sAuth: a hierarchical implicit authentication mechanism for service robots

Xiaomei Zhang1 · Pengming Zhang1 · Xin Zeng1 · Yajie Wang1 · Chi-hung Chi2

Accepted: 18 March 2022 / Published online: 30 April 2022 © The Author(s) 2022

Abstract
With advances in networks, Artificial Intelligence (AI), and the Internet of Things, humanoid robots are rising in many areas, including elderly care, companion, education, and services in public sectors. Given their sensing and communication functionality, information leakage and unauthorized access will be of big concern. Very often, authentication techniques for service robots, especially those related to behavioral identification, have been developed, in which behavior models are created using raw data from sensors. However, behavioral-based authentication and re-authentication is still an open area for research, including cold start problems, accuracy, and uncertainty. This paper proposes a hierarchical implicit authentication system by joint built-in sensors and trust evaluation, coined sAuth, which exploits sensor data-based sliding window trust model to identify the service robot and its expected users. In order to mitigate the fluctuations of identification results in the real world environment, the trust evaluation is computed via combining the weighted intermediate identification probability of various small sliding windows. The performance of sAuth is evaluated under different scenarios where we show that (i) approximately 5–7% higher accuracy and 2–18% lower equal error rate can be achieved by our method compared to other works; and (ii) the hierarchical scheme with joint sensors and trust sliding windows improves the authentication accuracy significantly by comparing it with only sensor-based authentication.

Keywords Behavioral-based identity · Implicit authentication · Trust evaluation · Sliding windows
1 Introduction

Service robots that store sensitive and private data have become increasingly popular in our daily life [1–3]. It covers areas from education, age care, to companion. Increasing intelligence in these robots further adds their potentials. Through interaction with their human partners, these robots not only do jobs that are usually considered as dirty, dull, distant, dangerous, and repetitive, but they also try to understand the cognitive aspect of human. This is usually done through sensors in these robots for data capturing and the analytic capabilities inside. Some good examples include Pepper from SoftBank Robotics, Quori from Immersive Kinematics Lab, and Zenbo from ASUS. Service robots and humanoid robots usually have an Android-based software interface through some kind of pad-like hardware interface. Furthermore, there are many situations where these robots are expected to interact only with a limited number of people.

With the deployment of service robots in our daily life, security is an increasing concern. To prevent their private data from unauthorized access, traditional explicit authentication methods are employed using a password, personal identification number (PIN), face, fingerprint, or secret pattern [4]. Previous work has shown that such solutions provide limited security because of several reasons. Firstly, they make these robots vulnerable to guessing, shoulder surfing, smudge, and spoofing attacks [5, 6]. Secondly, after the authorized user has started the initial interaction with the robot, it is challenging to detect intruders after login [7]. Thirdly, expenses of extra hardware, data acquisition time, and quality requirements of samples [8] are significantly high. To address these issues, it is essential to investigate a continuous authentication system that authenticates a user during the runtime of user-robot interaction.

The continuous authentication mechanisms aim to learn the characteristics of user-robot interaction in terms of their behavioral data. They essentially adopt behavioral biometrics-based measurements, typically including sensor-based authentication [9] and touchscreen-based authentication [10]. Touchscreen-based methods identify users through a screen touch input, which is the data generated by users’ swipe gestures on the touchscreen of service robots. And authentication verifies the user’s identity based on how he/she interacts with the robot services. However, this type of authentication takes a major expenditure of time to collect the desired data since users are constrained to carry out a specific task, e.g., swiping or typing. These studies cannot guarantee that intruders will be blocked in time before they access sensitive data or services. In this paper, we propose an unobtrusive authentication scheme, which utilizes built-in sensors to profile the hand(s) micro-movements on the interface pad without taking any extra action.

More recently, built-in sensors-based authentication schemes have attracted significant attention. Conti et al. [11] proposed an authentication method that utilizes accelerometer sensors and orientation sensors as the biometric measure tools. Burriro et al. [12] design a build-in unprivileged sensors-based authentication approach according to users’ hand movement ways. Most of those studies have achieved high accuracy and low EER. However, according to the European
standard for access-control systems [13], there is still a need for further effort to gain higher accuracy. In realistic scenarios, users interact with the robots through various activities, which can disturb the authentication models. However, only a few exiting works consider the real interference when they try to improve the accuracy of authentication. In this paper, we propose a sliding window approach for trust estimation, and this tries to take into consideration the dynamics of authentication. The aim of our solution is to evaluate the trust value of a user by leveraging the recent authentication result and historical trust levels. On the one hand, our integrated scheme achieves better accuracy and lower EER since it reduces some misjudgments caused by interference. On the other hand, the integrated scheme takes different situations and previous samples into consideration, and this increases the robustness of the authentication mechanism.

We implement and evaluate the authentication performance of our scheme in different scenarios. In the study of this paper, we use a mobile phone to approximate the pad hardware interface of the service robots. This is to allow us to perform and repeat experiments without loss of generality. The main contributions of this paper are summarized as follows:

- We present a sliding windows-based hierarchical user authentication architecture, which is expected to achieve a low error rate, high accuracy and good robustness. We utilize the build-in sensors to train the behavioral model as the front-level authentication, and then, design a back-level authentication, which combines the intermediate authentication probability from the front-level windows into a final decision.
- We propose a trust evaluation mechanism over sliding windows, associating with calculating the multiple historical trust values, which reflect dynamic variation of impacts from various interference. A map method leveraging sigmoid log function is introduced to reduce the fluctuation of authentication results.
- In our performance study, our front-level authentication method achieves high ACC up to 97.2%. Furthermore, compared to the front-level model, our hierarchical method has better accuracy and robustness of authentication where the FAR and FRR declined by 11% with impostors.

2 Problem formulation

In this section, we introduce behavioral features on 3-dimensional sensors, interference models and evaluation indicators.

2.1 Behavioral features on 3-dimensional sensors

Due to the differences in the physiological characteristics and behaviors, different individuals have different characteristics such as holding posture and strength when they operate the devices [14, 15]. These characteristics will lead to different shaking ranges of mobile phone and rate of change of vibration, which shows differences in
individual behavior. Some studies have shown that different behaviors can be recognized by sensors [16], indicating that sensors can detect physical behavior characteristics. In this section, we collect behavioral information based on the three sensors of the smart phone: accelerometer, gyroscope and magnetometer. We analyze the changes of the sensors in the three axes of X, Y, Z, and study how to characterize the behavior changes of users while using the device.

