# TouchPass: Towards Behavior-irrelevant on-touch User Authentication on Smartphones Leveraging Vibrations

Xiangyu Xu
Shanghai Jiao Tong University
chillex@sjtu.edu.cn

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

Yingying chen
Rutgers University
yingche@scarletmail.rutgers.edu

Qin Hua
Shanghai Jiao Tong University
huaqin@sjtu.edu.cn

Yanmin Zhu
Shanghai Jiao Tong University
yzhu@cs.sjtu.edu.cn

Yi-Chao Chen
Shanghai Jiao Tong University
yichao@cs.sjtu.edu.cn

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

## ABSTRACT

With increasing private and sensitive data stored in mobile devices, secure and effective mobile-based user authentication schemes are desired. As the most natural way to contact with mobile devices, finger touches have shown potentials for user authentication. Most existing approaches utilize finger touches as behavioral biometrics for identifying individuals, which are vulnerable to spoofer attacks. To resist attacks for on-touch user authentication on mobile devices, this paper exploits physical characters of touching fingers by investigating active vibration signal transmission through fingers, and we find that physical characters of touching fingers present unique patterns on active vibration signals for different individuals. Based on the observation, we propose a behavior-irrelevant on-touch user authentication system, *TouchPass*, which leverages active vibration signals on smartphones to extract only physical characters of touching fingers for user identification. *TouchPass* first extracts features that mix physical characters of touching fingers and behavior biometrics of touching behaviors from vibration signals generated and received by smartphones. Then, we design a Siamese network-based architecture with a specific training sample selection strategy to reconstruct the extracted signal features to behavior-irrelevant features and further build a behavior-irrelevant on-touch user authentication scheme leveraging knowledge distillation. Our extensive experiments validate that *TouchPass* can accurately authenticate users and defend various attacks.

## CCS CONCEPTS

• **Security and privacy** → **Authentication**;

## KEYWORDS

User authentication; Behavior-irrelevant; Vibration signals;

[*] Jiadi Yu is the corresponding author

*MobiCom '20, September 21–25, 2020, London, United Kingdom*
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7085-1/20/09…$15.00
https://doi.org/10.1145/3372224.3380901

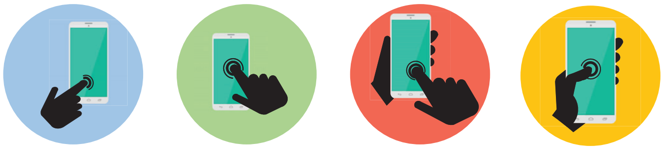


**Figure 1: Illustration of various finger touching modes on smartphones.**

**ACM Reference Format:**
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 *The 26th Annual International Conference on Mobile Computing and Networking (MobiCom '20), September 21–25, 2020, London, United Kingdom.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3372224.3380901

## 1 INTRODUCTION

With the widespread use of mobile devices, an increasing number of people use them to carry out commercial transactions, store sensitive personal information, etc. According to a report[16], nearly 41% of all data breach events from 2005 to 2015 were caused by lost mobile devices, which demands effective authentication schemes on mobile devices. Existing mobile-based user authentication schemes are either password-based[29][28] or explicit biometric-based (e.g., Fingerprint[17], Face[35], Voiceprint[8]), which are vulnerable to smudge attacks[39] and replay attack[44], respectively.

With the increasing popularity of touch screen on mobile devices, recent years have witnessed the emerging of behavioral biometric-based on-touch user authentication schemes[1][7][9][11], which could validate the identity of users based on the touching behavior on mobile devices. Although behavioral biometric-based on-touch authentication schemes provide a natural way for mobile user identification, these approaches are proved quite vulnerable[18] under mimic attacks. Moreover, current behavioral biometric-based schemes usually degrade user experiences, as the login touch needs to have similar behavioral biometrics as predefined touches. While in real scenario, finger touching of the same user on mobile devices could have various modes(e.g., different positions, different forces, supports for smartphones, etc.), as shown in Figure.1. To improve security and user experience, it is desired to build a behavior-irrelevant on-touch user authentication scheme on mobile devices.

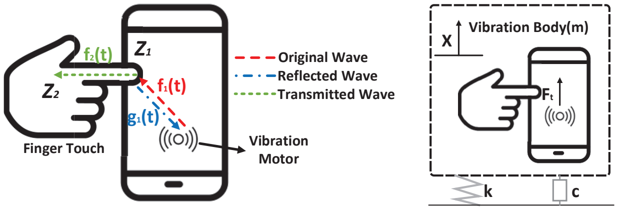
Fingers have unique physical characters(e.g., density, conductance, etc) that vary among individuals[34]. To realize behavior-irrelevant for on-touch user authentication, we exploit these physical properties of touching fingers. Since vibration signals have excellent short-range transmission characteristics for reflecting physical characters of different media, we consider vibration signals transmitted through touching fingers could embed physical characters of fingers. Therefore, this work aims to leverage active vibration signals to capture physical characters of fingers for realizing a behavior-irrelevant on-touch user authentication scheme.

To realize the behavior-irrelevant on-touch user authentication scheme on smartphones leveraging vibration signals, we face several challenges in practice. First, the influence from physical characters of touching fingers to vibration signals is unclear, so it is necessary to extract representative features from active vibration signals generated and received by smartphones. Second, behavioral biometrics of finger touching could also influence the vibration signals, thus the authentication scheme should eliminate the influence of behavioral biometrics for realizing behavior-irrelevant user authentication. Third, since smartphones are computational-restricted, the authentication scheme should be light-weighted for off-the-shelf smartphones without any additional hardware.

In this paper, we first investigate the feasibility of utilizing active vibration signals to capture the physical characters of touching fingers for user authentication, and find that the transmission of active vibration signals through touching fingers presents unique patterns for different individuals. Motivated by the observation, we propose a behavior-irrelevant on-touch user authentication system on smartphones, *TouchPass*, which leverages active vibration signals to extract unique physical characters of touching fingers for identifying different individuals. Specifically, when detecting a finger touching on a smartphone, *TouchPass* actively generates vibration signals through the built-in motor on the smartphone, and collects the vibration signals with IMU sensors on the same smartphone. Then, from the received vibration signals, *TouchPass* extracts features that mix physical characters of touching finger and behavioral biometrics of touching behavior. After that, we propose a Siamese network[5]-based architecture with a specific training sample selection strategy to reconstruct the extracted features to behavior-irrelevant features, and further design a light-weighted behavior-irrelevant on-touch user authentication scheme based on knowledge distillation[13]. Our extensive experiments demonstrate that *TouchPass* can accurately authenticate users under different touching modes and defend both mimic and replay attacks.

We highlight our main contributions as follows:

- We explore the transmission process of active vibration signals generated by the built-in motor on smartphones, and extract unique features that relate to the physical characters of touching fingers from vibration signals received by the IMU sensor on smartphones.
- We reconstruct features extracted from active vibration signals to behavior-irrelevant features based on Siamese network for building a behavior-irrelevant authentication scheme.
- We build a light-weighted on-touch user authentication scheme based on knowledge distillation for user authentication and spoofer detection on off-the-shelf smartphones.



(a) Transient-state. (b) Steady-state.

**Figure 2: Smartphone vibration model.**

- We conduct extensive experiments in real environments settings and the results show that our system is feasible to authenticate users and resist both mimic and replay attacks.

## 2 PRELIMINARY

Touching is one of the most natural way to interact with mobile devices, which makes it a perfect behavior for user authentication. In this section, we first introduce the attack model of touch-based user authentication, and then exploit the feasibility of an on-touch user authentication scheme leveraging active vibrations.

### 2.1 Attack Model

Touch-based user authentication on mobile devices is a process to identify a user by extracting the biometric (e.g., fingerprint) or behavioral (e.g., touching area, duration, force, etc.) characteristics. However, the above touch-based user authentication is threatened by two kinds of attacks, i.e., replay attack and mimic attack.

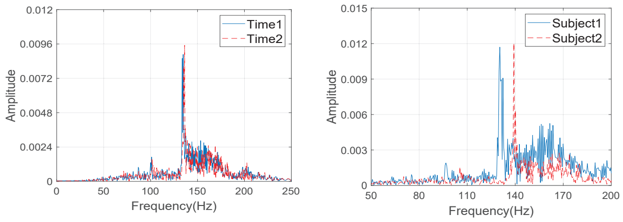
In a replay attack, a spoofer records the information that a legitimate user used to pass the authentication. For instance, a spoofer can take photos of the fingerprint of a registered user and fake it to perform a replay attack to the authentication system, which threats most of existing fingerprint-based authentication schemes.

In a mimic attack, a spoofer first observes the way that a legitimate user performs to the pass the authentication, and then practices to mimic the behavior for conducting the attack. This is a kind of attack that focuses on the behavioral biometrics-enabled touch-based user authentication. A recent work[18] have shown that mimic attack can break through the behavioral biometrics-enabled authentication in system level.

