revised 28 June 2021; Accepted 23 July 2021; Published 18 August 2021

Academic Editor: Chien-Ming Chen

Behavior-based continuous authentication is an increasingly popular methodology that utilizes behavior modeling and sensing for authentication and account access authorization. As an appearing behavioral biometric, user interaction patterns with mobile devices focus on verifying their identity in terms of their features or operating styles while interacting with devices. However, unimodal continuous authentication schemes, which are on the basis of a single source of interaction information, can only deal with a particular action or scenario. Hence, multimodal systems should be taken to suit for various environmental conditions especially in circumstances of attacks. In this paper, we propose a multimodal continuous authentication method both based on static interaction patterns and dynamic interaction patterns with mobile devices. Behavioral biometric features, HMHP, which is combined hand motion (HM) and hold posture (HP), are essentially established upon the touch screen and accelerator and capture the variation model of microhand motions and hold patterns generated in both dynamic and static scenes. By combining the features of HM and HP, the fusion feature HMHP achieves 97% accuracy with a 3.49% equal error rate.

1. Introduction

In Internet of things, mobile devices that store ample sensitive and private data have become increasingly popular in the daily life, with 400000 Apple and 1.3 million Android devices activated [1] each day. To protect this private data from unauthorized access, traditional explicit authentication methods for mobile devices are employed using a password, personal identification number (PIN), face, fingerprint, or secret pattern. Previous work has shown that such solutions provide limited security because of several reasons [2–5]. Firstly, they make devices vulnerable to guessing, shoulder surfing, smudge, and spoofing attacks. Secondly, since the user commonly perform them for initial interaction with mobile devices, it is challenging to detect intruders after login. Thirdly, expenses of extra hardware, data acquisition time, and quality requirement of the sample are significantly higher. With the purpose of overcoming these issues, it is essential to investigate techniques for continuous authentication as a secure protection to user data.

The majority of continuous authentication mechanisms sense the manner during the user's interaction with the device and monitor user identity in terms of behavioral data. They essentially adopt behavior-related measurements, making use of embedded sensors typically including the gyroscope, accelerator, touch screen, and orientation sensor, to measure behavioral biometric traits. In this paper, we consider the behavioral biometric dynamics based on the user's interaction patterns with the devices. However, most studies pay more attention to unimodal interaction patterns using a single source of information, which can cause nonuniversality and intraclass variations [6]. Therefore, behavior-based continuous authentication mechanisms must be able to extract what are the distinctive behavior traits of an individual for interacting with the mobile device while applicable to various scenes.

We make a few observations that people have two types of interaction patterns with mobile devices depending upon the use situation. The first one is the dynamic interaction pattern.

Here, relative locations of user's finger or hand on mobile devices change during a time interval, like touching, scrolling, typing, flicking, and rotation of the touch screen [7–9]. The way that users interact with devices in the dynamic pattern is utilized as a behavioral biometric to perform continuous authentication. This approach works well in precision-demanding task scenarios. In practice, the user may not undertake particular activities or not use targeted applications so that the system collects data under constrained environments and certain times. The second philosophy is the static interaction pattern. Here, relative locations of the user's body (part) with mobile devices remain unchanged during a time interval, such as holding a mobile phone or wearing a smart glass on his head. Users may stand, walk, sit, talk, swipe, and type when they hold a mobile device or wear a device. However, the continuance of the authentication system becomes limited when the sensor is not triggered or the phone usage is under various situations.

To improve the continuance and accuracy of continuous authentication, it is necessary to apply the static interaction pattern to complement the dynamic interaction pattern to result in secure authentication systems. Dynamic interactions, such as touch behavioral biometrics with swiping and key stroking, cope with intruders to be an effective method for continuous authentication on mobile devices. Swipe biometrics are more general than click and keystroke biometrics that are not always available since there is no virtual keyboard on lots of wearable devices [10]. Static interaction and motion sensors, such as accelerators and gyroscopes, could record distinctive movement and orientation behavior of users when they hold their phones with hands [11] or wear smart devices on their body [12].

Existing works [13, 14] show that there are big differences in the physiological features and behaviors between different individuals. In terms of physiological features, the influence of the user's hand structure [15] (the size of the palm and the length of the finger) and the difference in age [15, 16] can affect the sliding position of the finger, the strength of the hand, and the length of the sliding, which ultimately form different HP features. The usage habits of different hands holding the phone, speed, strength, and duration of the sliding screen cause difference in the phone's shaking amplitude and change rates of the vibration. In addition, different device shapes (size and weight), direction, and sliding screen can also cause different actions for HP. The differences of these features affect how the user operates, resulting in differences in operational behavior. All these reflect a user-specific usage behavior, so that HMHP features can identify owners and impostors.

In this paper, we propose a continuous user authentication method that integrates a static interaction pattern and dynamic interaction pattern for mobile devices with the example of smart phones. The dynamic interaction pattern is achieved by means of hand motion, while static interaction pattern is achieved by means of hand posture. A novel set of behavioral biometric features, HMHP features, is extracted from motion sensors and screen sensors. We take the variation range and rate of micromovement changes as HM features and take change boundary values and stable trend values as HP features. These features can be seamlessly extracted while users use phones or just hold phones. It does not require users' particular action or scenario. Thus, our system works in various scenarios. Moreover, the combination of two interaction patterns does not only improve classification accuracy but also effectively defend against spoof attacks aimed at individual behaviors or individual sensors. Note that in our paper, we focus on one specific case of mobile devices: smart phones. However, the authentication methods of the integrated static interaction pattern and dynamic interaction pattern introduced in our paper also apply to other mobile devices with a touch screen and built-in motion sensors, such as pad [17] and smart glasses [18]. Furthermore, there is an Android-based software interface via the "pad" as the hardware interface in a service robot or a human imitator, which is expected to interact with a limited number of people. Our unobtrusive authentication scheme uses sensors to prole the interaction on the interface "pad" without any extra action.

The major contributions of our paper are threefold. First, we develop a design concept of multimodel continuous authentication of an integrated static and dynamic interaction pattern for mobile devices, which is adaptable to various environmental conditions. We model behavioral biometric HM and HP using a motion sensor and screen behavior via extensive evaluation and systematically investigate the discrimination power and authentication permanence of HM, HP, and HMHP. Our authentication method with HMHP features can verify user's identity with low error rates using a short training time, without constrained environment and interactions.

Second, we optimize our algorithm by comparing three types of machine learning algorithms: K-nearest neighbor (KNN), support vector machine (SVM), and random forest (RF), to verify feature performance. The optimal parameter value for each algorithm is searched by a tenfold crossvalidation method and then is used to evaluate the fusion feature authentication accuracy on the same dataset. The accuracy of three algorithms is higher than 94%, which shows that HMHP features have strong discrimination. Furthermore, we found that the HMHP features combined with the random forest algorithm can achieve the best overall performance since its authentication accuracy is 97% and its EER is 3.49%.

Finally, we convened four researchers as impostors to imitate owners using mobile phones. Take each slide track as a single pass. The experiment found that the average success rate of 500 attacks per attacker (30 minutes required) was only 3%. This indicates that the behavior information captured by HMHP is difficult to imitate, indicating that the HMHP features can resist violent attacks to a certain extent.

We present the description of features and data acquisition in Section 2. In Section 3, we describe the feature evaluation on HMHP and then evaluate the performance of our multimodal authentication under different scenarios in Section 4. We review related research in Section 5. We conclude in Section 6.

2. Feature Description and Data Acquisition

2.1. Description of Features. We define two types of features: HP features and HM features, which are calculated by

recording x and y coordinates, pressure sensor, and accelerometer data when sliding. We draw and analyze a part of the collected data.

2.1.1. HP Features. The HP features capture the phone's orientation and state features from a macroscopic view. It can generally show the changes in the features of holding a mobile phone. Figure 1 shows that the difference between User1 and User2 since the maximum and minimum values are significantly different. The change is stable for the same user, yet the difference between different users is large. For example, in the XVelocity and YVelocity diagrams, the sliding velocity of User1 at the x coordinate is much higher than that of User2 and the sliding velocity of User2 is much higher than that of User1 at the y coordinate; the maximum value of User1 on the x-axis of the accelerometer is about 0.5, while User2 is basically below 0.5; User1 has the smallest x-axis acceleration at the beginning of the trajectory which reaches the maximum at the end point, while the performance of User2 is opposite. Both the acceleration and the change on the y-axis are relatively stable, with the acceleration of User1 being 10m/s^2 and that of User2 being around 6m/s^2. This phenomenon may be the results of the difference in users' holding posture. Hence, we select the maximum and minimum pressures, velocities, and accelerations as the features of different users when they are holding devices.

