FingerPass: Finger Gesture-based Continuous User Authentication for Smart Homes Using Commodity WiFi

Hao Kong*
Shanghai Jiao Tong University
hao.kong@sjtu.edu.cn

Li Lu*
Shanghai Jiao Tong University
luli_jtu@sjtu.edu.cn

Jiadi Yu†
Shanghai Jiao Tong University
jiadiyu@sjtu.edu.cn

Yingying Chen
WINLAB, Rutgers University
yingche@scarletmail.rutgers.edu

Linghe Kong
Shanghai Jiao Tong University
linghe.kong@sjtu.edu.cn

Minglu Li
Shanghai Jiao Tong University
mlli@sjtu.edu.cn

ABSTRACT

The development of smart homes has advanced the concept of user authentication to not only protecting user privacy but also facilitating personalized services to users. Along this direction, we propose to integrate user authentication with human-computer interactions between users and smart household appliances through widely-deployed WiFi infrastructures, which is non-intrusive and device-free. In this paper, we propose *FingerPass* which leverages channel state information (CSI) of surrounding WiFi signals to continuously authenticate users through finger gestures in smart homes. We investigate CSI of WiFi signals in depth and find CSI phase can be used to capture and distinguish the unique behavioral characteristics from different users. *FingerPass* separates the user authentication process into two stages, login and interaction, to achieve high authentication accuracy and low response latency simultaneously. In the login stage, we develop a deep learning-based approach to extract behavioral characteristics of finger gestures for highly accurate user identification. For the interaction stage, to provide continuous authentication in real time for satisfactory user experience, we design a verification mechanism with lightweight classifiers to continuously authenticate the user's identity during each interaction of finger gestures. Experiments in real environments show that *FingerPass* can achieve 91.4% authentication accuracy, and 186.6*ms* response time during interactions.

CCS CONCEPTS

• **Security and privacy → Mobile and wireless security**; • **Networks → Home networks**.

KEYWORDS

User authentication, finger gesture, WiFi signals, smart home

*Hao Kong and Li Lu are the co-first authors
†Jiadi Yu is the corresponding author

Mobihoc '19, July 2–5, 2019, Catania, Italy

ACM ISBN 978-1-4503-6764-6/19/07…$15.00
https://doi.org/10.1145/3323679.3326518

ACM Reference Format:
Hao Kong, Li Lu, Jiadi Yu, Yingying Chen, Linghe Kong, and Minglu Li. 2019. *FingerPass*: Finger Gesture-based Continuous User Authentication for Smart Homes Using Commodity WiFi. In *The Twentieth ACM International Symposium on Mobile Ad Hoc Networking and Computing (Mobihoc '19), July 2–5, 2019, Catania, Italy*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3323679.3326518

1 INTRODUCTION

With the development of Internet of Things (IoT), smart household appliances are increasingly pervasive and common in home environments, making smart homes a practical realization. According to a report [18], the deployment rate of smart household appliances is 32.0% in 2018, and expected to hit 53.1% by 2022. Smart household appliances store various sensitive information such as personal interests, hygiene habits, health status, which could facilitate a variety of customized services. However, such potentially leaked information could cause unauthorized access to personal data and derivation of personal lifestyles. Thus, it is essential to provide secure user access to smart appliances in home environments.

Personal Identification Number (PIN) [1] and biometric-based approaches (e.g., fingerprint [4], voiceprint [28], face recognition [15], etc.) are the most widely deployed user authentications. These approaches are successful, but they only provide one-off user authentication, and are not sufficient for scenarios where continuous privacy protection is necessary. Moreover, biometrics are vulnerable to replay attacks. To provide continuous protection, some works [12, 25] explore the human-computer interactions between users and appliances to implement continuous user authentication for smart homes. But these methods require either wearable wrist sensors or pre-deployed infrastructure, which are intrusive for users and induce a high cost. Recently, WiFi-based user authentication attracts considerable attention, because of the widespread deployment of WiFi infrastructures in indoor environments. Researches [16, 27] utilize WiFi signals to distinguish users based on daily human activities. However, these approaches are only realized based on coarse-grained movements (e.g., gaits, daily activities).

In smart homes, the fine-grained finger gesture rather than coarse-grained human movements is a natural and common way of human-computer interactions. Almost all information or service requests to smart household appliances are issued by finger gestures from users. Toward this end, our goal is to secure each request of finger gestures, i.e., enable continuous user authentication based

on each human-computer interaction, so as to achieve continuous privacy protection. Recently, many works extend the usage of WiFi signals rather than communications, such as indoor location [26], recognition of human activities [24], and breathing rate monitoring [10]. This inspires us to leverage WiFi signals for finger gesture-based user authentication, which is non-intrusive and device-free. To implement the finger gesture-based continuous user authentication through WiFi signals, we face several challenges in practice. First, we should mitigate the always-existed interference induced by unconscious finger motions in CSI of WiFi signals to extract robust features of finger gestures. Second, the authentication system needs to be capable of accurately identify each individual based on extracted unique behavioral characteristics for secure access control. Finally, the user authentication should provide a real-time response for satisfactory user experiences due to user authentication integrated with human-computer interactions.

In this paper, we first investigate the feasibility of leveraging WiFi signals for finger gesture-based user authentication in depth. By analyzing Channel State Information (CSI) of WiFi signals induced by finger gestures, we find that the behavioral characteristics of different users can be presented in CSI phase of WiFi signals. Also, since the user authentication is integrated with human-computer interactions, we are inspired to maintain the real-time response during each finger gesture-based user authentication for satisfactory user experiences. Toward this end, we propose a finger gesture-based continuous user authentication system, *FingerPass*, which leverages CSI of WiFi signals to continuously authenticate users during finger gesture-based interactions. First, *FingerPass* pre-processes the received CSI of WiFi signals and segments the signals into episodes for every finger gesture through *amplitude differential*, and then recognizes different finger gestures through *Support Vector Machine* (SVM). To achieve high authentication accuracy and satisfactory user experience simultaneously, the whole authentication process of *FingerPass* is divided into two stages, i.e., the login and interaction stages. The login stage identifies a user's identity based on a specific login finger gesture from multiple registered users. In order to ensure high user authentication accuracy in the stage, we propose a deep learning-based approach, i.e., *Long Short-Term Memory-based Deep Neural Network* (LSTM-based DNN), to mitigate the interference induced by unconscious motions and further capture unique behavioral characteristics of finger gestures from CSI phase of WiFi signals for user identification. After a successful login, the user can interact with the system in the interaction stage. To provide continuous protection in real time, *FingerPass* verifies the user's identity during each interaction of finger gestures. We propose a *verification mechanism* integrated with the lightweight *Support Vector Domain Description* (SVDD) to achieve the real-time continuous authentication. Experiments demonstrate that *FingerPass* is reliable for continuous user authentication in home environments.

We highlight our contributions as follow.

- We find that utilizing CSI phase of WiFi signals can authenticate user's identity based on each user's finger gestures, and propose a finger gesture-based user authentication system, *FingerPass*, which utilizes CSI of WiFi signals to authenticate users based on finger gesture interactions.
- We develop a deep learning-based method to mitigate the always-existed interference induced by unconscious finger motions for exploring robust sequential relationship of finger gestures, and further extract unique behavioral characteristics underlying the sequential relationship for user identification.
- We design a verification mechanism to secure each interaction of finger gestures, i.e., enable continuous authentication for human-computer interactions, which achieves real-time response and high authentication accuracy simultaneously.
- We conduct experiments in home environments. The results show that *FingerPass* can achieve an authentication accuracy of 91.4%, and a response time of 186.6*ms* during interactions.