### 2.2 Propagation Model of Active Vibration Signals on Mobile Devices

Fingers embed unique physical characters(e.g., density, conductance, etc.) that vary among different individuals[34] and thus can be leveraged for user authentication. Comparing to fingerprint and behavioral biometrics of finger touching, these physical characters are neither easily duplicated, nor vulnerable to mimic attack. To capture these characters, a stimulating signal is desired to propagate through the touching finger and then be received for analysis. For finger touching on mobile devices, active vibration signal could be a suitable stimulator, since it has been long severed as the natural touching feedback on mobile devices.

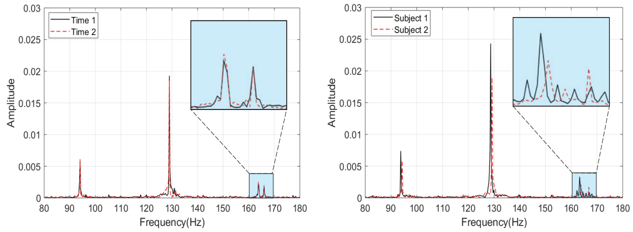
In active vibration, a mobile device, e.g., a smartphone, actively generates a vibration wave with the built-in motor, and the wave travels through the smartphone and other medium that is physically connected to the smartphone, with attenuation along the



(a) One user.

(b) Two users.

Figure 3: Vibration profile of two users in transient-state.



(a) One user.

(b) Two users.

Figure 4: Vibration profile of two users in steady-state.

propagation path and reflections/diffractions on the boundary of each medium. Specifically, the vibration propagation can be divided into two states: *transient-state* at the beginning of the vibration when the vibration is built, and *steady-state* when the vibration reaches an equilibrium and becomes stable. In the following, we discuss the vibration model in these two states, respectively.

**Transient-state:** Since transient-state of vibration signals is highly dynamic, it is intractable to build the full physical model. Therefore, we build a simplified model that focuses on the most significant process in the transient-state, i.e., vibration waveform propagation at the contacting area of different media. During this process, the original waveform hits the contacting area, parts of the waveform is reflected and other parts of the waveform transmits to the next medium, as shown in Figure.2(a). Given the original waveform, the reflected waveform and the transmitted waveform as $f_1(t)$, $g_1(t)$ and $f_2(t)$, and the *characteristic impedance*[12] of two connected media as $Z_1$ and $Z_2$, we have

$$g_1(t) = \frac{Z_1 - Z_2}{Z_1 + Z_2} f_1(t), \tag{1}$$

$$f_2(t) = \frac{2Z_1}{Z_1 + Z_2} f_1(t). \tag{2}$$

The *characteristic impedance* of a medium is to describe the impedance of a medium to wave propagation, which is mainly determined by the material and density of the medium. According to Eq.1 and Eq.2, given transmitted waveform $f_1(t)$, both reflected signal $g_1(t)$ and transmitted signal $f_2(t)$ could be very different according to the *characteristic impedance* of different media.

**Steady-state:** The model of active smartphone vibration in steady-state can be taken as a forced vibration with damping, which has a well-known spring-mass-damper model as shown in Figure.2(b). The model describes a body with mass $m$, damper coefficient $k$, spring coefficient $c$ moves a distance $x$ under a external force $F_t$:

$$m\frac{d^2x}{dt^2} = -k\frac{dx}{dt} - cx + F_t. \tag{3}$$

In the case of steady-state for active smartphone vibration, the body corresponds to the whole vibration system containing several media (including smartphone, finger, etc.). So the physical characters of the touching finger have a significant influence on the parameters $m$, $c$, $k$, and thus can be reflected in the vibration signals.

## 2.3 Feasibility of on-touch User Authentication Leveraging Active Vibration Signals

According to the propagation models of active vibration signals, the vibration propagation is strongly dependent on the physical characters of the medium contacting to the vibration source. Since the physical characters of fingers as a medium of vibration wave are different from person to person[34], we consider to leverage active vibration signals to enable on-touch user authentication on smartphones. Specifically, when a user touches a smartphone with his/her fingers, the smartphone generates active vibrations to sense the physical characters of fingers as a medium and authenticates the user accordingly.

To show the feasibility of on-touch user authentication scheme, we conduct an verification experiment. Specifically, we ask two volunteers to touch a smartphone (Samsung Galaxy S6) with their right index fingers. To rule out the influence of behavioral biometrics(e.g., touching force, location, etc.) as much as possible, we leverage the data collected by the touch screen sensor as real-time feedback to make sure that the behavioral biometrics are almost the same for each touching behavior. As soon as the smartphone detects a finger touching behavior with its touch screen sensor, it generates a vibration signal of $0.5s$ with the built-in vibration motor in the smartphone. During the procedure, the vibration signal is collected by the International Mathematical Union (IMU) in the same smartphone, and then roughly separated into the transient-state and steady-state according to their frequency changes.

We compares the collected vibration signals of the two users during the two states, respectively, and the results are shown in Figure.3 and Figure.4. Figure.3 shows the vibration frequency profile of two users in transient-state. It can be seen that the vibration frequency profile covers a large range of frequencies. Moreover, for the same user, the two touches present similar vibration patterns in frequency domain as shown in Figure.3(a), while for different users, the vibration patterns tend to have different patterns in frequency domain as shown in Figure.3(b). Figure.4 shows the vibration profile of two users in steady-state. It can be observed that the vibration profile has clear structure in frequency domain with a dominating frequency component at the resonance frequency of smartphone motor (i.e., $129Hz$ in this case) and several harmonic frequency components. Furthermore, for the same user, the two touches presents pattern in almost all frequency components as shown in Figure.4(a), while for different users, the vibration patterns tend to have different patterns in these frequency components as shown in Figure.4(b). These encouraging results demonstrate that active vibration signals (both in transient-state and steady-state) during finger touching to smartphones varies from person to person, which can be utilized to identify different individuals.

## 2.4 Towards Behavior-irrelevant

The vibration propagation model in Section.2.2 considers the connection between a smartphone and a touching finger as a *rigid link*

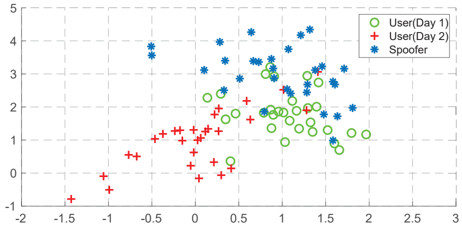


**Figure 5: Example for illustrating mimic attack vulnerability and intra-class variability of behavioral biometrics.**

so that only physical properties of the finger influence the signal. In fact, the smartphone-finger connection is more of an *elastic link*, which means besides physical characters, behavioral biometrics of touching could also influences the propagation of vibration signals.

However, involving behavioral biometrics in user authentication brings two concerns, i.e., vulnerability under mimic attack and high intra-class variability. We conduct an experiment to show these two concerns. Specifically, we ask a volunteer (denoted as user) to touch a smartphone several times in two different days (denoted as day 1 and day 2) and collect the multi-dimensional behavioral biometrics (including touching position, area size, duration, etc.) from touch screen sensor on the smartphone. Then, we further ask another volunteer (denoted as spoofer) to mimic the touching behavior of the user in day 1. After that, we compare the collected multi-dimensional behavioral biometrics in Figure.5, which plots the t-SNE[26] of the behavioral biometrics in 2-dimensional space. It can be seen from the figure that the touching behavior of the user in day 1 and day 2 are not very close in the t-SNE figure, which shows the high intra-class variability. Moreover, comparing to the user in day 2, the mimic touching behaviors of the spoofer are closer to the touching behaviors of the user in day 1, which shows that the spoofer could mimic the behavioral biometrics of touching behavior better than a user himself/herself.

We further investigate the vulnerability to attacks for user authentication systems involving behavioral biometrics. Concretely, in an authentication system leveraging both behavioral biometrics and physical characters, assuming $\alpha\%(\alpha \in (0, 100))$ of the authentication is based on behavioral biometrics, and other $(100 - \alpha)\%$ is based on physical characters. Then, the overall authentication score of a individual can be denoted as

$$s = a \times \alpha\% + b \times (100 - \alpha)\%, a, b \in [0, 1], \tag{4}$$

where $a$ and $b$ represent the score of physical characters and behavioral biometrics authentication for the individual, respectively. Then, if the authentication score $s$ is greater than a threshold $s_0$, then the authentication successes. Otherwise, the authentication fails. Consider the intra-class variability of a user, we further assume the value of $a$ for a registered user varies from $x$ to 1 $(0 < x \leq 1)$, and $b$ is always 1 as the physical characters are usually stable. Then, to ensure that the registered user can enter the system despite of the intra-class variability, the threshold $s_0$ needs to satisfy

$$s_0 < x \times \alpha\% + 1 \times (100 - \alpha)\% = 1 - (1 - x)\alpha\%. \tag{5}$$

For a spoofer, we assume he/she can mimic the behavioral biometrics of a registered user to get $a \geq y(x < y < 1)$, then, to pass the authentication $(s > s_0)$, the authentication score $b$ of the spoofer needs to satisfy

$$b > \frac{1 - (1 - x + y)\alpha\%}{1 - \alpha\%}. \tag{6}$$

Since we have $1 - x + y > 1$, it can be seen from Eq.6 that the required score of $b$ monotonically decreases when $\alpha$ increases, showing that with more reliance on behavioral biometrics, the user authentication system would be more vulnerable to spoofers.

