Taprint: Secure Text Input for Commodity Smart Wristbands

Wenqiang Chen
Shenzhen University
Shenzhen, China
chenwenqiang2016@email.szu.edu.cn

Lin Chen
Shenzhen University
Shenzhen, China
chenlin@email.szu.edu.cn

Yandao Huang
Shenzhen University
Shenzhen, China
huangyandao@email.szu.edu.cn

Xinyu Zhang
University of California San Diego
San Diego, USA
xyzhang@ucsd.edu

Lu Wang
Rukhsana Ruby
Shenzhen University
{wanglu,ruby}@szu.edu.cn

Kaishun Wu
Shenzhen University
Shenzhen, China
wu@szu.edu.cn

ABSTRACT

Smart wristband has become a dominant device in the wearable ecosystem, providing versatile functions such as fitness tracking, mobile payment, and transport ticketing. However, the small form-factor, low-profile hardware interfaces and computational resources limit their capabilities in security checking. Many wristband devices have recently witnessed alarming vulnerabilities, e.g., personal data leakage and payment fraud, due to the lack of authentication and access control. To fill this gap, we propose a secure text pin input system, namely Taprint, which extends a virtual number pad on the back of a user's hand. Taprint builds on the key observation that the hand "landmarks", especially finger knuckles, bear unique vibration characteristics when being tapped by the user herself. It thus uses the tapping vibrometry as biometrics to authenticate the user, while distinguishing the tapping locations. Taprint reuses the inertial measurement unit in the wristband, "overclocks" its sampling rate to extrapolate fine-grained features, and further refines the features to enhance the uniqueness and reliability. Extensive experiments on 128 users demonstrate that Taprint achieves a high accuracy (96%) of keystrokes recognition. It can authenticate users, even through a single-tap, at extremely low error rate (2.4%), and under various practical usage disturbances.

MobiCom '19, October 21–25, 2019, Los Cabos, Mexico

ACM ISBN 978-1-4503-6169-9/19/10…$15.00
https://doi.org/10.1145/3300061.3300124

CCS CONCEPTS

• **Security and privacy** → *Authentication*; • **Wearable device** → Text input.

KEYWORDS

Secure input; Authentication; Wearable Devices; Vibration Recognition

ACM Reference Format:
Wenqiang Chen, Lin Chen, Yandao Huang, Xinyu Zhang, Lu Wang, Rukhsana Ruby, and Kaishun Wu. 2019. Taprint: Secure Text Input for Commodity Smart Wristbands. In *The 25th Annual International Conference on Mobile Computing and Networking (MobiCom '19), October 21–25, 2019, Los Cabos, Mexico.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3300061.3300124

1 INTRODUCTION

Recently, wearable devices have gained momentum and witnessed a phenomenal growth in popularity. Gartner predicts that the revenue from wearable devices would exceed the smartphone market, reaching 61.7 billion revenue by 2020 [47]. Smartwatches and smart wristbands represent the dominant force in the wearable ecosystem, widely used as fitness trackers or smartphone companions, and more recently adopted in mobile payment, transportation ticketing, etc. As such devices become increasingly personalized, they carry a looming threat to impinge on users' privacy and security. One may question the necessity of securing these low-profile devices, which commonly generate insensitive fitness data. However, the raw sensor data has already been used by wearable apps to infer private activities, healthiness, etc., and many wearables can execute SMS messaging and online payment functions [39,40]. To ensure security, the wearable device itself must be able to authenticate the user and enforce access control.

Traditional authentication methods often rely on heavyweight hardware and user interfaces, which do not fit wear-

Figure 1: Taprint at a glance.

ables. For example, password keypads [1,2] often require a touchscreen, and are vulnerable to shoulder surfing and smudge attacks. Fingerprint[7] and face recognition[8] are vulnerable to forgery attack, and require specialized hardware which do not match wearables' cost and form-factor constraints. Certain behavioral activities such as hand gesture [17] and gait patterns [11] may be used to identify a user, but are still possible for adversaries to mimic.

In this paper, we propose Taprint, a lightweight wearable input method which can authenticate the user while simultaneously offering a keypad interface. As shown in Figure 1, Taprint extends a virtual 12-key numeric keypad on the 12 knuckles of a hand. To authenticate herself, a user simply needs to use a finger on the other hand to tap one key. This *single tap* suffices for authentication. In addition, the user can input numbers or even characters, as long as she has a typical smartphone keypad layout in mind (to map to the knuckles). Knuckles are natural "landmarks" on hand, requiring no projection or drawing.

The key premise behind Taprint is that the user's tapping-on-knuckle represents a consistent feature, which can be sniffed by the wristband's inertial measurement unit (IMU). We investigate this property through a model, which takes into account the hand biometry and finger tapping behavior. We further show through extensive experiments that the feature is unique and diverse across tapping positions. More importantly, the tapping is *unforgeable*, i.e., adversaries cannot reproduce the feature even if they know the authentic user's tapping position. Therefore, when the user types on the virtual keypad, Taprint can essentially *continuously authenticate* each of the user's tapping input.

To make Taprint reliable and usable, we need to address several key challenges. First, the IMU may be affected by arm motion and other ordinary activities, which corrupt the vibration feature from tapping. Second, unlike active vibrations from modulated acoustic signals, tapping signals are passive and unmodulated, comprised of a variety of frequencies which make the pattern matching much more challenging [30]. Third, the hand biometry and tapping behavior may vary even for the same user over time.

To cope with these challenges, we first denoise the vibration caused by user activities. Furthermore, we propose a fine-grained feature extraction mechanism to enhance the uniqueness of vibration features from different tapping positions, and maintain consistency of the vibration features at the same position. We also design a density-based one-class classifier, namely DenID, to distinguish different tapping vibration by different users. Last but not the least, a calibration scheme including a real-time manual calibration and a multi-threshold self-calibration is designed, thus enabling adaptation to varying typing behaviors and real-time feature update when error occurs.

We build Taprint as a prototype application for the Android smartwatch. Our implementation achieves real-time secure input without noticeable latency. To acquire IMU readings with fine time resolution, we modify the OS kernel of a commodity smartwatch and configure the sensor registers to improve its sampling time by 5×. We further implement two use cases: a one time validation app, and a typing app. An *demo video* of Taprint is available at https://youtu.be/4tfMuYc_AMo. To evaluate the performance of Taprint, we have recruited 128 users and repeated multiple validation experiments over one month. The results show that Taprint can correctly identify a user with a low error rate of 2.4%. It recognizes the 12 keys with a mean accuracy of 96%. We have also conducted an extensive evaluation of Taprint's robustness under various disturbances, such as wearing position of wristband, tapping strengths, user states and different environment (e.g., on the airplane). We have further validated Taprint's resilience against common attacks (e.g., adversaries tapping on the authorized user's hand while she is sleeping).

The remainder of this paper is organized as follows. In Section 2, we first briefly introduce the vibration model in our case and presents the feasibility study of using vibration profile to characterize users in Section 3. In Section 4, we provide the system overview of this work and define four threat model. Then, Section 5 explains how Taprint detects the vibration of a tap. We introduce the fine-grained vibration recognition mechanism for user characterization in Section 6 and a calibration mechanism in Section 7. Section 8 explains the implementation detail, followed by comprehensive experimental evaluations and user studies of our system. We discuss the related work in section 9 and conclude the paper in Section 10.

2 VIBRATION MODEL FOR ON-BODY TAPPING

Typically, the mathematical model of complex vibration systems such as human body is intractable. For the sake of simplicity, we first construct a single degree-of-freedom model as shown in Figure 2 to illustrate the basic mechanisms. In this model, the mass element is represented by a rigid

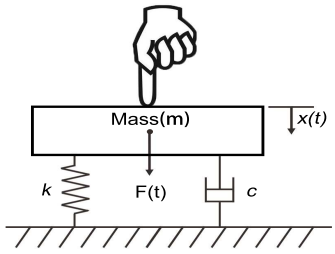


Figure 2: The vibration model of a tap.

body with constant mass m, the spring element is defined as a spring with negligible mass and constant k, and the damping element is represented by a damper with damping coefficient c.