2 RELATED WORK

WiFi Signal-based Applications. Recently, WiFi-based sensing attracts considerable attentions because of the wide deployment of WiFi infrastructures in home environments. Previous studies explore WiFi signals for crowd counting [30], breathing rate monitoring [10], indoor location [26], etc. Furthermore, due to the ability of capturing subtle movement with CSI of WiFi signals, more recent works propose to recognize human activity [23], even fine-grained finger gestures [9, 21], through WiFi signals.

User Authentication. Personal Identification Number (PIN) [1] is a common user authentication method, but it is easily leaked to others. To overcome the vulnerability of PIN-based authentication, previous works propose biometric-based authentications, such as fingerprint [4], voiceprint [28], and face recognition [15], etc. However, these works are all vulnerable to replay attacks, which are easy to carry out, requiring neither sophisticated equipment nor specific expertise. Moreover, the biometric-based authentications are not appropriate for continuous user authentication, due to poor user experiences under frequent active authentications.

User Authentication for Smart Homes. Recent works [12, 25] realize user authentication based on human movements for smart homes. These works either require users to wear intrusive sensors or pre-deploy infrastructures to sense human activity for user authentication. Such strong requirements hinder the wide deployment of these works in practical home scenarios. More recently, some works [16, 27] explore the widely-existed WiFi signals to identify the daily human activity (e.g., walking gait) to authenticate users. However, these works are limited to coarse-grained activities (e.g., walking), which cannot provide continuous authentication during human-computer interactions.

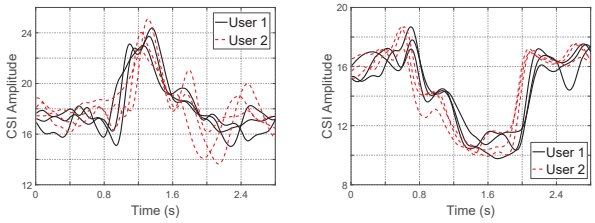
Unlike existing works, our work aims to capture the unique behavioral characteristics of fine-grained finger gestures during human-computer interactions, for continuous user authentication in smart homes, which is non-intrusive and device-free.

3 PRELIMINARY

In this section, we first describe the attack scenario in smart homes, and then explore commodity WiFi for continuous user authentication with satisfactory user experiences.

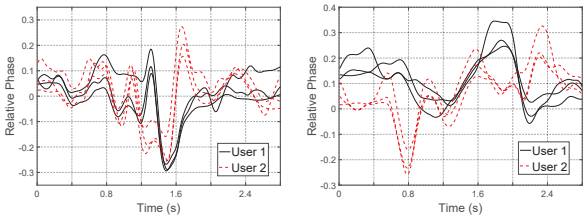
3.1 Attack Scenario in Smart Homes

As the popularity of smart homes, smart household appliances would not only store individual privacy, but also provide personalized services for specific family members. Traditional user authentication for smart household appliances is usually a one-off process,



(a) Waving right. (b) Circling left.

Figure 1: CSI Amplitude of two users when performing two finger gestures.



(a) Waving right. (b) Circling left.

Figure 2: Relative phase of two users when performing two finger gestures.

i.e., authenticate user's identity only once during login. However, users usually do not log out such smart household appliances when they are temporarily suspended. This may result in two representative attack scenarios in smart homes. The first scenario is that the individual information may be leaked to adversaries. For example, a malicious guest in the home may eavesdrop privacies of family members from smart household appliances. The second scenario is that the personalized services provided for specific users may be mistakenly provided to other unsuitable users. For example, children may interact with smart household appliances out of curiosity during the absence of adults, which makes it possible for children to use unsuitable services (e.g., adult movie, online shopping, etc.). The traditional user authentication cannot provide a continuous guarantee for privacy protection to prevent the two attack scenarios. Therefore, it is necessary to secure each interaction between users and smart household appliances, i.e., enable a continuous user authentication during the use of smart household appliances.

3.2 Authentication Feasibility via Finger Gesture Sensing using WiFi

The finger gesture is a natural and common way of human-computer interactions between users and smart household appliances. In smart homes, WiFi infrastructures are widely deployed in home environments. The Channel State Information (CSI) of WiFi signals [5] describes the channel properties of a WiFi signal's propagating path, which can be utilized to recognize different finger gestures through CSI amplitude [9, 21]. Hence, it is natural to first investigate the feasibility of utilizing CSI amplitude of WiFi signals for finger gesture-based user authentication.

To validate whether CSI amplitude can be used to distinguish different users, we conduct an experiment involving two volunteers in a lab. Each volunteer is required to perform two different finger gestures (i.e., waving right and circling left) three times. There is a wireless Access Point (TP-LINK-WDR5620) and a laptop (HP Pavilion 14) equipped with Intel WiFi Link 5300 NIC. The distance between the AP and the laptop is $1m$. The finger gestures are required to be conducted in the middle of the two devices. Figure 1 shows CSI amplitude at 20^{th} subcarrier when two volunteers perform two different finger gestures three times respectively. We can see that there are significant differences on CSI amplitudes for the two finger gestures, which is consistent with existing works[9, 21]. However, the differences between different users are not distinct enough to separate the curves, which indicates that the CSI amplitude cannot be used to distinguish different users. This is because subtle differences between users are overridden by the differences between different finger gestures. Specifically, a finger gesture blocks the propagating path of WiFi signals between a transmitter and a receiver, which induces significant energy attenuation of received WiFi signals. Thus, the CSI amplitude would change significantly due to the energy attenuation. However, since the finger blockage depicts the coarse-grained characteristics of finger gestures, it would override the fine-grained behavioral uniquenesses of different users, i.e., the subtle differences between users cannot be exhibited in CSI amplitude of WiFi signals.

In order to distinguish different users through CSI of WiFi signals, we consider another measure, i.e., CSI phase of WiFi signals, which can express the movement of an object in the propagating path of WiFi signals [23]. However, the absolute CSI phase has an unpredictable offset due to hardware imperfection [29]. Hence, we employ relative phase to eliminate the offset and further reveal fine-grained behavioral characteristics. The relative phase at the k^{th} subcarrier can be represented as:

$$\angle \hat{H}_k = -\frac{2\pi}{\lambda}\Delta d, \tag{1}$$

where Δd is the length difference of two transmitting paths, and λ is the signal wavelength. The relative phase can reveal more fine-grained behavioral characteristics due to the cm-scale λ [16].

To explore the feasibility of user authentication through CSI phase of WiFi signals, we analyze the data collected from the previous experiment. Figure 2 shows the relative phase of WiFi signals when the two volunteers perform the two finger gestures respectively. Different from CSI amplitude, there are differences between different users when performing the same finger gesture, which can be used to distinguish different users. The result indicates that utilizing CSI phase of WiFi signals is feasible to authenticate user's identity. This is because the differences between different users mainly depend on behavioral characteristics, such as moving distance, speed, and orientation of a user's fingers and palm. The moving fingers and palm reflect WiFi signals, which would change the length of a propagating path of the WiFi signals. Such a length change of propagating path can be exhibited as CSI phase shift of WiFi signals [2]. This indicates that the CSI phase of received WiFi signals would express how fingers and palm move according to users daily habits. Therefore, the phase shift induced by the path length changes can depict the fine-grained behavioral characteristics of users when performing finger gestures, which can be further used to authenticate users identities.

