A Low Latency On-Body Typing System through Single Vibration Sensor

Wenqiang Chen, *Student Member, IEEE*, Maoning Guan, Yandao Huang, Lu Wang, *Member, IEEE*, Rukhsana Ruby, *Member, IEEE*, Wen Hu, *Member, IEEE*, and Kaishun Wu, *Senior Member, IEEE*

Abstract—Nowadays, smart wristbands have become one of the most prevailing wearable devices, as they are small and portable. However, due to the limited size of the touch screens, smart wristbands typically have poor interactive experience. There are a few works appropriating the human body as a surface to type on. Yet, by using multiple sensors at high sampling rates, they are not portable and are energy-consuming in practice. To break this stalemate, we proposed a portable, cost efficient text-entry system, termed ViType, which first leverages a single small form factor sensor to achieve a practical user input with much lower sampling rates. To enhance the input accuracy with less vibration information introduced by lower sampling rates, ViType designs a set of novel mechanisms, including a fine-grained feature extraction to process the vibration signals, and a runtime calibration and adaptation scheme to recover from the error due to temporal instability. Extensive experiments have been conducted on 30 human subjects. The results demonstrate that ViType is robust against various confounding factors. The average recognition accuracy is 95 percent with an initial training sample size of 20 for each key. The accuracy is 1.54 times higher than the state-of-the-art on-body typing system. Furthermore, when turning on the runtime calibration and adaptation system to update and enlarge the training sample size, the accuracy can reach around 98 percent on average during one month.

Index Terms—Wireless smart sensing, wearable devices, activity recognition, vibration localization

1 Introduction

In the past few years, we have seen the take-off of wearable wristbands such as Fitbit and Apple iWatch for fitness applications. People begin to use more applications such as electronic payment and short message service (SMS) on smart wristbands instead of mobile phones. The size of smart wristbands has become smaller and lighter to provide better user experience. As a result, the touch screens on the wristbands also become smaller, while human fingers do not shrink accordingly, which are difficult to support text input.

Currently, to overcome the limitations of a small screen, speech recognition [1] is one of the methods but is sensitive to noise levels in the surrounding environments. Moreover, it is insecure for sensitive information (e.g., password input) since speech input is easy to be eavesdropped. For the same reason, it is also intrusive to the people surrounding the user. Recent works by FingerIO [2] and LLAP [3] achieves millimeter-scale localization accuracy for fingertip tracking, which enables users to write letters on ubiquitous surfaces instead of tiny touch screens. However, writing letters is significantly slower than typing them, which results in poor user experience [2], [3].

In this paper, we present a novel system termed ViType, as shown in Fig. 1, which enables a user to type on the back of one's hands (opisthenar) instead of a tiny touch screen of a smart wristband. And the interaction interface is designed to be a T9 keyboard, which can map 9 numerical keys onto the typing letters (A-Z). We leverage a vibration sensor on a wrist to collect the vibration signal by tapping in different locations on the opisthenar. As the signal carries diverse energy at different frequencies and over different distances, we reap its benefit as the unique input feature. ViType inherits the merits of vibration, such as resistance to acoustic noise and environmental dynamics. Moreover, the size of the opisthenar is larger than tiny touch screens, which enables the user to type more quickly and conveniently.

Motivated by this, we have designed a keystroke recognition system that leverages location-based vibration information derived from a small piezoelectric ceramic vibration sensor. The sensor can be easily embedded to a smart wristband. Although keystroke recognition via body vibration has been studied in Skinput [3], it takes 10 sensors of an armband with a very high sampling rate (e.g., 55 kHz) to collect the signal. On the contrary, as depicted in Fig. 2, ViType only uses a single small form factor sensor which makes it easier and more cost effective. Furthermore, it samples at lower sampling rates (e.g., 600 Hz) to make it more efficient for running on resource limited smart wristbands.

- W. Chen, M. Guan, Y. Huang, L. Wang, and R. Ruby are with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, P.R. China. E-mail: {chenwenqiang2016, guanmaoning, huangyandao}@email.szu.edu.cn, {wanglu, ruby}@szu.edu.cn.
- W. Hu is with the School of Computer Science and Engineering (CSE), The University of New South Wales (UNSW), Sydney, NSW 2052, Australia. E-mail: wen.hu@unsw.edu.au.
- K. Wu is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, P.R. China, with the PCL Research Center of Networks and Communications, Peng Cheng Laboratory, Shenzhen, China, and also with Guangzhou HKUST Fok Ying Tung Research Institute, Guangzhou 511458. E-mail: wu@szu.edu.cn.