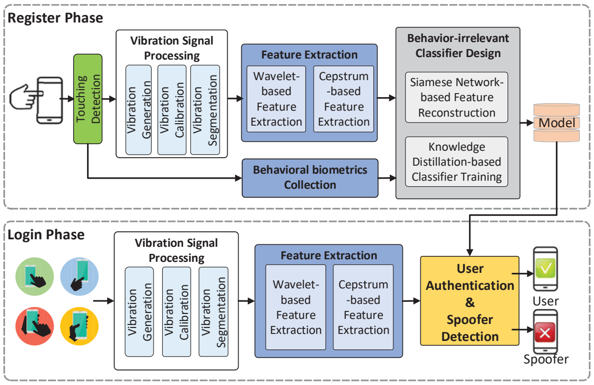

Register Phase
Touching Detection
Vibration Signal Processing
Vibration Generation
Vibration Calibration
Vibration Segmentation
Feature Extraction
Wavelet-based Feature Extraction
Cepstrum-based Feature Extraction
Behavioral biometrics Collection
Behavior-irrelevant Classifier Design
Siamese Network-based Feature Reconstruction
Knowledge Distillation-based Classifier Training
Model
Login Phase
Vibration Signal Processing
Vibration Generation
Vibration Calibration
Vibration Segmentation
Feature Extraction
Wavelet-based Feature Extraction
Cepstrum-based Feature Extraction
User Authentication & Spoofer Detection
User
Spoofer

**Figure 6: System architecture of *TouchPass*.**

Therefore, in order to build a secure on-touch user authentication scheme, we should reduce the reliance of behavioral biometrics as much as possible, and realize towards a behavior-irrelevant on-touch user authentication system.

## 3 SYSTEM OVERVIEW

We design a behavior-irrelevant on-touch user authentication system, *TouchPass*, which identifies different individuals through finger touching on smartphones. Figure.6 shows the architecture of *TouchPass*, which can be divided into two phases, i.e., register phase and login phase.

In register phase, *TouchPass* collects data from users. Once a touching behavior on smartphones is detected, *TouchPass* first generates a specific-designed vibration signal through the built-in vibration motor and collects the vibration signal with the IMU sensor in the smartphone. After the signal is collected, it further goes through calibration and segmentation to obtain segments in transient-state and steady-state of vibration propagation. Then, based on the segments, *TouchPass* utilizes wavelet-based method to extract features in transient-state of vibration signals, and cepstrum-based method to extract features in steady-state of vibration signals, respectively. After that, *TouchPass* leverages the extracted features and behavioral biometrics collected from touch screen sensor to design a behavior-irrelevant on-touch user authentication model. In particular, *TouchPass* develops a siamase network-based approach to reconstruct the extracted features to behavior-irrelevant features, and further proposes a knowledge distillation-based model to train a light-weighted behavior-irrelevant model for user authentication on mobile devices.

In login phase, *TouchPass* first captures the vibration signals generates by a smartphone's motor when a finger touching on the smartphone is detected, and then extracts the features in the same way as in register phase. After that, *TouchPass* performs user authentication and spoofer detection with the trained model from the register phase based on the extracted features.

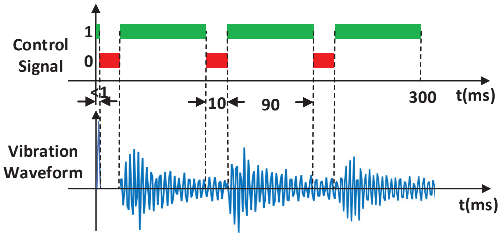


**Figure 7: Illustration of designed vibration signal.**

# 4 VIBRATION SIGNAL PROCESSING

For utilizing active vibration signals to realize user authentication on a smartphone, *TouchPass* first needs to generate vibration signals with the motor in the smartphone and receive the vibration signals with the IMU sensor in the same smartphone. In this section, we introduce details of the active vibration signal design, and further describe the synchronization and segmentation of received vibration signals.

## 4.1 Vibration Signal Design

To capture the finger's characteristics of a user who touches a smartphone for authentication, we first design the active vibration signal generated by smartphones. Once senses a touching behavior on a smartphone, *TouchPass* actively generates a vibration signal utilizing the vibration motor on the smartphone. The motors implemented in most smartphones are Linear Resonant Actuators (LRA), which allow for regulating both the magnitude and frequency of vibration signals. However, most current smartphone operator systems (i.e., Android, iOS, etc.) do not give the authority of changing the vibration frequency to users and developers, but only allow the motor to vibrate at its resonance frequency, which is the frequency of vibration signals at steady-state as described in Section 2.2.

Base on the characteristics of motors on smartphones, *TouchPass* generates the vibration signal by activating/deactivating the motor on a smartphone. Specifically, after the smartphone senses a finger touching, *TouchPass* activates the motor periodically to let the vibration goes through transient-state and steady-state alternately. Figure.7 illustrates the generated vibration signals in *TouchPass*. The signal begins with a very short impulse of vibration ($< 1ms$), and then a short delay with about $10ms$, which is used for signal synchronization. After generating the impulse and the following delay, the motor is activated for $90ms$ with a steady-state process and a transient-state process, and then deactivated for $10ms$ to decay the vibration. The activate-deactivate process continues for 3 times to generate a vibration signal with about $300ms$, which is a very short time period that a user could hardly feel uncomfortable.

## 4.2 Signal Synchronization and Segmentation

Since we use the vibration motor on a smartphone to generate active vibrations and the IMU sensor in the same smartphone to receive the vibration signals, the motor and the IMU sensor need to be synchronized. *TouchPass* realizes the synchronization by utilizing the short impulse of vibration at the beginning of the generated vibration signal. Since the transmitting speed of the vibration signal is larger that $2000m/s$ in smartphones, and the sampling rate of the IMU sensor on most off-the-shelf smartphones is no more than $1000Hz$(which brings the time resolution as $1ms$), the time that the motor starts generating vibration signals is within the $1ms$ sample that the IMU sensor receives the impulse. So the motor and the IMU sensor is synchronized at the level of $1ms$.

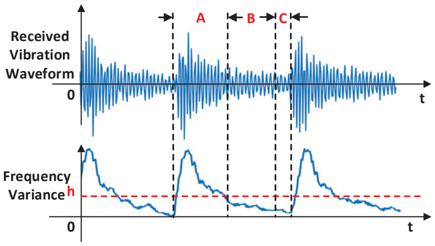


**Figure 8: Illustration of vibration signal segmentation.**

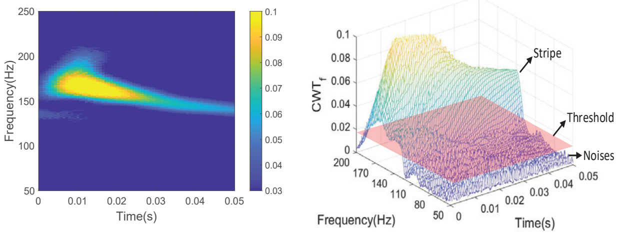
After synchronization, *TouchPass* further segments the received vibration signal into transient-state and steady-state based on the frequency changes of vibration signals. The vibration frequency changes over time in transient-state, while remains steady in steady-state. Specifically, *TouchPass* first applies a sliding window to computes the frequency variance within the window, and then uses a threshold $h$ to segment vibration signals. When the frequency variance becomes lower than $h$, we consider the signal is in the steady-state ($h$ is determined by empirical study). Figure.8 illustrates the vibration signal segmentation. Since the motor and the IMU sensor is synchronized, *TouchPass* first separates the $90ms$ vibration phase when the motor is activated from the $10ms$ vibration decaying phase when the motor is deactivated (denoted as $C$ in Figure.8). After that, base on the threshold $h$, *TouchPass* further segments the $90ms$ vibration transmitting phase into transient-state denoted as $A$ and steady-state denoted as $B$, as shown in Figure.8. After transient-state and steady-state of the received vibration signals are separated, we can extract features from each state of vibration signals for user authentication.

# 5 FEATURE EXTRACTION

In order to obtain representative characters from received vibration signals, we need to extract features from different states of active vibration signals. In this section, we present the approaches for extracting features from transient-state and steady-state of the received vibration signals, respectively.