Through the analysis above, we find CSI phase of WiFi signals can depict differences between different users due to the exhibition of behavioral characteristics. Therefore, we employ CSI amplitude to recognize different finger gestures, while leverage CSI phase to authenticate users' identities.

3.3 Finger Gesture-based Continuous User Authentication in Real Time

As mentioned above, we aim to integrate user authentication into the finger gesture-based interactions between users and smart household appliances, i.e., enable continuous user authentication. In such a scenario, the satisfactory user experience is an important aspect of designing the user authentication system. When interacting with smart household appliances, users always require a real-time response of each interaction rather than waiting for a long time. Thus, the continuous user authentication should meet the real-time response requirement for the satisfactory user experience. To design the user authentication within a real-time response, we first consider the workflow of a typical human-computer interaction system with one-off user authentication. The first step of such a system is an identification process of a user's identity for user login, which is actually a multi-class problem. After a successful login, all subsequent services of the system are provided based on the current login user's identity. To extend the typical system to continuous user authentication, we consider enabling a verification process during each interaction, i.e., verify whether the current user is the logged-in user or not. Thus, the user authentication during interactions is actually a binary classification.

Toward this end, to simultaneously ensure continuous authentication and real-time requirements, the whole authentication process is considered as two stages, i.e., the login stage and the interaction stage, which corresponds to user identification and verification respectively. The login stage is a one-off user identification which identifies a user's identity based on a specific login finger gesture. In the stage, there is almost no interaction request issued by users. Hence, a relatively long response time for a high accuracy identification would not suffer significant degradation of user experiences. The interaction stage verifies the identity of the current user according to each interaction of finger gestures. To ensure a satisfactory user experience during interactions, the system should respond to finger gesture interactions in real time.

In practice, challenges emerge when implementing the finger gesture-based continuous user authentication through WiFi signals. First, the CSI phase of WiFi signals affected by finger gestures contains not only behavioral characteristics, but also unconscious motions that cannot represent the characteristics of users. Hence, the unique features of behavioral characteristics should be extracted for robust user identification during login. Second, in order to satisfy user experience, user verification during interactions can be achieved by a lightweight method for a real-time response. However, using the lightweight method to verify users may affect authentication performance due to the limitations of the method itself. Thus, a verification mechanism should be explored for improving the verification accuracy as well as ensuring the real-time response during interactions.

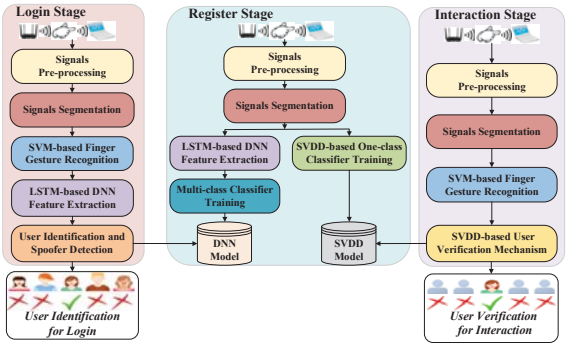


Figure 3: Architecture of *FingerPass*.

4 SYSTEM DESIGN

In this section, we present the design of a finger gesture-based user authentication system, *FingerPass*, which leverages CSI of WiFi signals to continuously authenticate users based on finger gestures with high accuracy and real-time response in smart homes.

4.1 System Overview

Figure 3 shows the architecture of *FingerPass*, which includes a register stage, a login stage, and an interaction stage. The register stage is an off-line training process, and the login and interaction stages are on-line authentication processes.

In the register stage, the system collects multiple finger gestures from family members as the training data for model construction. First, *FingerPass* processes the received WiFi signals to mitigate multipath effects through Inverse Fast Fourier Transform (IFFT) and Butterworth filter, and selects specific subcarriers which are sensitive to movements. Then, the pre-processed signals are segmented into episodes of finger gestures based on the amplitude differential of in received WiFi signals. Afterward, *FingerPass* apply Long Short-Term Memory-based Deep Neural Network (LSTM-based DNN) to construct a DNN model for user identification during login. Finally, Support Vector Domain Description (SVDD) is applied to construct lightweight SVDD model for user verification during interactions.

In the login stage, *FingerPass* identifies a user's identity based on a specific login finger gesture. Specifically, *FingerPass* first processes and segments signals, which is the same as that in register stage, and then recognizes the login finger gestures through Support Vector Machine (SVM) based on CSI amplitude. Next, *FingerPass* extracts unique behavioral characteristics of the user from CSI phase through LSTM-based DNN feature extraction, and applies the trained DNN model for user identification and spoofer detection.

In the interaction stage, each finger gesture-based interaction is first recognized as an interaction request, and then authenticated to provide a security guarantee. Specifically, *FingerPass* first processes and segments signals, then recognize finger gestures, which is similar to that in the login stage. Then, a verification mechanism with trained SVDD models is applied for continuous user authentication during the finger gesture-based interactions.

4.2 Signal Pre-processing

In this section, we first describe how to mitigate multipath effects, and then give the method of selecting sensitive subcarriers.

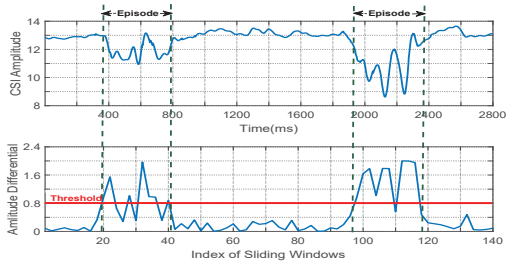


Figure 4: Illustration of signals segmentation leveraging amplitude differential.

4.2.1 Multipath Mitigation. Except for the targeted finger, the nearby moving objects (e.g., walking people) also reflect the omni-directional WiFi signals, i.e., the multipath effect. Such a multipath effect could interfere the received WiFi signals of finger gestures, and thus reduce the robustness of the finger geature-based user authentication. Therefore, it is necessary to mitigate the multipath effect from received WiFi signals. The signal reflections from the distant dynamic movements usually have longer propagation delays before arriving at a receiver. Hence, we remove these signals components with a large time delay to mitigate multipath effects [21]. Specifically, given CSI of WiFi signals at each subcarrier in the frequency domain, we obtain the power delay profile through the n-point Inverse Fast Fourier Transform (IFFT). The previous study shows indoor environments have the maximum delay less than $500ns$[7]. Hence, we remove the signal components with a delay longer than $500ns$ in the power delay profile, which mitigates multipath effects caused by distant dynamic movements. Note that although the IFFT operation reduces the time resolution of received WiFi signals, the mitigation of multipath effect based on IFFT contributes more on resisting the interference from ambient moving objects and improving the robustness of user authentication.