2.1.2. HM Features. The HM features can measure the change rate of features on the sliding screen and the rate of change of the characteristic amplitude when it behaves within the trajectory. The underlying changes in the original features are closely related to the various points within the trajectory. When the user slides the screen, the features of trajectories at different points show different amplitude changes (for example, the finger will form pressure on the screen and the different pressing force will cause that the screen output pressure of the mobile phone is different). In a sliding track, the pressure change value is recorded at different points. The result shows a trajectory of the original feature change. The rate of change of each feature point on this trajectory represents the user's habits of operating a mobile phone. As can be seen in Figure 1, the x-axis acceleration of the User1 trajectory fluctuates greatly, and that of User2 is relatively stable, which indicates that the vibration amplitude of the mobile phone is inconsistent during operation. We use equation (1) to calculate the corresponding rate of change in pressure, velocity, and acceleration:

$$\Delta_{\text{FVrate}} = \frac{\text{FV}_{\max} - \text{FV}_{\text{start}}}{T_{\max} - T_{\text{start}}}. \tag{1}$$

$\text{FV}_{\max}$ represents the maximum value of finger pressure, swiping velocity, and acceleration in the trajectory. FV_{start} represents the value at the start point of the track. And Δ_{FVrate} represents the rate of change from the start point to the feature of the value maximum point. Table 1 is the feature vector of the interaction behavior pattern extracted by analysis (the calculation of the variation rate feature in the table follows equation (1).

2.2. Data Collection. Since the Android system prohibits third-party applications from acquiring other APP-related data, we develop an APP to implement the basic functions of the reader and joined the data collection service. We use the Huawei glory V10 mobile phone as the collection tool to deploy the APP on the EMUI9.1 (based on Android 9.0) system to collect data. We select 10 students (6 boys and 4 girls) as experimental users and collected 1000 action tracks including more than 10000 metadata points for each user. The data recorded contains the following original features: relative event time (milliseconds), x and y coordinates of each point in the trajectory, screen pressure at x and y coordinates, sliding speed of the finger at x and y coordinates, and acceleration value on the x-, y- and z-axes (data format is shown in Table 2).

After preliminary research, each user experiment was at least 1 hour. We adopt a turn-to-turn strategy: each user uses 20 minutes in turn and 10 users are taken in turn.

3. Feature Evaluation

We evaluate the performance of features in this section: (1) feature screening, (2) three classifiers for comparison experiments, (3) classifier parameter selection, (4) model training and testing, (5) feature performance evaluation, and (6) data processing.

3.1. Feature Selection. In order to measure the validity of selected features, we use mutual information method (MI) to calculate the feature contribution rate. The mutual information method evaluates the correlation of qualitative independent variables to qualitative dependent variables, which describes the degree of correlation between features x and y . This method is often used for featuring variable screening of classification problems. The mutual information method is mathematically defined as follows:

$$I(\mathbf{x}\,;\mathbf{y}) = \sum_{x\in\mathbf{x}}\sum_{y\in\mathbf{y}} p(x,y)\,\log\,\frac{p(x,y)}{p(x)p(y)}. \tag{2}$$

Here, $\mathbf{x}$ is the feature vector and $\mathbf{y}$ is the label vector. It takes a value between 0 and 1, whereas 0 means the feature carries no information about the $\mathbf{y}$ label and 1 means the feature determines the $\mathbf{y}$. The result of this analysis is shown in Figure 2.

As can be seen in Figure 2, the mutual information of hand gesture feature map and hand micromotion feature is relatively high. In the two types of behavior pattern features, the stroke rate feature contains a large amount of information, which indicates that the user's finger motion rate has a high discrimination. Analysis of Figure 2(a) shows that the characteristics of the xy coordinate position, xy coordinate rate, and acceleration change rate are higher than 10%. This indicates that the tiny vibration is generated at different points of the trajectory when the finger swipes, indicating that the corresponding characteristic change rate contains more abundant information. The mutual information of the coordinate features of the start point and the end point is close to 18% and 14%, respectively, which indicates that the

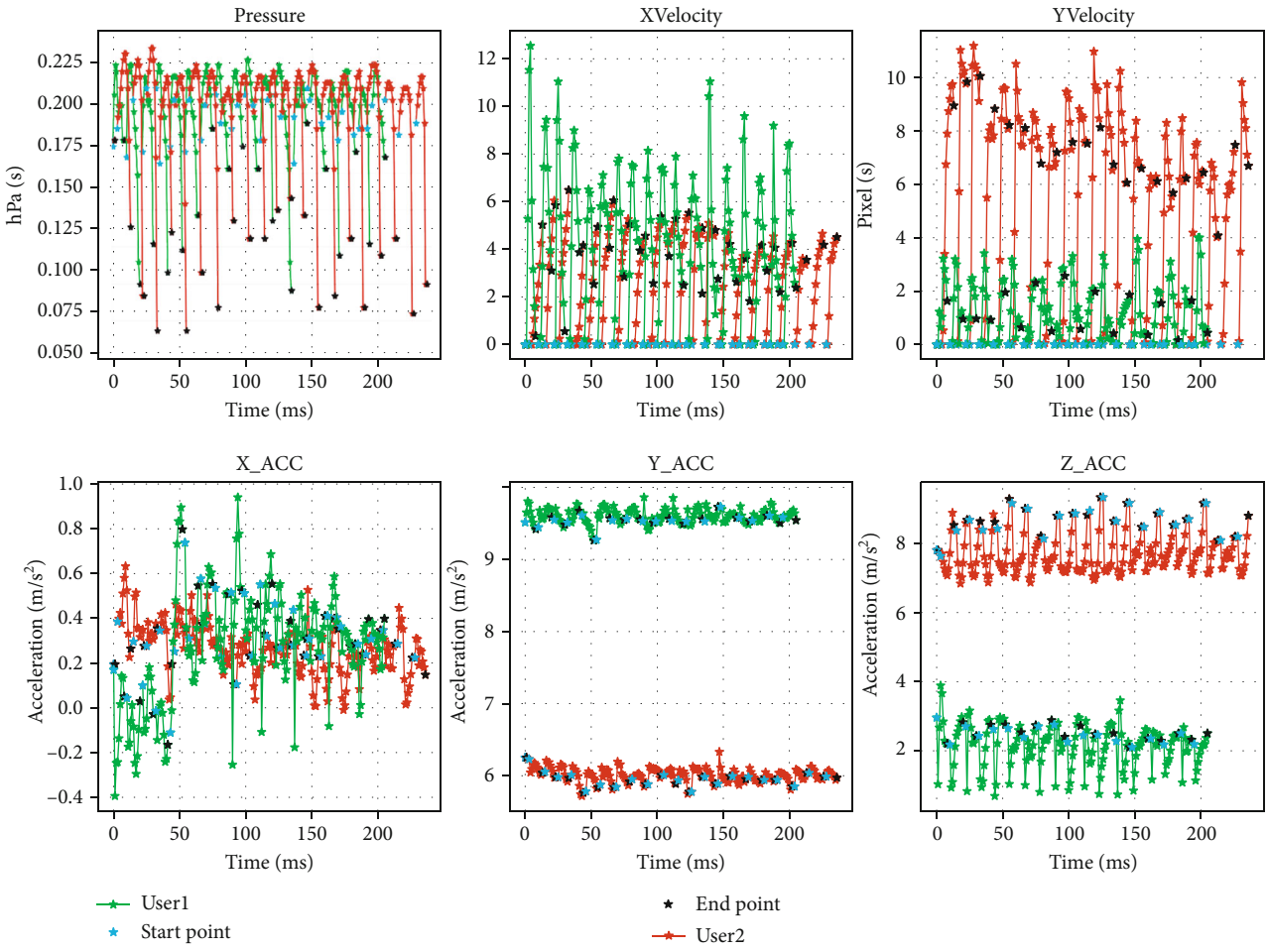


Figure 1: Change of original features.

start position feature of the stroke screen has obvious discrimination. However, the information content of the maximum and minimum pressure features is low and the information content of the minimum pressure feature is zero. It shows that the change of minimum pressure is not obvious and no useful information is provided. As can be seen in Figure 2(b), the information content of acceleration difference between the start point and the end point is less than 4%, which indicates that for different users, the maximum amplitude change of acceleration difference is not obvious. The maximum and minimum acceleration feature information is more than 14%, which contains more feature difference information.

In order to further analyze the correlation between features, we use a correlation matrix to analyze the influence of correlation between features and the results are shown in Figure 3. The red area is positive correlation, and the cyan area is negative correlation.

It can be seen in Figure 3 that there are correlations between features but the strength is different. In Figure 3(a), the correlation of mean, maximum, and median finger pressure features reaches 100%, forming feature information redundancy; and the features related to the speed of finger stroke are highly correlated. In Figure 3(b), the linear length of finger stroke trajectory is highly correlated with the actual length, while the maximum acceleration characteristics of the x-axis and z-axis of the accelerometer are negatively correlated. Although some features have high correlation, it is found in Figure 2 that their feature contribution rate is large, which indicates that the discrimination of their combined features may be higher. Through the above analysis, this chapter removes the minimum pressure, the median pressure, and the track straight line length, which are highly correlated and have little information.

