Listen to Your Fingers: User Authentication Based on Geometry Biometrics of Touch Gesture

HUIJIE CHEN and FAN LI*, School of Computer Science, Beijing Institute of Technology, China
WAN DU, Department of Computer Science and Engineering, The University of California, Merced, USA
SONG YANG, School of Computer Science, Beijing Institute of Technology, China
MATTHEW CONN, Department of Electrical Engineering and Computer Science, Vanderbilt University, USA
YU WANG, Department of Computer and Information Sciences, Temple University, USA

Inputting a pattern or PIN code on the touch screen is a popular method to prevent unauthorized access to mobile devices. However, these sensitive tokens are highly susceptible to being inferred by various types of side-channel attacks, which can compromise the security of the private data stored in the device. This paper presents a second-factor authentication method, TouchPrint, which relies on the user's hand posture shape traits (dependent on the individual different posture type and unique hand geometry biometrics) when the user inputs PIN or pattern. It is robust against the behavioral variability of inputting a passcode and places no restrictions on input manner (e.g., number of the finger touching the screen, moving speed, or pressure). To capture the spatial characteristic of the user's hand posture shape when input the PIN or pattern, TouchPrint performs active acoustic sensing to *scan* the user's hand posture when his/her finger remains static at some reference positions on the screen (e.g., turning points for the pattern and the number buttons for the PIN code), and extracts the multipath effect feature from the echo signals reflected by the hand. Then, TouchPrint fuses with the spatial multipath feature-based identification results generated from the multiple reference positions to facilitate a reliable and secure MFA system. We build a prototype on smartphone and then evaluate the performance of TouchPrint comprehensively in a variety of scenarios. The experiment results demonstrate that TouchPrint can effectively defend against the replay attacks and imitate attacks. Moreover, TouchPrint can achieve an authentication accuracy of about 92% with only ten training samples.

CCS Concepts: • **Human-centered computing → Ubiquitous and mobile computing**.

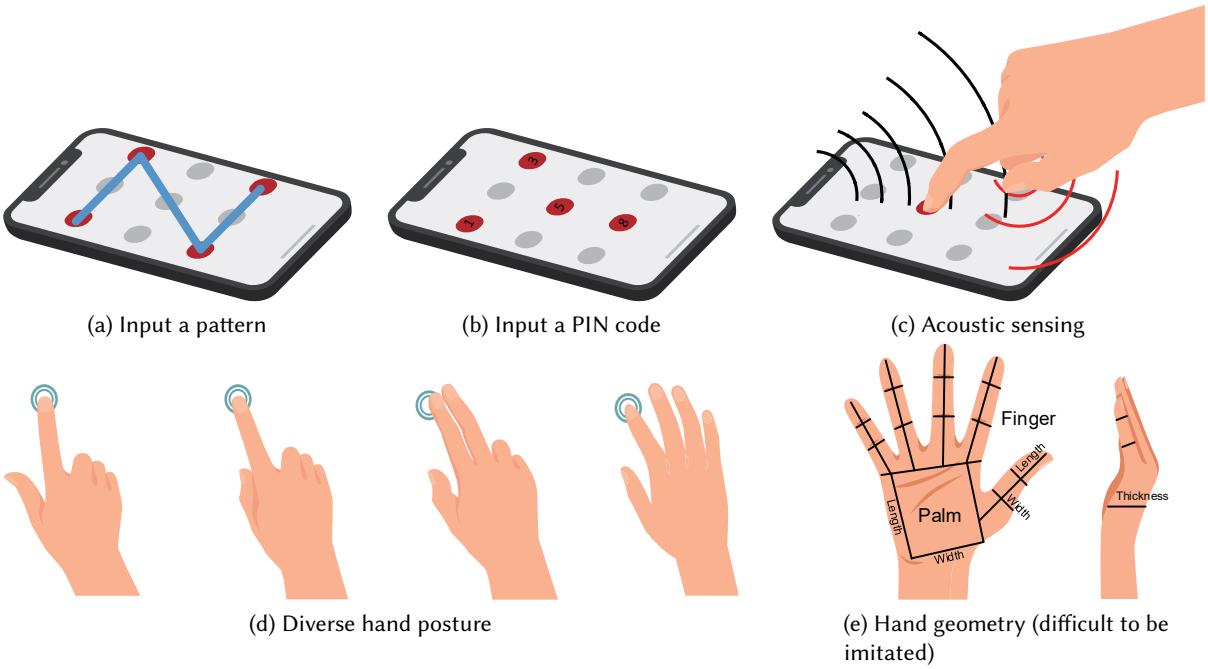
Additional Key Words and Phrases: User Authentication, Acoustic Sensing, Finger Touch Interaction

ACM Reference Format:
Huijie Chen, Fan Li, Wan Du, Song Yang, Matthew Conn, and Yu Wang. 2020. Listen to Your Fingers: User Authentication Based on Geometry Biometrics of Touch Gesture. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 3, Article 75 (September 2020), 23 pages. https://doi.org/10.1145/3411809

*F. Li and Y. Wang are the co-corresponding authors.

Authors' addresses: Huijie Chen; Fan Li, School of Computer Science, Beijing Institute of Technology, Beijing, China, chenhuijie,fli@bit.edu.cn; Wan Du, Department of Computer Science and Engineering, The University of California, Merced, Merced, USA, wdu3@ucmerced.edu; Song Yang, School of Computer Science, Beijing Institute of Technology, Beijing, China, S.Yang@bit.edu.cn; Matthew Conn, Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, USA, matthew.b.conn@vanderbilt.edu; Yu Wang, Department of Computer and Information Sciences, Temple University, Philadelphia, USA, wangyu@temple.edu.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM.
2474-9567/2020/9-ART75 $15.00
https://doi.org/10.1145/3411809



(a) Input a pattern (b) Input a PIN code (c) Acoustic sensing

(d) Diverse hand posture (e) Hand geometry (difficult to be imitated)

Fig. 1. Capturing the unique pattern of hand posture shape when input the passcode (i.e.,pattern and PIN code) via active acoustic sensing.

1 INTRODUCTION

Mobile devices (i.e., smartphones, wearables, and tablets) have become ubiquitous computing and communication platforms that play an essential role in everyday life. For example, users rely on these mobile devices to store personal information or to access sensitive online services (e.g., shopping, medical appointments, or banking). To prevent the smartphone from being used by an unauthorized person, various user verification mechanisms are proposed for user authentication. Specifically, camera-based facial recognition exhibits both convenience issues (the need to perform special motions like looking up or blinking) and environmental issues (sensitivity to brightness and phone orientation). Many commercial mobile devices have also been equipped with specialized hardware like fingerprint sensors, but they are easily fooled and do not support liveness verification. In contrast, passcodes (i.e., pattern lock shown in Fig. 1(a) or PIN code shown in Fig. 1(b)) based methods are faster, convenient and comfortable, which make them still most popular and be selected as the first choice by 73% consumers during mobile payment [22].

However, recent works show that it is possible to infer these sensitive passcodes via side-channel attacks. Specifically, the PIN code or pattern can be tracked by motion sensors embedded in wrist-worn devices [24, 25], nearby smartphones via acoustic sensing [16], or surrounding Wi-Fi signals [17]. Moreover, a camera placed far away from the user may also be able to infer the passcode [30] only by observing the passcode input motion. Fortunately, some effort has been made to provide additional second-factor user verification for PIN code or pattern-based authentication methods. These efforts may be grouped into two categories. The first type of method recognizes the user's on-screen behavioral biometrics and extracts the corresponding pressure and velocity information for user authentication [14, 33]. These methods often lead to less accurate authentication with a

high false-positive rate due to inevitable behavioral variability between inputs. The second kind of method uses the multi-touchpoint sensing ability of touchscreens to extract the geometric relationships between multiple fingers for user authentication [20]. This method is difficult to directly apply as existing passcode systems often only support single finger entry (e.g., PIN).

The above limitations motivate us to design a new user authentication method, called *TouchPrint*, which provides second-factor user authentication for the PIN code or pattern-based authentication without any additional specialized hardware. Our key observation is that the user moves a single finger to touch the screen with his/her preferred hand posture when she inputs a passcode (i.e., pattern or PIN). The hand posture is always diverse between users (Fig. 1(d)) and cannot be inferred by the non-visual side-channel attacks (i.e., the method based on the motion sensor, acoustic sensing, and Wi-Fi signal). Besides, different users have unique hand geometry biometrics [19], shown in Fig. 1(e), an adversary can still be detected even if imitating the user's hand posture to input the passcode after watching him/her inputs actions on-site or in a watch [30]. Therefore, when a user inputs the pattern or PIN on his/her touch screen, the $3D$ shape of the user's hand posture can be used as an identifier for authentication due to the distinct characteristics and difficult of imitation. Additionally, the microphone and speaker embedded on the smartphones can enable active acoustic sensing to *scan* the surrounding environment and depicts its spatial characteristics through the analysis of the multipath reflection signals. These principles inspire the basic idea of TouchPrint: to use active acoustic sensing on the smartphone to *scan* (Fig. 1(c)) the user's hand posture shape when inputting the passcode, and then to extract the unique spatial pattern (dependent on the hand geometry biometrics and posture type) from the multipath reflection signals for the second factor user authentication.

Despite its simple idea, three major challenges underlie the design of TouchPrint:

- *How to obtain the acoustic fragment when the finger is static in reference positions for reliable multipath feature extraction?* During the passcode input process (i.e., the pattern shown in Fig. 1(a) and the PIN code shown Fig. 1(b)), the on-screen finger always pauses in several fixed positions (i.e., turning points or digital numbers) for a short but measurable period. Therefore, the acoustic multipath feature when the finger is static at these positions (called *landmark positions*) can be used for authentication. Due to the uncontrollable delay of acoustic signal transmission, TouchPrint has to perform a continuous active acoustic sensing to avoid missing such short moments. However, it is insufficient to accurately determine whether the finger is static at a landmark position only by analyzing the acoustic characteristics (e.g., frequency and phase) of the collected audio signal. To address it, TouchPrint first uses the finger trace data to estimate the static finger period and then detects the finger tapping sound position in the acoustic signal. Finally, the acoustic fragment can be segmented out from the detected tapping sound position (used as the start point) with the length of the estimated static finger period.
- *How to extract the fine-grained multipath effect features from the segmented acoustic fragment to characterize the hand geometry biometrics and posture distinctness?* The segmented acoustic fragment contains various types of multipath signals that are reflected not only from the hand but also from surrounding objects in the environment (i.e., the user's body, other users, furniture and walls). The multipath propagation from other objects will significantly interfere the user authentication. To overcome this issue, we adopt a Zadoff-Chu (ZC) sequence-based acoustic signal to eliminate these acoustic multipath interferences and extract the fine-grained acoustic multipath response, which reflects the hand geometry biometrics and posture distinctness.
- *How to design a reliable, secure authentication method with limited samples?* In the actual application scenario, TouchPrint only requires the user to collect a limited number of samples (i.e., ten samples) in the registration stage. This is done to enable a friendly user experience, but it poses a challenge of being robust to the inconstant hand posture or inaccurate touch position in such a small number of training samples.