4.2.2 Subcarriers Selection. Although WiFi infrastructures usually provide multiple subcarriers for communication (e.g., 30 subcarriers of Intel WiFi Link 5300 NIC), not all of them contribute to capturing human movements [16], due to the insensitivity to specific environmental changes. In order to reduce the coverage of insensitive subcarriers on unique features of finger gestures, *FingerPass* needs to select sensitive subcarriers from the m subcarriers on CSI of WiFi signals. Specifically, we select k subcarriers whose variance values exceed mean variance values as sensitive subcarriers. For each sensitive subcarrier, we use the proportion of variance values w_i as weights to combine each subcarrier so as to get a combined sensitive subcarrier. Given $W = \sum_{i=1}^{k} w_i$ which represents the sum of sensitive subcarriers' variance values, the combined carrier H is calculated as $H = \sum_{i=1}^{k} \frac{w_i}{W} H_i$, where H_i denotes CSI of i^{th} subcarrier. The combined subcarrier with high sensitivity for finger gestures could enhance the feature extraction of unique behavioral characteristics from different users.

4.3 Finger Gesture Detection and Recognition

In this section, we describe the segmentation of CSI amplitude of received WiFi signals to detect each finger gesture, and the recognition of different finger gestures.

4.3.1 Signals Segmentation. Different finger gestures would induce different values on CSI amplitude. The top part of figure 4 shows CSI amplitude of WiFi signals induced by two different finger gestures. Compared with the second finger gesture, the amplitude value of the first finger gesture is not obvious, which is similar to the white noises in CSI amplitude. This may results in that some finger gestures cannot be detected, or some white noises in CSI amplitude would be mistakenly detected as a finger gesture. However, we can observe that CSI amplitude induced by different finger gestures all have significant changes. Hence, we consider utilizing the amplitude change for signals segmentation.

To depict the change of CSI amplitude, we define *amplitude differential*, i.e.,

$$D(n) = \sum_{t=nL}^{(n+1)L-1} |(C_{t+1} - C_t)|, \quad n \in [0, N-1], \tag{2}$$

where $D(n)$ denotes the amplitude differential of n^{th} sliding window, L is the length of a sliding window, C_t is a CSI amplitude value at time t, and N is the number of sliding windows. The bottom part of figure 4 shows the amplitude differentials of two different finger gestures. We can see that although the amplitude value of the first finger gesture is not obvious, the amplitude differential of the finger gesture is significantly different from that of white noises. Thus, we can utilize the amplitude differential for signal segmentation of finger gesture episodes. Specifically, *FingerPass* utilizes sliding windows to compare the amplitude differential with a predefined threshold for capturing the starting and ending point of a finger gesture. The threshold can be obtained through empirical studies.

4.3.2 Finger Gesture Recognition based on SVM. As mentioned in Section 3.3, the interaction between users and smart household appliances requires a real-time response for a satisfactory user experience. Existing works about finger gesture recognition based on CSI of WiFi signals mainly employs Dynamic Time Warping (DTW) [9, 21]. However, the computational complexity of DTW-based recognition is $O(Knm)$ [14], where K is the number of known finger gestures, n and m are the numbers of sampling points in the matching finger gesture episodes respectively. Such a computational complexity cannot meet the real-time requirement.

To meet the requirement of real-time response, we employ Support Vector Machine (SVM) [20] for finger gesture recognition. Combined with a one-versus-one strategy, *FingerPass* constructs a multi-classifier, whose computational complexity of finger gesture recognition is $O(K^2n)$. Usually, the number of finger gestures is limited in a human-computer interaction system, which leads to a far smaller K than the number of sampling points n. Therefore, the time consumption of the SVM-based method would be significantly lower than that of DTW, which results in a faster response during human-computer interactions for a satisfactory user experience.

4.4 User Identification through Deep Learning for Finger Gesture-based Access Control

In smart homes, before a user interacts with smart household appliances, *FingerPass* should first obtain the identity credential in login stage. The user authentication in the login stage can be considered

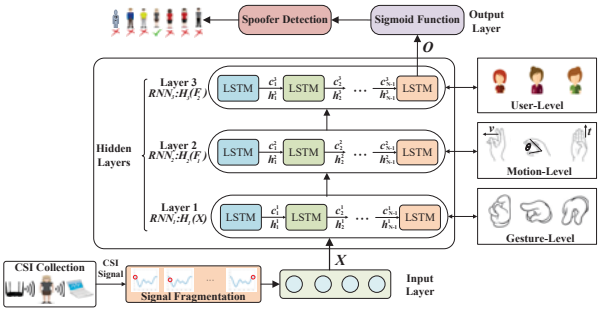


Figure 5: Architecture of a three-layer LSTM-based deep neural network.

as an identification problem, i.e., identify the user by a specific login finger gesture, which is actually a multi-class problem.

As mentioned in Section 3.2, CSI phase of WiFi signals depicts behavioral characteristics of each user through finger gestures. During performing a finger gesture, there always exist unconscious finger motions, which can interfere the received CSI of WiFi signals. Since the finger gesture is fine-grained movement, such a subtle interference induced by the unconscious motions could significantly affect the robustness of finger gesture-based user authentication. To extract unique features from the behavioral characteristics for robust user identification, it is necessary to reduce the interference of these unconscious motions. Usually, a finger gesture exhibits strong relationship between the previous and subsequent finger motions, i.e., the sequential relationship. The unconscious motions of finger gestures are different with normal sequential relationships, i.e., they induce instant significant shifts, which are neither related with previous motions, nor induce sequential effects to the subsequent motions. Thus, we can extract unique features from each user's finger gestures and eliminate the impact of such unconscious motions through the sequential relationships.

Feature Extraction. To utilize the sequential relationships in CSI phase induced by finger gestures, we propose a three-layer Long Short-term Memory-based Deep Neural Network (LSTM-based DNN) to extract features of each user's finger gestures for robust user identification. Figure 5 shows the architecture of user identification through the three-layer LSTM-based DNN.

In the proposed DNN model, each hidden layer consists of a Recurrent Neural Network (RNN) with LSTM units, which abstracts the input as a set of feature representations. Traditional RNN only maintains the short-term previous information, which leads to the loss of sequential relationships that occurred long ago [3]. Thus, traditional RNN cannot fully depict the unique features from behavioral characteristics. In order to capture all previous information, including long-term and short-term information of finger gestures, we employ LSTM units [6] rather than the typical neural unit in RNN for finger gesture-based user identification. For each hidden layer, LSTM unit can map the input Z_t^l in time slot t and layer l into a feature representation f_t, i.e., $f_t = g(PZ_t^l + b)$, where $g()$ is a activation function, P is the weight matrix, b is a bias. The input Z_t^l contains three informations, i.e., $Z = [x_t^l, h_{t-1}^l, c_{t-1}^l]^T$, where x_t^l is the current information, h_{t-1}^l is the short-term information, and c_{t-1}^l is the long-term information. Thus, RNN with LSTM units retains both long-term and short-term previous information of finger gestures, which could express more sequential relationships of finger gestures for robust user identification.

Given CSI phase profiles of a user's finger gestures, each layer of the DNN model contains an RNN H_i, which abstracts the input into a set of feature representations as output. We first partition the input CSI phase profiles X into N small fragments $x(t)$. The fragmented CSI phase profiles of finger gestures are represented as $X = [x(1), x(2), ..., x(N)]$, where $x(t)$ is the fragmented CSI phase profiles in time slot t ($t \in [1, N]$). The inputs of the first layer are the fragmented CSI phase profiles X of users' finger gestures, and the gesture-level features F_1 can be extracted as output by $H_1(X)$ in the first layer. Then, the output F_1 of the first layer is fed to the second layer. $H_2(F_1)$ in the second layer further extracts the motion-level features F_2 (e.g., speed, angle, and time). Finally, $H_3(F_2)$ in the last layer takes the output F_2 of the second layer as input, and extracts the user-level features as an output O, which represents user's unique features and can be used for user identification.