3.2. Classifier Selection. In order to achieve better classification results, we use three classifiers to verify the feature performance: K-nearest neighbor (KNN), support vector machine (SVM), and random forest (RF). We choose these three classifiers for the following reasons:

KNN can provide fast classification [7], which is a powerful classification tool. The KNN classifier obtains each new observation (here is a track feature) and positions it in the feature space according to all the training observations. The classifier first identifies the K training observations closest to the latest observation and then selects the tags that most of the K training observations have. It does not need a clear

Table 1: The HM and HP feature set.

Type	Feature	Description
HP	start_x, start_y	Starting point coordinates of x and y
	stop_x, stop_y	End points of x and y coordinates
	velocity_x,y_max	Maximum velocities of trajectory in x and y directions
	velocity_x,y_mean	Average rates of trajectory in x and y directions
	acc_x,y,z_max	Maximum accelerometers of x-, y- and z-axes
	acc_x,y,z_ min	x-, y- and z-axis acceleration minimum
	pressrue_max,min	Maximum and minimum pressures in trajectory
	pressure_median	Median pressure value
	pressure_mean	Pressure mean value
	pressrue_std	Pressure standard deviation value
	acc_x,y,z_mean	Average acceleration values of x-, y- and z-axes
	acc_x,y,z_std	x-, y-, and z-axes acceleration variance
HM	direct_distance	End-to-end straight distance of the track
	real_distance	Trajectory distance of actual sliding
	storke_duration	Sliding trajectory duration
	swpie_velocity	Finger sliding velocity
	acc_x,y,z_Begin_Max_rate	Change rates of acceleration of x-, y- and z-axes to the maximum value
	acc_x,y,z_max_to_min	Maximum and minimum differences of x-, y-, z-axis acceleration
	acc_x,y,z_End_Max_Rate	Change rates of acceleration of x-, y-, and z-axes to the minimum value
	pressure_start_to_end	Pressure difference between the start point and the end point
	pressure_Begin_Max_rate	Rate of change from pressure at the beginning of trajectory to maximum pressure
	pressure_End_Max_rate	Rate of change from maximum pressure of trajectory to end pressure
	velocity_x, y_Begin_Max_rate	Rates of change from x and y starting point rates to maximum rates
	velocity_x,y_max_to_end	The difference between maximum speed of x and y coordinates and the speed at the end point
	velocity_x,y_End_Max_rate	Rate sof change from x and y end point rates to maximum rates
	velocity_x,y_End_Min_rate	Rates of change from x and y end rates to minimum rates

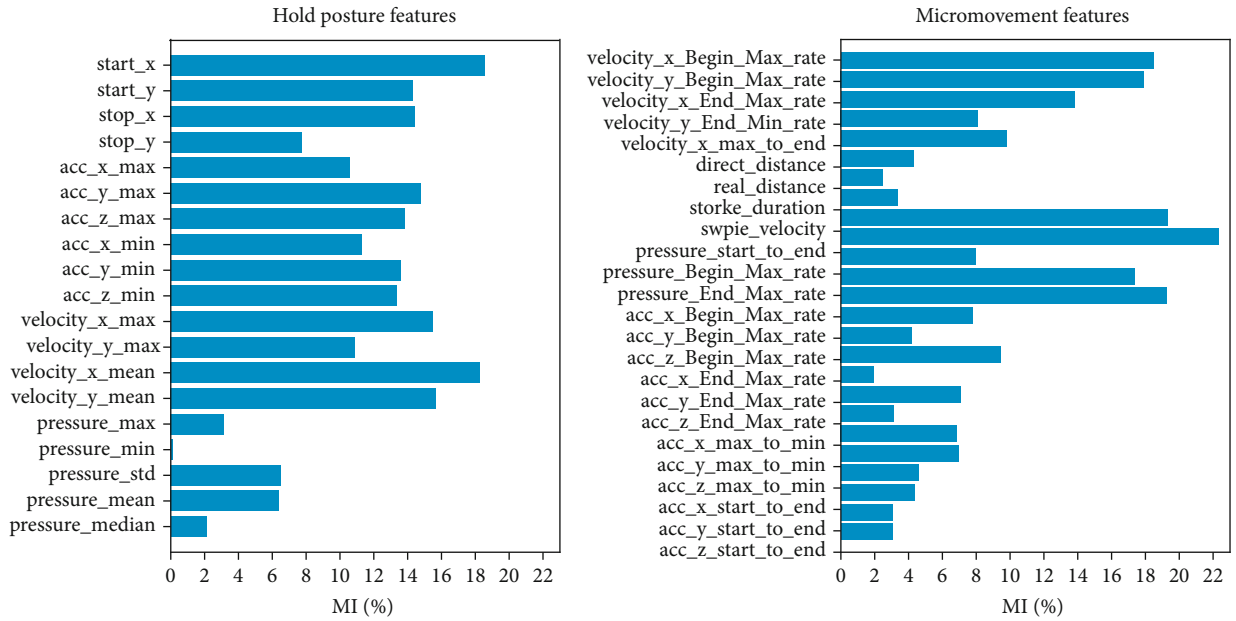
Table 2: Original feature set.

Features	Description
UserID	User ID
Time	Timing relatively
TouchType	Move, up, down flag
XCoordinate, YCoordinate	x and y coordinates
Pressure	Pressure caused by finger swiped on phone screen
XVelocity, YVelocity	x and y coordinate point velocities
X_ACC, Z_ACC, Z_ACC	x-, y-, z-axis values of acceleration

training stage and only stores all the training observations and tags, and the algorithm runs fast.

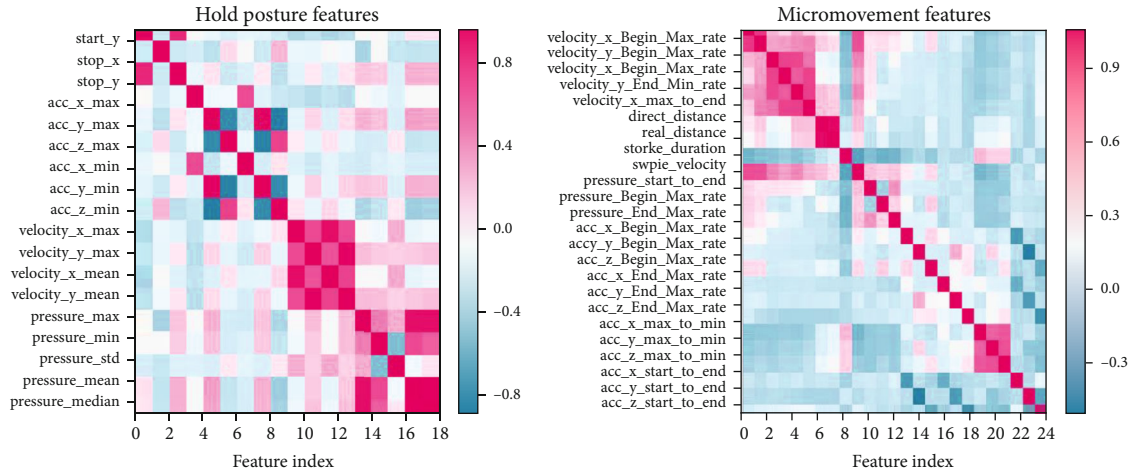
SVM is a popular and powerful binary classifier. SVM maximizes the feature space through a hyperplane, which maximizes the boundary between two classes; SVM compresses the thickest hyperblock between the boundary observations of two classes (the so-called support vector). SVM is characterized by induction from observed data. That is to say, it forgets individual observation after training and only saves the decision hyperplane. In order to eliminate the influence of outliers, a small number of boundary values are allowed in the boundary. There is a tradeoff between parametric profit maximization and exception minimization. For a class that cannot be linearly separated in the feature space, the so-called kernel can be used instead of the standard scalar product involved in hyperplane computation.

RF is an integrated algorithm, which belongs to the bagging type. The forest consists of many trees, which depend on the results of the classification of multiple decision trees. By combining multiple weak classifiers and finally voting or averaging, the result of the overall features has high accuracy and generalization performance. Among them, "random" makes it have



(a) MI (%) (b) MI (%)

Figure 2: Characteristic contribution rate analysis of HP and HM features.



(a) Feature index (b) Feature index

Figure 3: Correlation analysis of HM and HP characteristics.

the ability to resist overfitting and "forest" makes its classification more accurate. Because of the randomness, it is very useful to reduce the variance of features, so random forest does not need additional pruning but also can achieve better generalization ability and anti-overfitting ability.