To overcome this, TouchPrint adopts an occupancy map-based method to filter the outlier composition and leverages a vote-based method to combine the results of multiple reference positions for reliable authentication.

The main contributions of this paper are summarized as follows:

- We design a new authentication method, TouchPrint, which provides second-factor verification for the existing passcode-based authentication method. TouchPrint performs active acoustic sensing to extract the fine-grained acoustic multipath effect features corresponding with the hand geometry and postural information when the finger is static on the touch screen. It does not require any active user intervention in terms of behavior pattern (speed or pressure of the tapping, swiping) and does not restrict the number of fingers touching the screen, which makes it compatible with almost all the types of finger touch interaction-based authentication.
- Our method exploits the finger touch traces (coordinates and timestamps) on the screen for accurate acoustic signal segmentation (i.e., obtaining the acoustic fragment when the finger is static in reference positions). It uses the high autocorrelation characteristics of the ZC sequence-based acoustic signal to extract the multipath response that reflects the hand posture shape distinctness. Then, our method combines the acoustic multipath response-based results from multiple reference positions to ensure that it is robust towards inconsistent user gesture inputs, even with a limited training sample size.
- We implement a prototype of TouchPrint on commodity smartphone and conduct extensive experiments in a variety of scenarios. The experiment results demonstrate that our approach can effectively defend against replay attacks and observe-imitate attacks. It can achieve an authentication accuracy of about 92% with only ten training samples.

The remaining parts of this paper are organized as follows. Section 2 presents the design overview of our system. Section 3 shows how to use the collected touch trace for acoustic signal segmentation. Section 4 describes the technical details on the acoustic signal generation, hand multipath separation, and multipath effect enrichment. Section 5 shows the design details of feature extraction and the verification procedure. Section 6 presents our evaluation results. Section 7 and Section 8 describe the related work and discussion, respectively. Finally, Section 9 concludes the paper.

2 SYSTEM DESIGN

In this section, we first present several adversary models that need to be defended and then provide an overview of our proposed user authentication method.

2.1 Adversary Models

The proposed user authentication method is designed to resistant against the adversary models described below.

Replay Attack: the pattern or PIN code may be inferred by the non-visual based side-channel attacks (e.g., motion sensor, acoustic or Wi-Fi signal). In these types of attacks, the adversary knows only the passcode information but not the victim's hand posture and hand biometrics (i.e., finger length and palm-size). Then, the adversary only can perform a replay attack by inputting the inferred passcode using his/her hand in a random hand posture.

Observe and Imitate Attack: this attack can be performed when the adversary observes the victim inputting the pattern or PIN (e.g., direct observation or from the recoded video) and then tries to input the passcode by imitating the victim's hand posture.

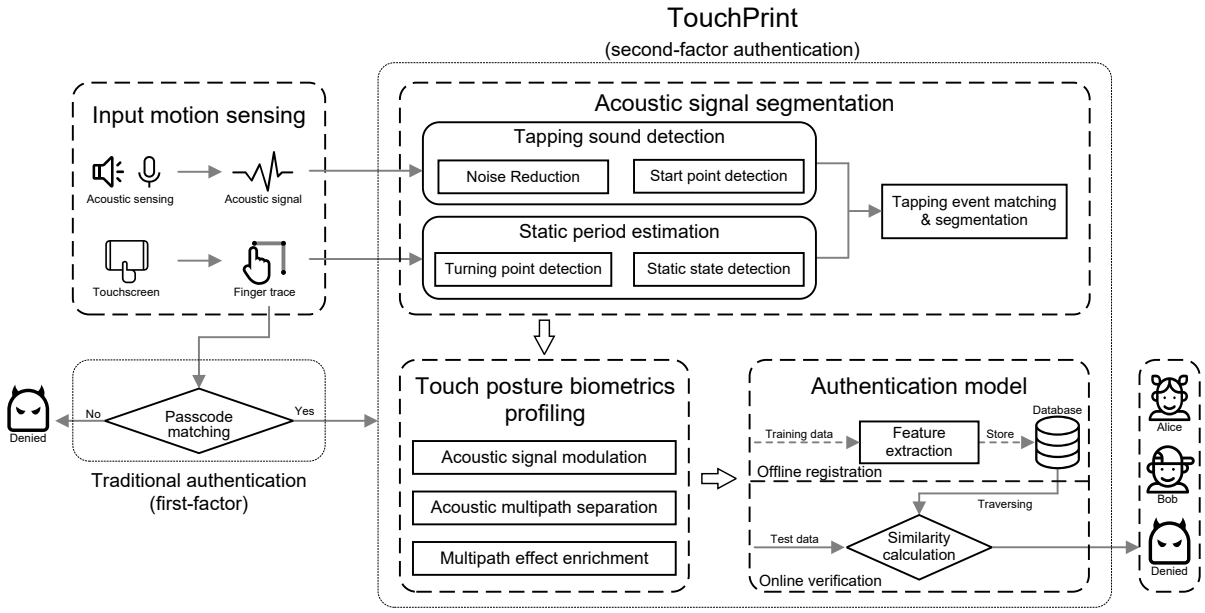


Fig. 2. System overview of TouchPrint.

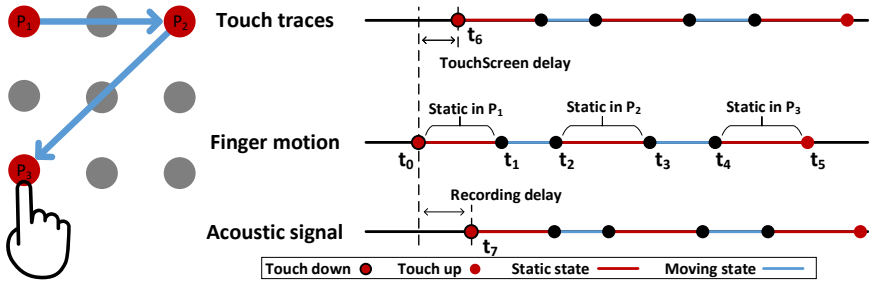
2.2 System Overview

The proposed method aims to provide second-factor verification for the passcode-based user authentication system. It uses the unique spatial characteristics of a user's hand posture during the passcode input procedure as biometric for user authentication. Note that the on-screen finger motions of the pattern and PIN inputs are different. Specifically, the on-screen finger movement for the pattern case always contains one $TouchDown$ event, one $TouchUp$ event, and many on-screen traces generated from finger swiping. In general, when arriving at the turning point of the pattern, the finger is inevitably static for a short period while changing direction. Moreover, the finger movement during the user input the PIN contains many pairs of $TouchDown$ and $TouchUp$ events. Similarly, the finger is inevitably static for a brief instant at each number's position in the PIN sequence, while the user taps the number.

To recognize the spatial character of the user's hand posture, TouchPrint performs continuous active acoustic sensing and segments the recorded acoustic signal to extract the multipath effect when the finger is static in the reference positions (i.e., turning points or number buttons). Since the finger movements between the pattern input and PIN input are different, the procedure to estimate the static finger period is also different. The next several paragraphs will clearly describe the system overview for the pattern case. The finger static period estimation for the PIN code case will be described in Section 3.4.

The system overview of TouchPrint is shown in Fig. 2, which includes the following components:

- **Input motion sensing:** during the pattern input procedure, the finger trace (consisting of the coordinate (x_i, y_i) and timestamp t_i) will be collected from the touch screen. TouchPrint uses the speaker and microphone (located at the top and bottom of the smartphone, respectively) to perform continuous active acoustic sensing. The trace information will be used to obtain the acoustic signal fragment when the finger is static in the landmark positions.



(a) Example (b) Motion event timeline

Fig. 3. Pattern input example and its corresponding motion event timeline.

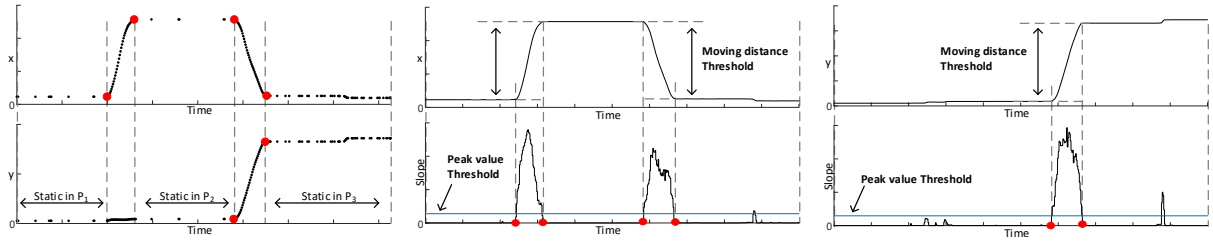
- **Acoustic signal segmentation:** this component segments out the acoustic fragment during the period when the finger is static on the landmark positions and comprises three steps. First, the turning point and the static state detection over the collected touch trace, are used to estimate the static period of the finger at the turning point. Second, applying a low pass filter helps process the recorded acoustic signal to accurately detect the sound of finger tapping and locate the event within the collected acoustic samples. At last, we isolate the acoustic fragment (from the recorded acoustic signal) along with the starting point of the tapping and estimated static period.
- **Touch posture biometrics profiling:** the multipath signal related to the input hand posture will be extracted from the acoustic fragment. Then, the hand geometry's feature space can be enriched by leveraging different combinations among the speakers and microphone on the phone.
- **Authentication model:** the last component comprises two stages: registration and verification. In the registration stage, TouchPrint requires the user to input a passcode (i.e., PIN or pattern) multiple times with a consistent hand posture that is comfortable for the user. The corresponding extracted acoustic multipath effect from these samples will be fed into the feature extraction process to generate the principal components as the user features that will be stored in the database. In the verification stage, TouchPrint traverses the feature profiling from the database and authenticates the user based on the similarity between these features and the new input sample.

3 ACOUSTIC SIGNAL SEGMENTATION

3.1 Intuition

We now describe how to accurately obtain the acoustic fragment when the finger is static on landmark positions. Fig. 3(a) and Fig. 3(b) are an example of inputting a pattern on the touch screen and the corresponding motion event timeline. Specifically, the finger first taps the position P_1 on the touchscreen at time t_0. Next, the finger moves to the position P_2 and then to P_3. Finally, it leaves the touchscreen at time t_5. The entire sequence during the pattern input procedure contains three static periods (i.e., $t_0 \sim t_1$,$t_2 \sim t_3$, and $t_4 \sim t_5$) and two moving periods (i.e., $t_1 \sim t_2$ and $t_3 \sim t_4$).