## 5.1 CWT-based Feature Extraction in Transient-state

We first focus on extracting features from transient-state of received vibration signals. In Section.2.4, we show that transient-state of vibration signals tend to have different patterns in frequency domain for different users, indicating that the frequency changing process in transient-state of vibration signals embeds unique physical characters of touching fingers. Since the transient-state of vibration signals is highly dynamic, we need to achieve high-resolution in both time and frequency domain of vibration signals for accurately capturing the frequency changing process. Therefore, instead of fast Fourier transformation[41] (FFT)-based approaches that inevitably have time-frequency trade-off, we apply continuous wavelet transform[6] (CWT) to transient-state of received vibration signals for obtaining a time-frequency spectrum with high-resolution in both time and frequency domain. Specifically,



(a) 2-D CWT spectrum

(b) 3-D CWT spectrum

**Figure 9: Illustration of CWT on a $50ms$ transient-state of vibration signals.**

CWT can be denoted as:

$$CWT_f(a, \tau) = \left\langle f(t), \psi_{a,\tau}(t) \right\rangle = a^{-1/2} \int_{\mathbb{R}} f(t)\psi\left(\frac{t-\tau}{a}\right) \mathrm{d}t, \quad (7)$$

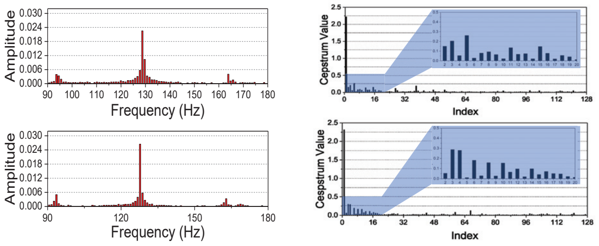
where $CWT_f(a, \tau)$ is the obtained spectrum, $f(t)$ is the signal function that corresponds to the transient-state of vibration signals in *TouchPass*, and $\psi_{a,\tau}(t)$ is the wavelet base function, with $a$ and $\tau$ representing the resolution of frequency and time domain, respectively. The choice of wavelet base function directly affects the wavelet transformation processing result to the signal. In *TouchPass*, we choose *complex Morlet function* as the wavelet base function because of its extremely high resolution in frequency and time domain.

Figure.9 illustrates the CWT result of *TouchPass* on a $50ms$ transient-state of vibration signals, both in 2-D plot form (Figure.9(a)) and 3-D plot form (Figure.9(b)). It can be seen from Figure.9 that the dominate part of vibration signals starts at a relatively high frequency around $170Hz$ and decreases to a steady frequency around $140Hz$ within $50ms$, which fits the model for transient-state of vibration signals in Section.2.2 well. Moreover, the high-resolution CWT spectrum reveals that transient-state of vibration signals spreads in a frequency range of about $20Hz$, which forms a time-frequency stripe as shown in Figure.9(b). The stripe describes the influence during propagation of active vibration signals with respect to the user's finger, and thus can be taken as the response of the user's finger to the active vibration signal. Hence, *TouchPass* extracts the stripe as time-frequency features in transient-state of active vibration signals. We could set a threshold (as shown in Figure.9(b)) to filter the noises and obtain the stripe.

## 5.2 Cepstrum-based Feature Extraction in Steady-state

Besides transient-state, steady-state of vibration signals also embeds characteristics of the user who touches a smartphone. So we further extract features from steady-state of active vibration signals. According to Section.2.2, vibration signals in steady-state are centralized to the resonance frequency and several harmonic frequencies. To realize the user authentication, we exploit these frequencies for extracting effective features.

We first take a close look at the frequency spectrum. Figure.10(a) plots two frequency spectrum for the steady-state of vibration signals, corresponding to two finger touching on a smartphone from two different users. It can be observed from Figure.10(a) that there are some less-powerful frequency components around the resonance frequency and harmonic frequencies, which are denoted as side-band frequencies[14]. The side-band frequencies in steady-state usually present the influence of interactions among different components in a system. In the case of *TouchPass*, it presents the interactions among smartphone and the touching finger. In order to capture the side-band frequencies, we further compute the cepstrum[4] for the steady-state of vibration signals. Cepstrum is widely utilized in the field of complex vibration system such as vibration-based gear fault detection and speech analysis, as it can provide the rate of changes in different spectrum bands and distinguish the influences of different components in a complex vibration system. Specifically, the cepstrum can be calculated as:



(a) Frequency spectrum.

(b) Cepstrum.

**Figure 10: Illustration of frequency spectrum & cepstrum for steady-state of vibration signals.**

$$\mathrm{C_y(q) = F^{-1}\left(\log S_y(f(t))\right)}, \quad (8)$$

where $f(t)$ is the signal function that represents steady-state of vibration signal in *TouchPass*, $\mathrm{S_y(f(t))}$ is the power spectral density[27] (PSD) of the signal, and $\mathrm{F^{-1}}$ is inverse fast Fourier transform[41] (IFFT).

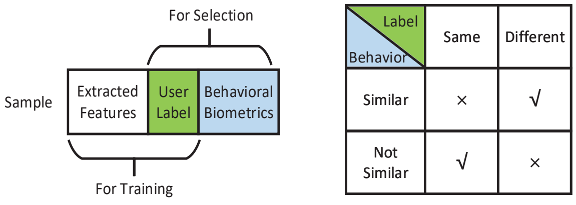
Through Eq.8, the cepstrum collects the side-band frequencies and forms a new frequency representation of the vibration signal. Figure.10(b) shows the cepstrum of vibration signals corresponding to the two touching in Figure.10(a). Although the frequency spectrums look similar in Figure.10(a), it can be seen from Figure.10(b) that two cepstrums present very different patterns. Therefore, *TouchPass* extracts cepstrum as features in the steady-state of vibration signals.

# 6 BEHAVIOR-IRRELEVANT AUTHENTICATION MODEL DESIGN

To realize behavioral-irrelevant user authentication, *TouchPass* needs to reconstruct extracted features to behavior-irrelevant features, and then build a light-weight behavior-irrelevant authentication model utilizing the reconstructed features.

## 6.1 Behavior-irrelevant Feature Reconstruction

The features extracted in Section.5 contain both physical characters of touching fingers and behavioral biometrics of touching behavior. To build a behavior-irrelevant user authentication scheme in *TouchPass*, we need to eliminate the influence of behavioral biometrics in the extracted features, i.e., reconstruct features to behavior-irrelevant features. *TouchPass* leverages the Siamese network for the feature reconstruction. The idea of Siamese network is basically using a pair of neural networks with same architecture and weights



| Label / Behavior | Same | Different |
|---|---|---|
| Similar | × | √ |
| Not Similar | √ | × |

(a) Samples in *TouchPass*.

(b) Selection Table.

**Figure 11: Illustration of Training samples selection.**

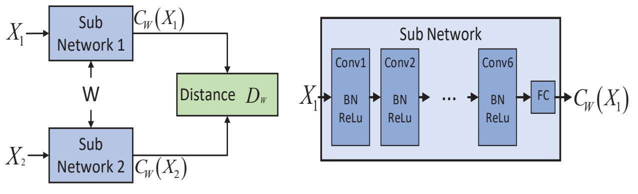
to compute a distance metric for two inputs[5][19]. The particular structure of Siamese network enables customized feature extraction by the selection of sample pairs during training, which supports the feature reconstruction in *TouchPass*.

Specifically, *TouchPass* applies a training sample selection scheme, as shown in Figure.11. Besides extracted features, each collected sample in *TouchPass* also contains behavioral biometrics collected by the built-in touching screen sensors on smartphones. The extracted features are used as training inputs to the Siamese network, while the behavioral biometrics are used for training sample selection. Concretely, for each pair of samples, *TouchPass* classifies it into one of the four classes based on their behavioral biometrics and user labels, as shown in Figure.11(b). If the user labels are the same, *TouchPass* selects the sample pairs with less similar behavioral biometrics as the training samples, so that the model could learn to ignore the behavioral differences for samples from the same user, while if the user labels are different, *TouchPass* selects the sample pairs with similar behavioral biometrics, so that the model could learn to distinguish samples from different users not based on the behavioral biometrics. Hence, the key part of the training sample selection is to determine whether two training samples have similar behavioral biometrics. *TouchPass* calculates the similarity of the two behavior features, $x_1$ and $x_2$, by Pearson Correlation Coefficient, which can be denoted as:

$$r_{x_1,x_2} = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}}, \tag{9}$$

where $n$ is the sample size of behavioral features. After calculation, we normalize the coefficient to $[0, 1]$ and set a threshold $\delta$ to determine the similarity of behaviors. Then, all pairs of training samples can be divided into two categories, i.e., pairs with similar behavioral biometrics, i.e., $\rho > \delta$, and pairs with less similar behavioral biometrics, i.e., $\rho < \delta$. Based on the similarity, *TouchPass* selects proper training samples as input to the Siamese network for reconstructing behavior-irrelevant features.

Figure.12 shows the architecture of Siamese network in *TouchPass*. Given the a pair of signals features as the input, Siamese network reconstructs the behavior-irrelevant feature representation from the signals features through two identical sub-networks, and computes the distance of the behavior-irrelevant feature representation as the similarity of the inputs. The structure of the sub-network in *TouchPass* is basically a 6-layer time-delay neural network (TDNN)[40], as illustrated in Figure.12(b). The sub-network is designed with two 1-D convolution (Conv) blocks[20] to extract the behavior-irrelevant features and a fully-connected (FC) layer to merge the features into a compact representation. Within each convolution block, we implement a 1-D convolution kernel followed by a batch-normalization (BN) layer[15] and a ReLU layer[30] to normalize the features. With the multiple-layer CNN architecture, the sub-network has capability to learn the mapping from extracted features to behavior-irrelevant feature representation.



(a) Architecture of Siamese Network in *TouchPass*.

(b) Sub-network Design in *TouchPass*.