3.3. Parameter Selection. In the same dataset, in order to get better classification, we need to adjust the parameters of different classification algorithms. For the KNN algorithm, we search a series of K values to find the optimal K. For SVM, because the sample size is large and the initial test shows that linear kernel classification is better, we use the linear kernel SVM classifier. The SVM penalty parameter C determines the classification boundary, and the larger the C is, the different parameter C determines the fitting degree of the model. We search a series of C-values to find the optimal parameters, so as to improve the accuracy of the boundary line division between user categories. The classification performance of random forest is closely related to the number of trees M. Too high M will lead to overfitting and low accuracy. For the above algorithm, the grid search method is used to search the optimal parameter value and the 10 times crossvalidation method is used to evaluate.

3.4. Feature Training and Testing. In the reading mode, we collected 10000 pieces of data for 10 users. After removing

noise data, the dataset is divided into the training set and test set according to 75% and 25%. Training sets are used to train models and optimize parameters; test sets are used to evaluate models and solve overfitting and underfitting problems. In order to further test the performance of the model, we use the training set to train 10 users (positive example) who mark legal users in turn and then test them with the test set. In the experimental evaluation section, we give 10 user authentication results.

3.5. Performance Evaluation Metrics. For the machine learning algorithm and feature selection, there are usually the following evaluation indexes: ROC curve, AUC value, F1-score, DET (detection error tradeoff) curve, and equal error rate (EER). AUC is defined as the area under ROC curve, where ROC curve is used to show the performance of two classifiers relative to their classification threshold. The x -axis of the ROC curve is a false-positive rate, and the y -axis is a true rate. The calculation method is as follows:

$$\mathrm{AUC} = \frac{1}{2}\sum_{i=1}^{M-1}(x_{i+1} - x_i)\bullet(y_{i+1} - y_i). \tag{3}$$

M is the number of thresholds and x_i, y_i are the true rate and the false-positive rate corresponding to the its threshold. The AUC value is between 0 and 1; the larger the value is, the better the model is. The F1-score is an index to measure the accuracy of binary features, which is widely used in machine learning algorithm performance evaluation. The F1-score can be regarded as a kind of harmonic average of feature accuracy and recall rate. It takes into account both the accuracy and recall rate of classification features, and the value is between 0 and 1. The higher the score, the better the performance of the model. Its mathematical definition is

$$\mathrm{F1} = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}}. \tag{4}$$

TP is the true-positive case, FP is the false-positive case, and FN is the false-negative case. In the exception detection, if the unauthorized user is classified as the owner, it is regarded as a false alarm. If a real user is classified as an owner, it is considered as a real-positive factor. The F1-score can be widely used in the field of information retrieval to measure the performance of retrieval classification and document classification. We use the DET curve as a graph of the bit error rate of the binary classification system, which draws the curve between FRR (false reject rate) and FAR (false accept rate) with the change of judgment threshold. The DET curve can be used as the evaluation index of the performance of pattern recognition classifier. Where EER is the value at the intersection of the FRR and FAR curves, the lower EER indicates the better performance of the algorithm.

In the aspect of algorithm parameter selection, we use the EER index to evaluate the influence of parameters on feature classification, find the best parameter value for each algorithm, and optimize to get the best learning algorithm; in the aspect of classifier selection, we use the ROC curve and DET curve to evaluate the algorithm's performance; in the part of fusion feature evaluation, we use accuracy and the F1-score to measure feature authentication performance; in the attack test, EER is used to evaluate model performance.

3.6. Data Processing. The original data obtained from mobile phones cannot be directly used for business processing, because the original data contains noise and missing values. Therefore, in order to extract more reliable and available data features for data business processing and meet the training requirements, we process the data from two aspects:

3.6.1. Data Filtering. It can be seen in Figure 4 that the data at the beginning and at the end of each characteristic curve are almost the same as those of adjacent points. For example, the acceleration of the x- and y-axes is zero at the starting point and the next point, while three values of acceleration have little change. This is because smart phone sensors are extremely sensitive, many values have been recorded before the fingers touch the screen, and there is a sudden change at the beginning and at the end of the x- and y-axes. The first track of User1 is too short to show the change of features in the track, so it is difficult to extract relevant feature information. For example, the track of records generated while clicking the screen is short, so the relevant feature information cannot be extracted. For the above cases, we do the following processing: (1) take the start and end points of the trajectory as noise points; we use a truncation method to remove the data of the start and end points, and replace them with adjacent points; (2) we remove the sliding track of less than n points when the trajectory noise can be filtered out at n equals 3; and (3) as the user's fingers will pause during the sliding and the screen has not been moved by pressing and holding all the time, if the x and y coordinate data at this time is invalid, the filtering process shall also be performed.

3.6.2. Data Dimension Reduction. There are three main purposes of data dimension reduction: (1) eliminating the correlation between features. According to the analysis in Section 3, there is still a correlation between the reserved features. The accuracy of classification is further improved by reducing the dimension and eliminating the influence of correlation and (2) improving the training speed of the model. We have collected tens of thousands of data with large data volume and dimensions, which leads to a long time in the model training stage, and we need to optimize the training efficiency through dimension reduction. Principal component analysis (PCA) is a very mature dimension reduction algorithm, which is widely used in a large number of and high-dimensional data dimension reduction. To sum up, we use the PCA method to reduce the dimension of data. According to PCA properties, data will lose some information after dimension reduction, which in turn affects model classification. In order to find a balance between classification efficiency and accuracy, we search for the reasonable number of dimensions from a series value of dimensions. In next section, we represent the analysis results.

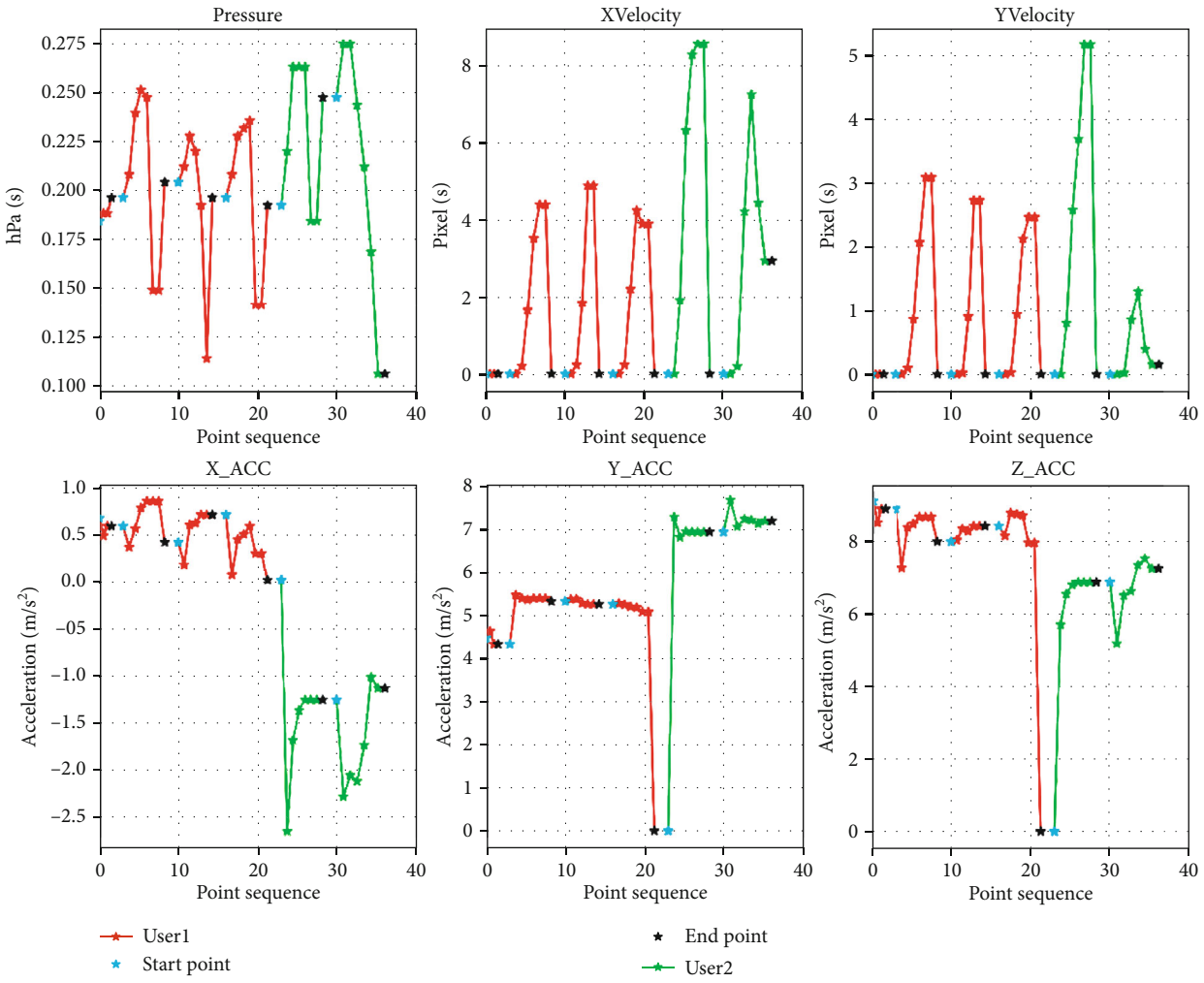


Figure 4: Noise data analysis of the accelerometer and touch screen sensor.