When an external force is applied to the rigid body, vertical displacement occurs. According to the Newton's second law of motion, we have,

$$F(t) = ma(t) + kx(t) + cv(t), \tag{1}$$

where $F(t)$ is the external force, $v(t)$ is the speed, $x(t)$ is the vertical displacement, c is the damping coefficient, k is the spring constant and m is the mass. The relation in (1) can further be explained as,

$$F(t) = m\frac{d^2x(t)}{dt^2} + kx(t) + c\frac{dx(t)}{dt}. \tag{2}$$

The vibration during a finger tapping operation can be separated into two phases. In the first phase, the finger has a transient contact to the rigid body within the duration of seconds, which is considered to be a forced vibration with constant force $F(0)$. After the disturbance of the initial transience, in the second phase, the contact between the finger and the rigid body disappears, which leaves the system to vibrate on its own and this is called free vibration. In the forced vibration phase, after applying the Fourier transform to both sides of (3), we have,

$$\frac{F(0)}{jw}(1 - e^{-jw\Delta t}) = -w^2mX(w) + kX(w) + jwcX(w), \tag{3}$$

that is,

$$X(w) = \frac{1 - e^{-jw\Delta t}}{-\frac{jm}{F(0)}w^3 - \frac{c}{F(0)}w^2 + \frac{jk}{F(0)}w}, \tag{4}$$

where $X(w)$ is the spectrum of the vertical vibration signal and w is the frequency. We then investigate the vibration in the horizontal direction. During the horizontal propagation of a vibration signal from the tapped location to the sensor, the vibration suffers from attenuation, and the corresponding model can be stated as follows,

$$y(t) = x(t)e^{-\alpha d}, \tag{5}$$

where $y(t)$ is the vertical displacement at the position that the vibration has propagated to, $x(t)$ is the vertical displacement at the finger tapped position, d is the propagation distance, and α is the attenuation coefficient. Again, after applying the Fourier transform to both sides of (6), we have,

$$Y(w) = X(w)e^{-\alpha d}. \tag{6}$$

Note that α is related to the propagation medium. Wave propagation in body is dispersive by nature, which implies that different frequencies propagate with different attenuation coefficients at different velocities. Roughly speaking, the attenuation is small when the vibration signal propagates through the hard bone, whereas the attenuation is large through the soft tissue. Therefore, vibration waves generated at different positions on the hand back result in different values of α and d, which make the vibration signals unique at different positions. After putting (4) into (6), we obtain,

$$Y(w) = \frac{(1 - e^{-jw\Delta t})e^{-\alpha d}}{-\frac{jm}{F(0)}w^3 - \frac{c}{F(0)}w^2 + \frac{jk}{F(0)}w}. \tag{7}$$

For the same location of the human body, m, c and k are stable and belong to the same biometric feature. Moreover, as part of the tapping habits of users, $F(0)$ and Δt have certain stability. However, we know that the attackers can arbitrarily adjust $F(0)$ and Δt. In (8), the numerator is an exponential function of α and d, and the denominator is a polynomial function of $\frac{m}{F(0)}$, $\frac{c}{F(0)}$ and $\frac{k}{F(0)}$.

Note that w is a vector rather than a scalar value. For all the frequency points of w, unless the four parameters ($\frac{m}{F(0)}$, $\frac{c}{F(0)}$, $\frac{k}{F(0)}$ and Δt) of a tapping vibration signal in a certain position are equal to the parameters of the vibration signal in another position at the same time, the frequency spectrum $X(w)$ of the corresponding two positions cannot be the same. Among them, m, c and k vary from person to person [42]. *Therefore, the tapping force of different people at a fixed position can uniquely be identified, which can be leveraged for authentication.*

3 FEASIBILITY STUDY

In this section, we conduct experiments to validate the insights from the foregoing analysis, particularly the uniqueness of the tapping feature and its diversity among locations/users.

3.1 Uniqueness of Vibration Signals

In this experiment, we control the variables $F(t)$, Δt and d, by using a motor to vibrate twice at the same position on the left-hand back of two users, respectively, so as to investigate the signal profile in the time domain. We plot the vibration signals of motion reading in Figure 3(a) from the Z-axis of the accelerometer in a commodity smartwatch. We can observe that the vibration waveforms from the same person are consistent and differ among the two users.

We then repeat the experiment, but substitute the motor by users' fingers. In Figure 3(b), we can observe that, for the same user (S1), the profile of two tapping signals match

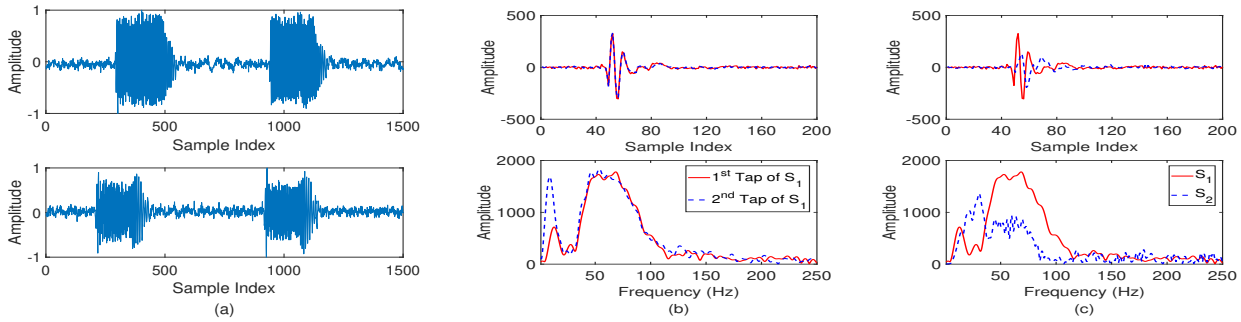


Figure 3: The vibration profile of two subjects in the time and frequency domains, where (a) is generated by a motor, and (b) and (c) are generated by a tapping force.

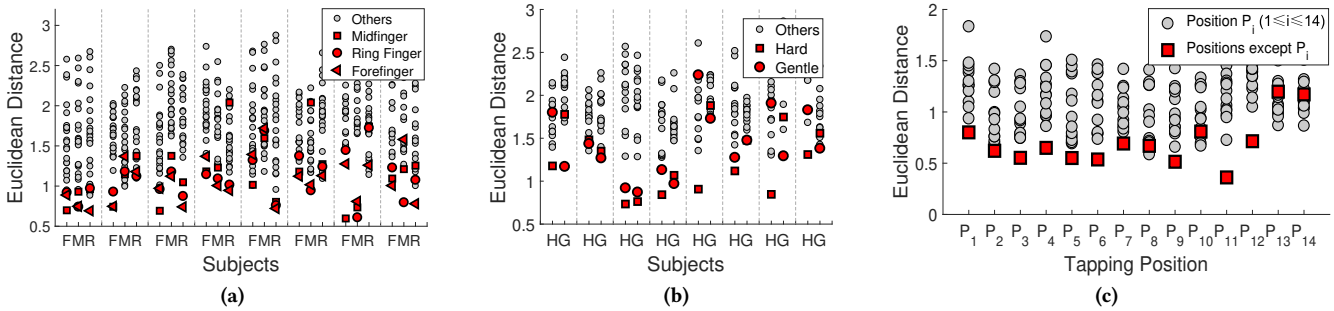


Figure 4: The Euclidean distance of samples generated by different (a) fingers, (b) tap strengths, and (c) locations.

each other well in both time and frequency domains, which indicates the consistency of m, c and k of the same person. In contrast, the profile of the two subjects differ significantly (Figure 3(c)). This verifies the feasibility of using the values of m, c and k to distinguish the legitimate user from attackers.

3.2 Vibration of Different Fingers

One natural question here is: does the distinction comes from the hand back or the finger? To answer this question, we asked 8 participants to tap on 14 knuckles of their left hands for 30 times using the forefinger, the middle fingers and the ring finger, respectively. In Figure 4(a), 'F' means that we take the mean value of 20 samples generated by the forefinger as the datum point. Similarly, 'M' and 'R' denote the middle finger and the ring finger, respectively. We calculate the average Euclidean distance between each datum point and other samples. Overall, we have three observations: 1) the Euclidean distance of most authorized samples is smaller than that of the unauthorized samples, 2) the Euclidean distance of different fingers is about the same, and 3) some of the authorized samples mix with the unauthorized ones, which might be caused by the variation of the initial tapping force. In conclusion, *the distinct signals are generated mainly by hand back associated with the values of m, c and k rather than a tapping finger itself.*

3.3 Vibration of Different Strengths

We now intentionally use different tapping forces and investigate the signal variation. We asked 8 participants to tap on 12 knuckles of a hand for 30 times with both heavy and gentle force. In Figure 4(b), 'H' means that we take the mean value of 20 samples generated by heavy force as datum point. Similarly, 'G' denotes the gentle force. We observe that the authorized samples with different tapping force may mix with the unauthorized ones. Therefore, the tapping force is an interference factor that needs to be resolved. We will design a calibration scheme to tackle this challenge (Section 7).

3.4 Vibration at Different Locations

In this experiment, we study the situation when the tapping positions change. Specifically, we asked one participant to tap on the 14 knuckles on the hand back each for 30 times. Then, we take the mean value of 20 samples collected from position P_i as datum point and calculate the average Euclidean distance between each datum point and other samples. The result (Figure 4(c)) shows that the Euclidean distance of samples on the same location is generally smaller than that of the others except 4 outliers. We conjecture that the instability of initial tapping force results in these outliers. Nevertheless, it is evident that the value of $y(t)$ is influenced by the variation of distance from the tapped position to a sen-

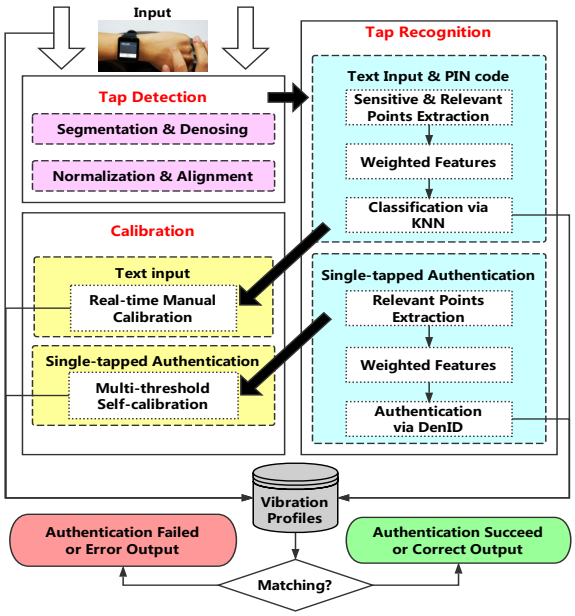


Figure 5: The workflow of Taprint.

sor. Such diversity allows Taprint to distinguish the tapping locations, thus creating a small pinpad.