One straightforward way to capture the acoustic multipath effect during the static period is to launch the active acoustic sensing once the finger stays static on a landmark position. However, the static period may be missed by the active acoustic sensing due to the uncontrollable touchscreen delay and acoustic sensing delay (i.e., $t_6 - t_0$ and $t_7 - t_0$ shown in Fig. 3(b)). To address it, TouchPrint will continuously perform the active acoustic sensing (i.e, emit the sensing signal and record the acoustic response) until the entire input procedure is completed (i.e.,



(a) Period division (b) Static period estimation for the X axis (c) Static period estimation for the Y axis

Fig. 4. Finger static period estimation using the touch traces.

the user lifts the finger from the touch screen), so that the entire finger static period can be covered. Therefore, the collected acoustic samples contain all the fragments that finger is static on landmark positions.

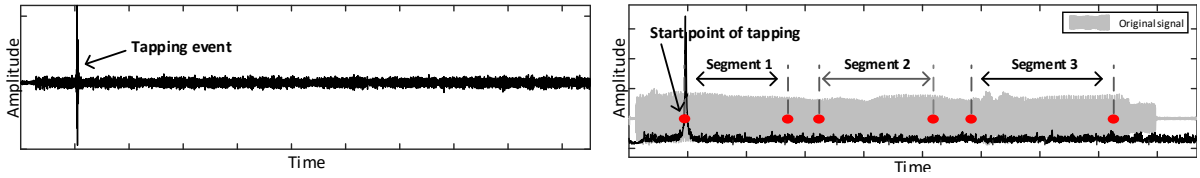
To segment out above acoustic fragments, the finger movement state is first to be determined. In general, the state of the finger (i.e., static/in motion) could be determined based on the Doppler frequency shift and phase information of the recorded acoustic signal. However, these methods cannot detect whether the finger stops at the landmark positions when static (e.g., pauses in the air). Moreover, finger touching can be detected and localized outside the screen area with the structure sound propagation [21]. But the coarse-grained localization accuracy is not sufficient to be applied in the scenario of TouchPrint. Therefore, it is unrealistic to get the acoustic fragment, corresponding the finger is static in reference positions, purely based on the recorded acoustic signal.

Instead, TouchPrint uses the finger trace data from the touch screen to estimate the static finger period for acoustic signal segmentation. Our reasoning for this is threefold. First, the finger trace data is naturally obtained in the scenario of finger-touching authentication. Second, the finger state (static/in motion) can be easily detected based on the touch coordinate changes. Specifically, the collected finger touch coordinate does not change during the period that the finger is static. Third, the duration time of the static period can be accurately estimated via touchscreen sensing since the touchscreen delay from interaction to feedback has millisecond-level lag (i.e., $t_6 - t_0$ shown in Fig. 3(b)) and stable delay jitter [5].

In addition, the finger motion of a $TouchDown$ event at time t_0 will generate an acoustic response when the finger hits the screen. Then, the $TouchDown$ event can be detected from the recorded audio based on this acoustic response. Therefore, the acoustic fragment can be segmented out from the detected $TouchDown$ event position (used as a start point) with the length of the estimated static finger period. Note that our segmentation method only refers to the detection of the $TouchDown$ event sound. We do not need to detect the $TouchUp$ sound since the timestamp in the recorded acoustic signal can be determined by combining the $TouchDown$ event timestamp and the duration of the static period (estimated from the finger traces). The following subsection will introduce the technical details of both finger static period estimation and finger tapping sound detection. The case of segmentation for the PIN input is also discussed.

3.2 Finger Static Period Estimation

The collected touch trace coordinates (x_i, y_i) and time samples $(t_i, 1 \leq i \leq n)$ are used to estimate the finger static periods. Since finger movement yields change for the trace coordinates, we need to detect the turning point of the time-position series for the X and Y axes separately. As shown in Fig. 4(a), the sampling frequency of touch samples is nonconstant. Specifically, the sampling frequency is decreased when the finger is static but increases when it is in motion. Therefore, we first apply a linear interpolation to the time-position series of the X and Y axes. These results are shown in the upper figure of Fig. 4(b) and Fig. 4(c). Next, we calculate the slope for the magnitude of interpolated series, which is shown in the lower figure of Fig. 4(b) and Fig. 4(c).



(a) Finger tapping event (b) Tapping start point detection

Fig. 5. Finger tapping sound detection and acoustic signal segmentation.

In the slope figure for the time-position series of the X and Y axes, the peak area corresponds with the period when the finger is in motion. Then, the start and end times of the period can be determined by detecting the root point (marked in red) before and after each peak. Notably, the finger's micro-shake motion always generates a peak in the slope of the time-position series, which can lead to an incorrectly detected period. To eliminate these errors, we apply a threshold-based method for determining peaks. Specifically, we first filter the peaks whose values are less than 10% of the highest detected peak, and then remove any detected root point pairs when the coordinate changes are less than a threshold, which set as half the distance between adjacent points in the pattern (roughly 200 pixels). Note that the detected turning points belong to the interpolated time-position series. We then transfer these identified turning points to the original time-position series by finding the nearest samples with the minimum Euclidean distance.

Next, we combine the turning point times T_x and T_y (detected from the original time-position series of the X and Y axes) and use these times for the static period division. These turning points are combined with the starting sample time T_1 ($TouchDown$ event) and ending sample time T_n ($TouchUp$ event) of the original series. The union of the set, denoted as $\{T_1, T_x, T_y, T_n\}$ will be used for static period division with the following process: 1) Remove duplicate elements and sort in ascending order. 2) Traverse the set using a sliding window (use step size of 1 and a window length of 2). In each sliding window, denote the left element as t_l and the right element as t_r. 3) The period $[t_l, t_r]$ will be detected as the finger static period when $|x_{t_l} - x_{t_r}| < \alpha$ and $|y_{t_l} - y_{t_r}| < \alpha$, where the threshold value α is set as 10 pixels for our implementation.

3.3 Tapping Sound Detection and Segmentation

The motion of a user tapping the touchscreen with a finger always generates a weak acoustic signal. Thus, the tapping sound may be used by an attacker to infer the input password [2] or text information [8]. Instead, we will use the tapping sound to match the $TouchDown$ event in the trace and extract the finger static acoustic segment. Next, the detail of finger tapping sound detection is described. First, the recorded acoustic signal for the example in Fig. 3(a) will be processed with a bandpass filter ($5Hz \sim 200Hz$) and the output signal is shown in Fig. 5(a). The significant pulse in the left part is the tapping sound. Second, the envelope $\mathrm{E}(s)$ of the filtered signal s is extracted with following equation:

$$\mathrm{E}(s) = H(|s|), \tag{1}$$

where $H(\cdot)$ is the Hilbert transform. To remove outliers, a sliding window-based data smoothing method is adopted for the envelope signal. The window size is set to 1/10 of the signal length and the step length is set to 1/4 of the window length. The smoothed envelope signal is shown in Fig. 5(b).

The sample index of the peak can be detected by adopting a peak finding method. We traverse the signal backward from the peak position to select which sample is the starting point of the tapping sound by using the condition that this sample value is less than the threshold value, defined as 1/5 of the peak height. Finally, the

acoustic fragment can be segmented out from the detected start point of the tapping sound with the length of the estimated static finger period.

3.4 For PIN Code Scenario

For the PIN-based user authentication method, the finger motion on the touch screen contains many event pairs of $TouchDown$ and $TouchUp$ events. The acoustic fragment during the static period can be directly isolated by detecting these $TouchDown$ and $TouchUp$ events in the recorded acoustic signal. However, the $TouchUp$ event sound (which is emitted when the finger leaves the screen) is too weak to be detected effectively. Therefore, TouchPrint still relies on the touch screen to record the timestamps of the $TouchDown$ and $TouchUp$ events for segmentation. Specifically, we first detect the starting point of each $TouchDown$ event using the method mentioned in Section 3.2. The timestamp of each $TouchDown$ event and $TouchUp$ event are collected by the touchscreen. The interval between these two timestamps will be considered as the static finger period. Finally, we cut out the finger static acoustic segment beginning from the starting point of each detected $TouchDown$ event with this newly estimated finger static period.

4 TOUCH POSTURE BIOMETRICS PROFILING

TouchPrint relies on the auto-correlation properties of the Zadoff-Chu sequence for fine-grained acoustic multipath separation. Thus, the multipath response that reflects the hand geometry and posture distinctness can be extracted accurately.

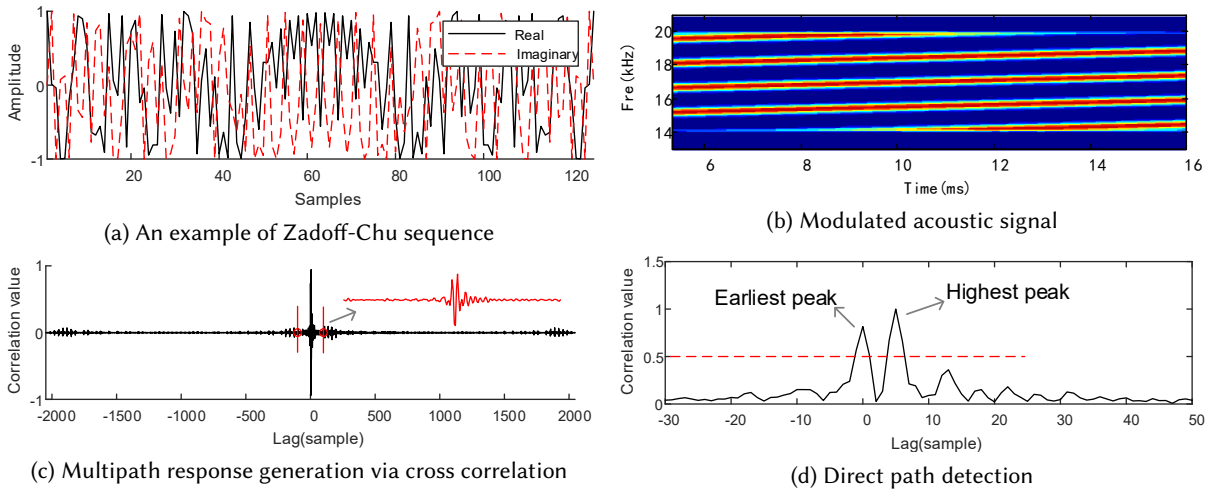
4.1 Active Acoustic Sensing Procedure

In general, a commercial smartphone is always equipped with the speaker and microphone, which are placed at the top and bottom of the phone as shown in Fig. 1(c). These two sensors will be used in our system for active acoustic sensing. The active acoustic sensing starts when the user enters the App to input the password or pattern and stops once the finger is lifted from the screen (for the pattern) or the *Enter* button is clicked (for the PIN). During the sensing procedure, the speaker continuously emits a pre-designed acoustic signal described in Section 4.2. Meanwhile, the microphone is continuously recording. The recorded acoustic signals will be processed as described by Section 3 to extract the acoustic fragment when the finger stays static in the landmark positions. These extracted audio fragments will be used to estimate the multipath effect corresponding with the hand part (described in Section 4.3).