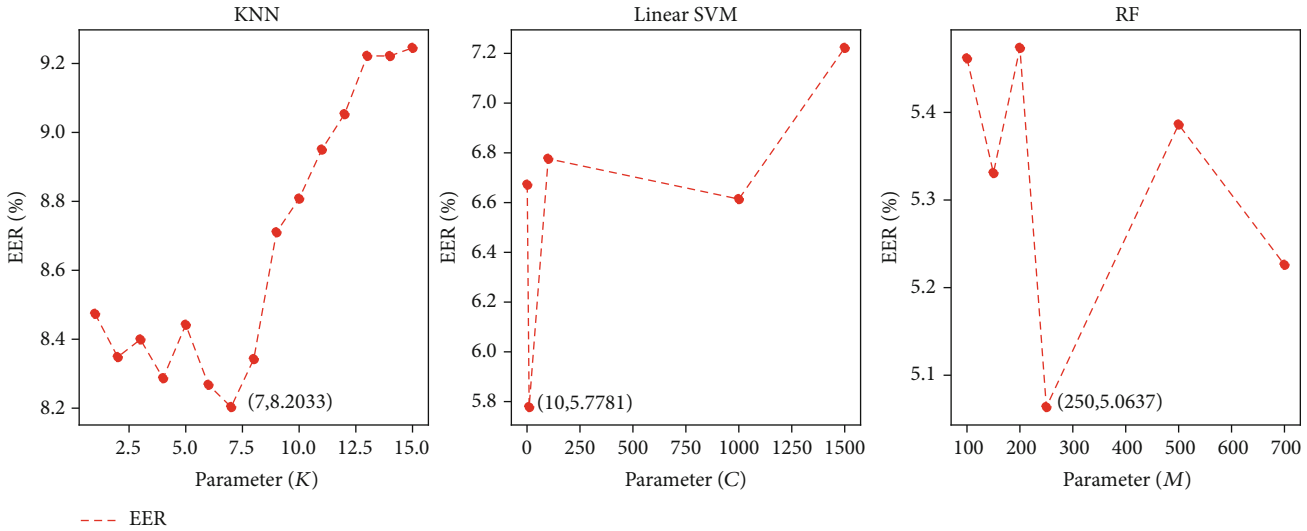


Figure 5: The optimal parameter search of each classification algorithm.

4. Experimental Evaluation

In this section, we adopt a variety of classification algorithms to verify the feature performance and analyze the performance of HM feature and HP feature authentication, respectively, as well as the authentication results after these features are combined. We compare HMHP features to other related works and analyze the advantages and disadvantages of the features.

4.1. Comparison of Different Algorithms

4.1.1. Parameter Optimization. Before the algorithms begin to be compared, we need to adjust the algorithm to achieve the perfect results. We choose the common value as the parameter adjustment option value to adjust KNN, SVM, and RF algorithm. For KNN, we search for a K value between 1 and 15; linear SVM only needs to adjust parameter C and we search for the C value in [1, 10, 100, 1000, 1500]; RF parameters are numerous, but the most influential is the number of trees M, so we only consider the optimal value of M and search for the M value in [100, 150, 200, 250, 500, 700]. For each selected parameter, the 10 times crossvalidation method is used to evaluate the impact of parameters on the classification performance of the algorithm. The classification performance results of each algorithm are shown in Figure 5.

In Figure 5, it can be seen that the EER value of KNN decreases with the increase of the selected K value and the performance improves gradually. EER increases rapidly after $K = 7$ and the performance of the algorithm decreases. EER drops to the lowest point at $K = 7$. At $C = 10$, EER of SVM changes greatly and EER of RF changes a little in the search interval, which is existing between 5% and 5.5%. At $M = 250$, the performance of the algorithm is optimal, which is about 5%. Through the above analysis, we selected $K = 7$, $C = 10$, and $M = 250$ as the optimal parameters of the three algorithms.

4.1.2. Algorithm Comparison. On the basis of the optimal parameters, the above three machine learning algorithms are used to classify HM features, HP features, and fusion features. We obtain the ROC curve and DET curve of three algorithms in Figures 6–8. We can see in Figure 6 that the real rate value corresponding to RF at low threshold rises rapidly on the ROC curve of the HM feature, while the ROC curve has the best response. However, the DET curve of the KNN algorithm decreases slowly and the corresponding EER value is more than 0.2, which is much higher than other algorithms. It indicates that KNN has poor classification effect for this kind of feature. In Figure 7, we find that the performance of KNN on HM is better than that of HP and EER is similar to other algorithms. By observing Figures 6 and 8, DET curves of RF decrease rapidly in different behavior patterns and EER values are all below 0.1, which is better than other algorithms. In summary, the AUC and EER evaluation indexes of the RF algorithm are better than those of the three algorithms so that the RF algorithm is used to train the model.

4.2. Fusion Feature Representation

4.2.1. Overall Performance. In order to compare the performance of three types of features, we mark 10 testers as legitimate users in turn and analyze the performance of three types of features on different users. The experimental results are shown in Figure 9.

In terms of accuracy, three types of features show high accuracy in different users. HM features are more than 84%, while HP features are more than 95%. The accuracy of fusion features is about 95% on average, on a part of user data. The accuracy can be up to 99%, which indicates that fusion features are excellent. Although the accuracy of some users' posture features is close to or even higher than the fusion features in terms of accuracy, the accuracy model results may be biased. Then, we compare the F1-score performance of the features. As can be seen in Figure 9(b), the F1-score of HM features is lower with an average of 0.5 or even lower than 0.3 on the user 3. It indicates that the features cannot perform well in the precision and recall ratios. The HP feature is 0.5 lower than User9, and other users are above 0.7. The fusion features have an F1-score of 0.93, since HP features contain more information and get better classification performance. The comprehensive accuracy and the F1-score show that the fusion features have outstanding performance, strong generalization ability, and high discrimination in authentication.

4.2.2. Attack Test. We simulate four experimenters to deliberately imitate legitimate user behavior habits (including finger sliding movements and posture features). In the experiment, each slip is regarded as one authentication and the relationship between the number of attackers and the success rate is analyzed. The experimental steps are as follows: first, we train the legal user data to generate an authentication model and then let the user reoperate and let the attacker observe it. After the attacker opens the same reading content, each attacker takes turns to imitate. Finally, we evaluate the data validation model of both legitimate users and attackers.

As shown in Figure 10, the number of successes per attacker increases with the number of attacks. However, the number of successes is 0 among the first 10 attacks, which means that fewer attacks cannot break the continuous authentication. When the number of attacks reaches 100, the first three attackers succeed for one or three times, while the success rate of the fourth hacker is still 0. It shows that the fusion features have a high degree of discrimination and the selected features are difficult to be imitated. Although the number of successful attackers increases, the success rate cannot rise much after being attacked for 100 times and can achieve only about 4%. Especially after the 400th attack, the success rate continues to be sharply reduced. It shows that the similarity of the attacker's behavioral imitation is gradually deviated. This may be their memory of behavior habits of the imitation object gradually becoming blurred, or it may be that the lack of patience of hackers leads to a reduction in the success rate. Our results show that the fusion features can resist the intentional imitation attack within a certain period of time. To prevent an attacker from continuing an attack through authentication, users can set different authentication time thresholds according to the privacy security level to avoid brute force attacks.