4 OVERVIEW OF TAPRINT

4.1 System overview

Taprint consists of three major components, with the following functionalities.

(1) Tapping vibration detection. Taprint first uses the energy-based threshold to segment the signals, and then the General Cross Correlation (GCC)-based algorithm to align the segments. Then, Taprint removes the noise caused by body motion. Finally, the Z-score normalization is used to reduce the variation of tapping force.

(2) Fine-grained vibration recognition. Taprint uses fine-grained features to distinguish different tapping vibration on different knuckles of different users. We design two kinds of weighted features for inputting numbers (text input and password input), and inputting single-tap for authentication, respectively. Given these features, we further customize the nearest neighbor method and a density-based one class classifier to recognize the inputs.

(3) Update and calibration. The vibration features may change due to user's tapping habit and tapping position variation. Taprint thus incorporates a simple real-time manual calibration and a multi-threshold self-calibration mechanism to adapt to these changes.

Figure 5 describes the work-flow of Taprint. In the initial training stage, a series of pre-processing is conducted that includes segmentation, denoising, normalization and alignment. Afterwards, fine-grained features are extracted to build the training vibration profile. When a new tapping vibration is detected, the number input function (text input & password input) extracts features weighted by their relevance, and then classifies them via the nearest neighbor classifier. The single-tap authentication works in a similar way, except that it uses different feature processing algorithms.

4.2 Threat Model

We consider the following attacks that may threaten the proposed authentication functionalities.

Zero-effort Attack. The attacker attempts to find a potential tapping location that can generate similar vibration signals to bypass the authentication, by tapping randomly without knowing either the PIN code or the location of the single-tap lock.

Credential-aware Attack. The attacker obtains the legitimate user's credentials, including the PIN code and the location of the single-tap lock. However, attacker does not know the behaviors of the legitimate user such as tapping force, tapping angle, gesture, contact duration.

Observer Attack. The attacker possesses the prior knowledge of legitimate user's PIN code and the location of single-tap lock, and tries to imitate the behavior of the legitimate user based on stealthy observations via shoulder surfing or camera recording.

Intimate Attack. The attacker, who may have an intimate relationship with the legitimate user, acquires knowledge of the legitimate user's PIN code and the location of the single-tap lock. The attacker attempts to pass the authentication by tapping on the legitimate user's hand when she is unaware of it (e.g., during sleeping).

5 DETECTION OF TAPPED VIBRATION SIGNAL

5.1 Segmentation and Denoising

To detect whether a tapping occurs, Taprint uses an energy-based approach, and segments the raw vibration signal into small windows. For each segment, the signal energy is calculated and forwarded to the tapping detection module. In terms of the cutoff point, we set it to 0.1 s after the start point as the duration of a keystroke tapping signal is usually around this value [41].

Human mobility such as walking often causes body vibration, which needs to be denoised. Based on the short time Fourier analysis, we observe that the vibration caused by human mobility is mostly less than 10 Hz, and hence a 20 Hz Butterworth high pass filter is sufficient to remove the noise from the captured vibration signal. Through this filter, the direct current component such as gravity can also be removed.

In the login authentication process, the users need to turn

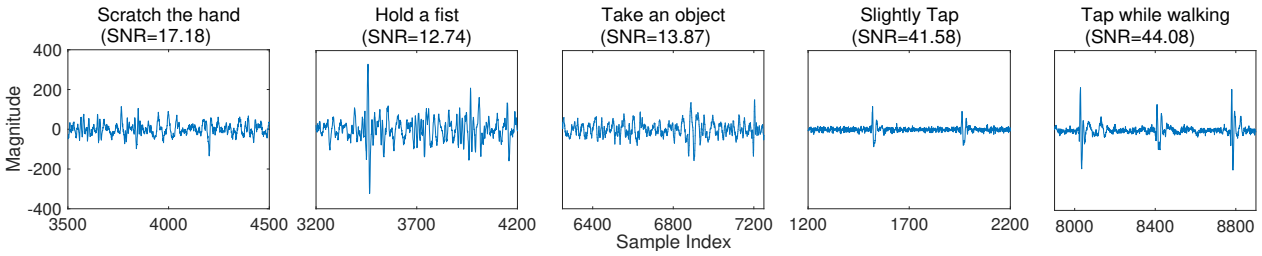


Figure 6: Comparison of SNR resultant from five different actions with a 20 Hz highpass filter.

on the touchscreen first. When a user types on a laptop keyboard or washes dishes, he/she may not turn on the touchscreen of the smartwatch. However, in the text input process, some actions when typing on the back of one's hand (e.g., picking up objects or scratching hands) may trigger false positives. Examples are shown in Figure 6, which plots the vibration signals detected during 5 types of user activities. Note that these vibration signals are all filtered by a 20 Hz Butterworth high pass filter. In this figure, we observe that the signal to noise ratio (SNR) of finger taps (even with slight taps) are obviously higher than that with other actions. Thus, we simply segment a tapping-induced vibration signal when the signal SNR exceeds a certain threshold (default to 20 dB).

5.2 Normalization and Alignment

Afterwards, Taprint normalizes the magnitude of both signals using the Z-score normalization technique, which is standardized based on the mean and standard deviation of the original data. The processed data conforms to the standard normal distribution, i.e., the mean value is 0 and the standard deviation is 1.

Once the tapping event is detected, Taprint can only achieve a rough estimate of the starting point of the tapping induced vibration signal. Consequently, Taprint aligns signals by finding the time shift with the GCC algorithm [46]. Note that Taprint does not utilize the Dynamic Time Warping (DTW) algorithm since it removes the timing information critical to the signal's pitch and requires much more intensive computation.

6 FINE-GRAINED VIBRATION RECOGNITION

In this section, we describe how Taprint extracts fine-grained features based on proposed position-sensitive points and position-relevant points. The reason why we do not adopt more sophisticated machine learning algorithms which may extract features automatically will be discussed in section 6.2.

6.1 Extracting Weighted Features

Due to the dispersion of the vibration signal, the tapping induced vibrations in different positions may have different spectra. Accordingly, we choose the amplitude spectral density (ASD) [38] as the basic feature. For a given segment of vibration signals, the ASD can be simply obtained through FFT. However, different frequency bins may have different contributions to the uniqueness of features (Figure 7). First, if the vibration amplitude of some frequency points shows notable differences at different tapping positions, these frequency points can better distinguish the vibration position. We refer to these frequency points as *position-sensitive frequency points*. On the other hand, if the amplitude discrepancies of certain frequency points at the same position are small, these frequency points can better describe the characteristics of this location. We refer to these frequencies as *position-relevant frequency points*. By weighting the position-sensitive frequency points or position-relevant frequency points from other frequency points, the characteristics of the vibration signal can be amplified. We refer the resulting feature points as *weighted features*.

Keypad input. In the keypad input scenario which needs to distinguish 12 tapping positions, the position-sensitive frequency points and the position-relevant frequency points jointly provide the feature information. We use the Fisher score technique to identify position-sensitive frequency points and position-relevant frequency points. The Fisher Score is designed as the weight for features and is given by

$$F_r = \frac{\sum_{i=1}^{l} n_i (u_i - u)^2}{\sum_{i=1}^{l} n_i \delta_i^2}, \tag{8}$$

where r denotes the feature number of each dimension, n_i denotes the number of samples at the ith class, u_i and δ_i^2 denote the mean and variance of the samples at the ith class, respectively. Moreover, u denotes the average of all classes associated with the dimension feature and l is the total number of classes.

Single tap authentication. In the single-tap lock authentication scenario, the system only needs to identify whether the vibration is generated by a certain fixed position. Consequently, the position-relevant frequency points exhibit better characteristics. We can set a weight for each frequency point according to the following relation

$$w = \frac{max(E(X_i)) - E(X_i)}{\sum max(E(X_i)) - E(X_i)}, \tag{9}$$

where $E(.)$ is the variance. With this weight design, frequency points with small variances represent positional correlations, and hence are given larger weights. On the other hand, frequency points with high variances are considered to be weak descriptive features and are multiplied with smaller weights.

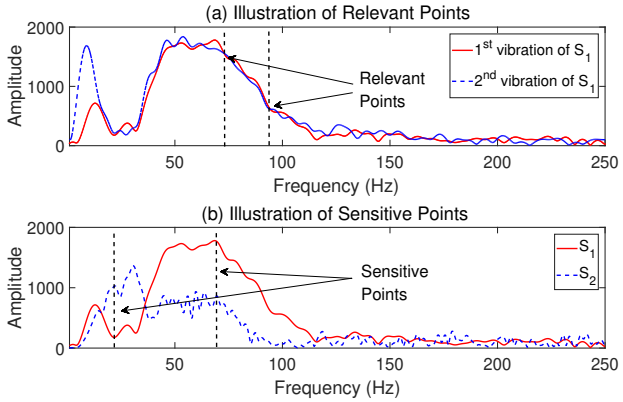


Figure 7: Illustration of relevant and sensitive points. The dotted black line shows the example points.