4.2 Acoustic Signal Selection and Modulation

During the pattern input procedure (as shown in Fig. 1(a)), the recorded acoustic signal via active acoustic sensing always contains two signal types: i) the direct path signal emitted from the top speaker and recorded by the bottom microphone, and ii) the multipath signal reflected off the hand and surrounding objects (i.e., user body, wall, ceiling or other furniture). Thus, we need to separate the reflection signals corresponding with the user's hand from the multipath signals for authentication. Existing works [9, 23, 34] separate the reflected signal corresponding to the user's face or surrounding environment with the chirp signal. In these scenarios, the propagation path of the reflected signal ($> 50cm$) is generally much longer than the direct path (about $15cm$). In contrast, the reflected signal's propagation path ($< 30cm$) in our scenario is much closer to the direct path's length, which leads to more difficulty for reflection signal separation.

The Zadoff-Chu sequence has been used in several works (e.g., acoustic ranging [1] and finger movement tracking [21]) due to the auto-correlation properties. Inspired by these works, we apply the Zadoff-Chu sequence to accurately extract the spatial multipath feature of the hand posture shape. A Zadoff-Chu sequence s_k of length N_{zc} is defined as follows:



(a) An example of Zadoff-Chu sequence

(b) Modulated acoustic signal

(c) Multipath response generation via cross correlation

(d) Direct path detection

Fig. 6. Hand multipath signal separation.

$$s_k = e^{flag \times i\alpha_k}, \quad k = 0, 1, 2, \cdots, N_{zc} - 1, flag = +1 \ or \ -1, \tag{2}$$

where α_k is defined as follows:

$$\alpha_k = \begin{cases} u\pi k^2 / N_{zc}, & N_{zc}\%2 = 0 \\ u\pi k\left(k+1\right) / N_{zc}, & N_{zc}\%2 = 1, \end{cases} \tag{3}$$

where u and N_{zc} are integers with u coprime to N_{zc} and % is the modulo operator. An example of a Zadoff-Chu sequence is shown in Fig. 6(a) with sequence length $N_{zc} = u \times N_{chirp} + 1 = 125$, when $u = 31$, $N_{chirp} = 4$ and $flag = +1$, meaning that modulated acoustic signals will contain $N_{chirp} = 4$ chirps.

The generated Zadoff-Chu sequence is a polyphase complex-valued sequence (containing the real part and imaginary part) and needs to be modulated into the acoustic signal (only contains the real value) for active acoustic sensing. The acoustic signal modulation procedure contains the following two steps:

Upsampling: this step increases the sampling rate f_s of the generated Zadoff-Chu sequence s_k (whose length is N_{zc}) to $48kHz$ through zero-padding in the frequency domain. After this step, the length of the generated signal s_k^I will be increased to $N_{zc}^I = N_{zc} f_s / B$ for the given target signal's bandwidth B. The procedure of zero padding in the frequency domain is denoted as follows:

$$s_k{}^I = \mathcal{F}^{-1}\left\{\mathcal{S}\left\{\mathcal{Z}\left\{\mathcal{S}\left\{\mathcal{F}\left\{s_k\right\}\right\}, N_{zc}^I\right\}\right\}\right\}, \tag{4}$$

where $\mathcal{F}$ and $\mathcal{F}^{-1}$ represent the Fourier transform and inverse Fourier transform operations. $\mathcal{S}$ moves the zero-frequency component to the center of the spectrum. For a vector, $\mathcal{S}$ will swap the left and right halves. For a sequence $X = [x_i]$ where $1 \le i \le N_{zc}$, $\mathcal{Z}\left\{X, N_{zc}^I\right\}$ is used to insert the zeros before and after the sequence X, to increase its length to the length of N_{zc}^I. Specifically, we denote the added zero number as $N_{zero} = N_{zc}^I - N_{zc}$, then $\mathcal{Z}\left\{X, N_{zc}^I\right\} = \left[0_{i_b}, x_i, 0_{i_a}\right]$ $1 \le i_a \le N_{zero}/2$, $1 \le i_b \le N_{zero}/2$ when $N_{zero}\%2 = 0$, otherwise, $1 \le i_a \le N_{zero}/2+1$, $1 \le i_b \le N_{zero}/2$.

Acoustic signal generation: the acoustic signal $A(t)$ used for active sensing should be inaudible to avoid disturbing users. Thus, the above Zadoff-Chu sequence should be multiplied by a carrier signal with a high frequency of f_c as follows:

$$A(t) = \cos(2\pi f_c t)\, Re\left({s_k}^I\right) - \sin(2\pi f_c t)\, Im\left({s_k}^I\right), \tag{5}$$

where $Re({s_k}^I)$ and $Im({s_k}^I)$ are the real and imaginary parts of ${s_k}^I$, respectively. An example of a generated acoustic signal is shown in Fig. 6(b).

Larger N_{chirp} value leads to that more chirps can be modulated in the generated acoustic signal, which benefits the multipath separation. Therefore, in order to accurately separate the reflection signals corresponding with the different hand parts, we select $u = 63$, $N_{chirp} = 6$, $flag = +1$, meaning that the length $N_{zc} = 379$. We set $N_{zc}^I = 2{,}048$ and the audio frequency bandwidth $B = N_{zc} f_s / N_{zc}^I \approx 8{,}883$. This is advantageous because a wider frequency bandwidth yields better autocorrelation. We also set the carrier signal frequency, f_c, to $17kHz$ so that the modulated signal ranges from 12,558.5Hz to 21,441.5Hz. This frequency range may be slightly audible to some people, so the signal volume is reduced to minimize annoyance for users.

4.3 Multipath Signal Separation

The extracted acoustic signal fragment always contains both the direct path signal and the multipath signals (reflected by nearby objects). Using the high auto-correlation characteristic of above ZC-sequence based acoustic signal, the multipath signals of hand can be separated from the recorded signal. This is done by applying the correlation function for the acoustic segment $R = \{r_i, i \in [1, M]\}$ and the designed active sensing signal $A = \{a_i, i \in [1, N]\}$. The correlation result $C = \{c_k, k \in [1, M + N - 1]\}$ of R and A is calculated by following equation:

$$c_k = \sum_{j=1}^{N} a_j r_{k+j}. \tag{6}$$

Fig. 6(c) is an example of the correlation between the acoustic segment and the designed active signal. Because the hand is closer to the smartphone than nearby objects, it is necessary to locate the direct path signal in the correlation result and to extract only the near signals, which can be seen as the multipath signal reflected by the hand part.

In general, the highest peak in the correlation result of C corresponds to the direct path because the surrounding object and loss will absorb the reflected signal's energy through the longer propagation path. However, for the No-Line-of-Sight (NLOS) scenario where the hand impedes the direct path signal, the reflection signal may be stronger. To solve this, the earliest peak will be considered the direct path. Fig. 6(d) shows an example of direct path detection. Specifically, we first obtain the envelope by applying a Hilbert transform to the positive section of the correlation result. The highest recorded peak can be viewed as the candidate result. Then, we check if any peaks above the threshold value come prior to the candidate one. If so, the earliest peak should be chosen as the true direct path. In our system, the threshold value is set to one half of the maximum peak value. Finally, the fragment near the direct path peak will be used as the *multipath response* corresponding with the hand. In our implementation, the fragment length is set with 70 samples.

In practice, the length of the extracted acoustic segment may be shorter than the designed acoustic signal, which means $M < N$. This type of acoustic segment will be discarded. Thus, this segment will be used to generate the multipath response only when $N \leq M$. However, when $N \leq M \leq 2N$, the segment may not contain a complete active sensing signal. In this case, we apply the circle cross-correlation function to generate the multipath response. First, a subsegment $V = \{v_i, i \in [0, N - 1]\}$ will be extracted from the center position of the acoustic segment R. The circle cross-correlation result can be calculated as:

$$C_{\text{circle}}(k) = \sum_{j=1}^{N} v_j a_j^k, \tag{7}$$

where $\mathrm{A^k} = \left\{a_j^k, i \in [1, N]\right\}$ circularly shifts each element in A by k positions.

For the scenario where $M \geq 2N$, we also use above steps to generate the multipath response, which is more computationally efficient than directly computing the cross correlation between R and A.

4.4 Multipath Effect Enrichment

In recent years, mobile phone hardware increasingly supports high definition audio capabilities targeted at audiophiles. Specifically, the phone supported stereo output spans a wide variety of off-the-shelf phones and tablets, such as Samsung Nexus 10 tablets, HTC One M8, Sony Xperia Z2, Google Nexus 6P and Google Pixel 4. Compared with the traditional smartphone (mono channel playback only), this type of phones is equipped with two homogeneous speakers located at the top and bottom of the phones. Moreover, this type of phone enables the two speakers to play different audio streams independently. It supports to sense the various sides of the input gesture to enrich the hand multipath effect during the passcode input procedure.i

TouchPrint uses the speaker array embedded in the smartphone (especially for the phone supporting stereo output) to enrich the hand multipath effect during the passcode input procedure. To avoid the conflict between the two speakers, TouchPrint enables the top and bottom speakers to independently emit two orthogonal ZC sequences-based acoustic signals (i.e., $flag = +1$ and $flag = -1$ in Equ. 2), respectively. These two types of ZC sequences based acoustic signals have lower cross-correlation characteristics. So, the multipath response generation for these two signals does not interfere with each other.