Figure 1 shows the changes of the three sensors, accelerometer, gyroscope and magnetometer, in XYZ axis when the user operates the mobile phone. Each segment in the figure denotes the data recorded by the sensor in each axis when a user is swiping the screen. We can see that different users have different behavior changes. On the acceleration sensor, the data presents layered changes, and most of the data generated by different users are within a certain range. On the Y axis of the accelerometer, the variation of user 0 is below 4, while that of user 1 is almost above 4.5, and that of user 2 is the value between them. On the Z axis of accelerometer, the change of user 0 is at about 9.0 and that user 2 is at about 8.5, which indicates that there are differences in user behavior data on each axis of the accelerometer. On the gyroscope sensor, the data amplitude of different users is more obvious. On the X axis, user 2 has the smallest amplitude change, in about 0.2; User 0 has a small variation in the order of 0.3, while user 1 has the largest amplitude in the order of 0.5. This indicates that the mobile phone has a large lateral sway and obvious changes in features when the user is using it. Then on the Y axis, user 0 varies between 0 and 0.2, user 1 data varies mostly between -0.2 and 0.2, and user 2 varies

![Fig. 1 Analysis of axial data of different sensors](image-url)
between -0.1 and 0.1. The maximum value of the user’s data varies by a wide range, indicating that the posture of holding phone also changes slightly along the vertical direction. Additionally, the magnetometer is the equipment that senses the intensity and angle of geomagnetic and determines the direction of device. The variation of magnetic force in different orientations indicates the posture discrepancy of holding devices with different orientations, thus it could reflect the behavioral differences among users. In this paper, it is used to detect magnetic change. As shown in the subgraphs from G to I of Fig. 1, the stratified difference of the magnetometer data in each axial direction is very obvious, indicating that the magnetometer data are more differentiated.

Here we also use the cosine similarity to identify the correlation between users. On the X axis of the acceleration sensor, the values of the cosine similarity are 0.3569, 0.6998, 0.476 between user 0 and user 1, between user 0 and user 2, between user 1 and user 2, respectively. On the X axis of the gyroscope sensor, the values of the cosine similarity are 0.044, -0.1107, -0.0318 between user 0 and user 1, between user 0 and user 2, between user 1 and user 2, respectively. On the X axis of the magnetometer sensor, the values of the cosine similarity are 0.056, -0.0687, -0.0457 between user 0 and user 1, between user 0 and user 2, between user 1 and user 2, respectively. According to the results, there is little correlation between the behavior of users.

The above analysis shows that there are obvious differences among various users in the holding posture, mobile phone movements and other behavioral characteristics. Therefore, sensors can be used to continuously track user behavior so that the authentication application can detect user authenticity throughout the entire interaction process. It is difficult for others to impersonate legitimate users because the behavior characteristics are difficult to imitate, which can avoid the problems of feature forgery and password guessing. And the continuous authentication process is carried out in the background, so that this method can improve users’ experience since the process is unobtrusive for users.

2.2 Interference models

The red box of Fig. 2 shows that the classifier misidentifies imposters as legitimate users, while in the green box, the legitimate users are misidentified as imposters. The unexpected operation behavior of the impostor may be similar to that of legitimate users, which may lead to the misjudgment of the classifier, or it may be the intentional attack of the impostor. Moreover, the probability value of user authentication fluctuates greatly, and some authentication results over at the threshold of the classifier. It may appear that the classifier classifies legitimate users into impostors at some time, because the trust probability value of legitimate users in the classifier is relatively low. However, looking at the window before the false identification area appeared, the authentication results for multiple actions by legitimate and impostors are all within the normal range. In this paper, we propose a method, which considers the history value of the adjacent authentication probability into the classifier’s
current decision, so that the user authentication result can become accurate through decreasing the influence.

2.3 Evaluation indicators

We used three indicators to evaluate algorithm performance: False Accept Rate (FAR), False Rejection Rate (FRR), and Accuracy (ACC). FAR represents the proportion of negative cases misclassified as positive cases in the total number of negative cases. The definition is shown in Formula 1, where FP is the false positive sample number and TN is the true negative sample number. FRR represents the percentage of positive cases that are wrongly classified as negative cases in the total number of positive cases. The definition is shown in Formula 2, where TP is the true number of cases and FN is the false negative number of cases. The former measures the reliability of the algorithm while the latter measures the ease-of-use of the algorithm. The higher FAR is, the more vulnerable the model is to attack. Meanwhile, the higher FRR is, the more unfriendly the model authentication is to users. ACC represents the classification accuracy of positive and negative samples (Formula 3), which is used to evaluate the overall classification performance of the algorithm. Obviously, the lower FAR and FRR values and the higher ACC values, the better performance of the model.

\[
FAR = \frac{FP}{FP + TN} \quad (1)
\]

\[
FRR = \frac{FN}{TP + FN} \quad (2)
\]
3 System design

Firstly, we develop an application to collect three kinds of sensor data through the system API. To avoid environmental constraints, application runs in the background as a separate service. For each sample of sensor data, timestamp measurement is collected through the instruction code when the user uses the application, and the information is recorded as soon as the system event occurs. Then, the original data is processed with data denoising, feature extraction and screening, etc. And the classification model is used with a machine learning algorithm. Its output is set as the probability of authentication. According to the historical authentication result output by the classifier, the probability value of historical authentication is recorded through the weighted-sliding-window-based mechanism, and the weighted trust value was projected inside it. Finally, users with trust value that is higher than the threshold are legitimate, while those with trust value that is lower than those are imposters. We have collected data of three kinds of sensors in the initial stage, however, it needs to be further analyzed for the combination characteristics of sensors, and finally, we could determine which kinds of sensor data to collect. We discuss it in the evaluation section. The implicit authentication framework is shown in Fig. 3.