**Figure 12: Siamese network architecture.**

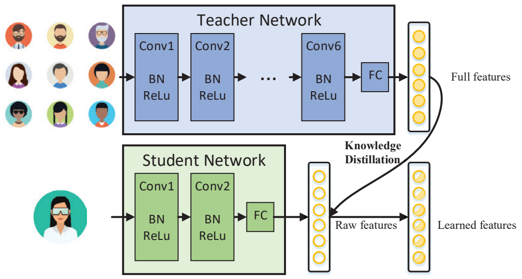


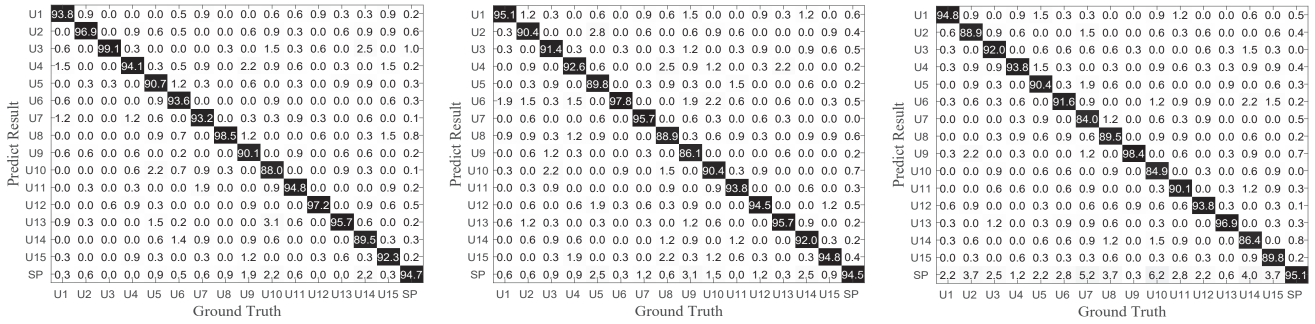
**Figure 13: Architecture of knowledge distillation in *TouchPass*.**

By denoting the parameters of the sub-network as $W$, the loss function $L(W)$ is designed as

$$L(W) = \sum_{i=1}^{N} Y(D_W^i)^2 + (1 - Y)\max(M - D_W^i, 0)^2, \tag{10}$$

where $Y$ is a indicator to show if the two input samples are related to the same user, i.e., if they are from the same user, then $Y = 1$, otherwise, $Y = 0$. $D_W^i$ is the Euclid distance of the $i^{th}$ input samples, $M$ is the margin that represents the decrease interval. The intuition behind Eq.10 is that the loss $L(W)$ is monotonically increasing by the distance $D_W$ if the two input samples are from the same user, while $L(W)$ is monotonically decreasing by $D_W$ if they are from different users. Then, the designed Siamese network aims at minimizing the loss $L(W)$, i.e., minimizing the distance between samples of the same user and maximizes the distance between samples of different users.

To train the Siamese network, we recruit 20 volunteers (12 males, 8 females with ages in $[19, 55]$) to perform finger touching behavior to three different types of smartphones(Xiaomi 6, Galaxy S6 and Honor 7x). Each volunteer is required to touch each smartphone for 20 times, with different force (light, normal or hard) in different area (as shown in Figure.15), and different supports for smartphones (wooden desk, book or hand with different gestures). Based on the collected data, the Siamese network is trained according to Eq.10. After training, the designed Siamese network could reconstruct the extracted features to behavior-irrelevant features.



(a) Confusion matrix in laboratory environment.

(b) Confusion matrix in bar environment.

(c) Confusion matrix in bus environment.

**Figure 14: Confusion matrix of *TouchPass* for user authentication and spoofer detection.**

### 6.2 Light-weighted Behavior-irrelevant Model

To realize the behavior-irrelevant user authentication on off-the-shelf smartphones, we further build a light-weighted behavior-irrelevant user authentication model upon the reconstructed behavior-irrelevant feature representation.

For a smartphone with limited computing power, realizing the generalized feature reconstruction model with multiple-layers CNN implementation as described in Section 6.1 is too computational expensive.

To achieve a light-weighted authentication model, while keeping the generalization capability, *TouchPass* exploits the idea of knowledge distillation, which transfers the knowledge from a large and cumbersome model to a small model that is more adequate for deployment[13]. In *TouchPass*, the reconstruction network in Figure.12(b) is set as a trained teacher network to teach the knowledge of behavior-irrelevant feature transformation to a smaller student network with only two convolutional layers, as illustrated in Figure.13. Specifically, the training of the student network is a two-folded process. In the first training stage, the outputted features of teacher network, denoted as $F_T$, is utilized as the training goal of the output of student network(denoted as $F_S$) with respect to the training data from the teacher network. And the loss function $L_{TS}$ is calculated as the cross entropy of $F_T$ and $F_S$:

$$L_{TS} = H_{F_T}(F_S) = -\sum_i F_T \times \log(F_S). \tag{11}$$

In the second training stage, the student network refines the parameters of the network with its own training data from specific user registered to the smartphone. And the loss function is in the same form of Eq.10.

After the two-folded training, the student network is able to extract behavior-irrelevant features (denoted as $F_L$) of users that registered to the smartphone. *TouchPass* then builds a behavior-irrelevant authentication model to distinguish different registered users and spoofer by determining a template $T_i$ for each registered user $i$. Specifically, the template of a registered user $i$ is designed as the average of the extracted behavior-irrelevant features of user $i$:

$$T_i = \frac{1}{N}\sum_{i=1}^{N} F_{L_i}. \tag{12}$$

With the template, *TouchPass* could authenticate users and detect spoofers.

### 6.3 User Authentication & Spoofer Detection

When a user logs in *TouchPass* system, the user authentication is performed based on the built-in templates for registered users in the behavior-irrelevant authentication model. Specifically, assuming there are $n$ registered users, there should be $n$ templates corresponding to each registered user, i.e., $T_1, T_2, \ldots, T_n$ in the authentication model. Then, the extracted behavior-irrelevant feature of the login user, $F_x$, is compared with each of the $n$ templates, respectively, and output $n$ distances for each registered users, i.e., $D_1, D_2, ..., D_n$. And the distance is calculated as:

$$D_i = ||T_i - F_x||_2, \tag{13}$$

where $||x||$ denotes the 2-norm of the vector $x$. After the distances are calculated, the login user is identified as user $i$ with smallest distance $D_i (i \in 1, 2, \ldots, n)$ among the $n$ distances.

A login user can be an adversary that attempts to enter the system. Hence, *TouchPass* needs to detect the adversary in the login phase. Since the authentication model is trained to maximize the distance between different inputs, the sample collected from the adversary tend to have a large distance to any registered user. Thus, if the smallest distance $D_i$ is larger than a pre-determined threshold $\delta$ (which is normally obtained by empirical study), *TouchPass* would authenticate the login user as an adversary.

## 7 EVALUATION

In this section, we evaluate the performance of *TouchPass* in real environments with 75 volunteers.

### 7.1 Setup & Methodology

The prototype of *TouchPass* is implemented as an Android App in three different types of smartphones, which are Xiaomi 6, Galaxy S6 and Honor 7x, respectively. Our experiments are conducted in three different environments, i.e., laboratory (static and quiet), bar (static and noisy) and bus (dynamic and noisy). For each environment, we randomly recruit 25 volunteers (16 males, 9 females with ages in [20, 52]) to conduct experiments for evaluating the performance of *TouchPass*, i.e., 75 volunteers in total are involved in the evaluation. Among the 25 volunteers in each environment, 15 of them register *TouchPass* system as legitimate users, the rest 10 volunteers play the role of spoofers. Specifically, in register phase, each of the 15 legitimate users are asked to touch the smartphone with a

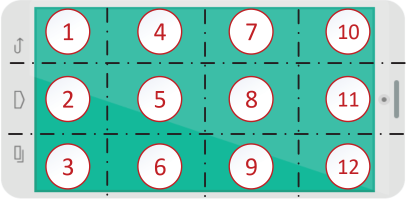


**Figure 15: Illustration of different touching positions on a smartphone.**

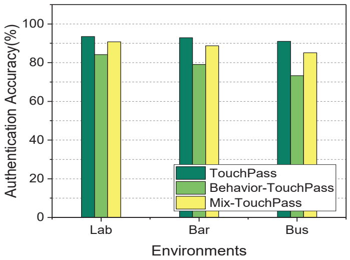


**Figure 16: Authentication accuracy under different environments.**

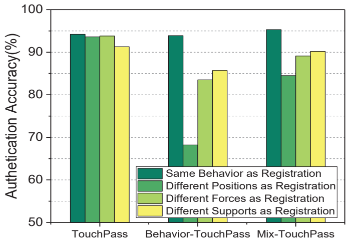


**Figure 17: Sensitivity of three versions of *TouchPass* to behavioral biometrics.**

randomly generated requirement for touching position (one of the 12 positions as shown in Figure.15), one of the touching forces (i.e., soft, normal or hard) and one of the supports for smartphones (i.e., hand, book or wooden desk (not available for bus environment)), while the 10 spoofers do not touch with smartphones. Then, during the login phase, all 25 volunteers in each environment are required to touch smartphones with all possible combinations of 12 touching positions, 3 touching forces and 3 supports for smartphones.