4.5 Case Study

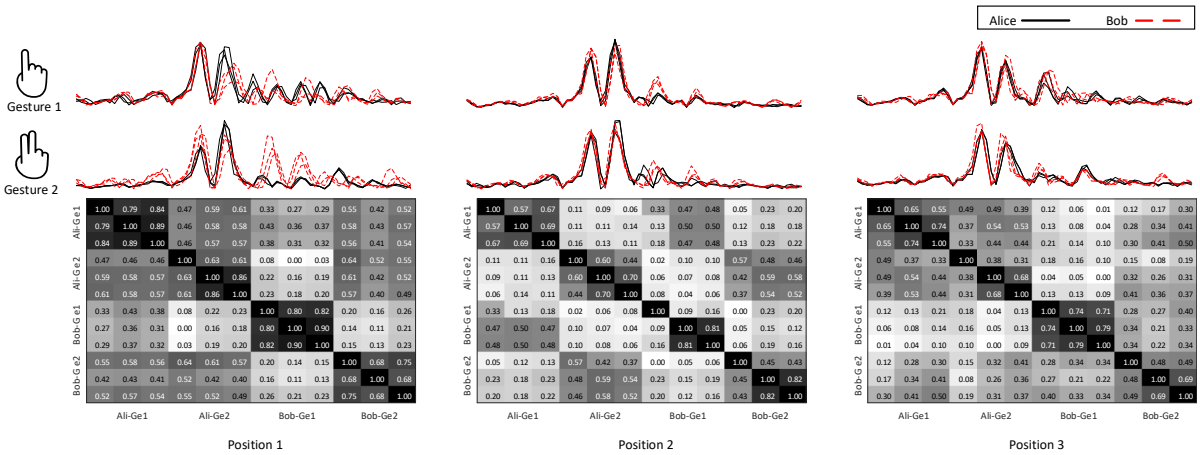
A case study is performed to verify whether the generated multipath response can reflect the individual difference on the hand biometrics (i.e., finer length and palm-size) and posture types. The hypothesis is that the difference between samples from the same user with different postures indicates that the generated multipath response can distinguish the posture types. Moreover, the difference between the samples from the different users with the same posture indicates that it can reflect the individual difference on the hand biometrics (i.e., finer length and palm-size). Thus, two users (called Alice and Bob) participate in the case study and each of them will input the pattern in Fig. 3. These two users have different hand sizes (Alice: $17cm \times 9cm$ and Bob: $15cm \times 8.5cm$). During the pattern input procedure, the touch traces on the screen and recorded acoustic signal are used to generate the hand multipath response. Each user will input the pattern with two gestures shown in Fig. 7. With the $Gesture1$, the user will only use the forefinger to touch the screen and grip other fingers. In contrast, with the $Gesture2$, the user will outstretch both the forefinger and middle finger, but only uses the middle finger to touch the screen. With each gesture, the user will input the pattern three times.

The generated multipath response and the corresponding similarity matrix are shown in Fig. 7. We can discover the following findings. (1) For each user, the multipath response corresponding various gestures is different even in the same landmark positions on the screen. (2) The multipath response will be unique for various types of gestures, and (3) it shows observable differences between users, even when they use the same gesture and touch positions. (4) The hand multipath response can be effectively enriched through sensing with multiple speakers.

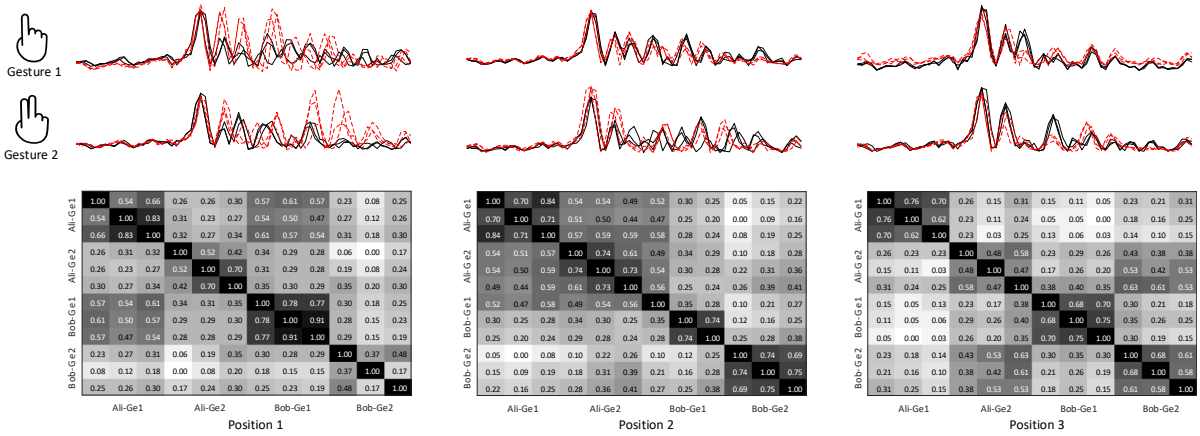
Based on the above findings, we can conclude that the generated hand multipath response can distinguish the hand posture and geometry biometrics from different users, thus being applicable as the identifier for passcode based user authentication.

5 AUTHENTICATION MODEL

The user should perform registration before using the system for the first time. In the registration state, the user is required to input his/her passcode with a consistent hand posture (e.g., Gesture 1 in Fig. 7) for multiple trials. Then, for successful authentication in the verification stage, the user must input the passcode using the same



(a) Sensing with the top speaker and bottom microphone



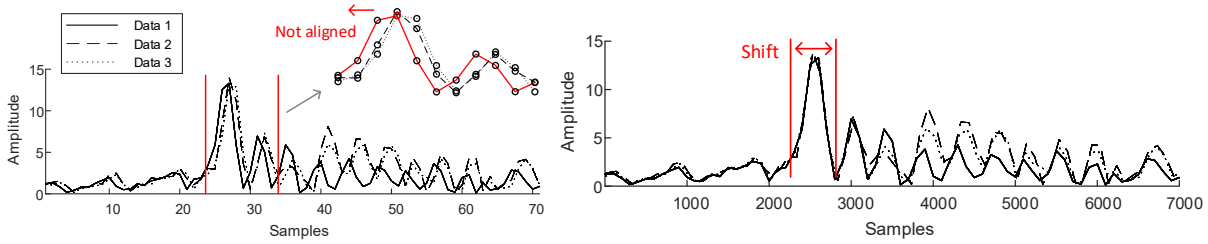
(b) Sensing with the bottom speaker and bottom microphone

Fig. 7. Generated hand multipath responses and the corresponding similarity matrix between two users who input the pattern (shown in Fig. 3(a)) with two gestures.

hand posture as in the registration stage. This section will describe how to generate the multipath response based feature for each landmark position and each sensing channel in the registration state as well as how to authenticate the user for the single-user model and multi-user model.

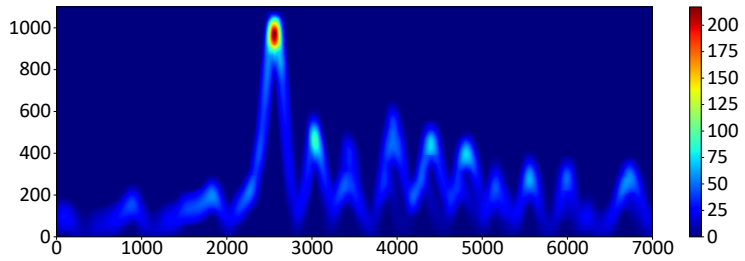
5.1 Feature Generation

We assume that m records are collected from a user during the registration state. For the kth record, the multipath response $R(i,j)_k$ of the ith landmark position with the jth sensing channel will be generated as described in Section 4.3. This subsection will discuss how to generate the feature for ith landmark position with the jth sensing channel based on the multipath responses $R(i,j) = \{R(i,j)_k, k \in [1,m]\}$. The multipath response difference between Alice and Bob (Fig. 7) can be observed visually. In this way, the users can be distinguished



(a) Three unaligned multipath response samples

(b) Alignment via shifting



(c) Generated occupancy map (2D matrix)

Fig. 8. Feature extraction with the generated multipath responses in the registration state.

based on the similarity of their multipath response. However, we find that directly using Euclidean distance will not accurately calculate the similarity. For example, Fig. 8(a) shows the multipath responses of landmark position 1. In Fig. 8(a), Alice has inputted the pattern with gesture 1. The highest peak (direct path) of data 1 is not aligned with data 2 and 3 due to the limited sampling rate.

To overcome this issue, the multipath responses $R(i, j)$ first should be realigned. Specifically, we first adopt a linear interpolation for each multipath response $R(i, j)_k$ since the offset may be less than the sample interval distance. Then, the 1st multipath response $R(i, j)_1$ is selected as the base. The direct path peak segment from the other multipath responses $R(i, j)_k, k \in [2, m]$ will be shifted by $s \in [-w, w]$ samples and matched with the direct path peak segment of base $R(i, j)_1$. The offset $\hat{s}$ occurs at the minimum distance between the shifted peak segment and base peak segment. Finally, the aligned multipath responses $\widetilde{R}(i, j)$ (as shown in Fig. 8(b)) are generated by shifting these multipath responses $R(i, j)_k, k \in [2, m]$ with the corresponding determined offset $\hat{s}$.

Secondly, an occupancy map-based method is adopted to filter the outlier data from an inconsistent user hand posture. The occupancy map can be regarded as a confidence map made up of two-dimensional matrix $M_{i,j}$, whose column number is set with the length of the generated multipath response. The raw number is set at 1100 empirically. The cell in the matrix reflects the similarity among the aligned multipath responses $\widetilde{R}(i, j)$ and will be set with the confidence level according to the vertical distance between the multipath responses. The initial value of each cell is set to zero. The following steps can construct the occupancy grid map: 1) the cell $M(x, y)$ covered by each multipath response $\widetilde{R}(i, j)_k$ will be added with the current amplitude a (i.e., the yth element amplitude in $\widetilde{R}(i, j)_k$). 2) the surrounding cells $M(x_t, y_t), x - r < x_t < x + r, y - r < y_t < y + r$ of the cell $M(x, y)$ will be added by $a \times (1 - \sqrt{(x_i - x)^2 + (y_j - y)^2})$. Thus, a higher amplitude of the multipath response will lead to a higher confidence level. 3) Any outlier cells, defined as having a confidence level less than 30% of the highest

confidence level in the occupancy grid (as shown in Fig. 8(c)), are removed. Finally, the direct path peak segment (other parts will be set to zero) of base $R(i,j)_1$ and corresponding occupancy map based matrix $M_{i,j}$ will be stored for verification.

5.2 Similarity Based Verification

The multipath response in different landmark positions and with different sensing channels can be seen as an independent event. Thus, we adopt a vote-based method to authenticate the user. In the voting procedure, each position with a different channel can be treated as an independent elector.