3.1 Data processing and feature engineering

Original Data Format. Sensors built in the phone automatically record original data and report data back to the operating system as raw events. Taking the Android system for example, each raw event includes the instantaneous attitude change of the phone measured by the accelerometer, the tiny rotation of the phone measured by the gyroscope, and the magnetic field change of the phone measured by the magnetometer.

\[
Acc = \frac{TP + TN}{TP + FP + TN + FN}
\]
by the gyroscope in radians per second, and the magnetic induction strength in the Angle and direction of the phone recorded by the magnetometer during user operation. The original data format recorded by the sensor is shown in Table 1.

**Data Denoising.** The data analysis shows that the sensor has noise data in X, Y and Z axes. This is because: a) The sensors are so sensitive that even when you hold the phone stably, the data from the sensors will change, causing interference. b) In the preparation stage of the experimenter, the sensor will also record many values, resulting in sudden changes at the beginning and end of the scratching track. c) During the experiment period, the sudden and abnormal movements of the experimenter will also cause the abnormal changes in the data. These anomalies will not only fail to extract relevant feature information, but also affect the extraction of other features. To deal with those problems, the following processing is done: for the redundant noise points generated at the beginning of the experiment, the truncation method is adopted to remove the data of the starting point and the ending point; for noise data generated by abnormal behavior, the smooth filtering method is used to remove noise. We show the formula as follows:

\[
 x_k = \frac{1}{H} \sum_{n=1}^{H-1} x_n
\]

where \( x_k \) is the kth data point along the X axis of the sensor, and \( H \) is the total number of sensor data points collected in a period of time t.

**Feature Extraction.** Through the analysis of the three kinds of sensor data, it is found that the change rules of different sensors are not the same. Therefore, through combination with the characteristics of the data of each sensor, multiple features are extracted from different sensors to represent the behavior changes of users. Features extracted by each sensor are different, but they are all shown in X, Y and Z axis directions. Different behaviors perform differently in the three axes of the sensor. Therefore, in addition to extracting the types of corresponding behavioral characteristics of each sensor, the data of hand movements in different axial directions are also recorded. Table 2 lists the feature types corresponding to each sensor and the number of extracted features.

**Feature Subset Selection.** The features selected above are processed by mutual information method. According to the behavioral feature sets of different sensors, the contribution rates of each feature on the sensor are calculated and

| Feature     | Definition                               | Unit  |
|-------------|------------------------------------------|-------|
| UserID      | User ID                                  | No    |
| Timestamp   | point of time                            | ms    |
| Mag_X, Mag_Y, Mag_Z | induction value of magnetometer on the three axes of X, Y and Z | T/s   |
| Gyro_X, Gyro_Y, Gyro_Z | Rotation rate of gyroscope on the three axes of X, Y and Z | rad/s |
| Acc_X, Acc_Y, Acc_Z | Acceleration value of accelerometer on the three axes of X, Y and Z | m/s²  |
the advantages and disadvantages of the selected features are analyzed. That is to analyze the importance of a user’s behavior in different feature sets. The calculated results of the mutual information method are shown in Fig. 4, in which Fig. 4a is the characteristic contribution rate of the accelerometer, Fig. 4b is the characteristic contribution rate of the gyroscope, and Fig. 4c is the characteristic contribution rate of the magnetometer.

The results are shown in Fig. 4. Firstly, the results of accelerometer are shown in the Fig. 4a, it is obvious that its maximum and minimum values of each axis have a high characteristic contribution rate, the contribution rate of the amplitude difference is also relatively large, and the contribution rate of features related to X axis rank first, while that of features related to Y axis and Z axis is lower. It indicates that the X-axis related features are the most obvious changes, and users are more likely to swipe the screen horizontally. However, the contribution rate of the difference is small between the minimum value point and the end point, which ranks lower. The difference between the initial point and the maximum point on the Y axis is the smallest, indicating that in terms of longitudinal attitude changes of the equipment, most users’ operating behaviors change little from the initial point to the maximum point, and the difference is low. Analysis for the gyroscope
is found in Fig. 4b, the contribution rate of features related to the Y axis is relatively high, that of the rate of difference changing between the initial point and the maximum point on the Y axis is up to 12.09%, leading to the contribution of it becomes the highest among all features. This indicates that the users shake differently in the longitudinal direction when operating the mobile phone, which may be caused by the different swiping speed of the hand and the different tapping intensity of the screen. For the results of magnetometer in the graph Fig. 4c, it is found that the characteristic contribution rate of the standard deviation is very small for all axes, and the contribution value is even zero on the Z axis. This indicates that the average variation of users’ behavior characteristics on the magnetometer is relatively stable, and is lack of information, so it does not have much differentiation. In general, features on magnetometer have higher contribution rate, followed by gyroscope, and then, accelerometer, indicating that the three kinds of sensors have different data changes when combining features in different dimensions. What’s more, we can see from Fig. 4c, for all sensors, data on the Z-axis change steadily too. But among the overall features, the degree of relevant features differentiation on the Z-axis is higher than that of others. In view of specific features on each dimension, the change rate of sensor data is marked with a high contribution value, while the amplitude difference is the second one, and the maximum and minimum values come last.

The importance of each feature can be concluded by their contribution rates, but the final selection of features is still not available when only the contribution rate is known, while some features may correlate with each other. In this chapter, we use
the Features Correlation Matrix (FCM) to calculate the relation between features, extendedly, analyzes the correlation of different sensor data on each dimension to reduce feature redundancy.

Figure 5 is the thermodynamic diagram of the correlation matrix between different sensors’ behavior characteristics. Heat map matrix corresponding to the behavior characteristics of the sensor. Red represents positive correlation, while blue represents oppositely, and the darker the color, the more relevant it is. As can be seen from Fig. 4, there is a correlation between the features, but the strength is different. For accelerometer, for example, there are significantly positive connection among the maximum, the minimum and the average that features of both the X axis and the Y axis, and the features of the Y axis and Z axis change in the characteristics of strong negative correlation, correlation between some characteristics and even close to 1.0, it shows that more than one characteristic changes under some action, and the associated features also change causally, forming the characteristic information redundancy. Among the features related to the gyroscope, they are correlated and redundant little except for the two features: the starting point and the ending point, because of a strong correlation between them. Combined with the contribution rate of gyroscope in Fig. 4, it can be observed that the change rate reflects accurately the stable trend of the changes in action, which is clearly differentiated between users. As for the features relative with the magnetometer, it is found in the Fig. 4c that, the other corresponding features have strong correlations except for the standard deviation feature, whose correlation degree can reach 100% in some features, and the redundancy is great. However, it is found in Fig. 5 that the contribution rate of some features is relatively large, it can be indicted that features with high correlation cannot be simply excluded, while it is possible to have a higher degree of discrepancy differentiation in combination features subsequently. Through all the analysis, the features with minimal contribution or high redundancy can be removed.