Intuitively, raw vibration signals exhibit time-domain characteristics of tapping operations at different locations. As shown in Figure 3, the tapping of the original signal differs between two positions. Therefore, before applying the weighted operation, we utilize the fusion features with ASD of the raw vibration data.

6.2 Recognizing Tapping Position

Keypad input. After the extraction of fine-grained features, Taprint runs a nearest-neighbor-based pattern matching algorithm that compares the extracted features with those in the training set. The training data with minimum distance is declared as the current key position and displayed on the user interface. In the simplest form, we use the Manhattan distance as the metric of comparison.

Note that Taprint cannot use sophisticated pattern recognition methods, like neural network, or support vector machines. Because these algorithms require substantial training operations to construct the classification model, and often incurs formidable computational cost which hampers real-time tapping recognition. In addition, Taprint needs to update the training set as a user taps. Performing such real-time update is simple for nearest-neighbor methods, but infeasible for computationally intensive machine learning algorithms.

Single tap authentication. To realize one-time validation with only one tap, the simplest form is to draw the boundary of the sample distribution. If the test sample is out of the edge, it is regarded as the unauthorized sample. Unfortunately, our examination of collected data shows that, most of the data from the same position has an irregular distribution instead of a circle distribution which leads to difficulty of drawing the edge of the samples distribution. Interestingly, we observe that the density of a sample set from the same position of a hand of the same user remains stable, which means that the pairwise distance between samples is similar. Based on this observation, we propose a *density-based one-class classifier*, called DenID. An unauthorized test sample is far away from the authorized sample sets. Thus, if an unauthorized test sample is added to the authorized samples set, the density of the sample set decreases. More specifically, the density of a samples set can be represented by the following relation,

$$D_1 = \frac{\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} d_{ij}}{C_N^2}, \tag{10}$$

where d_{ij} is the distance between sample i^th and j^th. N is the number of samples in the training set. When a test sample is added to the samples set, the density of the new set becomes,

$$D_2 = \frac{\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} d_{ij} + \sum_{i=1}^{N-1} d_{i(N+1)}}{C_N^2 + N}, \tag{11}$$

where $N+1$ is the index of the test sample. Finally, if $D_1 > D2 + th$, the test sample is considered as unauthorized. Note that th is a threshold. A larger threshold means more false acceptance rate (FAR) while a smaller threshold means more false rejection rate (FRR).

7 CALIBRATION

We design a calibration mechanism to ensure robustness against tapping behavior variations, location deviations, and feature changes over time.

7.1 Real Time Manual Calibration

The vibration features may change due to user's tapping habit and tapping position variation. To cope with this challenge, we introduce a real-time manual calibration mechanism for the keypad use case. To sustain false recognitions, Taprint displays top 2 candidate keys on the touch screen when a user types. Users can touch to choose the candidate key showing on the touchscreen and update this key to the training set when an error key occurs. Owing to the simple nearest neighbor matching algorithm, Taprint is able to update the training set over time without the hassle of retraining a model.

It is worth noting that we do not update the initial training set but update a copied training set in the memory. This implies that once the system is used, the copied training set in the memory is cleared. There are several motivations behind this design. First, if we update the initial training set continuously over time, users may update an error key into the initial training set incautiously, which pollutes the initial training set. Second, adversaries may insert the unauthorized samples into the training set. Based on these observations, we also leverage a reset mechanism for the training set when users feel that the initial training set does not work anymore over time and requires one-time validation. This is similar to the reset mechanism for the fingerprint sensor on smartphones.

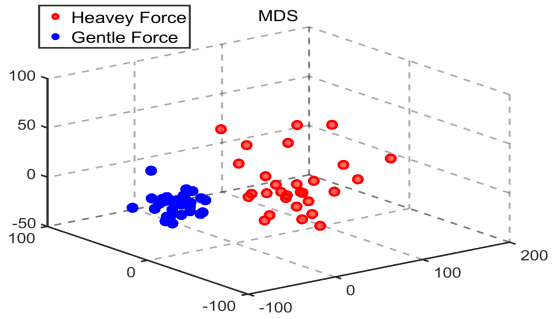


Figure 8: Illustration of sample distribution of different tap strength using the MDS technique.

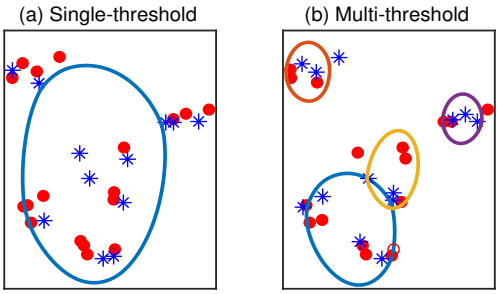


Figure 9: Sample result of the multi-threshold calibration mechanism.

7.2 Multi-Threshold Self Calibration

In case of single-tap authentication, Taprint cannot adopt the real-time update mechanism as adversaries may update and insert the unauthorized samples into the training set. Therefore, it guides users to input a variety of samples into the initial training set with different tapping force and slight deviations from the target position. This is again inspired by the fingerprint sensor calibration on smarpthones, which requires a user to touch the center of the finger several times and then touch the surrounding finger multiple times, so that the system obtains different variations of training samples. However, as various training samples are divided into different clusters, they have disparate densities, which brings in a great complexity into the DenID algorithm. For example, a cluster with heavy tapping force has a small density whereas a cluster with gentle tapping force has a higher density, as shown in Figure 8. Hence, we are unable to determine a threshold to draw an edge of the samples distribution of multiple clusters. Therefore, we further propose a multi-threshold calibration mechanism to draw multiple boundaries of different clusters. Figure 9 compares a sample outcome between the single threshold and the multi-threshold mechanism. Note that, we use the Multiple Dimensional Scaling (MDS) technique for visualization purpose and reduce the dimension in Figure 8 and Figure 9. We now proceed to describe the multi-threshold calibration.

Rough Clustering: First, we need to find mutations in the distance between two samples. This sudden change in pair distance is the first basic idea of distinguishing different clusters. More specifically, we find two samples with the closest distance in the training set U at the beginning, and then add them into the visited set R. In order to find the next sample P_i, we design

$$P_i = \arg\min_{P_i \in \complement_u R} D_{iR}, \tag{12}$$

where D_{iR} is the distance between P_i and R. We define

$$D_{iR} = mind_{ij} \tag{13}$$

where d_{ij} is the distance between P_i and P_j, $P_i \in R$, $P_j \in \complement_u R$. We store D_{iR} into array θ. Going through all the samples, we set the threshold to find the location of the mutated distance. We set the threshold to $\mu + \sigma$. μ and σ are the mean and variance of array θ, respectively. When the distance between two samples is less than this value, we consider that the corresponding samples belong to the same cluster. Besides, if the distance value in the distance array D is smaller than the previous value, we consider that the corresponding samples should be in the same cluster.

Fine Clustering: To determine whether two clusters are similar, we use the density concept. When two clusters are merged and the density is hardly changed, we consider that the resultant two clusters are similar. Specifically, before the merging operation, the density of cluster $C1$ is $d1$, and the density of the new cluster after merging $C1$ and $C2$ is $d2$. When $d2/d1$ is less than a certain threshold, these two clusters are considered as similar enough to be merged. In accordance with our experience, we set the threshold to 1.05. While merging two clusters, it is stipulated that a cluster with a small number of samples should be combined with a cluster with a large number of samples. In the merging process, one cluster may be similar to the other two clusters. In this case, the cluster is preferentially merged to a cluster that is most similar to that cluster.

8 EVALUATION

8.1 Implementation and Experimental Setup

We have implemented Taprint as a standalone application program on the LG G Watch W100 with a 37.9×46.5 mm screen, 1.2 GHz Quad-Core processor, a RAM with 512 MB, 400 mAh of battery and Android Wear 1.0.5. Taprint utilizes the built-in accelerometer and gyroscope (InvenSense MPU6515) in the smartwatch, and acquires the motion readings through existing Android Wear APIs to detect the finger tapped vibration. The maximum sampling rate through the APIs is only 100 Hz. Through the short time Fourier frequency spectral analysis, we observe that the frequency features of vibration signals caused by finger tapping should be from 10 Hz to 250 Hz, which requires at least 500 Hz Nyquist sampling rate. To address this challenge, we have modified the Linux kernel on the smartwatch. Specifically, our kernel

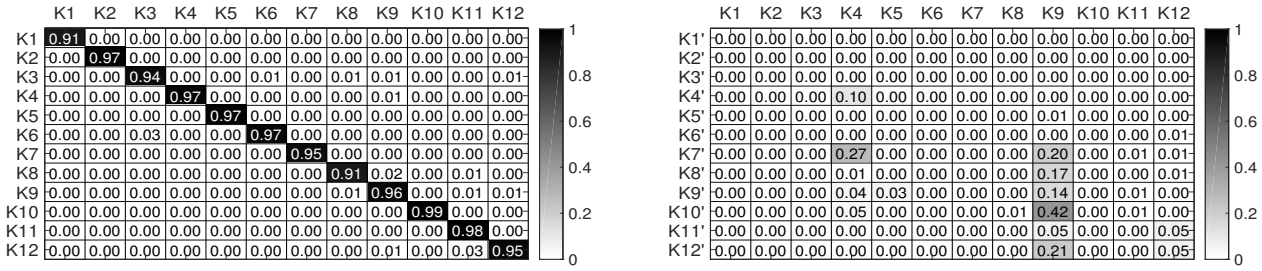


Figure 10: Confusion matrix of 12 keys, where (a) both the training and test samples are from the legitimate users (b) the training samples are from the legitimate users while the test samples are from the attackers.

driver interface configures the IMU registers to realize the documented high-speed sampling allowable by the low-layer hardware. Noted that the IMU we used in LG G Watch is used in many other popular smartwatches, including the LG Watch Urbane, Moto 360, Samsung Gear 2 and Gear Fit.