For the single-user model, we assume that the database (shown in Fig. 2) has stored a candidate's occupancy map based matrix $M_{i,j,U}, U = [u_1]$, which corresponds to the finger touch in the ith position with the jth sensing channel. The elector $E_{i,j}$, which corresponds to the ith landmark position with the jth sensing channel, will vote based on the similarity between the multipath response of the input sample and the stored feature. To do this, the input multipath response, $R(i,j)_t$, must first be interpolated and aligned with the corresponding stored direct path peak segment for similarity calculation with the stored matrix M_{i,j,u_1}. The multipath response after interpolated and aligned is denoted as $\widetilde{R}(i,j)_t$. Then, the similarity φ can be calculated by the following equation:

$$\varphi = \frac{S}{P_{zero}}, \tag{8}$$

where S is the average value of confidence level of all cells in M_{i,j,u_1} covered by the multipath response $\widetilde{R}(i,j)_t$ and P_{zero} is the percent of the cells whose confidence level is equal to zero. To avoid the exception of dividing by 0, P_{zero} will be reset to $1/(1.2 \times length(\widetilde{R}(i,j)_t))$, where $length(x)$ is the length of the vector x. The vote will be discarded if the similarity φ is less than a threshold value (this will be discussed later). Finally, the user can be identified successfully if at least half the electorate votes in favor of authentication.

The threshold value mentioned above is determined in the registration stage. Here, collected samples will be divided into two parts, which are used to generate the matrix-based features and determine the threshold value. Each sample in the second part will generate a similarity value for each stored matrix M_{i,j,u_1}. Thus, the threshold value for matrix M_{i,j,u_1} will be set as 1.2 times of the average value of the generated similarity values.

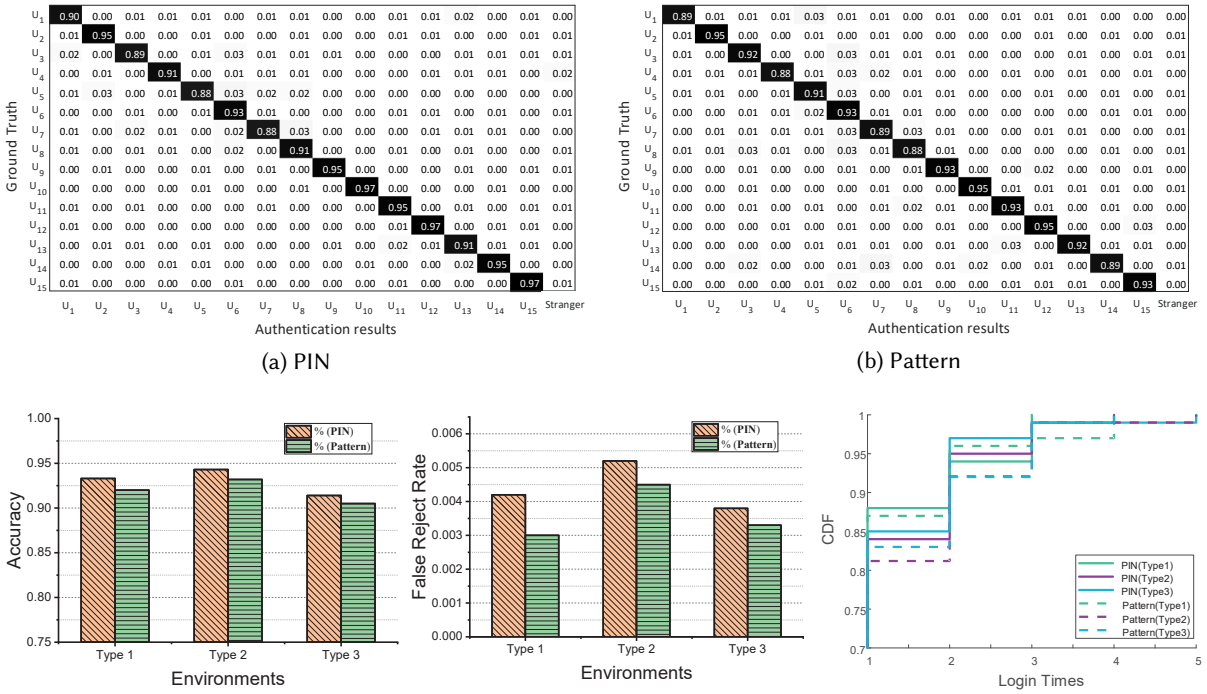
For the multi-user model, we assume that the database has n candidates' occupancy map based matrix $M_{i,j,U}, U = [u_1, \cdots, u_n]$. In the voting procedure, each elector $E_{i,j}$ will traverse all stored candidates $U = [u_1, \cdots, u_n]$ and vote for the candidate with the highest similarity value. At last, the user to be authenticated will be the one who receives the most votes. Besides, the user will be denied access to the device once more than half the elector discards their vote.

6 EVALUATION

This section will evaluate the performance of TouchPrint and study the impact of training dataset size, number of landmark positions, and applied multipath enrichment method. Additionally, it will study the effectiveness of defending against the replay attacks and the observe-imitate attacks.

6.1 Experiment Setup

We have recruited 30 users (i.e., 18 males and 12 females whose ages range from 24 to 58) to participate in our experiment and provided a gift (electronic accessories) to each user as an incentive. No volunteer drops out the sample collection since it does not take them much time in each day. All users are divided into two groups (15 registered users and 15 attackers) due to different instructions. All registered users are told what the biometrics used in TouchPrint before the sample collection is. Therefore, they will input the pattern and PIN with a preferred hand posture and keep the same hand posture during the sample collection. Besides, all attackers are also told



(a) PIN (b) Pattern

(c) Authentication accuracy (d) False reject rate (e) Login times until successful authentication

Fig. 9. Overall Performance of TouchPrint.

the biometrics used in TouchPrint, but the attackers who perform the replay attack do not know any information (i.e., hand sizes and posture types) about the biometrics of the victim.

The evaluation is performed in different rooms, with three types of environment: Type 1 (a quiet, open setting with few furniture), Type 2 (a crowded, setting with more furniture and slightly more ambient noise from the human walking, opening or closing the door, or talking), and Type 3 (quite noisy due to a playing TV). We ask each registered user to input a specified pattern (as in Fig. 3(a)) and PIN code (as in Fig. 1(b)) since our approach is mainly used as a second-factor authentication for the Passcode based method. Thus, the registered user inputs the above Patten and PIN ten times on the Google Nexus 6P (supports stereo playback and the multipath effect enrichment method, mentioned in Section 4.4) in each environment. Besides, the collection lasts for ten days to evaluate system performance over time.

The used metrics for performance evaluation contain (1) authentication accuracy: the percentage of correct identification (U_i is correctly authenticated as U_i), (2) false accept rate: the percentage of the unregistered user who is identified as a registered user. (3) false reject rate: the percentage of the registered user who is identified as a stranger.

6.2 Overall Performance

In the evaluation of the authentication accuracy for registered users, only the samples collected on the first day are used for training. Other samples are used for testing. The confusion matrixes of results for PIN and Pattern scenarios are shown in Fig. 9(a) and Fig. 9(b), respectively. We observe that the authentication accuracy

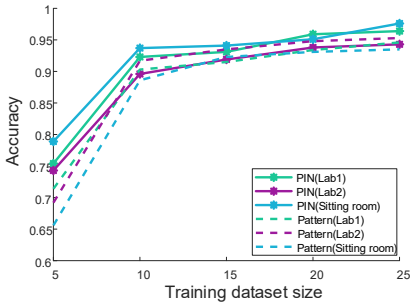


Fig. 10. Impact of training dataset size.

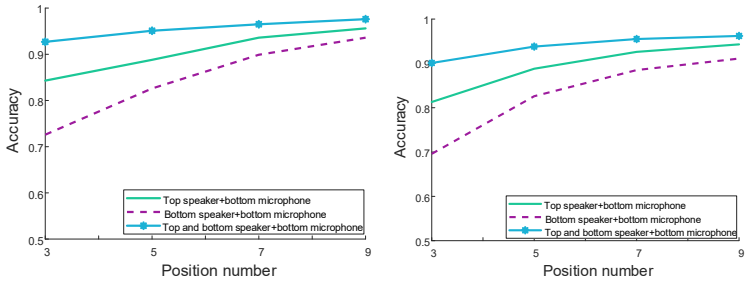


Fig. 11. Impact of landmark position number and multipath effect enrichment.

for registered users in each case achieves over 88% accuracy. The average authentication accuracies of PIN and Pattern scenarios are 92.8% and 91.7%, respectively.

Fig. 9(c) shows the authentication accuracy of TouchPrint in three types of environment. We can find that the accuracy of PIN or Pattern is similar to different kinds of environments, which shows that our approach is robust to various types of physical layouts and environmental noise, including moving people. The authentication accuracy of the PIN scenario is better than that of the Pattern scenario in each case. We believe this is because the PIN scenario has more landmark positions (four vs. three), and a larger number of landmark positions may be beneficial for overall accuracy, evidenced by the study in Section 6.4.

User experience in the authentication stage will be impacted once falsely authenticated as a stranger. Therefore, we first evaluate the user experience using the false reject rate. The results are shown in Fig. 9(d). Results show that the average false reject for registered users in each type of environment is below 0.6%. Besides, we also observe that the user can be authenticated to other users from the above confusion matrixes. Therefore, we further evaluate the user experience with the number of passcode input times until the successful authentication for each registered user. Fig. 9(e) shows the CDF of input times until successful authentication. We can see that more than 90% of registered users can be accurately authenticated through inputting the passcode less than three times. This result demonstrates that the impact on the user experience is limited.

6.3 Impact of Training Dataset Size

Only limited samples can be collected for training in the application scenario since collecting too many samples will hurt the user experience during the registration stage. To study the impact of training dataset size, we use different sample scales at each training day. Here, data from the first three days are used as training data, and the samples of the other seven days are used as test data. Fig. 10 shows the accuracy with different training dataset sizes in the three types of environment. We find that the gain for accuracy is limited when the training dataset size exceeds the above ten samples. Specifically, when five samples are used for training, the accuracy of all cases is under 80%. An average accuracy level of about 92% can be achieved when ten samples are used for training. Furthermore, accuracy can reach approximately 96% when 25 samples are used for training. Considering the tradeoff between the user experience and the authentication accuracy, we suggest that the user at least collect ten samples for training in the registration stage.

6.4 Impact of Landmark Positions Number

Our approach combines the identification results in the multiple landmark positions for user authentication. Therefore, more landmark positions will lead to better accuracy. To study the impact of the landmark position number in the pattern or PIN, five registered users are invited to input a specified pattern and PIN, which cover

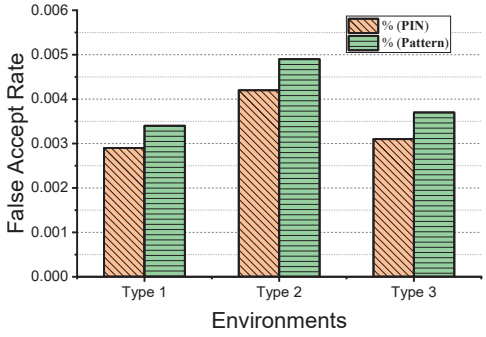


Fig. 12. False accept rates against replay attacks.

all the nine landmark positions in Fig. 3. Each user inputs both the pattern and PIN for 40 times in the sample collection procedure. In this experiment, the first ten samples will be used for training, and the remaining 30 samples are used for testing.