Since *TouchPass* is an authentication scheme towards behavior-irrelevance, we also implement two extra version of *TouchPass*, i.e., *Behavior-TouchPass* and *Mix-TouchPass*, to evaluate the effectiveness of behavior-irrelevant for *TouchPass*. The main differences between these versions and the original *TouchPass* are the reconstructed features used for user authentication. For the original *TouchPass*, we has the training sample selection strategy as Figure.11(b) for constructing behavior-irrelevant features. For *Behavior-TouchPass*, we utilize the opposite strategy of original *TouchPass* to construct behavior-relevant features. While for *Mix-TouchPass*, we do not perform training sample selection, but let all the Siamese network train all the samples, so that the reconstructed features contain both behavioral biometrics and physical characters.

To evaluate *TouchPass*, we define several metrics as follows:

- **Authentication Accuracy:** the probability that user $A$ is correctly authenticated as user $A$ among all users.
- **Confusion Matrix:** Each row of the matrix represents the instances in a predicted class while each column represents the instances in an actual class. The $i^{th}$-row and $j^{th}$-column entry of the matrix shows the percentage of samples that are classified as the $i^{th}$ class while actually the $j^{th}$ class.
- **False Accept Rate (FAR):** The probability that system authenticates a spoofer as a legitimate user.
- **False Reject Rate (FRR):** The probability that system authenticates a legitimate user as a a spoofer.

## 7.2 Overall Performance

We first evaluate the overall authentication performance of *TouchPass* for 15 legitimate users (denoted as $U1$, $U2$, ..., $U15$) and 10 spoofers (denoted as $SP$) in each of three different environments, and show the results as confusion matrix in Figure.14. It can be observed that *TouchPass* can achieve average accuracy of 93.5%, 92.9% and 91.0% for legitimate users authentication, and 94.7%, 94.5% and 95.1% for spoofer detection in lab, bar and bus environments, respectively. The result indicates that *TouchPass* can accurately authenticate legitimate users and detect spoofers in different environments. In bus environment, the accuracy for legitimate users authentication is a little bit lower than other two environments due to the influence of external vibration noises to the vibration-based system, but the accuracy for spoofer detection remains high. Then, we further compare the authentication accuracy of original *TouchPass* with *Behavior-TouchPass* and *Mix-TouchPass* in three different, and show the result in Figure.16. It can be seen that the original *TouchPass* outperforms *Behavior-TouchPass* and *Mix-TouchPass* for each environment, showing that behavior-irrelevant authentication scheme is proper for cross-environment user authentication.

We then evaluate the performance of *TouchPass* under different behavioral biometrics, including touching positions, touching forces and supports for smartphones. Table.1 gives the authentication accuracy for each behavioral biometric. It can be seen that there is no obvious difference in authentication accuracy of *TouchPass* for different touching positions, touching forces and supports for smartphones. We further evaluate the sensitivity of the three versions of *TouchPass* to the changes of behavioral biometrics(especially between registration phase and login phase). The result is show in Figure.17. It can be seen that for the original *TouchPass*, the authentication is robust to behavior changes, while for *Behavior-TouchPass* and *Mix-TouchPass*, the authentication accuracy drops evidently when the behavior changes. The result shows that the by extracting behavior-irrelevant features, *TouchPass* is not sensitive to the changes of behavioral biometrics and can reach a good authentication performance even when the login touching behaviors are different from registration.

## 7.3 Feature Validation

In the design of *TouchPass*, we utilize features extracted from both transient-state and steady-sate of vibration signals, i.e., CWT-based

**Table 1: Authentication accuracy of *TouchPass* under different touching behaviors.**

| Position(#) | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| Accuracy(%) | 92.8 | 93.1 | 93.0 | 93.2 | 93.3 | 92.9 |
| Position(#) | 7 | 8 | 9 | 10 | 11 | 12 |
| Accuracy(%) | 93.0 | 93.3 | 93.1 | 92.7 | 93.1 | 92.8 |
| Force | soft | | normal | | hard | |
| Accuracy(%) | 93.1 | | 93.3 | | 93.2 | |
| Supports | hand | | book | | desk | |
| Accuracy(%) | 93.3 | | 92.9 | | 93.1 | |

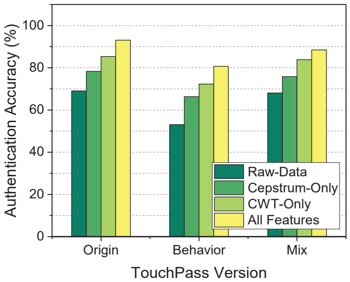


**Figure 18: Authentication accuracy of *TouchPass* under different features.**

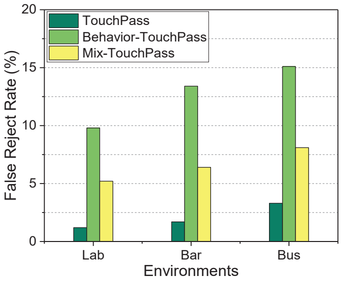


**Figure 19: FRR of *TouchPass* for authentication approaches.**

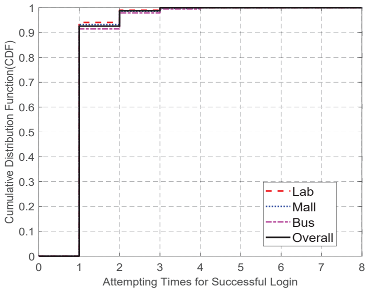


**Figure 20: CDF of attempting times for successful login.**

features and cepstrum-based features, to realize user authentication. To show the individual contribution of these features, we conduct an experiment that builds three extra user authentication systems leveraging the raw vibration data, CWT-based feature only and cepstrum-based feature only, respectively. Then we compare the performance of these systems to *TouchPass*. The result is shown in Figure.18. The average authentication accuracy of the *TouchPass* with raw data can only achieve 69.1%, *TouchPass* with only CWT-based features or with only cepstrum-based features improve the average authentication accuracy to 85.4% and 78.3%, respectively. And *TouchPass* with all features further improves the average authentication accuracy to 93.2%. This pattern remains for Behavior-TouchPass and Mix-TouchPass. The results indicates that both CWT-based feature and cepstrum-based feature contributes to the performance of *TouchPass*.

## 7.4 Performance on User Experience

Since *TouchPass* is designed as a smartphone user authentication system, we evaluate the experience of *TouchPass* for registered users through false reject rate (FRR), attempting times for successful login and the time cost for each authentication.

**False Reject Rate:** We show the FRR for three versions of *TouchPass* under three different environments in Figure.19. It can be seen that the FRRs of *TouchPass* are the lowest under all environments, with the FRRs lower than 4% for all environments. However, the FRRs of *Behavior-TouchPass* and the FRRs of *Mix-TouchPass* are higher than 9% and 5% under all environments, respectively. The result demonstrates that *TouchPass* has a low probability to reject legitimate in various environments, which provides a satisfactory user experience.

**Attempting Times for Successful Login:** We evaluate the the attempting times for successful login and show the result in Figure.20. It can be seen from the figure that more than 92% login operations are successful with only one attempt under different environments, and more than 99.5% login operations are successful with three attempts, which is user-friendly in real scenarios.

**Authentication Time Cost:** We also evaluate the user experience through the time cost for each authentication process under different types of smartphones, and the result is shown in Table.2. It can be seen that the overall time during each authentication is lower than 1.2$s$ with all three different types of smartphones. For comparison, Touch ID on iPhone 7 Plus costs approximately 0.91$s$ and Face ID on iPhone X costs approximately 1.5$s$ for user authentication[37]. So *TouchPass* could achieve a satisfactory user experience.

## 7.5 Performance on Attack Resistance

To show that *TouchPass* can resist attacks described in Section.2.1, we conduct experiments under mimic attack and replay attack, respectively. For mimic attack, a spoofer tries his/her best to mimic the touching behavior of a legitimate user during login. For replay attack, we consider the case when the smartphone is supported by a plane, and a spoofer places a smartphone besides the legitimate user's phone on the same plane to record the vibration signals, and then replays the recorded samples to the authentication system.

In each experiments, we divide the 25 volunteers into 3 groups in each environments(5 volunteers in the first group, 10 volunteers in the second and third group). The volunteers in the first group register to the system as legitimate users. Volunteers in the second group and third group performs mimic attack and replay attack to the system, respectively, in which each volunteer performs the attack for 20 times.

Figure.21 shows the false accept rate for mimic attack. It can be seen that for *TouchPass*, the false accept rates are lower than 1.8% for different touching modes in different environments, which indicates that *TouchPass* can resist mimic attack well in various environments. While the false accept rates of *Behavior-TouchPass* and *Mix-TouchPass* are higher than 4.7% and 2.3% in different environments, respectively, which are much more vulnerable to mimic attacks than original *TouchPass*. Figure.22 shows the false accept rate of all three versions of *TouchPass* for replay attack. The false accept rates for all three versions of *TouchPass* are lower than 2% under three different environments, showing that *TouchPass* can resist the replay attack with low false accept rate.