We have implemented all the components of our system including signal detection, feature extraction and the recognition algorithm on a COTS smartwatch. A user interface was designed to guide the user to train and use Taprint. We have also implemented a T9 input method based on a trie [48] to support English input, which maps 12 numbers to 26 characters. For the current implementation, the average end-to-end latency is 232 ms with a standard deviation of 26.5 ms from inputting the key to the display of the input. The initial training process lasts for about 1 mins. We measure the power consumption of the smartwatch using "Battery Historian" from Google. Specifically, three states are measured: 1) idle with the display on, 2) Taprint with power on but without key input, 3) Taprint with power on and continuous tap input. Since the platform is only able to measure the percentage of the battery consumption, we record the time duration for consuming 1% battery for each state. The resulting time duration in each state is 215 s, 188 s, 180 s, respectively. Given the 400 mAh battery capacity and the 3.7 V working voltage, we calculate the resulting power consumption of each state, which is 247.8 mW, 283.4 mW, 296 mW, respectively. Thus, Taprint only consumes an additional 48.2 mW level of power on top of the base power consumption. For comparison, we also conduct the measurement when running a step calculation application, resulting in a power consumption of 288 mW. The power consumption of Taprint is similar to the typical application running on the smartwatch.

We have recruited 128 participants (43 of them are female) from our university in the age range between [19, 26]. Besides, their body mass indexes (BMIs) are ranging from 17.16 (lean) to 29.28 (obese). To demonstrate the basic performance of Taprint, 113 participants were asked to tap on four random locations (4 knuckles) for the PIN code password, and 30 participants were asked to tap on all locations (12 knuckles), each for 30 times to generate the basic data set (with $113 \times 4 \times 30 + 30 \times 12 \times 30 = 24360$ samples in total). Furthermore, in order to examine the robustness of our system, some of the participants were asked to record the data under different experimental conditions, which will be specified in the following relevant sections. Each of the participants was designated as the legitimate user in turn to train the system, in the meanwhile the other subjects were set to be attackers for testing. The length of the training and test data sets are 20 samples and 10 samples for each key of a person, respectively. We repeated all the experiments for 10 times and calculated the mean value. The default experimental environment is a typical office room with around 44 dB ambient noise level. The smartwatch is worn on the left wrist in a comfortable manner with hand floating in the air.

Unless otherwise specified, all the experiments are launched based on the default setting discussed above. Note that the experiments were approved by the relevant institutes in our university.

8.2 Evaluation Metrics

VSR and AFR of PIN Code Authentication. The verification success rate (VSR) is defined as the success rate of inputting a complete PIN sequence by a legitimate user, while the attack failure rate (AFR) is defined as the failure rate of inputting a complete PIN sequence by an attacker.

ROC Curve of Single-tap Authentication. We define FAR as the ratio between the number of falsely accepted attacker samples and the total number of attacker test samples, and FRR as the ratio between the number of falsely rejected legitimate samples and the total number of legitimate test samples. We obtain the corresponding FAR and FRR pairs by adjusting the identification threshold and depict the receiver operating characteristic (ROC) curve, which shows the equal-error rate (EER) where the FAR is equal to the FRR.

8.3 Accuracy

In this section, we evaluate the baseline performance of Taprint which includes the verification of legitimate users while recognizing the users' tapping input. We also investigate the impact of different training set size and different sampling rates.

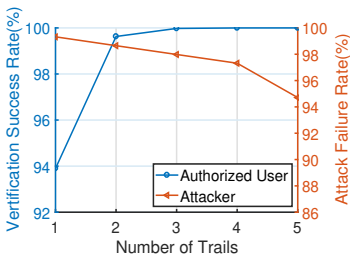


Figure 11: The VSR and AFR with multiple trials.

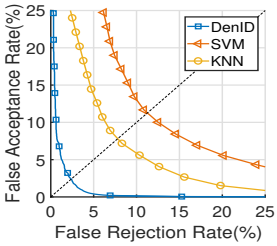


Figure 12: Sample ROC curves of single-tap authentication.

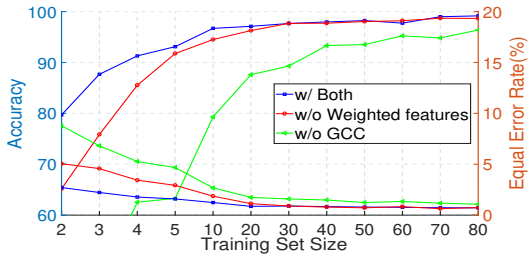


Figure 13: Impact of initial training set size.

8.3.1 Baseline classification accuracy. To mark a baseline accuracy of Taprint, we ask 30 participants to tap on twelve designated locations (see Figure 1) each for 30 times. The confusion matrix of recognition accuracy for 30 participants is shown in Figure 10(a). It demonstrates that Taprint obtains an average accuracy of 95.64% for twelve keys. In contrast, when the training and test samples are from different users as shown in Figure 10(b), the accuracy is very low, which makes it difficult for adversaries to type with Taprint. Note that there will always be an output due to the classification method even though the distances between training samples and the test sample are large. To improve the security, we set a threshold to prevent from showing a result when the distance is bigger than the threshold in the real time system (default to 20 in Euclidean Distance).

8.3.2 Verification accuracy.

PIN Code Authentication. Since a user need to input a complete PIN sequence to pass through the authentication, the verification success rate (VSR) of one trial is calculated as the product of the individual accuracy of 4 random keys. Figure 11 shows the VSR with respect to different numbers of trials. The VSR reaches 94% with a single trial, rises to around 99.5% with two trials and then it approaches nearly 100%. This indicates that a legitimate user can definitely pass through the authentication within 2 trails using the PIN code lock. Overall, the results show the effectiveness of the PIN code lock in authenticating legitimate subjects. The AFR is also provided in this figure, but the details will be elaborated in Section 8.4.

Single-tap Authentication. Figure 12 demonstrates the effectiveness of verifying a legitimate user through single-tap with ROC curve using a basic data set of 113 subjects. The single-tap location was set to a random location on 12 knuckles. On average, single-tap authentication obtains 1.29% false rejection rate (FRR) when the false acceptance rate (FAR) is 5%, which indicates that only 1.29% of the authorized subjects are rejected and 5% of the total subjects gain unauthorized access.

We also compare the outcome of our designed classifier DenID with that of the one class classifiers K-Nearest Neighbor (KNN) and Support Vector Machine (SVM). As shown in Figure 12, when applying DenID, Taprint obtains a much lower EER at 2.4% than that of traditional classifiers (at around 8% for KNN and 11% for SVM), which suggests that our feature optimization method is highly effective.

8.3.3 Impact of Training Set Size. In this experiment, 8 of the participants were asked to tap on all locations (12 knuckles) each for 100 times ($8 \times 12 \times 100 = 96000$ samples in total) in order to study the impact of different sizes of training sets. Figure 13 plots the resulting variance of accuracy as the number of initial training samples increases from 2 to 5 initially and then 5 and 10 afterward.

PIN Code Authentication. Figure 13 shows that the average classification accuracy is around 80% even with 2 initial training sample. The accuracy then escalates to above 98% on average when the training set size enlarges to 20. It can show only a marginal improvement on the further enlargement of training set size. We can also observe that the employment of both GCC and weighted features provides better performance.

Single-tap Authentication. As for the single-tap authentication, 2.7% equal error rate (EER) can be achievable on average with only 2 training instances for each user, and it further declines to below 1.0% when we enlarge the size of the training set to 20 or more. From the declining EER curve, we learn that a larger size of training set gives more information to the identifier and characterizes a user more precisely. Note that GCC and weighted features decrease the EER as shown in Figure 13.

8.3.4 Sampling Rate. Figure 14 (a) shows the average EER and classification accuracy at different sampling rates. The average EER decreases from 5.4% at 100 Hz to 1.9% at 500 Hz. Moreover, the average classification accuracy are 94% and 96.8% with the 100 Hz and 500 Hz sampling rate, respectively. We believe that more distinctive vibration features are incorporated owing to our implementation of sampling rate on the smartwatch. In Section 8.5, we will verify that the improved sampling rate also contributes to the robustness.