Fig. 11 shows the authentication accuracy for the PIN and pattern, respectively. It demonstrates that the accuracy can be significantly improved when covering more landmark positions, especially for the sensing channel using the bottom speaker and bottom microphone. Specifically, when three landmark positions are used, the authentication accuracy for the pattern case is about 70%. Besides, it can be improved to 88% when nine landmarks are used. We believe the underlying reason is that information about the user hand's geometry can be sensed more comprehensively with more landmark positions.

6.5 Impact of Mutlipath Effect Enrichment

Fig. 11 also shows the impact of applying the multipath effect enrichment method. First, it illustrates clearly that the accuracy is significantly improved when synchronous sensing from the top and bottom speakers and microphone is used so that more sides of the hand can be sensed. Second, the accuracy when using the top speaker-bottom microphone pair is better than the accuracy when using the bottom microphone-bottom speaker pair. One explanation is that the former combination has stronger multipath sensing ability (e.g., the channel can be blocked by the hand).

6.6 Resilience to Replay Attack

The *replay attack* is highly possible to be performed by an attacker once the passcode is inferred. It makes the traditional finger-touching authentication method less reliable. To evaluate the reliability of TouchPrint against the replay attack, we invited 15 volunteers (beyond the above 15 registered users) to perform the replay attack in all three types of environments. Specifically, the attackers attempt to fool the authentication system, which stores the user biometric feature generated with the first-day samples collected from 15 registered users in each environment. If the passcode is inferred via the non-visual sensors (e.g., motion sensor [24, 25], acoustic [16] or Wi-Fi signal [17]), the attacker does not have any information on the registered user's hand size and input posture. Thus, the attacker will input the pattern (as in Fig. 3(a)) and PIN code (as in Fig. 1(b)) with a random hand posture for ten times in each type of environment. Fig. 12 shows the average false accept rate. We find that even if the adversary knows the Pattern or PIN, our approach can still decrease the success rate of the replay attack to below 0.5%. It also means that TouchPrint is reliable for various types of non-visual based side-channel attacks (e.g., motion sensor, acoustic or Wi-Fi signal).

Table 1. False Acceptance Rates Against Observe and Imitate Attacks.

Attack	Type	1	2	3	4	5	6	7	Avg
Observe from the video	Pattern	0.0	0.0	0.1	0.0	0.1	0.1	0.0	0.042
	PIN	0.0	0.0	0.0	0.1	0.0	0.1	0.0	0.028
Observe near the victim	Pattern	0.0	0.1	0.0	0.1	0.1	0.1	0.0	0.057
	PIN	0.0	0.0	0.1	0.1	0.1	0.0	0.0	0.042

6.7 Resilience to Observe and Imitate Attack

In this study, two types of *observe and imitate attacks* are performed to fool the authentication system, which stores the user biometric feature generated with ten samples collected from victims. Since the attacker with the camera [30] can infer the passcode, the adversary will first imitate the victim's hand posture through watching the video, which records the proceedings of the victim's inputting the pattern and PIN with a fixed point of view. In this scenario, the adversary is allowed to watch the video multiple times. Moreover, the user hand posture during the passcode input procedure can also be snooped by the near adversaries with direct visual observation. Thus, the second type of observe and imitate attack allows the adversary to sit near the victim so that he/she can observe the victim's hand posture directly. To make this type of attack more challenging, the adversary is allowed to observe the hand posture of the victim's inputting from multiple angles.

To study the effectiveness of TouchPrint defense against the above two types of observe and imitate attacks, we randomly select seven volunteers (also participated the study in Section 6.6) as the adversaries and select seven registered users as the victims. For the above two types of observe and imitate attacks, the adversaries will input the pattern and PIN ten times with the observed hand posture seriously.

The false accept rate of both attack scenarios are shown in Table 1. The result demonstrates that the success rate is very low when the adversary performs the observe and imitate attack via a camera. Further, even if the adversary is close to the victim and can directly observe him/her inputting, our approach can still effectively thwart the imitate attack.

7 RELATED WORK

Over the past decade, various approaches have provided insights for passcode inferring, user authentication, and acoustic sensing. In this section, these systems and approaches are reviewed.

Passcode inferring via side-channel sensing: Recent years, many works demonstrate that various types of side-channel based attacks can infer users' personal information (e.g., PIN, patterns). For example, the motion sensor embedded on the wearable devices (e.g., smartwatch) can be used to infer the sensitive information when the user inputs the PIN on the physical keyboard [24, 26] or smartphone touchscreen [13, 25]. Patternlistener [35] cracks the pattern lock leveraging the speaker and microphone embedded in the victim's smartphone. These methods are known as intrusion attacks and can be defended against through sensor authority management. By contrast, other non-invasive attacks are more challenging to prevent. For example, Wi-Fi signals [17] surrounding a user have been proven as one way to infer the user's online payment passcode; this method is very difficult for users to detect. Further, sensitive information also can be inferred by surrounding smartphones. Specifically, the keystroke sound can be used to locate the click location using the microphones in the smartphone [12, 36]. Additionally, the acoustic phase and Doppler shift have been used to improve the PIN inferring accuracy [16]. Multiple side-channel information can also be combined to achieve more sophisticated attacks [10]. These threats motivate us to design a second-factor authentication for the PIN/pattern-based authentication method to further prevent the private data in the device from leakage.

User authentication with smartphone and wearable device: Smartphones and wearable devices have been widely used for user authentication since they contain various types of sensors that can sense the user's biometric information. For example, FaceID is a very popular facial recognition-based authentication method and has been applied in many commercial smartphones. PPGPass [4] uses signals from Photoplethysmography (PPG) sensors in wrist-worn wearables to perform two-factor authentication. However, these methods need to add additional sensors in the device, which increases the hardware cost. Another system, EchoPrint [34], uses active acoustic sensing to achieve the liveness verification for face identification (like FaceID) but does not adopt any additional hardware. Besides, the LipPass [15] system provides the liveness verification for voiceprint authentication by extracting the Doppler feature of a lip-reading motion. When interacting with IoT devices, P2Auth [11] has designed a uniform authentication method by comparing the interaction motion sensed by devices with those captured by the smartwatch. It does not require the IoT device to be equipped with any special sensors. The prior work [18] focuses on the gait-based authentication on smartphones considering the impact of the phone holding positions. It uses the accelerometer embedded in smartphones to track the user's body movement, and thus can be used for continuous authentication. In contrast, TouchPrint works in an instant scenario such as online payment and screen unlocking. Other works provide second-factor authentication for the interaction on the smartphone touchscreen by identifying the user's behavior pattern (i.e., pressure and velocity infromation [3, 14, 33]) and the geometric information between multiple fingers [20]. However, these two methods incur the limitations on their accuracy (which may be reduced due to the behavioral variability) and restricted interaction scenarios (i.e., many cannot be applied to the single-finger interaction scenario). By contrast, our approach is more robust than the behavior pattern-based method since it extracts more stable features (i.e., spatial information on the hand posture shape). It can be applied to a broader range of on-screen interaction scenarios (i.e., does not restrict the finger number touched on the screen). TouchPass [29] uses the active vibration signals to capture the physical characters (e.g., density, conductance, etc.) of touching fingers for user authentication. In contrast, TouchPrint adopts the acoustic sensing to capture another type of finger-touching biometrics (i.e. user's hand posture shape traits) for user authentication.

Acoustic sensing on smartphone: The accurate acoustic ranging on smartphones has been adopted for many mobile applications such as: D2D ranging [7], hand motion tracking [6], gesture recognition [28], and encounter detection [31, 32]. For these applications, the multipath effect is always seen as interference and needs to be removed. However, the multipath effect, when captured with active acoustic sensing, can identify the environment's spatial features and has also been adopted in various mobile applications. For example, EchoPrint [34] separates the echo signal, reflected by the user's face, for user authentication. EchoTag [23] locates the device by extract environmental-dependent frequency features. Another application, CondioSense [9], extracts the distribution of the echo signal for device context position identification. UbiK [27] extracts the multipath feature for keystroke recognition. To extract the multipath feature, these works generally separate the indirect path signal for the macro scenario (interval distance $> 50cm$) or directly extract the frequency feature of the entire recorded signal for the micro scenario (interval distance $< 30cm$). In contrast, our approach will carefully separate the fine-grained indirect path signal for a micro scenario, which is more challenging. VSkin [21] also adopts the Zadoff-Chu sequence-based acoustic sensing to design a finger-touching based interaction method. It aims to detect the finger tapping event, estimate the tapping position (possible outside screen area), and track the moving finger only via acoustic sensing. In contrast, TouchPrint detects the finger tapping event on the screen and obtains the tapping position directly from the touch screen rather than the acoustic sensing. Moreover, TouchPrint focuses on extracting the acoustic multipath response corresponding with the hand geometry when the finger stays static in the reference positions for authentication via the acoustic sensing. TouchPrint also utilizes multiple speakers to enrich the generated multipath feature, while VSkin only uses one speaker.

8 DISCUSSIONS

We now provide some further discussions on the limitation and possible extension of TouchPrint.

Continuous authentication manner: TouchPrint is an instant authentication method. It depends on the acoustic multipath feature when the finger is static on multiple specified positions on the screen. Once applied in the continuous scenario, it will become more challenging to extract the hand biometrics feature independent of the finger-touching positions. We leave such an extension as possible future work.

Multiple fingers scenario: TouchPrint focuses on identifying individual diversity on both hand biometrics (e.g., finger length and palm-size) and preferred posture types for user authentication. Meanwhile, TouchPrint depicts the 3D shape of the above hand posture by analyzing the acoustic multipath reflection signals. Comparing with the hand biometrics (e.g., finger length and palm-size), the users can select the posture type when inputting the passcode (i.e., PIN, Pattern) as their preferred. Therefore, TouchPrint still works when the user uses multiple fingers, even from different hands, to input the Pattern or PIN code as long as the same posture can be presented in the verification procedure.

Authentication accuracy: We study the impact of training data size on the authentication accuracy. The experimental results show that the authentication accuracy can be improved to 96% when 25 samples are used for training. However, larger training data size leads to more efforts from the user during the registration stage. Since we only leverage TouchPrint as the second-factor user authentication for the PIN code or pattern-based authentication, a 90% accuracy may be sufficient. We suggest that the system collects at least ten samples from the user during the registration stage. In case of higher accuracy is desired, more training samples and passcodes with more landmark positions are needed.