3.2 Machine learning models

Optimization of Super Parameters. The super parameters also have an important impact on the performance of classifiers. Before the classification algorithm starts to compare, the parameters need to be adjusted to make the classifier perform optimally under the same data set. Since KNN is only related to the number of adjacent points K, only one parameter needs to be adjusted for KNN. SVM can improve
performance by selecting different Kernel functions and penalty parameter C, so two parameters need to be adjusted for SVM. RF is related to the number N of trees and the depth D of the decision tree, and two parameters need to be adjusted. In this section, we use the grid search method to search parameters, and analyze the influence of parameters on the accuracy of the algorithm through the five-fold cross-validation method. In order to search the optimal parameters quickly, some sample data were randomly selected for training and the parameter adjustment of the model. The experimental results are shown in Fig. 6.

It can be seen that the accuracy of KNN algorithm is high when K is 1, and then, decreases gradually with the increase of K. However, with the increase of penalty parameter C, the accuracy curve of RBF kernel increases rapidly, while the linear kernel accuracy curve increases slowly. However, the linear kernel is always more accurate than the RBF kernel, and the linear kernel reaches the maximum when C is 100. In the RF graph of Fig. 6, the random forest algorithm is more sensitive to the number of initialization trees N. When the N is 10, the accuracy of RF is lower than that of other parameter values selected by other algorithms. When the N is 100 and D is 10, the classification accuracy of the model is higher than that of other parameters. It shows that increasing the number of trees cannot improve the model performance while the tree depth is low. The accuracy of RF algorithm is very low when the tree depth D is less than 10, and then, rises rapidly with the increase of the tree depth. When the D is 25 and the N is 200, the RF algorithm is optimal.

Comparison of Model Performance. After the optimal parameters are selected for the algorithm, some other sample data are randomly selected to train the classification model, and it is evaluated by AUC and accuracy. The results are shown in Fig. 7.

As can be seen from Fig. 7, the AUC area of the classifier is all above 0.96, indicating that the selected behavioral characteristics are highly discriminative, which can effectively identify real or imposters. Before the True Positive Rate (TPR) reaching 0.6, the ROC curve of the three classifiers shows an upward trend, indicating that the algorithm responds quickly. When the TPR is after 0.7, the ROC curve of KNN algorithm changes slowly, indicating that its misclassification begins to increase and

Fig. 6 Super parameter analysis of different classifiers
its performance is unstable. Moreover, its overall AUC value is lower than that of the other two algorithms and its performance is poorer. The AUC value of SVM is slightly lower than that of RF, and the change of its curve of ROC gradually slows down, as well as the corresponding accuracy is lower than that of RF. So, the performance of SVM is between KNN and RF. In the algorithm of RF, the accuracy is the highest and the ROC curve is relatively sensitive to features. Consequently, it can be analyzed comprehensively that the RF is rather stable and accurate, so it is more suitable for the feature sets.

### 3.3 Cumulative weighted sliding window-based authentication

The experimental results show that the fusion features based on multi-sensors perform better in the authentication. However, it is important to note that legitimate users sometimes have abnormal usage behaviors, which leads to the results of the classification model are biased. Moreover, the accuracy of the model itself is limited, and some behaviors of users will be misjudged. In most cases, the user’s operation behavior is continuous in a period of time, and the same user’s behavior is not only relatively similar, but also its change is fairly stable. Therefore, the influence caused by users’ misoperation or model’s misjudgment can be corrected according to the recent historical authentication results, which can reduce users’ misidentification rate and the False Rejection Rate, and improve model accuracy and user experience.

First of all, we set a sliding window with a certain size to record the user’s recent authentication results. At the same time, the set of fixed authentication times is converted into a window as the probability value of the current window. Then, the authentication results in multiple windows are combined through the sliding window mechanism. We observe that the closer the authentication results are to the present, the more likely they are to reflect the behavioral changes of the current user.

![Performance of different classifiers in feature set](image)

**Fig. 7** Performance of different classifiers in feature set
Therefore, we introduce the weighting factor and assign different weights to the trust value of each window to get the trust value after the weighted operation. In other words, within a certain length of the window, the real behavior changes of users in a continuous period can be judged according to the comprehensive change state of the historical trust value. According to the trust value distribution interval of the two types of users, a threshold is set to detect whether the current operation is derived from an imposter.

1) Sliding-window-based Trust Value Authentication Principle: First, we collect the behavior feature vector of user, and according to them, the classifier trained before is capable to output every probability of passing the authentication in each time slot, and the multiple authentication results are recorded in the array of the window with size T, to take the average probability of window and get primary trust values in each window, then we endow the sliding window of length K with different weighting factors. The importance of Windows increases gradually from far to near, so the weighting factor F of the window becomes larger and larger. In the current latest authentication window, the corresponding F can reach the maximum of 1, on the contrary, in the oldest window, the minimum is 1/K. After that, the weighted probability values of the sliding window with the length of K are accumulated to obtain the originally integrated value of the weighted trust values. By averaging the integrated trust value, the average weighted trust value is obtained, and then, it is mapped to the interval from negative one to positive one, and the user authentication result in the current window is judged by the threshold value. During continuous authentication, the window discards the probability value in the previous window, while sliding forward. At the same time, the above process is repeated to continuously detect the user. The sliding-window-based trust value model framework is shown in Fig. 8.