8.4 Security Analysis

In this section, we validate that our proposed authentication methods are secure under the 4 types of threats mentioned in Section 2.2.

Table 1: EER(%) and AFR(%) of four threat models with 20-sample training

Type of Attack	EER	AFR (1 trial)	AFR (5 trials)
Zero-effort Attack	0.80	99.92	99.60
Credential-aware Attack	2.40	99.65	98.27
Observer Attack	1.12	99.72	98.60
Intimate Attack	1.74	99.32	96.65

Zero-effort Attack. We have recruited another 20 participants from our university as attackers to attack the 113 subjects mentioned in Section 8.1. Each attacker was asked to randomly tap on his hand back for 40 times to generate the attack sample set. In Table 1, the results are consistent with our intuition that the random guesses are nearly impossible to pass through either the single-tap-based or the PIN code-based authentication even with 5 trials.

Credential-aware Attack. We conduct this experiment using the data collected from the 113 participants. Each participant was alternatively taken as the victim, and the remaining 112 participants playing as attackers. From Table 1, for PIN code authentication, there is a 99.65% chance for attackers to be blocked outside the system within a single trial, and the AFR only drops a little to 98.27% within 5 trials. For the single-tap authentication, the EER is 2.40%. The results show that our methods guarantee the security under credential-aware attack.

Observer Attack. We video-record the data collection process of 20 out of 113 participants, and demonstrate the videos to other 10 participants who act as attackers. The attackers were asked to mimic the tapping motion of victims. They were required to practice at least 10 times before generating 10 mimicry tap for each designated location. Under the observer attacks, our system also maintains high security (i.e., 98.6% five-trial AFR and 1.12% EER). Although the attackers tried their best to mimic the authentication motion of legitimate users, there are still intrinsic physiological features that are impossible to be identical among users.

Intimate Attack. We have asked 5 attackers to tap on the four designated locations on the hand back of victims (each for 10 times) to generate the intimate attack dataset. The results are shown in Table 1, which verifies that our method is secure to the intimate attack. We speculate that this result comes from the impact of behavioral variation which is difficult for attackers to mimic via their observation. Note that Taprint keep robust from the variations of behavioral usage based on the update and multi-threshold calibration. As shown in figure 4(a) and figure 4 (b), the points with the lowest Euclidean distance are always from the same users (red points). Note that the gray small circles are the distance from different users, which are always with bigger distance. Thus, even behaviors are different, the tapping vibration from the same person still keeps unique.

8.5 Robustness

In this section, we examine the robustness of our system under various disturbances in practical use cases. The experiments were conducted during one month and under different situations, such as walking, hand-wash, a noisy office, subways, and airplanes.

8.5.1 Strength of Tap. To examine how the result is affected by different initial tapping force, we have asked 15 participants to tap on each key 30 times both gently and heavily, resulting in 1920 responses [(4 keys $\times$ 15 users $\times$ 30 times $\times$ 2 ways).]

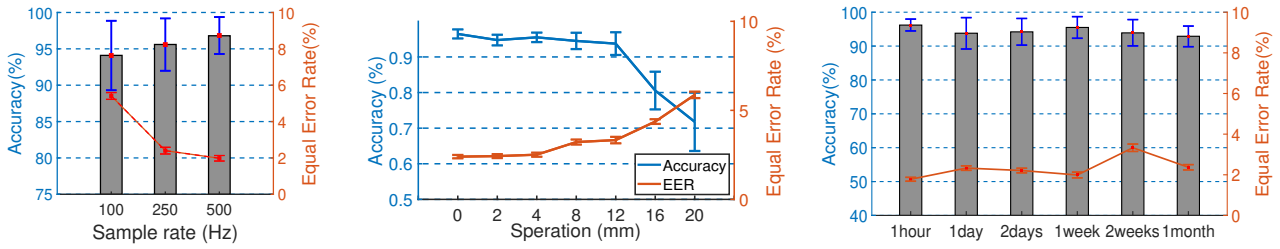
The results show that the classification accuracy drops to around 50% when the test tap force is different from the training tap force. Nonetheless, when our classifier is built with both "heavy" and "gentle" data, the accuracy remains the same level with the accuracy of using the same force. Therefore, we require users to apply different tap strength for the initial training set. However, for one-time validation of single-tap unlock, the EER is only 8% even the initial training set contains the samples with varying tap strength. Consequently, we utilize the multi-threshold calibration mechanism and the result shows that the EER reduces to 2%.

8.5.2 Resilience to Displacement. We further measure the sensor displacement and tapping location displacement, which might impact the reliability of Taprint.

We first form an anchoring group when the smartwatch is worn at a fixed position, and then generate other 6 test positions, which deviate from the anchoring position by 2 mm to 20 mm. We have asked 5 participants to tap on 12 keys each for 30 times with the increasing deviation to generate the dataset ($5 \times 12 \times 30 \times 7 = 12600$ responses in total). The resulting impact on the performance is shown in Figure 14 (b). On average, the EER and classification accuracy suffers minor impact when the deviation of the smartwatch is less than 12 mm. Further deviation over 12 mm leads to an unacceptable degradation of system performance.

As for the tap deviation, we know that even for tapping on the same key, the slight deviation of each tap occurs all the time. To investigate the impact on the performance of deviation of taps, we ask 17 participants to consecutively tap on each key for 30 times, and we take this group of keys as the center point. Then, they were asked to tap another 30 times respectively with an interval of 5 mm from the center point. The results show that our system is robust and suffer no impact with the spatial separation of 5 mm.

Note that we then reduce the sampling rate to 100 Hz (which modern smartwatches' APIs generally limit the sam-



(a) (b) (c)

Figure 14: Variation of EER and average accuracy (a) for different sample rate, (b) with different device location, and (c) over the time.

Table 2: Accuracy(%) and EER(%) of different user state & environment with 20-sample training.

Item&	Acc.	EER
Mobility	92.1	4.51
After Hand-wash	97.72	1.65
Quite office (44 dB)	96.43	2.40
Noisy office (85 dB)	97.44	1.76
Subway (65 dB)	96.65	2.47
Airplane (77 dB)	94.73	3.35

pling rate to it) and conduct the same aforementioned experiments. The accuracy reduces to about 50% when a 2 mm wristband deviation or 5 mm tapping displacement happen. These indicate that the the sampling rate increase not only improves the accuracy (see Section 8.3), but also enhances the robustness.

8.5.3 Arm Rotation. In practice, users might maintain different gestures when they are tapping. To evaluate the impact of such variations, we collected data under three different gestures of wrist rotated: (1) gesture 0 indicates the plane of hand back parallels to the ground, (2) gesture 1 indicates the arm rotate 45 degrees outwards from gesture 0, (3) gesture 2 indicates the arm rotate 45 degrees inwards from gesture 0. We collect the corresponding data of 18 participants from each location for 30 taps ($3 \times 18 \times 4 \times 30 = 6480$ responses in total). The results show that the accuracy is not compromised by different arm rotations.

8.5.4 User State. In this part, we examine the system robustness when users are walking or after having washed hands. We have asked 20 of 113 participants to tap on four locations, each for 30 times after washing their hand and when they were walking on the running machine (2.5 km/h).

A. Mobility: A noise interference occurs due to the physical movement that users make in the usage process of Taprint. To investigate how mobility affects the performance of Taprint, i.e., the error detection rate and the EER, we have conducted the following experiment and collect the data when participants are walking and tapping simultaneously. The result in Table 2 shows the individual statistics of every participant. Compared with that in an office, Taprint still obtains a good performance with average accuracy at 92.1% and EER at 4.5%. The reason of good performance is that the human mobility only caused low-frequency noise (less than 10 Hz)[36] and we already have removed it through a Butterworth high pass filter with 20 Hz cutoff frequency (see Section 5.1).

B. Hand-wash: Having hands washed is also a typical activity that users do many times in a day. We wonder if it causes a slight drift of underlying muscle tissue, which successively causes a change in the damping and elasticity coefficients. However, in Table 2, there is no impact on the performance of Taprint after having the users' hand washed.

8.5.5 Different Environment. We evaluate Taprint in four scenarios: a quiet office environment (i.e., control group), noisy office with loud music, subway, and flying airplane. These four environments are the typical representation of our daily lives (ranging from casual to tough) in which Taprint could be used. Our test scenarios represent a variety of noise levels, ranging from 44 to 85 dB.

Table 2 enumerates the average EER and accuracy in different environments. The results demonstrate remarkably good performance comparing to the control group in a quiet office (96.43% average accuracy and 2.40% EER). It reveals that the authentication system is robust and reliable in different environments.

8.5.6 Temporal Stability. It is crucial to verify that Taprint maintains the temporal stability (i.e., the model of a user is trained once and the resultant system keeps operating in a stable manner over the time). Two experiments were conducted spanning over a month with 6 times recorded data at different time periods as shown in Figure 14 (c). In the first experiment, 8 participants were asked to tap 12 knuckles for 6 times, respectively, to initialize the speed dial for number input. Then, participants test the speed dial by tapping each key 30 times. The results show that the accuracy is more than 95% over one month. We observe that when an error key appeared, the participants could get the correct keys by entering 1 or 2 keys under the real-time manual calibration mechanism. In the second experience, 8 participants were asked to tap on four designated location, each for 30 times with varying tapping force. Each time, they were required to wear smart wristbands in the rightest of the wrist and tap 30 times for the purpose of test. To summarize, from the figure, it is obvious that there is no significant change of EER over the one-month period under the multi-threshold calibration

Table 3: The average ranking of different authentication method.