User Identification. *FingerPass* employ the sigmoid function [8] in output layer through the extracted features to identify a user's identity. When a user attempts to login, *FingerPass* calculates the posterior probability $P(U_k \mid O)$ with the output feature O of the user and every registered user's features. For output O of the current user from user-level and each registered user U_k, the posterior probability $P(U_k \mid O)$ is calculated as:

$$P(U_k \mid O) = \frac{P(O \mid U_k)P(U_k)}{P(O \mid U_k)P(U_k) + P(O \mid \overline{U}_k)P(\overline{U}_k)}, \tag{3}$$

where $P(U_k)$ is the prior probability of user U_k, and $P(O \mid U_k)$ is the likelihood of the feature O given label U_k. The posterior probabilities is under the constraints that $0 \leqslant P(U_k \mid O) \leqslant 1$ and $P(U_k \mid O) + P(\overline{U}_k \mid O) = 1$. Given K user classes, sigmoid function outputs K posterior probabilities. With the objective function $k = \arg\max_{k \in K} P(U_k \mid O)$, the user with feature O will be identified as the user U_k.

Spoofer and Unexpected Body Movements Detection. Except for identifying the user's identity from multiple register users, *FingerPass* needs to detect unexpected spoofers in the login stage. Through the proposed DNN, the subtle differences between finger gestures of a spoofer and all registered users can be extracted, which can be utilized for spoofer detection. Specifically, when a spoofer attempts to login the system through a finger gesture, the features of the spoofer are extracted by LSTM-based DNN. Since the feature of spoofer would not match that of any registered user, the posterior probability of spoofer in output layer should be significantly lower than that of all registered users. Hence, we use a threshold to detect spoofers. Specifically, we define $score_k$ as the similarity between the feature O of attempted login finger gesture and features U_k of all registered finger gestures, i.e., $score_k = \{1 \mid P(U_k \mid O) > \lambda, 0 \mid P(U_k \mid O) <= \lambda\}$. If $\sum_{k=1}^{K} score_k = 0$, the attempted login user with the feature O is identified as a spoofer. Such an approach is also able to detect the unexpected body movements issued by the user. Since the body movement is quite different from a valid finger gesture, the features extracted by DNN would exhibit significant differences. Therefore, even though an unexpected body movement is recognized as a finger gesture, *FingerPass* can further identify

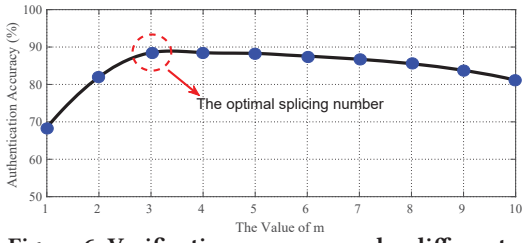


Figure 6: Verification accuracy under different *m*.

such a body movement as a spoofer's request, and thus trigger no further permission and interaction response.

4.5 User Verification through SVDD for Finger Gesture-based Interaction

To provide continuous privacy protection and consistent personal services delivery, *FingerPass* should perform user authentication during each interaction of finger gestures in the interaction stage. As mentioned in Section 3.3, the system needs to ensure real-time response so as to achieve a satisfactory user experience. Thus, user authentication in the interaction stage can be simplified as a verification problem, i.e., regards a current user as a valid user or an invalid user, which is actually a binary classification.

Verification Mechanism. In order to ensure a real-time response, we utilize a one-class classifier, Support Vector Domain Description (SVDD) [22], to verify the current user's identity. However, the SVDD classifier based on single finger gesture has lower classification accuracy since it is a light-weight method. Usually, a user tends to interact with smart household appliances multiple times after the user successfully logins. Based on such an intuition, we propose a verification mechanism, which leverages not only the current finger gesture interaction, but also the previous finger gestures that passed the verification during the ongoing interactions, to continuously authenticate the user's identity for each interaction. Thus, the verification mechanism gradually improves the accuracy of SVDD-based classifier as more interactions occur.

In the training process of SVDD-based classifier, *FingerPass* not only trains single-gesture classifier for every single finger gesture, but also splices finger gestures to train multi-gesture classifiers. Specifically, we first align all CSI phases of finger gestures through the interpolation method to make the input's length consistent, and then feed the aligned relative phases into the classifier for training. Assume there are n finger gestures, i.e., $g_{(0)}$,..., $g_{(n-1)}$, *FingerPass* trains finger gesture classifiers for single finger gesture and splicing finger gestures, i.e., for each finger gesture, *FingerPass* trains n single-gesture classifiers, $c_{g_{(0)}}$,..., $c_{g_{(n-1)}}$; for the splicing of two gestures, *FingerPass* trains n^2 two-gesture classifiers, $c_{g_{(0)}g_{(0)}}$, $c_{g_{(0)}g_{(1)}}$,..., $c_{g_{(0)}g_{(n)}}$, $c_{g_{(1)}g_{(0)}}$,..., $c_{g_{(n-1)}g_{(n-1)}}$; ... ; for the splicing of m finger gestures, *FingerPass* trains n^m m-gesture classifiers, $c_{g_{(0)}g_{(0)}\cdots g_{(0)}}$, $c_{g_{(0)}g_{(0)}\cdots g_{(1)}}$,..., $c_{g_{(n-1)}g_{(n-1)}\cdots g_{(n-1)}}$. Through the training above, we can obtain $\sum_{i=1}^{m} n^i = \frac{n(1-n^m)}{1-n}$ classifiers including the single-gesture classifiers and the multi-gesture classifiers.

In the verification process, when a user performs the t^{th} finger gesture interaction $g_{(t)}$, *FingerPass* utilizes classifiers of $m-1$ previous finger gestures that passed the verification during the ongoing interactions combined with current finger gesture, i.e., $c_{g_{(t)}}$, $c_{g_{(t-1)}g_{(t)}}$,..., $c_{g_{(t-m+1)}g_{(t-m+2)}\cdots g_{(t)}}$, to get m preliminary verification results. Then, *FingerPass* utilizes a voting mechanism that leverages the results of each classifier to obtain a final user verification result. Specifically, for each classifier, a user verification result $\mu_i \in \{1, 0\}$ is obtained, where $\mu_i = 1$ and $\mu_i = 0$ denote a successful and unsuccessful verifications respectively. Then, *FingerPass* votes on all the results, and decides the final user verification result according to the maximum voting results, i.e., $result = \{1 \mid \sum_{i=1}^{m} \mu_i > \frac{m}{2}, 0 \mid \sum_{i=1}^{m} \mu_i \leqslant \frac{m}{2}\}$, where $result = 1$ and $result = 0$ represent successful and unsuccessful verifications respectively. Similar to the unexpected body movement detection in the login stage, the interaction stage can also recognize an unexpected body movement as a spoofer's request, due to the significant difference between a valid finger gesture and the unexpected body movement.