2) Trust Value Generation and Update: The trust value represents the change of the user’s recent authentication results, and its generation is inseparable from the mean value \( P \) of \( T \) authentication probability in the window, the length K of the sliding window and the weight factor F assigned to each window. In Fig. 8, after the classifier outputs the probability of authentication, the mean \( P_r \) of the probability of authentication in each window is calculated. The calculation formula is as follows:

\[
P_r = \frac{\sum_{n=1}^{H-1} P_k}{T}
\]

According to the analysis of the authentication principle in the above section, less weight should be assigned to the historical observation value in the past according to the time point when the window is generated. According to this requirement, the weighting factor \( F_r \) of the \( r-th \) window of a sliding window with length K should decrease with the decrease of \( r \). Therefore, \( F_r \) is defined as:

\[
F_r = \frac{r}{K}, 1 \leq r \leq K
\]

After that, \( P_1 \) to \( P_K \) is multiplied by the weight factor \( F_r \) corresponding to the current window \( r \), and the weighted trust values within the sliding length K are accumulated.
to obtain the initial weighted trust value $WT$, which is averaged on the sum of factors, and the calculation formula is defined as:

$$P_r = \frac{\sum_{k=1}^{T} P_k}{T}$$  \hspace{1cm} (7)$$

$$TR_c = \frac{\sum_{r=1}^{K} P_r \times F_r}{\sum_{r=1}^{K} F_r}$$  \hspace{1cm} (8)$$

The cumulative weighted moving average (CWMA) mechanism helps to analyze legitimate users’ long-term behavior without causing immediate loss of information...
over time. We use this trust value as a metric to determine whether the current user is an imposter.

3) Threshold Detection and Imposter Identification: In the proposed trust value authentication scheme based on the weighted sliding window, it is necessary to set an appropriate threshold to identify the two types of users. Generally speaking, the trust value should drop rapidly when the current user is an imposter. On the contrary, when the legitimate user operates, the trust value should rise rapidly. Therefore, we use the log function to map the trust value only on interval between zero to infinite to the real number line, to amplify the subtle changes of the trust value and improve the trust value sensitivity [17]. The log function is defined as follows:

\[ WR_c = \log_2 \left( \frac{TR_c}{1 - TR_c} \right) \]  

(9)

After the mapping value is obtained, the sigmoid function is used to normalize the final trust value to the interval of [-1,1]. The sigmoid scale function is defined as follows:

Then, comparing the trust value with the threshold, we set the following detection logic (the threshold value will be determined by experiment in the next section):

(a) If \( TH_c > \text{Threshold} \), it determines that the current device user is a legitimate use and unlocks the device.

(b) If \( TH_c \leq \text{Threshold} \), the current device user is determined to be an imposter, and the device is locked.

4 Experimental results

4.1 Multi-sensor behavior characteristics

As mentioned above, we select the corresponding behavior characteristics for different sensors. The respective authentication performance of these features needs further experimental verification. Therefore, we evaluate the performance of different combinations of sensors’ features. There are seven combinations of the three kinds of sensor data, and 70% of the combined data sets are used for training while 30% for model validation. We adopt evaluation indexes such as the Accuracy (ACC), the Equal Error Rate (ERR) and F1-score for analysis and verification. The results are shown in Table 3, where S1 represents the accelerometer, S2 represents the gyroscope, and S3 represents the magnetometer.

| Sensor type | S1          | S2          | S3          | S1,S2       | S2,S3       | S1,S3       | S1,S2,S3    |
|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| ACC         | 92.32%      | 93.66%      | 95.62%      | 94.34%      | 96.91%      | 97.20%      | 96.36%      |
| ERR         | 7.94%       | 6.90%       | 3.58%       | 5.86%       | 3.74%       | 2.09%       | 3.49%       |
| F1 Score    | 83.09%      | 86.93%      | 91.67%      | 86.22%      | 93.85%      | 97.99%      | 92.66%      |

Table 3: The effect of feature authentication and performance of fusion feature of sensor

[Springer]
As can be seen from Table 3, in terms of single sensor, S3 has the best effect in regard to the selected criteria, indicating that the behavioral characteristics selected by the magnetic perform greatest. Then, the overall performance of the behavior features on S2 is greater than that on S1, indicating that the performance of features selected by the gyroscope is higher than that of the accelerometer, and the change rate of features related to the gyroscope are more differentiated. It can also be observed from the table that among the pairwise combination features, the overall performance of all sensor features combination has been improved, in which the combination of features on S1 and S3 has the best performance, and its accuracy is higher than that of a single sensor S3. This indicates that combined features integrate more information of behavior differences and enhance the differentiation of authentication model. However, when we combine the behavior features of all sensors, the accuracy was found to be only 96.36%, which was lower than the combination of S1 and S3, and the F1 score and the EER (the Equal Error Rate) were also lower. This indicates that different features may interfere with each other when the behavioral features of S2 are added, leading to poor classification performance of the trained model, and instead, the overall performance of the features is reduced. It means that S2 is no longer important and needed in sAuth. Therefore, we select the combined behavioral characteristics of S1 (accelerometer) and S3 (magnetometer) to train the implicit authentication model.

4.2 Comparison with the existing schemes

To analyze the performance of our selected features, we use two metrics, ACC and ERR, to compare the feature sets of different combinations with other works. As shown in Table 4, implications of $s_1$, $s_2$ and $s_3$ have been represented in Section 4.1, $s_4$ is represented as the orientation sensor, ”×” means that the research paper has not evaluated this item.

We can see that the fusion behavior characteristics based on S1 and S3 sensors combined with a random forest algorithm have the highest accuracy and the lowest equal error rate. At the same time, compare Zdeňka [14], Lee [18] and Ours found that under the same type of sensor behavior characteristics, the model authentication

| Works     | Modeling Algorithm | Sensor Type   | ACC   | EER  |
|-----------|---------------------|---------------|-------|------|
| Zdenka [14] | OneClsss-SVM        | S1,S2,S3      | -     | 7.16%|
| Lee [18]   | SVM                 | S1,S2,S3      | 90.23%| -    |
| Ours$_1$   | RF                  | S1,S2,S3      | 96.36%| 3.49%|
| Rahman [19] | DTW                | S1            | -     | 20.36%|
| Ours$_2$   | RF                  | S1            | 92.34%| 7.94%|
| Lee [20]   | KRR                 | S1,S2         | 92.1% | -    |
| Ours$_3$   | RF                  | S1,S2         | 93.34%| 5.86%|
| Chao [21]  | HMM                 | S1,S2,S3,S4   | -     | 4.74%|
| Ours$_4$   | RF                  | S1,S3         | 97.20%| 2.09%|
performance trained with random forest algorithm is better, indicating that different algorithms will also affect the accuracy of the model. Comparing Lee et al. [18] and [20], it is found that too many sensor features will also interfere with each other, resulting in the decline of authentication accuracy. The authentication model trained by Rahman [19] et al. has a high error rate and lower performance than our authentication model. Chao [21] et al. Combined more sensor data. The error rate of the trained model is 4.74 %, which is better than other literature work, but the authentication performance is lower than that of ours. Overall, our authentication model based on multi-sensor adaptive behavior features has better performance.