Item	1)	2)	3)	4)	Ranking
PIN code	3.8	3	4	3.8	3.65
Password	5	5	5	5	5
Pattern	2.8	3.2	2.8	2.6	2.85
PIN code(Taprint)	2.4	2.8	2.2	2.2	2.4
Single-tap(Taprint)	1	1	1	1.4	1.1

mechanism.

8.6 User Study

One of the design goals of Taprint is to make it universally applicable and user-friendly. We have investigated the usability of Taprint, , by comparing it to the common authentication (i.e., PIN code, password, and pattern lock) and input methods employed on COTS smartwatch.

The user study involves ten participants. We asked the participants to initialize their PIN code lock (i.e., 4-digit numbers), password lock (i.e., 4-digit character), pattern lock (as they prefer). The password was selected from the random numbers or the character table. Each participant then answers the questions after performing all the methods in a random order. Then, we asked them to rank the five methods regarding the following four perspectives: 1) the speed of login, 2) the easiness to memorize, 3) the convenience to perform, and 4) the difficulty to cause the error. On the other hand, in order to measure the text input accuracy and speed, we required participants to enroll in two sessions of testing, using Taprint and Huawei Watch2, respectively but in a random order, each involves typing 120 random characters of twelve keys (ten for each).

The average ranking of the five authentication methods is shown in Table 3. Moreover, the statistics for the input method are shown in Table 4, where the user experience scores from 0 to 5. In the cases of PIN code, password and pattern lock, the system incurs a certain cognitive load on users. On the contrary, users can tap precisely on the designated location without any thought when applying a single-tapped lock. Furthermore, in the cases of the existing methods, the fatter finger of a user results in more errors on the small size screen of the smartwatch. To summarize, Taprint is regarded as the faster, easier, more convenient and less error-prone one. Besides, it also earns higher acceptance compared to the existing methods.

Taprint utilizes the accelerometer and gyroscope on COTS smartwatch to detect the finger tap vibration signal, which usually requires the sensor to be well-contacted with users' skin to obtain good quality data. Therefore, we asked every participant to grade their feeling about the tightness and comfort degree by launching two Likert-scale question: participants response an average tightness degree of 2.8 (1 = loose, 5 = tight) and an average comfort degree of 4.8 (1 = uncomfortable, 5 = comfortable).

Table 4: The input accuracy, speed and user experience of Taprint and Huawei Watch2

Item	Accuracy	Speed(s)	Score
Taprint	95%	170	4.5
Huawei Watch 2	83%	218	2.2
Tightness	Comfort	Traning accept	
2.8	4.8	positive	

9 RELATED WORK

9.1 Authentication Methods

A variety of traditional authentication methods were studied but do not fit well for smartwatches due to the constraints of screen size, battery capacity and computing power. Among these, password [1,2] and gridlock recognition [3,4] are well known. However, the passwords and patterns are easy to forget by users. The password is also quite obtrusive for users to enter when accessing the applications such as e-mails, mobile payment and so on. Moreover, these technologies are vulnerable to shoulder surfing and smudge attacks. Another critical problem is that the screen size of the smartwatches are too small to interact and some even do not have touch screens. On the other hand, voice authentication [5,6] is susceptible to noise and can easily be simulated by professional software. Moreover, one can also record and playback the voice of legitimate users.

Fingerprint [7], facial recognition [8] and iris scans [9] can be obtained or replicated by adversaries via a camera. Another critical concern is that these technologies usually stir privacy concerns of the users and require the installation of expensive equipment which typically do not exist in commodity smartwatches. Using tissue response obtained by electrical current, [10] requires an extra device, whereas Taprint does not require any extra device to detect vibration signal. Other identification methods such as gait recognition [11-15] and in-air signatures [16-18] can still be simulated. In addition, such continuous action is inconvenient since the users need to walk while unlocking the screen. On the other hand, Taprint is always ready to be used in real-time. Recently, there also exist some methods that conduct the authentication operation in a constant manner for wearable devices. Cardiac Scan [43] authenticated users by monitoring the cardiac motion feature in real time. Vauth [44] paired speech by a user between the voice from the air and the acoustic from the body to authenticate users, and the voice may sensitive to the surrounding noise level. However, Taprint authenticated users based on the hand biometry and tapping behaviors. Therefore, Taprint is more practical and reliable.

9.2 Keystoke Recognition

Certain authentication techniques recognize users based on typing behavior, e.g., manner and rhythm of typing characters. Early work used the typing behavior on the PC keyboard for authentication [19,20]. Authentication based on keystroke dynamics have also been implemented on mobile phones. Utilizing the typing behavior on mobile phone screens, a series of research work proposed different authentication schemes [21-23]. Multi-touch [24] proved that these types of technologies relying solely on the behavioral information are not stable, and proposed to combine hand geometry information to achieve authentication. However, these approaches rely on touch screens, and are hard to be implemented on wearable devices. Since the touch screen of a smartwatch is too small, some studies have proposed to use the surface of the hand's skin as an extended input interface, including the use of lasers [25], vibration sensors [26][41], pressure sensors [27], cameras [28] or implanting sensors under the skin [29], to identify the keystrokes. [26, 41] used additional vibration sensors to detect the tapping vibration, while Taprint used existing inertial sensors on the commodity smartwatch. TapSkin [45] and iDial [30] use the microphones to recognize different keystrokes on the back of the hand. The primary purpose of these studies was to achieve text input. However, a microphone is sensitive to noise, and thus users cannot type and speak simultaneously. In contrast, Taprint is the first to use modified IMU on the commodity smartwatch for text input while simultaneously achieving user authentication.

9.3 Vibration Recognition

Viband [31] leveraged accelerometers of a smartwatch to classify hand gestures, such as flicks, claps, scratches, taps, and can also recognize grasped motor-powered objects. Serendipity [32] leveraged accelerometer and gyroscope to recognize five finger gestures, such as pinch, tap, rub, squeeze and wave. SpiPhone [33] also used the accelerometer to decipher keystrokes. AGIS [34] used the accelerometer to recognize tools, such as a drill, grinder, rotary hammer based on their reduced vibration. It has also been shown that mechanical vibration frequencies can be recognized through a "wireless vibrometry" technique [49]. On the other hand, Taprint enables the authentication functionalities through the recognition of finger vibration signals in different positions of different users by utilizing the accelerometer and gyroscope. Vibration-based authentication methods have also been investigated. Headbanger [35] employed accelerometers on smart glasses to analyze the behavior of head movement for authentication. VibID [36] generated vibration signals via a motor on the wrist and then collected the vibration response as a user identification using an accelerometer. However, the vibration motor on the body is obtrusive to users. Vibwrite [37] also utilized a vibration motor on the door. When users touch the door, the vibration propagation scenarios vary, which is used as a feature to recognize users. Compared to these two active vibration techniques, Taprint is a passive system which only uses the motion sensor in COTS smartwatches to detect the vibration from finger tapping.

10 CONCLUSIONS

To summarize, the main contribution of this paper is to address a secure text input system for a commodity smartwatch by exploiting the fine-grained feature of dispersion attenuation of the tapping-induced vibration signal. This contribution is broken down into the following main aspects. First, to the best of our knowledge, we are the first to propose a novel secure text input system for smart wristbands solely relying on the motion sensors on the commodity smartwatch, without requiring any extra dedicated hardware. We believe that Taprint paves the way for secure human-wearable interactions. Second, we have built an on-body tapping induced vibration model and verify its feasibility for secure input. We have proposed a set of novel vibration detection/classification mechanisms, including a fine-grained features extraction scheme and a calibration scheme to ensure the robustness and temporal stability. Third, we have implemented Taprint as an efficient application running on COTS Android smartwatch and validated its performance through comprehensive examinations under some realistic attack scenarios.

ACKNOWLEDGMENTS

This research was supported in part by the China NSFC Grant 61872248, 61472259, 61872246, 61502313, Joint Key Project of the National Natural Science Foundation of China (Grant No. U1736207), Guangdong Natural Science Foundation 2017A030312008, Shenzhen Science and Technology Foundation (No. JCYJ20170302140946299, JCYJ20170412110753954, JCYJ20170817095418831), Fok Ying-Tong Education Foundation for Young Teachers in the Higher Education Institutions of China(Grant No.161064), Guangdong Talent Project 2015TX01X111, Tencent"Rhinoceros Birds"-Scientific Research Foundation for Young Teachers of Shenzhen University and GDUPS (2015). This work is partially supported by Tianjin Key Laboratory of Advanced Networking (TANK), School of Computer Science and Technology, Tianjin University, Tianjin China, 300350. Kaishun Wu is the corresponding author.

REFERENCES

[1] R. Morris and K. Thompson. "Password security: A case history." In Communications of the ACM, 1979: 594-597.

[2] X. Suo, Y. Zhu, and G. Owen. “Graphical passwords: A survey." In Proc. IEEE Computer security applications conference, 2005.