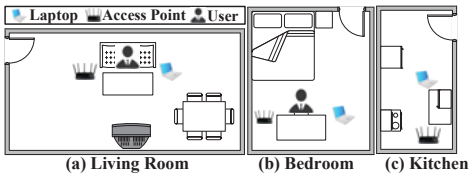


Figure 7: Experimental setup of three home environments.

The computational complexity of SVDD-based classifier is $O(n)$, where n is the size of each finger gesture sample. Considering $m-1$ previous finger gestures are utilized in our verification mechanism, i.e, m classifiers are applied in total, the computational complexity of user verification in the interaction stage is $O(nm^2)$. To explore the optimal value of m, we conduct experiments with data collected from real environments. Figure 6 shows the verification accuracy under different m. When we only use a single gesture to verify a user, i.e., $m = 1$, the verification accuracy is only 68.5%. As more previous finger gestures are utilized, the verification accuracy first increases and reaches the highest accuracy of 89.2% when m is 3, and then decreases due to over-fitting. Since the optimal value m is much less than n, the computational complexity of user verification in the interaction stage can be regarded as $O(n)$. Therefore, through the SVDD-based verification mechanism with a small splicing number, *FingerPass* can realize high user authentication accuracy and real-time response in the interaction stage.

5 PERFORMANCE EVALUATION

In this section, we evaluate the performance of *FingerPass* in real home environments.

5.1 Experimental Setup and Methodology

We implement *FingerPass* on a laptop, i.e., a HP Pavilion 14, which is equipped with an Intel WiFi Link 5300 NIC for providing 30 subcarriers on CSI of WiFi signals. A commercial wireless access point (AP), i.e., a TP-LINK-WDR5620, is employed as the WiFi signal transmitter, which continuously emits the 802.11n WiFi signals. We conduct the experiments in three different home environments, i.e., a living room, a bedroom, and a kitchen. The sizes of the three rooms are $5.8m \times 4.2m$, $3.8m \times 3.4m$, and $3.4m \times 2.2m$ respectively. The distances between AP and laptop in the three rooms are $3.0m$, $2.0m$, and $1.0m$ respectively. Figure 7 shows the layouts of the AP, laptop and other furniture in the three home environments.

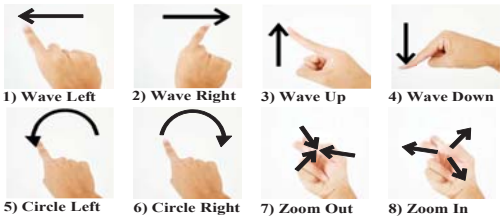


Figure 8: Illustration of eight different finger gestures.

We select 8 commonly used finger gestures, which are widely used in human-computer interaction systems, as shown in figure 8. Our experiments involve 7 volunteers. Since the average number of people per family in the US is 3.14 [19], we ask 5 of the volunteers as registered users in the experiments, and the rest 2 as spoofers, which could meet the needs of most families. All of the volunteers are required to perform finger gestures towards the laptop with a distance of $0.5m$. This is because users are natural to stand/sit in front of the smart appliance (i.e., laptop in our experiment) during the interaction between users and appliances.

We define several evaluation metrics:

• *Response Accuracy.* The probability that both the finger gesture and user's identity are recognized and authenticated correctly.

• *Response Time.* Assume the CSI of WiFi signals induced by a user's finger gesture is derived at time T_e, and the time that the system responds the user's interaction is T_{dev}. The response time of the system is defined as $T = T_{dev} - T_e$.

• *Confusion Matrix.* Each row and each column of the matrix denotes the ground truth and the authentication result of *FingerPass* respectively. The i^{th}-row and j^{th}-column entry of the matrix shows the percentage of samples that are authenticated as the j^{th} user while actually are the i^{th} user.

• *Authentication/Recognition Accuracy.* The probability that a user/finger gesture who is A is exactly identified as A.

• *False Accept Rate.* The probability that a user not a registered user is authenticated as a registered user.

• *False Reject Rate.* The probability that a user not a spoofer is authenticated as a spoofer.

5.2 Overall Performance

Figure 9 shows response accuracies in the login and interaction stages. We can see that the average response accuracies of the login and interaction stages are 91.3% and 88.6% respectively. The overall response accuracy of *FingerPass* is 90.0%. This result demonstrates that *FingerPass* can achieve satisfactory performance for both interaction and authentication. Moreover, it can be observed that the response accuracies in the three home environments are similar, which indicates *FingerPass* is robust to different distances between transmitter and receiver as well as different home environments.

Figure 10 shows CDFs of response time in the login and interaction stages respectively. Note that the response time contains both finger gesture recognition and user authentication. We can see that for 85% finger gestures, the response times are less than $200ms$ in the interaction stage, while that are in the range of $[800, 1200]ms$ in the login stage. Previous research validates that the latency of $[50, 200]ms$ in modern touch systems is an appropriate response time for a satisfactory user experience [11]. Since the login stage has no interaction requests, the relatively long response time would not degrade user experiences. For the interaction stage, it achieves a real-time response of touch system level, which satisfies good user experiences of human-computer interaction.

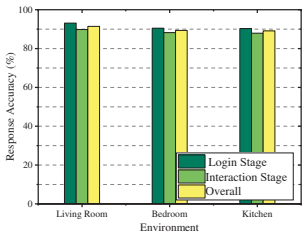


Figure 9: Response accuracy in different environments.

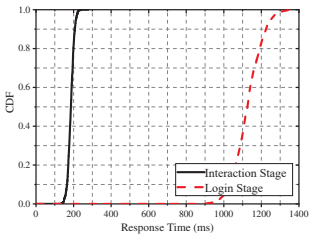


Figure 10: CDF of time in login and interaction stages.

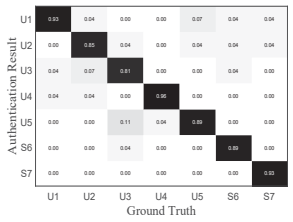


Figure 11: Confusion matrix of user authentication in login stage.

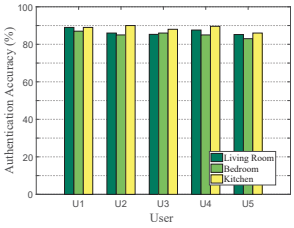


Figure 12: Authentication accuracy of *FingerPass* in interaction stage.

5.3 Performance of User Authentication

Figure 11 shows the confusion matrix of *FingerPass* in the login stage. We can see that *FingerPass* can achieve an average authentication accuracy of 93.3% in identifying the registered user, and that of 90.0% in spoofer detection. The average accuracy of the login stage in user authentication is 92.6% with a standard derivation of 4.43%. This indicates *FingerPass* can achieve a high accuracy of user authentication in the login stage, which validates the reliability and efficiency of user identification.

Figure 12 shows authentication accuracies of *FingerPass* in the interaction stage under the three home environments. We can observe that *FingerPass* can achieve average authentication accuracies of 91.3%, 89.8% and 89.2% under the three home environments respectively, and the standard derivations are 1.6%, 1.7% and 1.3% respectively. This result indicates that *FingerPass* can accurately authenticate the logged-in user's identity in the interaction stage.

Figure 13 shows false accept rates and false reject rates of *FingerPass* in the login stage under different environments. We can see that the average false accept rate of *FingerPass* under the three environments is only 3.5%, which demonstrates that *FingerPass* is reliable to identify a spoofer in the login stage. Moreover, the average false reject rate under the three home environments is 3.8%, which indicates that *FingerPass* hardly misidentifies a registered user in the login stage, which ensures a satisfactory user experience.