## 7.6 Impact of Threshold for Determining Similarity of Behavior Features

As described in Section 6.1, to obtain a behavior-irrelevant user authentication model, *TouchPass* selects specific training data to the

**Table 2: Running time of *TouchPass* during an authentication.**

| Devices | Vib | Signal | CWT | Cepstrum | Authen | Total |
|---|---|---|---|---|---|---|
| Xiaomi 6 | 0.3s | 0.06s | 0.35s | 0.18s | 0.31s | 1.20s |
| Galaxy S6 | 0.3s | 0.03s | 0.27s | 0.13s | 0.26s | 0.99s |
| Honor 7 | 0.3s | 0.05s | 0.34s | 0.14s | 0.28s | 1.11s |

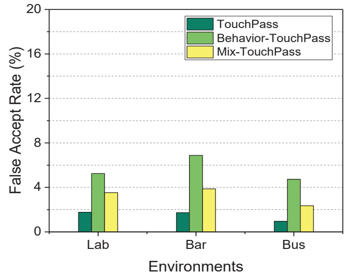


Figure 21: FAR of *TouchPass* for mimic attack.

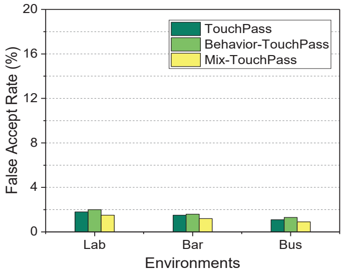


Figure 22: FAR of *TouchPass* for replay attack.

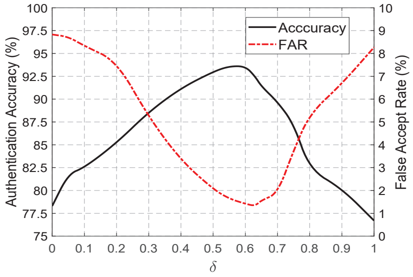


Figure 23: Impact of threshold.

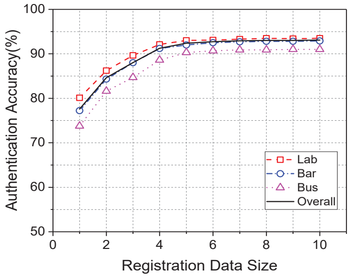


Figure 24: Impact of registration data size.

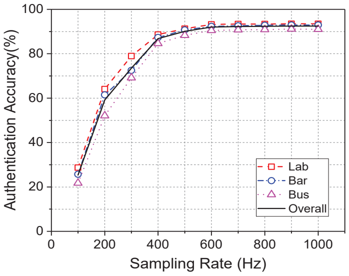


Figure 25: Impact of sensor sampling rate.

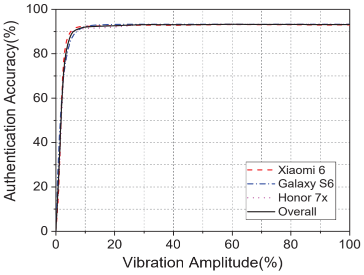


Figure 26: Impact of vibration strength.

Siamese network by calculating the similarity of behavior biometrics, which is controlled by a threshold $\delta$. Therefore, the value of $\delta$ directly influences the training data selection and further impacts the performance of *TouchPass*. Figure.23 shows the accuracy and false accept rate of *TouchPass* under different thresholds $\delta$. It can be seen from the figure that as the threshold increases form 0 to 1, the accuracy first increases and then decreases, while the false accept rate first decreases and then increases. The reason is that a small $\delta$ results in few training samples for reducing the distance of samples from the same user, while a large $\delta$ results in few training samples for increasing the distance of samples from different users. Hence, both small and large $\delta$ could add biases to the training data. Moreover, we find that the accuracies of *TouchPass* is above 92.5% when $\delta$ is in the range of $[0.47, 0.63]$ and the false accept rates of *TouchPass* is below 2% when $\delta$ is in the range of $[0.55, 0.68]$. Therefore, a practical deployment can select a threshold in the range of $[0.51, 0.69]$ to meet the specific requirements. Therefore, to achieve the best performance, we set $\delta = 0.6$ for *TouchPass* during the evaluation.

## 7.7 Impact of Registration Data Size

The size of registration data for a user is the number of touches during register phase. A larger registration data size usually improve the performance of the system, but leads to tedious finger work. Hence, we evaluate the performance of *TouchPass* under different sizes of registration data, and show the result in Figure.24. We can observe that as the size of registration data for each user increases from 1 to 10, the accuracy first increases and then remains stable when the size reach about 7. Moreover, when a user touches the smartphone only 2 times for registration, *TouchPass* can achieve an accuracy of about 85% for authentication in all three different environments.

## 7.8 Impact of Sampling Rate

We evaluate the performance of *TouchPass* under different sampling rates of IMU sensors on smartphones. Figure.25 shows the authentication accuracy of *TouchPass* under different sampling rates. It can be observed that as the sampling rate increases, the accuracy first increases sharply and then remains stable. Specifically, the authentication accuracy is relatively low when the sampling rate of IMU sensors is lower than $400Hz$. The reason is that the frequency components of active vibration signals are mainly in the range of $50Hz$ to $200Hz$ in frequency domain, which means that IMU sensors can not capture all valid vibration signals if the sampling rate is less than $400Hz$. Furthermore, it can be observed that after the sampling rate is larger than $600Hz$, increasing sampling rate do not contribute much to the authentication accuracy. Since most IMU sensors used in commercial smartphones could support a sampling rate up to $1000Hz$, the result shows that *TouchPass* can be widely employed in off-the-shelf smartphones.

## 7.9 Impact of Vibration Strength

Since *TouchPass* actively generates vibration signals to authenticate users, the strength of vibration signals could impact the authentication performance. Hence, we evaluate the performance of *TouchPass* on user authentication under different levels of vibration strength and show the result in Figure.26. We can observe from the figure that for Xiaomi 6 Galaxy S6 and Honor 7, as the vibration strength increases from 0 to 100%, the authentication accuracy of *TouchPass* first increases and then remains stable. This is because that a stronger vibration signal leads to higher Signal-Noise-Ratio (SNR) in the received signal and thus brings more robust features to train the user authentication model. Moreover, with only 8% vibration strength, *TouchPass* can achieve an accuracy over 90% for user authentication.

## 8 DISCUSSION

We discuss several issues of *TouchPass*, including the advanced attack resistance and the robustness to physical character changes.

**Advanced Attack Resistance:** Besides replay attack and mimic attack, there could be skilled spoofers that can obtain the collected data of the IMU in a registered user's smartphone. In that case, the spoofer could perform advanced replay attack by injecting the IMU sensor data collected from a login operation of a registered user. To prevent such advanced replay attack, *TouchPass* could give a small and random turbulence to the duration of active vibration signal generated by the motor smartphones, and the duration of transient-state and steady-state for the collected vibration signal will change accordingly. Then, by checking the duration of collected vibration signal, *TouchPass* can resist the advanced replay attack, while keep the performance of user authentication.

**Robustness to Physical Character Changes:** The physical characters of fingers utilized by *TouchPass* are usually stable under short-time environment changes such as weathers. But in a relatively long time period, i.e., years, these physical characters could change. For instance, the fat content changes when a user gains or loss weight, and the bone density changes when a user grows. These changes could influence the performance of *TouchPass*. To deal with physical character changes, *TouchPass* could update the user template periodically based on the login data.

## 9 RELATED WORK

We review the related works of *TouchPass* as follows:

**Smartphone User Authentication** Password-based user authentication including PIN number[29], lock pattern[28] and other graphical passwords[38] are the most widely used approaches on smartphones user authentication. Although low-cost and easy to implement, these approaches are vulnerable under shoulder-surfing and smudge attacks[39]. To overcome the shortcomings of password, previous works exploit physical biometric-based authentication approaches on smartphones, such as fingerprint[17], face recognition[35], iris recognition[21], and voice-print recognition[8]. However, due to lack of liveness detection[44] and sensitive to environments, these approaches are vulnerable to replay attacks.

**Touch-based User Authentication** As one of the most pervasive and natural behavior to interact with smartphones, finger touching presents unique behavioral biometrics among different users, which has been exploited for touch screen-implemented smartphone user authentication[1][7][10][9][11][23]. Among these approaches, some leverages the behavioral biometrics including touching pressure, finger area, duration and location for realizing touch-to-access authentication[1][7][10], while others exploit the changes of behavioral biometrics between successive touches for realizing continuous user authentication[9][11][23]. However, Khan et al.[18] show that these approaches are vulnerable under shoulder surfing and offline training.

More Recently, a touch-based user authentication scheme combining physical characters of a hand size and behavioral biometrics of the touching behavior is proposed[36], and shows higher authentication accuracy than behavioral biometrics-alone and physical characters-alone approaches. However, since the approach is still partly dependent on behavioral biometrics, it is less secure under attacks against behavioral biometrics like statistical attack[33]. Moreover, the approach requires users to perform a fixed touching gesture, which is an extra burden since users have to remember that pattern as a ‘password’.