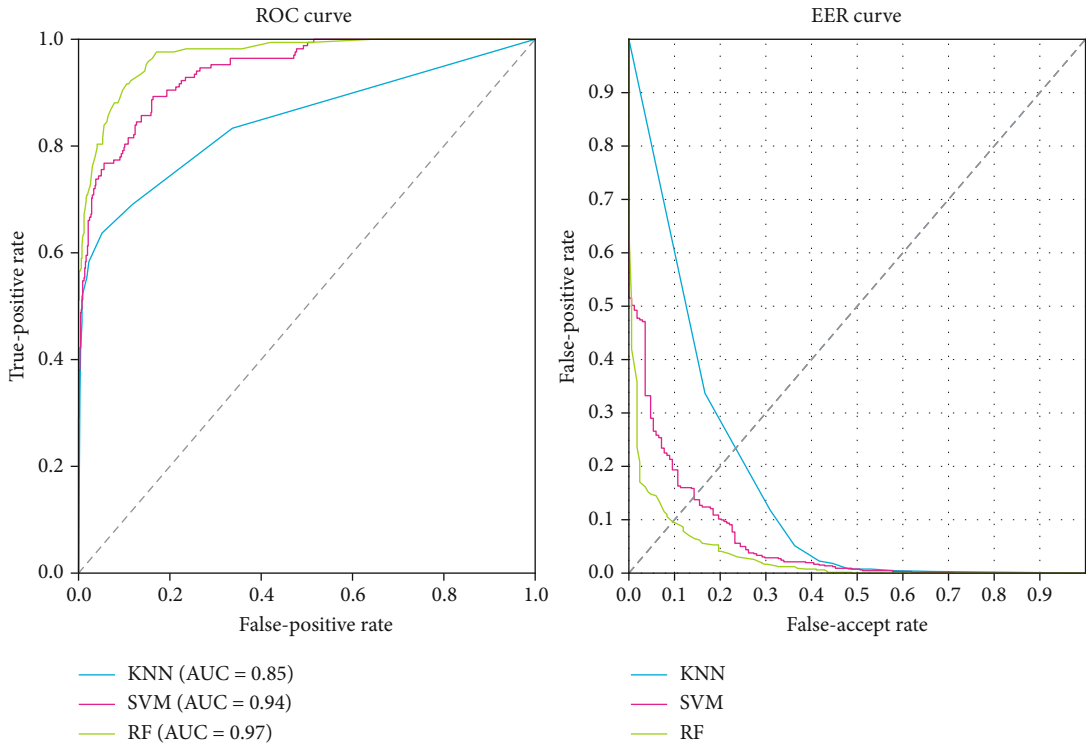


Figure 6: HM feature classification performance.

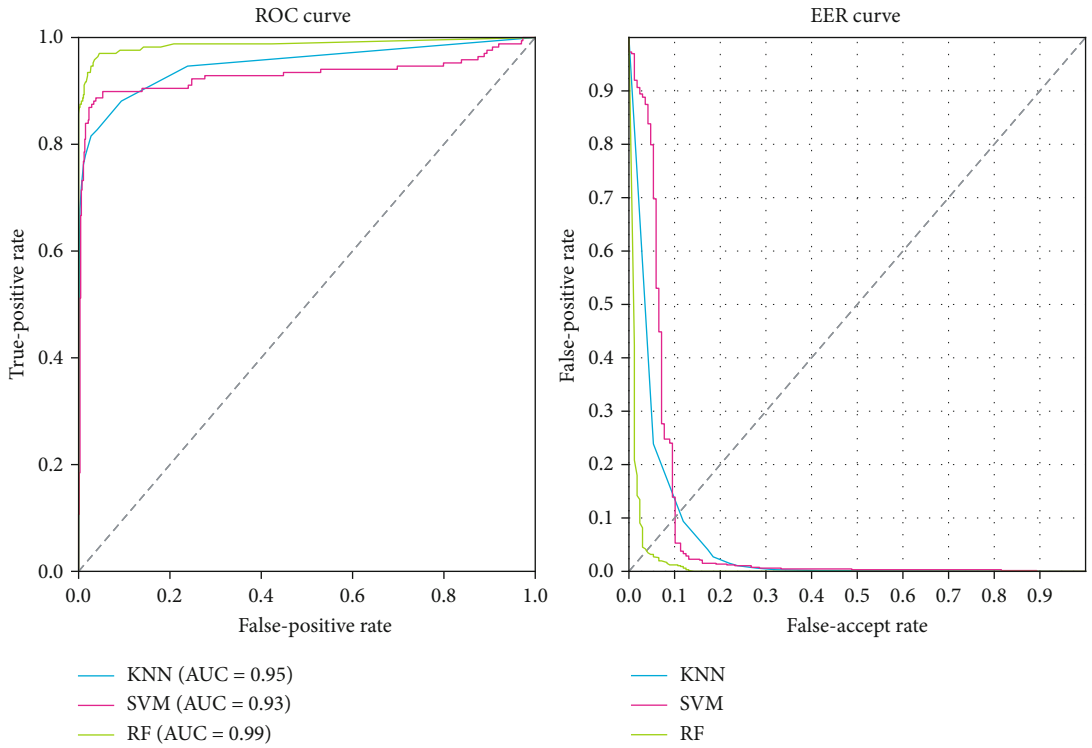


Figure 7: HP feature classification performance.

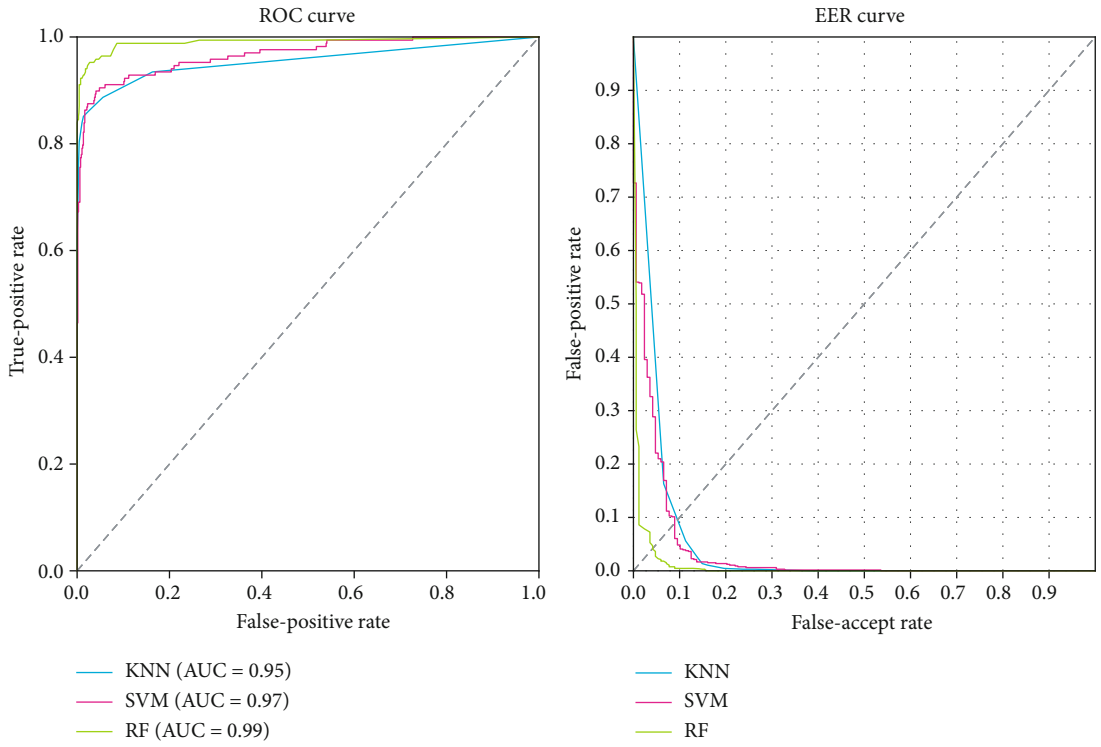


Figure 8: HMHP feature classification performance.

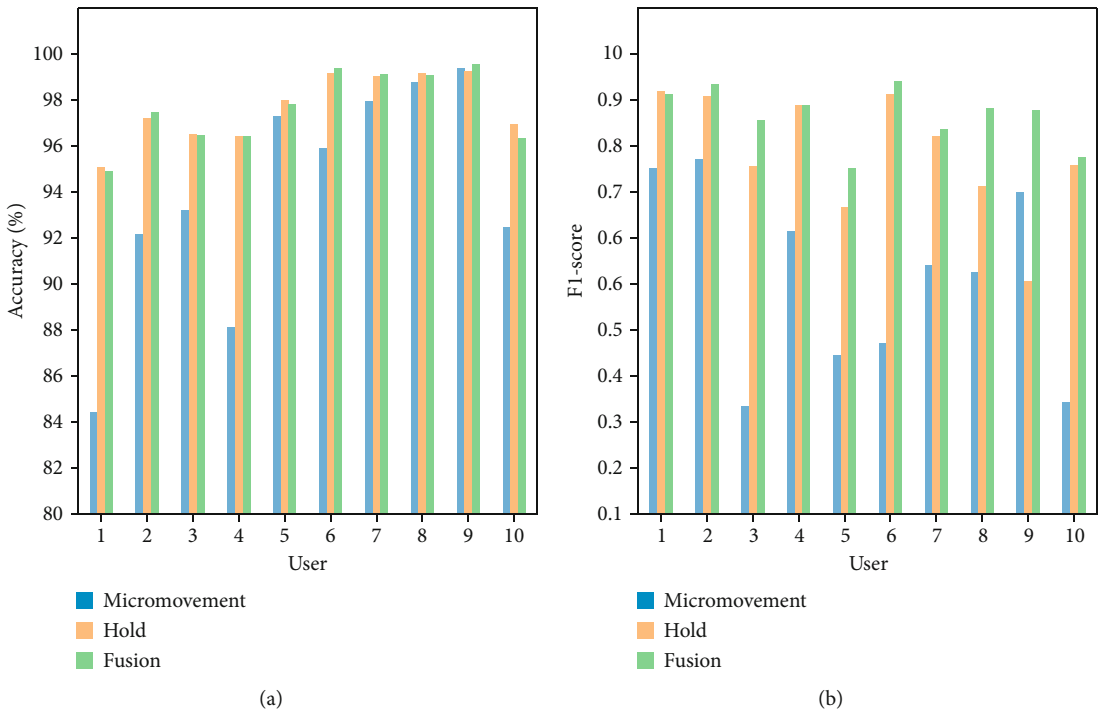


Figure 9: Performance of various features on different users.