Among them, S1, S2 and S3 represent the same as the above section, S4 represents the direction sensor, and '-' indicates that the evaluation was not carried out in the research work.

4.3 Performance of trust value evaluation

In order to evaluate the validity of trust value, we respectively collect the behavior information of using the device by legitimate users and imposters in a continuous period of time, set the output result of classifier as probability value, and then, obtain the distribution of the dynamic change of trust value on two types of users through weighted-sliding-window-based mechanism.

As can be seen from Fig. 9, the distribution of trust values between imposters and legitimate users is significantly different. For legitimate users, almost all their dynamic trust values greater than 0, and most of the trust value in 0.75 above, and even some certification results close to the maximum value of 1.0, legitimate users can enhance the trust value to (0, 1] interval, and the trust value of imposters almost are changing below zero, most of the certification result is less than -0.5, and the trust value that close to -1.0 is about 30%, which shows the trust value of imposters can quickly reduce the interval between negative one to zero. Therefore, the threshold value can be set as 0. If the trust value is above the threshold, users will be regarded as legitimate users, and if the trust value is below the threshold, users will be regarded as imposters. In conclusion, the dynamic trust value can effectively identify legitimate users and imposters.

![Fig. 9 Performance of trust value on impostors and real users](image-url)
As shown in Fig. 9, the trust value of some authentication results of legitimate users is close to zero, or even lower than zero. For the imposters, some of the trust values were higher than 0, and some of the trust values are higher than the legitimate users. According to the historical authentication results, misidentifications of classifier caused by misoperation of legitimate users or accidental behaviors of imposters will lead to abnormal change of probability value, which further cause deviation of trust value. Therefore, we analyze the historical change trend of trust values over a longer period of time to evaluate the effect of these trust values. We analyzed the usage behavior of two types of users over a long period of time, and the change trend of trust value is shown in Fig. 10.

As can be seen from Fig. 10, at the beginning, the trust value of the real user is close to 0.5, and then, with the gradual increase of authentication times, the trust value rapidly increases to more than 0.75 in a short time, and then, stabilized at about 1.0. The trust value of fake users continued to decline, mostly below - 0.5, and then, gradually approached - 1.0. This shows that in the case of long-term use, with the increase of authentication times, the trust value of the two types of users changes more and more, and the user distinction is more obvious. The effectiveness of the mechanism is verified in the continuous authentication environment. Compared with Fig. 9, with the increase of authentication times, the number of identification errors of the two types of users decreases accordingly. This shows that the weighted sliding window authentication mechanism can infer and identify the current user according to the historical authentication results, so as to reduce the system error rate and improve the accuracy of user recognition.

4.4 Robustness test under multiple scenarios

The change of trust value between legitimate users and imposters is independently analyzed above, and the legitimate and impostors can be distinguished according to the threshold. However, there are more complicated situations to consider sometimes: in real-world scenario, there are intentional or unintentional actions by other users after legitimate users using the device, or imposters’ sudden attacks. Sensitivity reflects whether the system can quickly and accurately perceive the change of

![Fig. 10 Performance of trust value on impostors and real users](image)
user behavior, so that it is necessary to analyze the change of trust value based on sliding window under multiple usage scenarios. We analyzed the change of dynamic trust value and the sensitivity of the trust value responses under the scenarios in Table 5. Trust value changes under the four scenarios are shown in Fig. 11.

It can be found in Fig. 11 that when switching between legitimate users and imposters, the trust value fluctuates sharply and changes rapidly within several window periods, indicating that the trust value can perceive abnormal behavior changes within a relatively short period of switching between true and impostors. Fig. 11a, b are the test scenarios of A and B in Table 5. In Fig. 11a, the authentication probability value output by the classifier during the operation of imposter in the initial stage is small, the trust value is below the threshold line, and is close to the lowest value −1.0. When the operation is suddenly switched to the legitimate use, the authentication probability value from the classifier increases and becomes the recent history window value. Under the accumulation of weighted trust value, the trust value rises rapidly and approaches the maximum value. Figure 11b shows a similar trend: when switching to the imposter action, the trust value drops rapidly from near the maximum to the lowest after several window periods. Figure 11c, d are test scenarios C and D in Table 5 to verify the sensitivity of the change of trust value under complex scenarios. When we observe Fig. 11c, when the operation is performed by the imposter, the trust value rises rapidly in a short period of time when it is suddenly switched to a short operation by the legitimate use. Then, the operation is performed by the imposter, in turn, the trust value decreases rapidly again, and similar changes occur after the operation is performed by the legitimate use once again, indicating that this mechanism has high robustness and rapid response. However, in the Fig. 11d, when the legitimate use suddenly switches to the imposter, the trust value rises after descending and even exceeds the threshold. However, it decreases sharply after that, which indicates that the weighted trust value can timely perceive the abrupt behavior of users and track the changes synchronously. In conclusion, the proposed method is robust and can cope with various realistic scenarios.