Figure 14 shows CDFs of interaction numbers for misidentifying a logged-in user and identifying a non-logged-in user in the interaction stage. We see that over 90% non-logged-in users can be authenticated within 3 interactions. It indicates that *FingerPass* is not vulnerable to other users or spoofers, which demonstrates the

Figure 13: False accept rate and false reject rate in login stage under different home environments.

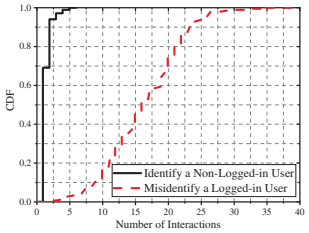


Figure 14: CDFs of interaction numbers to misidentify a logged-in user and identify a non-logged-in user.

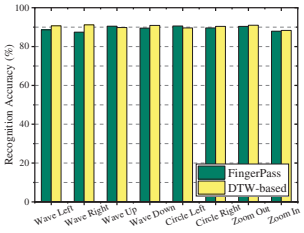


Figure 15: Recognition accuracy of *FingerPass* and DTW-based method for different finger gestures.

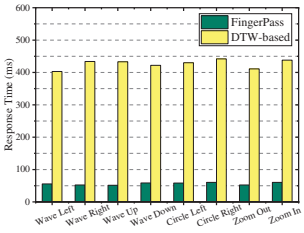
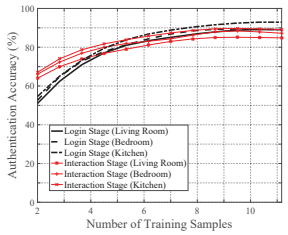
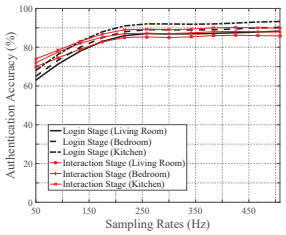


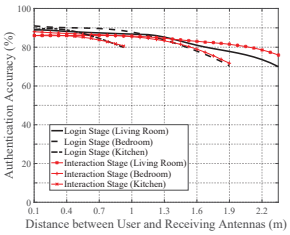
Figure 16: Response time of *FingerPass* and DTW-based method for different finger gestures.



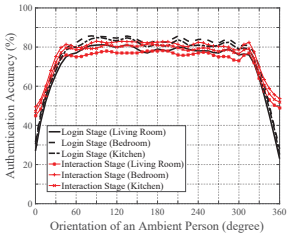
(a) Size of training sets.



(b) Sampling rates.



(c) Distance between user and antennas.



(d) Orientation of an ambient person.

Figure 17: Authentication accuracy under different impacts.

reliability of *FingerPass*. We also observe that when the interaction numbers are below 10 times, about 90% users are still not mistakenly logged out by *FingerPass*, which indicates that the continuous authentication rarely affects the normal use of the logged-in user.

5.4 Performance of Finger Gesture Recognition

To evaluate the performance of finger gesture recognition, we implement the DTW-based method [9, 21] as a baseline, and compare the recognition accuracy and response time between DTW-based method and *FingerPass*. We can see from figure 15 that the average recognition accuracy of *FingerPass* is quite similar to that of DTW-based method. *FingerPass* can achieve an average recognition accuracy of 88.7%, which indicates its reliability of interaction through recognizing finger gestures. Moreover, from figure 16, it can be observed that the average response time of *FingerPass* is $56.1ms$, which meets the real-time requirement. But the response time of the baseline is above $350ms$, which exceeds the time range for a satisfactory user experience (i.e., $[50, 200]ms$). The results show that *FingerPass* can achieve high accuracy of finger gesture recognition while satisfying the real-time requirement.

5.5 Performance under Different Impacts

Impact of Training Set Size. The size of the training set is the number of users' performing times of finger gestures for registering. Too much performing times would affect user experiences in the register stage. Figure 17(a) shows authentication accuracies in the login and interaction stages with different training set sizes in the three home environments. We can see that as the size of the training set increases, the authentication accuracies first increase, and then go stable in both stages. When users perform above 8 finger gestures for registering, *FingerPass* can achieve average authentication accuracies of over 80% in the two stages. More finger gesture performing times would not contribute an improvement in authentication accuracy. Since the performing times of 8 is acceptable for users, it demonstrates that the register stage of *FingerPass* is also consistent with user experiences.

Impact of Sampling Rate. In our experiment, the laptop is equipped with an Intel WiFi Link 5300 NIC as the WiFi signal receiver, which can receive WiFi signals under a sampling rate ranging in $[10, 2000]Hz$. To explore the appropriate sampling rates for *FingerPass*, we evaluate the performance of *FingerPass* under different sampling rates. Figure 17(b) shows the authentication accuracy of the login and interaction stages in the three home environments respectively. We can see that the authentication accuracy of *FingerPass* first increases and then goes stable as the sampling rate increases. When the sampling rate approaches $250Hz$, *FingerPass* can achieve an authentication accuracy of over 85% in the three home environments respectively. Such a sampling rate is capable for most smart household appliances. Thus, *FingerPass* can be widely applied for user authentication in smart homes.

Impact of Distance between User and Receiving Antennas. To explore whether *FingerPass* could achieve a satisfactory performance in a comfortable interaction distance, we first investigate the common interaction distances between user and appliances for smart homes. Usually, smart TV has the longest interaction distance, whose average size is 42.8 inches in 2017 [17], and the distance between such a smart TV and users with the best viewing experience is below $1.6m$ [13]. Hence, we conduct experiments to explore the performance of *FingerPass* in the distance within $2.4m$ that can meet the requirements for most appliances. Since the user is required to experiment between the AP and the laptop, and the distances between the two are fixed at $1m$, $2m$, and $3m$ respectively in the three homes, the experimental distance range between user and laptop in each home is set separately. Figure 17(c) shows the

authentication accuracy under different distances between a user and the laptop in the three homes. We can see that *FingerPass* can achieve over 75% authentication accuracy when the distance is less than $1.6m$ in the three homes. The result shows that *FingerPass* achieves satisfactory performance for comfortable interaction.

Impact of Ambient Persons. Since the multipath effect cannot be completely eliminated, the existence of an ambient person actually affects the performance of *FingerPass*. We study the performance of *FingerPass* under different orientations of an ambient person, because the orientation of an ambient person has a greater impact on WiFi signals' transmission than the distance. The experimental layout is set as shown in figure 7. We define the orientation of an ambient person as the angle between AP and the ambient person relative to the laptop. Specifically, the orientation of AP-laptop connection is $0°$, and other orientations, i.e., $[0°, 360°]$, are followed through a counterclockwise rotation relative to the AP-laptop connection. The current user is required to perform finger gestures $0.5m$ away from the laptop, and the distance between the ambient person and the laptop is fixed as half of the distance between AP and laptop, i.e., $0.5m$, $1m$, and $1.5m$, in the three environments respectively. Figure 17(d) shows the authentication accuracy of the login stage and the interaction stage under different orientations of the ambient person in the three home environments respectively. We can see that the authentication accuracies on the orientations of $[40°, 320°]$ (i.e., relatively far away from the line-of-sight transmission of WiFi signals) are over 75% and remain stable. This shows that *FingerPass* could achieve a satisfactory performance as long as an ambient person is not so close to the line-of-sight transmission of WiFi signals. However, the authentication accuracy significantly degrade under the orientations of $[0°, 40°]$ and $[320°, 360°]$. This is because such orientations depict a relatively close distance between the ambient person and the line-of-sight WiFi transmission, which leads to an intense interference to the signal transmission. The result indicates that *FingerPass* would not be significantly affected during the existence of an ambient person in most orientations.