4.3. Dimensional Analysis. We use the PCA algorithm to determine the dimension between 13 and 40 and analyze the influence of different dimensions on the feature's classification. It can be seen in Figure 11 that the EER of the dimension is 5.889% when the dimension equals 13, which is higher than the EER value of the original features, indicating that the dimension is too low. This method results in excessive loss of the fusion feature information, while the discrimination is

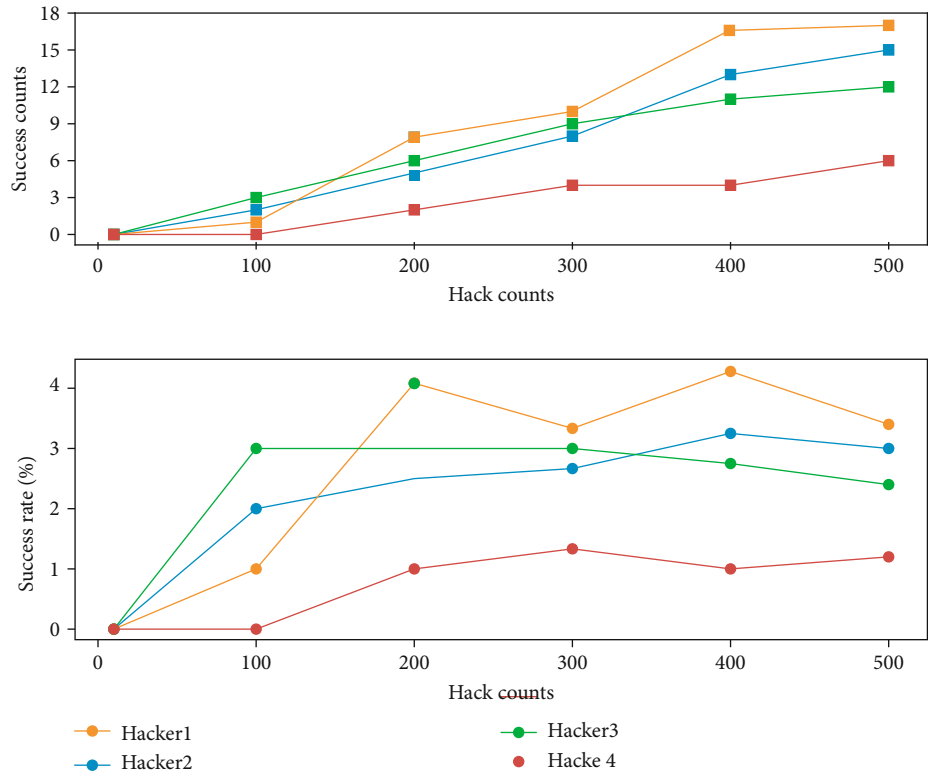


Figure 10: Attack test and analysis of the fusion feature model.

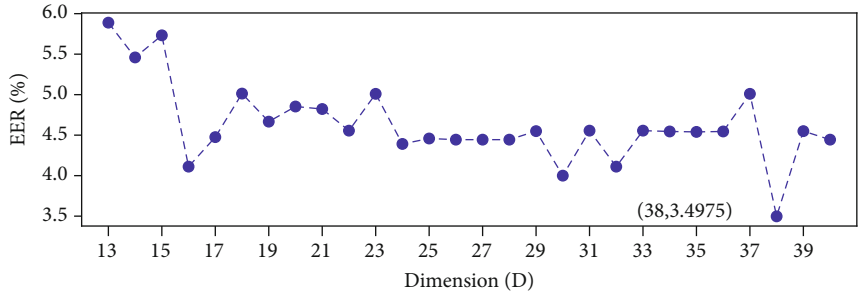


Figure 11: PCA dimension reduction analysis.

Table 3: Comparison of our authentication experiments with the previous work on a smartphone.

Work	Verifier	AUC	EER	F1-score
Wang et al. [17]	Random Forest (400)	0.903	×	0.944
Frank et al. [19]	SVM(RBF)	×	3%	×
Sitova et al. [20]	(1-CLASS) SVM	×	7.6%	×
This work	Random forest (250)	0.99	3.49%	0.937

decreased and the classification performance is deteriorated. The EER gradually decreases with the dimension increasing. EER stabilized at about 4.8% but still is higher than that of the original features. When the dimension equals 38, the EER reaches a minimum of 3.4975%, which is better than the original features. It shows that the noise data of the original features was removed by dimension reduction, which further improves the performance of the fusion features. And the model training speed after dimension reduction is 2.4 s, which is better than before.

4.4. Previous Work Comparison. We compare four research works in terms of different evaluation metrics. As can be seen in Table 3, each work uses a different algorithm to verify the feature effect (the × symbol indicates that the work has not been evaluated).

The final evaluation indicates the combination with the best algorithm. Our fusion features outperformed the first work on the AUC and F1-score performance. The second study had the lowest EER of all studies. The EER of our work is almost in line with that of the second work. Our work achieves a lower EER by comparison to the third work. Overall, our HMHP features are similar to or slightly higher than other works under different metrics. Moreover, our feature

calculation is easy to extract, which is more suitable for continuous authentication scenarios.

5. Discussion

Our proposed scheme achieves better classification results on collected dataset but still suffers from the following shortcomings: (1) the dataset is small and insufficient to support the training of an authentication network with strong generalization capability, and we will subsequently collect more data using smartphones to expand the number of datasets in a future work and (2) because of the limitation of cell phone security, we are unable to evaluate the impact on the energy consumption and memory of mobile devices after the model is deployed to them.

6. Related Works

As built-in sensors and accessories become an available source of sensing data on the current mobile devices [13, 21, 22], researchers have begun to study users' interactions with the devices to establish users' continuous authentication models through behavioral data [14, 23], for employing ordinary protection and mitigating the threats [24, 25].

The line of study starts from continuous authentication by users' interaction with the touchscreen, including touch sliding, tapping, swiping, and scrolling. Frank et al. [19] investigated in 2012 how users perform each finger gesture on the touchscreen when they interact with their mobile devices. They introduced a touchalytics-based approach that authenticates users by behavioral touch features and achieved satisfactory accuracy results. Feng et al. [26] focused on the typing habit of users and experimented on mobile devices with typing dynamics. Similarly, Zheng et al. [27] proposed a user verification and authentication mechanism through tapping a behavioral biometric. Later, Wang et al. [28] designed a differential evolution mechanism to select a comprehensive feature set of keystroke biometrics and improved the reliability and robustness of authentication systems. However, this kind of biometric information is not always available when the user undertakes other activities and the touchscreen is not triggered. Usually, these methods may not work well in uncontrolled and unsupervised testing environments.

The continuous authentication schemes also make use of built-in sensors such as the gyroscope, accelerator, and orientation sensor, to leverage sensory data and profile users' interaction for continuously distinguishing user identity. Conti et al. [29] proposed a system that uses accelerometers, gyroscopes, and magnetometers to authenticate users. Buriro et al. [30] profiled users in terms of their hand movement patterns when they unlock their smart phones. Filina and Kogos [31] showed that four movements of hand waving are sufficient to user authentication. However, most of existing hand motion pattern-based authentication methods are not completely transparent since they require users to do an appointed action repeatedly. In addition, data from motion sensors may be unavailable if users are accustomed to putting the smart phone on the table when using it.

The studies of unimodal continuous authentication focus on a single method to facilitate the collection of raw information from sensors. Such unimodal methods need to face some practical problems, such as limited application scenarios, long training time, and high resource consumption, which are addressed by utilizing multimodal continuous authentication methods. Multimodal systems can classify users in terms of information fusion from various modalities. Mahbub et al. [32] studied a multimodal continuous authentication method based on face, touch dynamics, and location information which is captured from a front-facing camera, touchscreen, built-in sensors, GPS, Bluetooth, Wi-Fi, and so on. However, this method is not widely applicable to mobile devices since not all mobile devices have these components. Moreover, mobile devices have limited storage and battery which cannot support the data collection and process from so many components. Sitova et al. [20] proposed a set of biometric features, called hand movement, orientation, and grasp (HMOG) to almost achieve EERs of 7.16%. Our work differs from the work in [20] on the following aspects: (i) we mainly use swiping data rather typing data which make our method able to generally be extended to other mobile devices because of the lake of the virtual keyboard on lots of wearable devices and (ii) our hand motion features are based on the acceleration patterns generated by users while using devices.