4.5 Effect of the size and length of sliding window

Although the trust value can respond quickly under test scenarios C and D in Table 5, there is still a certain delay compared with the trust value changes in Fig. 11a, b, and the trust value is oscillating with a small range of changes under repeating to switch user. In these two scenarios, it is fewer of authentication times for users who are suddenly switched in, and the sliding window needs to be combined with the historical value to judge the current behavior. Therefore, if the current window is too large, each window update takes longer time and requires more authentication results to be logged, which means the trust value cannot be updated in time, resulting in partial delay. When the detection interval of the sliding window is too long, the fluctuation range of the trust value will be reduced, and the sensitivity will be reduced as the trust value cannot change rapidly. The longer the sliding length, the smaller the fluctuation of the trust value and the slower the response; On the contrary, the smaller the window and the shorter the sliding length, the higher
Table 5  Experimental description of trust value in multiple scenarios

| Scene type | Scene change                          | Scene definition                                                                 |
|------------|---------------------------------------|----------------------------------------------------------------------------------|
| A          | Real user → Imposter                  | The legitimate use switches to the imposter                                      |
| B          | Imposter → Real user                  | The imposter switches to the legitimate use                                      |
| C          | Imposter → Real user → Imposter → Real user → Imposter | The imposter switches to the legitimate use, the legitimate user switches to the imposter, with repeating one more time |
| D          | Real user → Imposter                  | The imposter switches to the legitimate use, the legitimate user switches to the imposter |
the sensitivity of the trust value and the greater the fluctuation. Therefore, we analyzed the influence of window length $K$ and size $T$ on the change of trust value, and the results are shown in Fig. 12.

Fig. 11 Changing trend of trust value in multiple scenarios

Fig. 12 Changing trend of trust value in multiple scenarios
By observing Fig. 12, it is found that under the influence of window length \( K \) and size \( T \), the trust value changes obviously. In Fig. 12a, the change curve of trust value of legitimate users is analyzed. It is found that when the trust value decreases due to the accidental behavior of legitimate users, it decreases successively with the increase of window length. When \( K = 4 \), the trust value decreases by 0.5 and reaches below 0.25, which means the trust value fluctuates greatly. When \( K = 9 \), the trust value only decreases by 0.25, and the fluctuation range of the trust value is small around 0.5. Similarly, for imposters, the fluctuation of trust value decreases with the increase of window length. When \( K < 9 \), the authentication probability output by the classifier is large, resulting in a sudden rise in the trust value of the imposter, which exceeds the threshold. Obviously, the smaller the window, the higher the sensitivity of the trust value, but the misidentification rate increases accordingly. The larger the window, the smaller the fluctuation of trust, but its change is slower. By observing Fig. 12b, it is found that as the window size \( T \) gradually increases, the trust value changes slightly for the legitimate users, but significantly for the imposters, and decreases successively with the increase of the window \( T \). In conclusion, setting appropriate window size and length can reduce system misjudgment. In the next section, we will discuss in detail the variation trend of cumulative-weighted-sliding-window-based trust value authentication performance under the joint action of \( T \) and \( K \).

Firstly, the trust value of current window is obtained according to the sliding window mechanism, and the final authentication result is output after threshold detection. In order to make a comparative analysis, the threshold detection results are reversed at the same time, that is, according to the threshold judgment result, the authentication vector in the window is marked as the predicted value of the positive example or the negative example. And then compared with the real value of the vector marked, we analyze the authentication performance of the mechanism.

From the previous analysis, it is found that the different sizes and length of a sliding window can affect the change of trust value, and the trust value determines the authentication result of the current window. Figure 13 shows the variation trend of FAR, FRR and ACC indexes of the algorithm under the influence of these two factors.

![Figure 13](image-url) **Fig. 13** Changing trend of trust value in multiple scenarios
As can be seen from the Fig. 13a, the FAR of the model is greatly affected by the sliding length and decreases rapidly with the decrease of K. However, when the window $T$ decreases, the FAR changes irregularly with a small range of decrease or increase. The model of FAR reached the optimal value when $T = 5$ and $K = 3$, and the lowest value was 1.5%. It can be observed from the Fig. 13b that under the joint action of $T$ and $K$, FRR changes in a mountain-valley style, which indicates that it will cause a sharp increase in FRR when the sliding window is too short or too long and the window is too large or too small, leading to a decline in authentication performance. When $K=8$ and $T = 7$, FRR reaches a minimum of 1.26%. In Fig. 13c of Fig. 12, we observe that this mechanism shows a good authentication performance, and its accuracy rate is above 97.2%, and when $K = 3$ and $T = 8$ the accuracy rate is as high as 98%. It can be seen that the model focuses on different optimal values with different combinations. In this chapter, accuracy is selected as the focus, and $K = 3$ and $T = 8$ is set as the best value of the sliding window. As a future work, the model will be tested using an auto tuning method in order to choose the best value of $k$ and $T$ for various objectives.

### 4.6 Performance of hierarchical authentication scheme

After setting the optimal parameters, we compare the RF algorithm with the trust value method based on the weighted sliding window. We first selected 2882 legitimate use data sets, and then, added imposter data sets to take incremental testing, each increment is 10% of real users. When the number of attacks increases, the performance of the two algorithms in the dense and continuous authentication scenario is studied. The experimental results are shown in Fig. 14.

As can be seen from Fig. 14, the trust value mechanism-based on the weighted sliding window has better performance. The FAR and FRR curves of this mechanism are both lower than that of RF algorithm, which indicates that this mechanism can correct the misjudgment caused by classifier, reduce the misjudgment rate and improve the reliability of the system. In the Fig. 14, the attack probability of the imposter is in the range of 10–90, and the FAR of this mechanism decreases rapidly, with the maximum difference of 11 percentage points from the RF algorithm. By observing the Fig. 14b, it can be found that when the number of attacks is relatively small, the FRR of the two algorithms changes little, both below 0.65%, and the difference is not significant. When a large number of imposters’ attack, the FRR of RF algorithm increases gradually. The results show that under the interference of imposter, the performance of RF algorithm for legitimate use recognition declines rapidly, while the performance of our proposed authentication mechanism declines slowly. Moreover, the FRR of the authentication mechanism does not change after the attack probability increases from 200, and the authentication performance remains at 3.8%. However, the FRR curve of RF algorithm is still rising continuously, and the classification error rate of its legitimate users is getting higher and higher, indicating that the security of implicit authentication based on RF algorithm is getting lower and lower under the continuous attack of imposters. In the Fig. 14c, the authentication accuracy of this mechanism is higher than that of RF algorithm, and its effect is better in the continuous attack scenario. However, it can also be seen that when the evaluation index of the RF algorithm
decreases or increases, the weighted-sliding-window-based trust value based on also decreases and increases simultaneously. This is because the authentication result in the sliding window still depends on the probability value output by the previous RF classifier, which can strengthen the training of the previous RF classifier and improve the authentication effect.