Besides touch screen scenario, there are other touch-based user authentication approaches focusing on IoT[25] and smartwatch[3] scenarios. VibWrite[25] develops a vibration-based finger-input authentication system that can be implemented on various planes in IoT scenarios. However, this approach uses stand-alone vibrator motor and piezoelectric sensor, which are not available on COTS smartphones. Taprint[3] leverages the IMU on smartwatches to sense the passive vibrations generated by typing on the fix knots of hand for user authentication, which is not practical for on-touch user authentication on mobile devices with different touching modes. Moreover, the above approaches still rely on behavioral biometrics, which brings security risk to the authentication system and not practical for finger touching on smartphones with various touching modes.

**Vibration-based Applications** Due to the excellent transmission characteristics in short-range, recent years have witnessed huge development for vibration-based applications. Previous studies explore vibration signals in gesture recognition[22][42], keystroke detection[24][2], and near-field communication[32][31]. Most recently, VibID[43] actively generates vibration signals on a smartwatch to sense the physical characters for authenticating users. However, this approaches is designed only for smartwatch scenario with the device fixed to a certain area of body. Any movements of the device could influence the performance of the authentication scheme. So it is not practical for on-touch user authentication scheme on mobile devices.

Different from these works, *TouchPass* leverages active vibration signals to build a behavior-irrelevant on-touch user authentication system for smartphones, which can effectively resist both mimic and replay attacks, while providing satisfactory user experiences.

## 10 CONCLUSIONS

In this paper, we propose a behavior-irrelevant on-touch user authentication system on smartphones, *TouchPass*, which leverages active vibration signals to extract physical characters of touching fingers for identifying individuals. *TouchPass* first extracts features that mix physical characters of touching fingers and behavioral biometrics of touching from active vibration signals. Then, a Siamese network-based architecture is designed to reconstruct the extracted features to behavior-irrelevant features, and a behavior-irrelevant user authentication model is further built leveraging knowledge distillation. Our extensive experiments demonstrate that *TouchPass* can accurately authenticate users and defend various attacks.

## ACKNOWLEDGMENT

This research was sponsored by NSFC (No.61772338).

# REFERENCES

[1] C. Bo, L. Zhang, X.-Y. Li, Q. Huang, and Y. Wang. Silentsense: silent user identification via touch and movement behavioral biometrics. In *Proc. MobiCom'13*, pages 187–190. ACM, 2013.

[2] W. Chen, M. Guan, Y. Huang, L. Wang, R. Ruby, W. Hu, and K. Wu. Vitype: A cost efficient on-body typing system through vibration. In *Proc.SECON'18*. IEEE, 2018.

[3] W. C. Chen, L. Chen, Y. Huang, X. Zhang, et al. Taprint: Secure text input for commodity smart wristbands. In *Proc. MobiCom'19(Preprint)*, 2019.

[4] D. G. Childers, D. P. Skinner, and R. C. Kemerait. The cepstrum: A guide to processing. *Proceedings of the IEEE*, 65(10):1428–1443, 1977.

[5] S. Chopra, R. Hadsell, Y. LeCun, et al. Learning a similarity metric discriminatively, with application to face verification. In *Proc.CVPR (1)'05*. IEEE, 2005.

[6] I. Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE transactions on information theory*, 36(5):961–1005, 1990.

[7] A. De Luca, A. Hang, F. Brudy, C. Lindner, and H. Hussmann. Touch me once and i know it's you!: implicit authentication based on touch screen patterns. In *proc. CHI'12*, pages 987–996. ACM, 2012.

[8] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet. Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):788–798, 2011.

[9] T. Feng, Z. Liu, K.-A. Kwon, W. Shi, B. Carbunar, Y. Jiang, and N. Nguyen. Continuous mobile authentication using touchscreen gestures. In *Proc. HST'12*, pages 451–456. IEEE, 2012.

[10] T. Feng, J. Yang, Z. Yan, E. M. Tapia, and W. Shi. Tips: Context-aware implicit user identification using touch screen in uncontrolled environments. In *Proc. HotMobile'14*, page 9. ACM, 2014.

[11] M. Frank, R. Biedert, E. Ma, I. Martinovic, and D. Song. Touchalytics: On the applicability of touchscreen input as a behavioral biometric for continuous authentication. *IEEE transactions on information forensics and security*, 8(1):136–148, 2013.

[12] A. P. French. *Vibrations and waves*. CRC press, 2017.

[13] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[14] M. Inalpolat and A. Kahraman. A theoretical and experimental investigation of modulation sidebands of planetary gear sets. *Journal of sound and vibration*, 323(3-5):677–696, 2009.

[15] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[16] ITRC. Lost electronic devices can lead to data breaches. [online]. avaliable: https://www.idtheftcenter.org/lost-electronic-devices-can-lead-to-data-breaches/, 2015.

[17] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti. Filterbank-based fingerprint matching. *IEEE transactions on Image Processing*, 9(5):846–859, 2000.

[18] H. Khan, U. Hengartner, and D. Vogel. Targeted mimicry attacks on touch input based implicit authentication schemes. In *Proc. MobiSys'16*, pages 387–398. ACM, 2016.

[19] G. Koch, R. Zemel, and R. Salakhutdinov. Siamese neural networks for one-shot image recognition. In *Proc.ICML'15*, 2015.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[21] A. Kumar and A. Passi. Comparison and combination of iris matchers for reliable personal authentication. *Pattern recognition*, 43:1016–1026, 2010.

[22] G. Laput, R. Xiao, and C. Harrison. Viband: High-fidelity bio-acoustic sensing using commodity smartwatch accelerometers. In *Proc.UIST'16*. ACM, 2016.

[23] L. Li, X. Zhao, and G. Xue. Unobservable re-authentication for smartphones. In *Proc. NDSS'13*, volume 56, pages 57–59, 2013.

[24] J. Liu, Y. Chen, M. Gruteser, and Y. Wang. Vibsense: Sensing touches on ubiquitous surfaces through vibration. In *Proc. SECON'17*. IEEE, 2017.

[25] J. Liu, C. Wang, Y. Chen, and N. Saxena. Vibwrite: Towards finger-input authentication on ubiquitous surfaces via physical vibration. In *Proc.CCS'17*. ACM, 2017.

[26] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

[27] R. Martin. Noise power spectral density estimation based on optimal smoothing and minimum statistics. *IEEE Transactions on speech and audio processing*, 9(5):504–512, 2001.

[28] W. Meng, W. Li, L. Jiang, and L. Meng. On multiple password interference of touch screen patterns and text passwords. In *Proc. CHI'16*, pages 4818–4822. ACM, 2016.

[29] R. Morris and K. Thompson. Password security: A case history. *Communications of the ACM*, 22(11):594–597, 1979.

[30] V. Romanuke. Appropriate number and allocation of relus in convolutional neural networks. *Naukovi Visti NTUU KPI*, (1):69–78, 2017.

[31] N. Roy and R. R. Choudhury. Ripple {II}: Faster communication through physical vibration. In *Proc. NSDI'16*, pages 671–684, 2016.

[32] N. Roy, M. Gowda, and R. R. Choudhury. Ripple: Communicating through physical vibration. In *Proc.NSDI'15*. USENIX, 2015.

[33] A. Serwadda and V. V. Phoha. When kids' toys breach mobile phone security. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pages 599–610. ACM, 2013.

[34] W. E. Siri. The gross composition of the body. In *Advances in biological and medical physics*, volume 4, pages 239–280. Elsevier, 1956.

[35] I. Song, H.-J. Kim, and P. B. Jeon. Deep learning for real-time robust facial expression recognition on a smartphone. In *Proc. ICCE'14*, pages 564–567. IEEE, 2014.

[36] Y. Song, Z. Cai, and Z.-L. Zhang. Multi-touch authentication using hand geometry and behavioral information. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 357–372. IEEE, 2017.

[37] M. Spoonauer. iphone x face id slower than touch id. [online]. avaliable: https://www.tomsguide.com/us/iphone-x-face-id-speed-up,news-26060.html, 2017.

[38] X. Suo, Y. Zhu, and G. S. Owen. Graphical passwords: A survey. In *Proc. ACSAC'05*, pages 10–pp. IEEE, 2005.

[39] F. Tari, A. A. Ozok, and S. H. Holden. A comparison of perceived and real shoulder-surfing risks between alphanumeric and graphical passwords. In *Proc.SOUPS'06*. ACM, 2006.

[40] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. J. Lang. Phoneme recognition using time-delay neural networks. *Backpropagation: Theory, Architectures and Applications*, pages 35–61, 1995.

[41] P. Welch. The use of fast fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms. *IEEE Transactions on audio and electroacoustics*, 15(2):70–73, 1967.

[42] H. Wen, J. Ramos Rojas, and A. K. Dey. Serendipity: Finger gesture recognition using an off-the-shelf smartwatch. In *Proc.CHI'16*. ACM, 2016.

[43] L. Yang, W. Wang, and Q. Zhang. Vibid: user identification through bio-vibrometry. In *Proc. IPSN'16*, page 11. IEEE Press, 2016.

[44] L. Zhang, S. Tan, J. Yang, and Y. Chen. Voicelive: A phoneme localization based liveness detection for voice authentication on smartphones. In *Proc. CCS'16*, pages 1080–1091. ACM, 2016.