7. Conclusion and Future Work

We present a multimodal continuous authentication method which contains static interaction patterns and dynamic interaction patterns on mobile devices. We use HM and HP fusion features for authentication and to study the accuracy and efficiency of new features in continuous authentication. The relationship between fusion features and behavioral habits is analyzed. We collect more than 10000 pieces of data from 10 testers. Experiments show that the average accuracy of fusion feature authentication is about 97% and our algorithm can achieve good authentication results under the verification of multiple classification algorithms. This shows that the fusion features effectively capture user's behavior habit information.

Although the accuracy of feature authentication is high, the daily behavior of users changes. Over time, for example, after a few months, the behavior of the user to pick up the phone, the way to slide, the speed of sliding, etc. will change slightly over time. The authentication of features will make a difference, resulting in a decrease in accuracy. When this difference is gradually accumulated to a certain extent, the trained model will not be available. In fact, since the user performs continuous authentication every day, the actual interaction data of the user can be collected every time and the user's real behavior habits can be recorded. We believe that although behavioral habits change gradually over time, the model can gradually learn the changes in behavioral habits and establish a continuous dynamic update model. That is, when the accuracy of the existing authentication is lower than the set threshold, the model is automatically updated to ensure that the authentication is valid. This will avoid

showing the security risks caused by the registration authentication again and avoid the trouble of repeated registrations. We believe that more in-depth research can be done in future work.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant 61802252.

References

[1] How smartphones are on the verge of taking over the world, 2013, http://www.nydailynews.com/life-style/smartphones-world-article-1.1295927.

[2] K. Alrowaily, M. Alrubaian, and D. A. Mirza, "Smart phones security-touch screen smudge attack," in *Proceedings of the International Conference on Security and Management (SAM)*, 2012.

[3] B. Schneier, "Two-factor authentication: too little, too late," *Communications of the ACM*, vol. 48, no. 4, p. 136, 2005.

[4] X. Liu, Y. Li, R. H. Deng, B. Chang, and S. Li, "When human cognitive modeling meets PINs: user-independent inter-keystroke timing attacks," *Computers & Security*, vol. 80, pp. 90–107, 2019.

[5] J. Bonneau, S. Preibusch, and R. Anderson, "A birthday present every eleven wallets? The security of customer-chosen banking PINs," in *International Conference on Financial Cryptography and Data Security*, pp. 25–40, Berlin, Heidelberg, 2012.

[6] V. M. Patel, R. Chellappa, D. Chandra, and B. Bargello, "Continuous user authentication on mobile devices: recent progress and remaining challenges," *IEEE Signal Processing Magazine*, vol. 33, no. 4, pp. 49–61, 2016.

[7] H. Lee, J. Hwang, S. Lee et al., "A parameterized model to select discriminating features on keystroke dynamics authentication on smartphones," *Pervasive and Mobile Computing*, vol. 54, pp. 45–57, 2019.

[8] N. Z. Gong, M. Payer, R. Moazzezi, and M. Frank, "Forgery-resistant touch-based authentication on mobile devices," in *11th ACM on Asia Conference on Computer and Communications Security-ASIA CCS'16*, pp. 499–510, New York, New York, USA, 2016.

[9] Y. Yang, B. Guo, Z. Wang, M. Li, Z. Yu, and X. Zhou, "BehaveSense: continuous authentication for security-sensitive mobile apps using behavioral biometrics," *Ad Hoc Networks*, vol. 84, pp. 9–18, 2019.

[10] A. Messenia, S. Sur-Kolay, A. Raghunathan, and N. Jha, "CABA: continuous authentication based on BioAura," *IEEE Transactions on Computers*, vol. 66, no. 5, pp. 759–772, 2017.

[11] C. Bo, L. Zhang, T. Jung, J. Han, X.-Y. Li, and Y. Wang, "Continuous user identification via touch and movement behavioral biometrics," in *2014 IEEE 33rd International Performance Computing and Communications Conference (IPCCC)*, pp. 1–8, Austin, TX, USA, 2014.

[12] 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'17 Abstracts*, pp. 67–78, USA, 2017.

[13] Z. Syed, J. Helmick, S. Banerjee, and B. Cukic, "Effect of user posture and device size on the performance of touch-based authentication systems," in *2015 IEEE 16th International Symposium on High Assurance Systems Engineering*, pp. 10–17, Daytona Beach Shores, FL, USA, 2015.

[14] H.-H. Hsu, K.-C. Tsai, Z. Cheng, and T. Huang, "Posture recognition with G-sensors on smart phones," in *2012 15th International Conference on Network-Based Information Systems*, pp. 588–591, Melbourne, VIC, Australia, 2012.

[15] J. Tutkuviene and W. Schiefenhövel, "Laterality of handgrip strength: age- and physical training-related changes in Lithuanian schoolchildren and conscripts," *Annals of the New York Academy of Sciences*, vol. 1288, pp. 124–134, 2013.

[16] H. Kubota, S. Demura, and H. Kawabata, "Laterality and age-level differences between young women and elderly women in controlled force exertion (CFE)," *Archives of Gerontology and Geriatrics*, vol. 54, no. 2, pp. e68–e72, 2012.

[17] X. Wang, T. Yu, O. Mengshoel, and P. Tague, "Towards continuous and passive authentication across mobile devices," in *10th ACM Conference on Security and Privacy in Wireless and Mobile Networks-WiSec'17*, pp. 35–45, New York, New York, USA, 2017.

[18] G. Peng, G. Zhou, D. Nguyen, X. Qi, Q. Yang, and S. Wang, "Continuous authentication with touch behavioral biometrics and voice on wearable glasses," *IEEE Transactions on Human-Machine Systems*, vol. 47, no. 3, pp. 404–416, 2017.

[19] M. Frank, R. Biedert, E. Ma, I. Martinovic, and D. Song, "Touchalytics: on the applicability of touchscreen input as a behavioral biometric for continuous authentication," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 1, pp. 136–148, 2013.

[20] Z. Sitova, J. Ledenka, Q. Yang et al., "HMOG: new behavioral biometric features for continuous authentication of smartphone users," *IEEE Transactions on Information Forensics and Security*, vol. 11, pp. 877–892, 2016.

[21] Y. Li, M. Zhuoru, Z. Chenghui, and P. Qingqi, "Mobile platform continuous authentication scheme based on gait characteristics," *Journal on Communications*, vol. 40, no. 7, 2019.

[22] J. Ho and D.-K. Kang, "Mini-batch bagging and attribute ranking for accurate user authentication in keystroke dynamics," *Pattern Recognition*, vol. 70, pp. 139–151, 2017.

[23] A. N. Putri, Y. D. W. Asnar, and S. Akbar, "A continuous fusion authentication for android based on keystroke dynamics and touch gesture," in *2016 International Conference on Data and Software Engineering (ICoDSE)*, pp. 1–6, Denpasar, Indonesia, 2016.

[24] Q. Wang, X. Lin, M. Zhou, Y. Chen, and X. Luo, "VoicePop: a pop noise based anti-spoofing system for voice authentication on smartphones," in *IEEE INFOCOM 2019- IEEE Conference on Computer Communications*, pp. 2062–2070, Paris, France, 2019.

[25] J. Galbally and R. Satta, "Three-dimensional and two-and-a-half-dimensional face recognition spoofing using three-dimensional printed models," *IET Biometrics*, vol. 5, no. 2, pp. 83–91, 2016.

[26] T. Feng, X. Zhao, B. Carbunar, and W. Shi, "Continuous mobile authentication using virtual key typing biometrics," in *2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, Melbourne, VIC, Australia, 2013.

[27] N. Zheng, K. Bai, H. Huang, and H. Wang, "You are how you touch: user verification on smartphones via tapping behaviors," in *2014 IEEE 22nd International Conference on Network Protocols*, pp. 221–232, Raleigh, NC, USA, 2014.

[28] Y. Wang, C. Wu, K. Zheng, and X. Wang, "Improving reliability: user authentication on smartphones using keystroke biometrics," *IEEE Access*, vol. 7, pp. 26218–26228, 2019.

[29] 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 *2011 Acm Symposium on Information*, ACM, 2011.

[30] A. Buriro, B. Crispo, and Y. Zhauniarovich, "Please hold on: Unobtrusive user authentication using smartphone's built-in sensors," in *2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA)*, New Delhi, India, 2017.

[31] A. N. Filina and K. K. Kogos, "Continuous authentication over hand-waving for android smartphones," in *2020 Advanced Technologies in Robotics and Intelligent Systems*, pp. 413–424, Springer, Cham, 2020.

[32] U. Mahbub, S. Sarkar, V. M. Patel, and R. Chellappa, "Active user authentication for smartphones: a challenge data set and benchmark results," in *2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS)*, pp. 1–8, Niagara Falls, NY, USA, 2016.