9 CONCLUSION

To prevent the exposure of private data stored in the mobile device to an adversary who learns the passcode, traditional methods add a second-factor authentication that relies on the behavior biometrics or geometry constraint among the on-screen fingers. However, these methods generally have poor accuracy due to behavioral variability in users or may only be applied in the multi-finger touch scenario. Therefore, we propose a robust, reliable, and secure authentication method, called TouchPrint, which is robust to the touch behavioral variabilities and does not restrict the on-screen interaction manner. The key insight of TouchPrint is to identify the hand posture shape traits as the user inputs PIN or pattern by depicting the fine-grained multipath effect when the finger is static on the screen. We overcome several challenges, including accurate acoustic signal segmentation (i.e., obtaining the acoustic fragment when the finger is static in reference positions), fine-grained hand multipath separation, and reliable authentication with limited training samples. The evaluation is performed in the practical environments, and the results demonstrate that TouchPrint can effectively repel many attacks (e.g., replay attack and observe-imitate attack). Finally, our approach can achieve an authentication accuracy of about 92% with only ten training samples. We leave further improvements in the user experience and the authentication accuracy of the proposed approach as our future work.

ACKNOWLEDGMENTS

This work of Huijie Chen was partially done when he visited the Department of Computer Science, University of North Carolina at Charlotte, with a scholarship from the China Scholarship Council. The work of Fan Li is partially supported by the Beijing Natural Science Foundation under Grant No. 4192051 and National Natural Science Foundation of China (NSFC) under Grant No. 61772077. The work of Song Yang is partially supported by NSFC under Grant No. 61802018.

REFERENCES

[1] Mohammed H AlSharif, Mohamed Saad, Mohamed Siala, Tarig Ballal, Hatem Boujemaa, and Tareq Y Al-Naffouri. 2017. Zadoff-chu coded ultrasonic signal for accurate range estimation. In *2017 25th European Signal Processing Conference (EUSIPCO)*. IEEE, 1250–1254.

[2] S Abhishek Anand, Prakash Shrestha, and Nitesh Saxena. 2015. Bad sounds good sounds: Attacking and defending tap-based rhythmic passwords using acoustic signals. In *International Conference on Cryptology and Network Security*. Springer, 95–110.

[3] Cheng Bo, Lan Zhang, Xiang-Yang Li, Qiuyuan Huang, and Yu Wang. 2013. Silentsense: silent user identification via touch and movement behavioral biometrics. In *Proceedings of the 19th annual international conference on Mobile computing and networking (MOBICOM)*. ACM, 187–190.

[4] Yetong Cao, Qian Zhang, Fan Li, Song Yang, and Yu Wang. 2020. PPGPass: Nonintrusive and Secure Mobile Two-Factor Authentication via Wearables. In *Proceedings of IEEE 39th Conference on Computer Communications (INFOCOM)*.

[5] Géry Casiez, Thomas Pietrzak, Damien Marchal, Sébastien Poulmane, Matthieu Falce, and Nicolas Roussel. 2017. Characterizing latency in touch and button-equipped interactive systems. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST)*. ACM, 29–39.

[6] Huijie Chen, Fan Li, and Yu Wang. 2017. EchoTrack: Acoustic Device-free Hand Tracking on Smart Phones. In *2017 IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 1422–1430.

[7] Huijie Chen, Fan Li, and Yu Wang. 2018. SoundMark: Accurate indoor localization via peer-assisted dead reckoning. *IEEE Internet of Things Journal* 5, 6 (2018), 4803–4815.

[8] Haritabh Gupta, Shamik Sural, Vijayalakshmi Atluri, and Jaideep Vaidya. 2018. A side-channel attack on smartphones: Deciphering key taps using built-in microphones. *Journal of Computer Security* 26, 2 (2018), 255–281.

[9] Fan Li, Huijie Chen, Xiaoyu Song, Qian Zhang, Youqi Li, and Yu Wang. 2017. CondioSense: high-quality context-aware service for audio sensing system via active sonar. *Personal and Ubiquitous Computing* 21, 1 (2017), 17–29.

[10] Fan Li, Xiuxiu Wang, Huijie Chen, Kashif Sharif, and Yu Wang. 2017. ClickLeak: Keystroke Leaks Through Multimodal Sensors in Cyber-Physical Social Networks. *IEEE Access* 5 (2017), 27311?27321.

[11] Xiaopeng Li, Fengyao Yan, Fei Zuo, Qiang Zeng, and Lannan Luo. 2019. Touch Well Before Use: Intuitive and Secure Authentication for IoT Devices. In *The 25th Annual International Conference on Mobile Computing and Networking (MOBICOM)*. ACM, 1–17.

[12] Jian Liu, Yan Wang, Gorkem Kar, Yingying Chen, Jie Yang, and Marco Gruteser. 2015. Snooping keystrokes with mm-level audio ranging on a single phone. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MOBICOM)*. ACM, 142–154.

[13] Yang Liu and Zhenjiang Li. 2018. Aleak: Privacy leakage through context-free wearable side-channel. In *IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 1232–1240.

[14] Li Lu and Yongshuai Liu. 2015. Safeguard: User reauthentication on smartphones via behavioral biometrics. *IEEE Transactions on Computational Social Systems* 2, 3 (2015), 53–64.

[15] Li Lu, Jiadi Yu, Yingying Chen, Hongbo Liu, Yanmin Zhu, Yunfei Liu, and Minglu Li. 2018. Lippass: Lip reading-based user authentication on smartphones leveraging acoustic signals. In *IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 1466–1474.

[16] Li Lu, Jiadi Yu, Yingying Chen, Yanmin Zhu, Xiangyu Xu, Guangtao Xue, and Minglu Li. 2019. KeyLiSterber: Inferring Keystrokes on QWERTY Keyboard of Touch Screen through Acoustic Signals. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 775–783.

[17] Yan Meng, Jinlei Li, Haojin Zhu, Xiaohui Liang, Yao Liu, and Na Ruan. 2019. Revealing Your Mobile Password via WiFi Signals: Attacks and Countermeasures. *IEEE Transactions on Mobile Computing* 19, 2 (2019), 432–449.

[18] Abena Primo, Vir V Phoha, Rajesh Kumar, and Abdul Serwadda. 2014. Context-aware active authentication using smartphone accelerometer measurements. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. IEEE, 98–105.

[19] Raul Sanchez-Reillo, Carmen Sanchez-Avila, and Ana Gonzalez-Marcos. 2000. Biometric identification through hand geometry measurements. *IEEE Transactions on pattern analysis and machine intelligence* 22, 10 (2000), 1168–1171.

[20] Yunpeng Song, Zhongmin Cai, and Zhi-Li Zhang. 2017. Multi-touch authentication using hand geometry and behavioral information. In *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 357–372.

[21] Ke Sun, Ting Zhao, Wei Wang, and Lei Xie. 2018. Vskin: Sensing touch gestures on surfaces of mobile devices using acoustic signals. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking (MOBICOM)*. ACM, 591–605.

[22] TSYS. 2018. U.S. Consumer Payment Study. (2018). https://www.tsys.com/Assets/TSYS/downloads/rs_2018-us-consumer-payment-study.pdf/,Accessed November 10, 2019.

[23] Yu-Chih Tung and Kang G Shin. 2015. EchoTag: accurate infrastructure-free indoor location tagging with smartphones. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MOBICOM)*. ACM, 525–536.

[24] Chen Wang, Xiaonan Guo, Yan Wang, Yingying Chen, and Bo Liu. 2016. Friend or foe?: Your wearable devices reveal your personal pin. In *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*. ACM, 189–200.

[25] Chen Wang, Jian Liu, Xiaonan Guo, Yan Wang, and Yingying Chen. 2019. WristSpy: Snooping Passcodes in Mobile Payment Using Wrist-worn Wearables. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2071–2079.

[26] He Wang, Ted Tsung-Te Lai, and Romit Roy Choudhury. 2015. Mole: Motion leaks through smartwatch sensors. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking (MOBICOM)*. ACM, 155–166.

[27] Junjue Wang, Kaichen Zhao, Xinyu Zhang, and Chunyi Peng. 2014. Ubiquitous keyboard for small mobile devices: harnessing multipath fading for fine-grained keystroke localization. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services (MOBISYS)*. ACM, 14–27.

[28] Yanwen Wang, Jiaxing Shen, and Yuanqing Zheng. 2020. Push the Limit of Acoustic Gesture Recognition. In *IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 1–10.

[29] Xiangyu Xu, Jiadi Yu, Yingying Chen, Qin Hua, Yanmin Zhu, Yi-Chao Chen, and Minglu Li. 2020. TouchPass: towards behavior-irrelevant on-touch user authentication on smartphones leveraging vibrations. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking (MOBICOM)*. ACM, 1–13.

[30] Guixin Ye, Zhanyong Tang, Dingyi Fang, Xiaojiang Chen, Kwang In Kim, Ben Taylor, and Zheng Wang. 2017. Cracking Android pattern lock in five attempts. In *Proceedings 2017 Network and Distributed System Security Symposium (NDSS)*.

[31] Huanle Zhang, Wan Du, Pengfei Zhou, Mo Li, and Prasant Mohapatra. 2016. DopEnc: acoustic-based encounter profiling using smartphones. In *Proceedings of the 22nd Annual International Conference on Mobile Computing and Networking (MobiCom)*. ACM, 294–307.

[32] Huanle Zhang, Wan Du, Pengfei Zhou, Mo Li, and Prasant Mohapatra. 2017. An acoustic-based encounter profiling system. *IEEE Transactions on Mobile Computing* 17, 8 (2017), 1750–1763.

[33] Nan Zheng, Kun Bai, Hai Huang, and Haining Wang. 2014. You are how you touch: User verification on smartphones via tapping behaviors. In *2014 IEEE 22nd International Conference on Network Protocols (ICNP)*. IEEE, 221–232.

[34] Bing Zhou, Jay Lohokare, Ruipeng Gao, and Fan Ye. 2018. EchoPrint: Two-factor Authentication using Acoustics and Vision on Smartphones. In *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking (MOBICOM)*. ACM, 321–336.

[35] Man Zhou, Qian Wang, Jingxiao Yang, Qi Li, Feng Xiao, Zhibo Wang, and Xiaofen Chen. 2018. Patternlistener: Cracking android pattern lock using acoustic signals. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, 1775–1787.

[36] Tong Zhu, Qiang Ma, Shanfeng Zhang, and Yunhao Liu. 2014. Context-free attacks using keyboard acoustic emanations. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS)*. ACM, 453–464.