5 Conclusion

In this paper, we propose sAuth, an implicit authentication mechanism to enhance service robots’ security by improving authentication accuracy and robustness. sAuth exploits user behavioral model created by sensor data as the initial authentication and sliding window trust model as the final authentication. To capture the behavior of the user, the built-in sensors are utilized for recording behavioral traits when the user uses the robot pad. In sAuth, sliding windows are responsible for calculating a final decision result. Our results on sAuth show that: 1) the proposed first authentication scheme can achieve higher accuracy when compared with the existing approaches; 2) the hierarchical scheme can reduce FAR and FRR by 11% with unknown impostors.
Acknowledgements The research leading to these results has received funding from the "STUDY ON IMPPLICIT AUTHENTICATION OF MULTIPLE INTELLIGENT MOBILE DEVICES" project 61802252 funded by the National Natural Science Foundation of China. Xiaomei Zhang is the corresponding author: xmzhang@sues.edu.cn.

Author Contributions XZ set the objectives of the research, contributed to the design of the algorithms, guided the data acquisition, validated the data processing and results, executed the detailed analysis, designed the paper, wrote some sections, and performed the final corrections. PZ performed the data gathering and preprocessing, contributed to the development of the data processing algorithms, and wrote some sections. XZ, YW and CC contributed to the design of the algorithms and the data acquisition. All authors have read and agreed to the published version of the manuscript.

Funding the research leading to these results has received funding from the "STUDY ON IMPPLICIT AUTHENTICATION OF MULTIPLE INTELLIGENT MOBILE DEVICES" project 61802252 funded by the National Natural Science Foundation of China.

Declarations

Conflict of interest/Competing interests: the authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

1. Neves AJ (2018) Service robots. IntechOpen, London
2. Chiang AH, Trimi S (2020) Impacts of service robots on service quality. Serv Bus 14:439–459
3. Bartneck C, Belpaeme T, Eyssel F, Kanda T, Keijzers M, abanovi S. Human-robot interaction - An introduction. Cambridge Univerisity, London, (2020)
4. Chinchu S, Mohammed A, Mahesh, B (2017) A novel method for real time face spoof recognition for single and multiple user authentication. In: Proceedings of 2017 IEEE International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT) 376–380
5. Agarwal A, Yadav D, Kohli N, Singh R, Vatsa M, Noore A (2017) Face presentation attack with latex masks in multispectral videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops 81–89
6. Ariharan V (2015) Oracle model to validate shoulder-surfing resistance of virtual keyboards. In: Proceedings of 2015 International Conference on Computing and Network Communications (CoCoNet), 33–40
7. Shen C, Li Y, Chen Y, Guan X, Maxion RA (2017) Performance analysis of multimotion sensor behavior for active smartphone authentication. IEEE Trans Inf Forensics Secur 13(1):48–62
8. Keykhaie S, Pierre S (2020) Mobile match on card active authentication using touchscreen biometric. IEEE Trans Consum Electron 66(4):376–385
9. Zhu T, Qu Z, Xu H, Zhang J, Shao Z, Chen Y, Prabhakar S, Yang J (2019) Riskcog: Unobtrusive real-time user authentication on mobile devices in the wild. IEEE Trans Mob Comput 19(2):466–483
10. Rehman AU, Awais M, Shah MA (2017) Authentication analysis using input gestures in touch-based mobile devices. In: Proceedings of 2017 23rd IEEE International Conference on Automation and Computing (ICAC) 1–5

11. Conti M, Zachia-Zlatea I, Crispo B (2011) Mind how you answer me!: Transparently authenticating the user of a smartphone when answering or placing a call. In: Proceedings of 2011 ACM Symposium on Information

12. Burriro A, Crispo B, Zhauniarovich Y (2017) Please hold on: Unobtrusive user authentication using smartphone’s built-in sensors. In: Proceedings of 2017 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA). New Delhi, India

13. European standard en 50133-1. Alarm systems. access control systems for use in security applications. part 1: System requirements. standard number en 50133-1:1996/a1:2002

14. Sitova Z, Sedenka J, Yang Q, Peng G, Zhou G, Gasti P, Balagani KS (2015) Hmog: New behavioral biometric features for continuous authentication of smartphone users. IEEE Trans Inf Forensics Secur 11(5):877–892

15. Wang R, Tao D (2019) Context-aware implicit authentication of smartphone users based on multi-sensor behavior. IEEE Access 7:119654–119667

16. Shen C, Chen Y, Yang G (2016) On motion-sensor behavior analysis for human-activity recognition via smartphones. In: Proceedings of 2016 IEEE International Conference on Identity, Security and Behavior Analysis (ISBA) 1-6

17. Bhattacharjee S, Chatterjee M (2014) Trust based channel preference in cognitive radio networks under collaborative selfish attacks. In: Proceedings of 2014 IEEE 25th Annual International Symposium on Personal, Indoor, and Mobile Radio Communication (PIMRC), 1502–1507

18. Lee W, Lee R (2015) Multi-sensor authentication to improve smartphone security. In: Proceedings of 2015 International Conference on Information Systems Security and Privacy (ICISSP), 1–11

19. Rahman K A, Tubbs D J, Hossain MS (2018) Movement pattern based authentication for smart mobile devices. In: Proceedings of 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), 1054–1058

20. Lee W, Lee R (2016) Implicit Sensor-based Authentication of Smartphone Users with Smartwatch. In: Proceedings of 2016 Hardware and Architectural Support for Security and Privacy (HASP)

21. Shen C, Li Y, Chen Y, Guan X, Maxion R (2017) Performance analysis of multi-motion sensor behavior for active smartphone authentication. IEEE Trans Inf Forensics Secur 13(1):48–62

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Xiaomei Zhang1 · Pengming Zhang1 · Xin Zeng1 · Yajie Wang1 · Chi-hung Chi2

Pengming Zhang
1084105436@qq.com

Xin Zeng
ericzengxin@outlook.com

Yajie Wang
24665107@qq.com

Chi-hung Chi
chihungchi@gmail.com

1 School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Longteng Road 333, Shanghai 201620, China

2 Strategic Centre for Research in Privacy-Preserving Technologies and Systems, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798, Singapore