[3] W. Meng, W. Li, L. Jiang, and L. Meng. “On Multiple Password Interference of Touch Screen Patterns and Text Passwords." In Proc. ACM CHI, 2016:4818-4822.

[4] A. Timons and O. Altan. “Grid unlock." US Patent App. 2010, 12/698,321.

[5] A. C. Morris, S. Jassim, H. Sellahewa, L. Allano, J. Ehlers, D. Wu, J. Koreman, S. Garcia-Salicetti, B. Ly-Van, and B. Dorizzi. “Multimodal person authentication on a smartphone under realistic conditions." In Proc. SPIE, vol. 6250, 2006, pp. 120-131.

[6] R. Brunelli and D. Falavigna. “Person identification using multiple cues." Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 17, no. 10, pp. 955-966, 1995.

[7] A. Arakala, J. Jeffers, and K. Horadam. “Fuzzy extractors for minutiae-based fingerprint authentication." In International Conference on Biometrics. Springer, 2007: 760-769.

[8] B. Duc, S. Fischer, and J. Biun. “ Face authentication with Gabor information on deformable graphs." In IEEE Transactions on Image Processing, 1999:504-516.

[9] A. Kumar and A. Passi. “Comparison and combination of iris-matchers for reliable personal authentication." Pattern recognition ,2010: 1016-1026.

[10] C. Cornelius, R. Peterson, J. Skinner, R. Halter,and D. Kotz. “A wearable system that knows who wears it." In Proc. ACM Mobisys, 2014, pp. 55-67.

[11] W. Wang, A. X. Liu, and M. Shahzad. “Gait Recognition Using WiFi Signals." In Proc. ACM UbiComp, 2016.

[12] Y. Zeng, P. H. Pathak, and P. Mohapatra. “WiWho: wifi-based person identification in smart spaces." In Proc. IEEE IPSN, 2016.

[13] Y. Ren, Y. Chen, M. Chuah, and J. Yang. “Smartphone based user verification leveraging gait recognition for mobile healthcare systems." In Proc. IEEE Secon, 2013.

[14] J. Mantyjarvi, M. Lindholm, E. Vildjiounaite, S. M. Makela, and H. A. Ailisto. “Identifying users of portable devices from gait pattern with accelerometers." In Proc. IEEE ICASSP, 2005, pp. ii/973-ii/976 Vol. 2.

[15] H. Lu, J. Huang, T. Saha, and L. Nachman. “Unobtrusive gait verification for mobile phones." In Proc. ACM International Symposium on Wearable Computers, 2014.

[16] G. Bailador, C. Sanchez-Avila, J. Guerra-Casanova, and A. de Santos Sierra. “Analysis of pattern recognition techniques for in-air signature biometrics." Pattern Recognition,2011, vol. 44, no. 10, pp.2468-2478.

[17] Y.Yang, G. D. Clark, J.Lindqvist, and A. Oulasvirta. “Free-form gesture authentication in the wild." In Proc. ACM CHI, 2016, pp. 3722-3735.

[18] M. Sherman, G. Clark, Y. Yang, S. Sugrim, A. Modig, J. Lindqvist, A. Oulasvirta, and T. Roos. “User-generated Free-form Gestures for Authentication: Security and Memorability." In Proc. ACM MobiSys, 2014, pp.176-189.

[19] F. Monrose and A. Rubin. “Authentication via keystroke dynamics." In Proc. ACM CCS, 1997.

[20] F. Monrose, K. M. Reiter, and S. Wetzel. “Password hardening based on keystroke dynamics." International Journal of Information Security, vol. 1, pp. 69-83.

[21] N. L. Clarke, S. M. Furnell, B. M. Lines, and P. L. Reynolds. “Keystroke dynamics on a mobile handset: a feasibility study." Information Management and Computer Security, vol. 11, pp.161 - 166, 2003.

[22] N. Zheng, K. Bai, H. Huang, and H. Wang. “You Are How You Touch: User Verification on Smartphones via Tapping Behaviors." In Proc. IEEE ICNP, 2014, pp. 221 - 232.

[23] C. Giuffrida, K. Majdanik, M. Conti, and H. Bos. “I sensed it was you: authenticating mobile users with sensor-enhanced keystroke dynamics." In International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, 2014, pp. 92-111.

[24] Y. Song, Z. Cai, and Z. L. Zhang. “Multi-touch Authentication Using Hand Geometry and Behavioral Information." In Proc. IEEE SP, 2017,pp. 357 - 372.

[25] G. Laput, R. Xiao, X. A. Chen, S. E. Hudson, and C. Harrison. “Skin buttons: cheap, small, low-powered and clickable fixed-icon laser projectors." In Proc. ACM UIST,2014, pp. 389-394.

[26] C. Harrison, D. Tan, and D. Morris. “Skinput: appropriating the body as an input surface." In Proc. ACM CHI,2010, pp. 453 - 462.

[27] P. C. Wong, K. Zhu, and H. Fu. “ FingerT9: Leveraging thumb-to-finger interaction for one-handed text entry on smartwatches." In Proc. ACM CHI, 2018.

[28] C. Harrison, H. Benko, and A. D.Wilson. “OmniTouch: wearable multitouch interaction everywhere." In Proc. ACM UIST ,2011,pp. 441 - 450.

[29] C. Holz, T. Grossman, G. Fitzmaurice, and A. Agur. “Implanted user interfaces." In Proc. ACM CHI, 2012, pp. 503 - 512.

[30] M. Zhang, Q. Dai, P. Yang, J. Xiong, C. Tian, and C. Xiang. “iDial: Enabling a Virtual Dial Plate on the Hand Back for Around-Device." In Proc. ACM Ubicomp, 2018.

[31] G. Laput, R. Xiao, and C. Harrison. “ViBand: High-Fidelity Bio-Acoustic Sensing Using Commodity Smartwatch Accelerometers." In Proc. ACM UIST ,2016, pp. 321-333.

[32] H. Wen, J. Ramos Rojas, and A. K. Dey. “Serendipity: Finger gesture recognition using an off-the-shelf smartwatch." In Proc. ACM CHI ,2016, pp. 3847-3851.

[33] P. Marquardt, A. Verma, H. Carter, and P. Traynor. “(sp)iphone: Decoding vibrations from nearby keyboards using mobile phone accelerometers." In Proc. ACM CCS, 2011, pp. 551-562.

[34] D. J. Matthies, G. Bieber, and U. Kaulbars. “AGIS: automated tool detection and hand-arm vibration estimation using an unmodified smartwatch." In Proc. ACM 3rd International Workshop on Sensor-based Activity Recognition and Interaction ,2016.

[35] S. Li, A. Ashok, Y. Zhang, C. Xu, J. Lindqvist, and M. Gruteser. “Whose move is it anyway? Authenticating smart wearable devices using unique head movement patterns." In Proc. IEEE Percom, 2016, pp. 1-9.

[36] L. Yang, W. Wang, and Q. Zhang. “VibID: user identification through bio-vibrometry." In Proc. IEEE IPSN, 2016, pp. 1-12.

[37] J. Liu, C. Wang, Y. Chen, and N. Saxena. “VibWrite: Towards Finger-input Authentication on Ubiquitous Surfaces via Physical Vibration." In Proc. ACM CCS, 2017, pp. 73-87.

[38] J. Wang, K. Zhao and X. Zhang. “Ubiquitous Keyboard for

Small Mobile Devices: Harnessing Multipath Fading for Fine-Grained Keystroke Localization." In Proc. ACM MobiSys, 2014.

[39] Jacob B. “Surveillance Society: Wearable fitness devices often carry security risks". http://www.post-gazette.com/news/surveillance-society/2015/08/03/Surveillance-Society-Wearable-fitness-devices-often-carry-security-risks/stories/201508030023, 2015.

[40] Rahman, M., B. Carbunar, and M. Banik, “Fit and vulnerable: Attacks and defenses for a health monitoring device". arXiv preprint arXiv:1304.5672, 2013.

[41] 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. IEEE Secon, 2018.

[42] W. E. Siri. “The gross composition of the body." Adv Biol Med Phys, 1956, vol. 4, no. 239-279, pp. 513.

[43] C. Song, F. Lin, Y. Zhuang, W. Xu, C. Li and K. Ren. “Cardiac Scan: A Non-Contact and Continuous Heart-Based User Authentication System." In Proc. ACM MobiCom, 2017.

[44] H. Feng, K. Fawaz, K. G. Shin. “Continuous Authentication for Voice Assistants." In Proc. ACM MobiCom, 2017.

[45] C. Zhang, A. Bedri, G. Reyes, B. Bercik, O. T. Inan, T. E. Starner and G. D. Abowd. “TapSkin: Recognizing On-Skin Input for Smartwatches" In Proc. ACM ISS, 2016.

[46] C. Knapp and G. Carter. “The generalized correlation method for estimation of time delay" IEEE Trans. Acoust., Speech, Signal Processing, 1976, 24:320-327.

[47] http://tech.qq.com/a/20160222/006556.htm

[48] https://blog.csdn.net/li4951/article/details/7410511?utm_source

[49] T. Wei, S. Wang, A. Zhou and X. Zhang “Acoustic Eavesdropping through Wireless Vibrometry”, In Proc. of ACM MobiCom, 2015