6 CONCLUSION

In this paper, we propose a finger gesture-based user authentication system, *FingerPass*, which leverages CSI of WiFi signals to continuously authenticate users during human-computer interactions. First, we pre-process and segment CSI of WiFi signals through amplitude differential, and then recognize finger gestures by Support Vector Machine. For highly accurate and real-time user authentication, *FingerPass* divides the whole authentication into two stages, i.e., login and interaction stages. For the login stage, we propose a deep learning-based approach, i.e., Long Short-Term Memory Deep Neural Network, for highly accurate user identification. For the interaction stage, to provide continuous user authentication in real time, a verification mechanism with lightweight classifiers is proposed to continuously authenticate the user during each interaction of finger gestures. Experiments show that *FingerPass* is reliable for continuous user authentication in smart homes.

ACKNOWLEDGMENTS

This work is supported by National Natural Science Foundation of China (No. 61772338), and sponsored by China Scholarship Council.

REFERENCES

[1] Carlisle Adams. 2011. Personal Identification Number (PIN). In *Encyclopedia of Cryptography and Security*. Springer, 927–927.
[2] Bo Chen, Vivek Yenamandra, and Kannan Srinivasan. 2015. Tracking keystrokes using wireless signals. In *Proc. ACM MobiSys'15*. New York, USA.
[3] Georg Dorffner. 1996. Neural networks for time series processing. In *Neural network world*. Citeseer.
[4] Diego Gragnaniello, Giovanni Poggi, Carlo Sansone, and Luisa Verdoliva. 2015. Local contrast phase descriptor for fingerprint liveness detection. *Pattern Recognition* 48, 4 (2015), 1050–1058.
[5] Daniel Halperin, Wenjun Hu, Anmol Sheth, and David Wetherall. 2011. Tool release: Gathering 802.11 n traces with channel state information. *ACM SIGCOMM Computer Communication Review* 41, 1 (2011), 53–53.
[6] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.
[7] Yunye Jin, Wee-Seng Soh, and Wai-Choong Wong. 2010. Indoor localization with channel impulse response based fingerprint and nonparametric regression. *IEEE Transactions on Wireless Communications* 9, 3 (2010), 1120.
[8] John F. Kros, Mike Lin, and Marvin L. Brown. 2006. *Effects of the neural network s-sigmoid function on KDD in the presence of imprecise data.*
[9] Hong Li, Wei Yang, Jianxin Wang, Yang Xu, and Liusheng Huang. 2016. WiFinger: talk to your smart devices with finger-grained gesture. In *Proc. ACM Ubicomp'16*. Heidelberg, Germany.
[10] Jian Liu, Yan Wang, Yingying Chen, Jie Yang, Xu Chen, and Jerry Cheng. 2015. Tracking vital signs during sleep leveraging off-the-shelf wifi. In *Proc. ACM MobiHoc'15*. Hangzhou, China.
[11] Albert Ng, Julian Lepinski, Daniel Wigdor, Steven Sanders, and Paul Dietz. 2012. Designing for low-latency direct-touch input. In *Proc. ACM UIST'12*. Cambridge, MA, USA.
[12] Juhi Ranjan and Kamin Whitehouse. 2015. Object hallmarks: Identifying object users using wearable wrist sensors. In *Proc. ACM Ubicomp'15*. Osaka, Japan.
[13] RTINGS. [n. d.]. TV Size to Distance Calculator and Science. [Online]. Available: https://www.rtings.com/tv/reviews/by-size/size-to-distance-relationship.
[14] Stan Salvador and Philip Chan. 2007. Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis* 11, 5 (2007), 561–580.
[15] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proc. IEEE CVPR'15*. Boston, MA, USA.
[16] Cong Shi, Jian Liu, Hongbo Liu, and Yingying Chen. 2017. Smart user authentication through actuation of daily activities leveraging WiFi-enabled IoT. In *Proc. ACM MobiHoc'17*. Chennai, India.
[17] Statista. [n. d.]. Average size of LCD TV screens worldwide from 2015 to 2021 (in inches). [Online]. Available: https://www.statista.com/statistics/760288/average-tv-screen-size-worldwide/.
[18] Statista. [n. d.]. Smart Home - United States | Statista Market Forecast. [Online]. Available: https://www.statista.com/outlook/279/109/smart-home/united-states.
[19] Statista. [n. d.]. Society-Demographics-Average size of a family in the US 1960-2017. [Online]. Available: https://www.statista.com/statistics/183657/average-size-of-a-family-in-the-us/.
[20] J. A. K. Suykens and J. Vandewalle. 1999. Least Squares Support Vector Machine Classifiers. *Neural Processing Letters* 9, 3 (1999), 293–300.
[21] Sheng Tan and Jie Yang. 2016. WiFinger: leveraging commodity WiFi for fine-grained finger gesture recognition. In *Proc. ACM MobiHoc'16*. Germany.
[22] David M. J. Tax and Robert P. W. Duin. 1999. Support vector domain description. *Pattern Recognit Lett* 20, 11–13 (1999), 1191–1199.
[23] Wei Wang, Alex X Liu, Muhammad Shahzad, Kang Ling, and Sanglu Lu. 2015. Understanding and modeling of wifi signal based human activity recognition. In *Proc. ACM MobiCom'15*. New York, USA.
[24] Yan Wang, Jian Liu, Yingying Chen, Marco Gruteser, Jie Yang, and Hongbo Liu. 2014. E-eyes: device-free location-oriented activity identification using fine-grained wifi signatures. In *Proc. ACM MobiCom'14*. Maui, Hawaii, USA.
[25] Jonathan Wu, Janusz Konrad, and Prakash Ishwar. 2013. Dynamic time warping for gesture-based user identification and authentication with Kinect. In *Proc. IEEE ICASSP'13*. Vancouver, Canada.
[26] Chouchang Yang and Huai-Rong Shao. 2015. WiFi-based indoor positioning. *IEEE Communications Magazine* 53, 3 (2015), 150–157.
[27] Yunze Zeng, Parth H Pathak, and Prasant Mohapatra. 2016. WiWho: wifi-based person identification in smart spaces. In *Proc. IEEE IPSN'16*. Vienna, Austria.
[28] Linghan Zhang, Sheng Tan, Jie Yang, and Yingying Chen. 2016. Voicelive: A phoneme localization based liveness detection for voice authentication on smartphones. In *Proc. ACM CCS'16*. Vienna, Austria.
[29] Yiwei Zhuo, Hongzi Zhu, Hua Xue, and Shan Chang. 2017. Perceiving accurate CSI phases with commodity WiFi devices. In *Proc. IEEE InfoCom'17*. GA, USA.
[30] Han Zou, Yuxun Zhou, Jianfei Yang, Weixi Gu, Lihua Xie, and Costas Spanos. 2017. Freecount: Device-free crowd counting with commodity wifi. In *Proc. IEEE GLOBECOM'17*. Singapore.