Manuscript received 30 Aug. 2018; revised 24 June 2019; accepted 2 July 2019. Date of publication 18 July 2019; date of current version 1 Oct. 2020.
(Corresponding author: Kaishun Wu.)
Digital Object Identifier no. 10.1109/TMC.2019.2928549

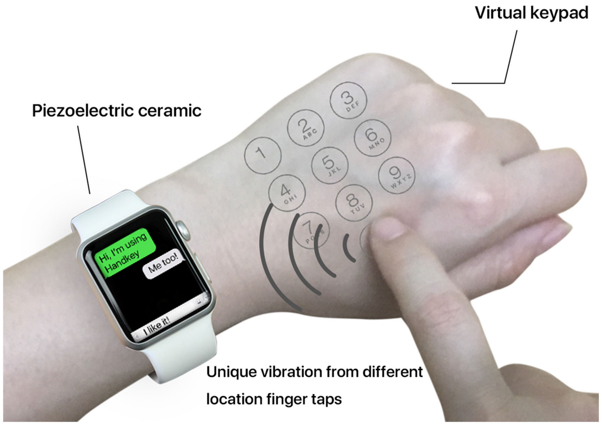

Fig. 1. A sample example of ViType.

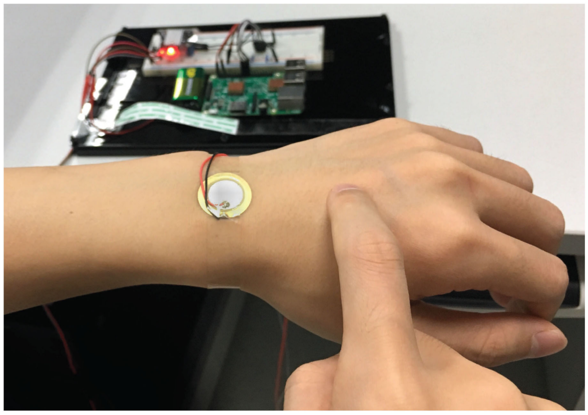

Fig. 2. A sample prototype of ViType.

It is nontrivial to embrace the above vision, as sampling at a lower rate produces significantly less vibration information. Hence, we need to investigate novel methods for keystroke recognition via body vibrations. Furthermore, in our vision, ViType should not only attain high accuracy but also be robust to many practical issues. For example, although ViType is a location-based training system, users' wristbands may have a little displacement while typing over the time. Second, users may type with different force or different finger/hand posture. Third, ViType is expected to be functional when users are walking which may cause vibration noise to the system. Fourth, it should be convenient for users to train the system at the time of first usage and then use it successively.

To cope with these challenges, we studied a set of novel keystroke detection/ classification mechanisms on different vibration patterns produced by keystrokes on users' opisthenar. We find that keystroke recognition scheme via location-based vibration depends on the vibration amplitude and frequency, which can be characterized by the waveforms and power spectral density (PSD). A more important observation is that the waveforms and PSD of different keystroke locations reveal highly distinguishable profiles and can be conveniently used as a location signature. We removed the noise signal caused by human mobility from the original signal via a filter and then used an online dual-threshold start point detection algorithm to detect keystroke signals. In addition, we conducted a feature optimization scheme with reliefF algorithm [38] to extract fine-grained weighted features. Last but not the least, we design a runtime calibration and adaptation system and provide a special scheme to update and enlarge the training set to enhance the robustness in practical situations such as variations of finger posture or displacement of wristbands and tap position.

We implement ViType as a prototype on a Raspberry Pi with a small form factor piezoelectric ceramic in real time system [36]. A demonstration video is attached in the link.[1] Our baseline evaluation shows that classification accuracy is 95 percent on average for 30 human subjects with an initial training sample size of 20 for each key, which is 1.54 times higher than the state-of-the-art system of Skinput [4]. We have also conducted a series of studies in realistic settings such as wristband or tapping displacement, variations of tapping force, and found that the performance of ViType degrades significantly in non-ideal circumstances. Thus, we design a runtime calibration and adaptation scheme to address these challenges and our results show that the proposed scheme can mitigate the degradation.

Our contributions in this work lie in the following aspects.

- ViType is the first attempt in the literature to recognize the keystrokes typing on a user's opisthenar via a single small size vibration sensor. It samples at an order of magnitude lower rates to achieve a more efficient text-input method on resource limited smart wristbands.
- We president the entire design of ViType, which extracts a fine-grained weighted features to recognize the keystroke vibrations, and harnesses a runtime calibration and adaptation scheme to achieve a desirable recognition accuracy.
- We comprehensively evaluate the performance of ViType under different scenarios. The recognition accuracy is 95 percent, which is 1.54 times higher than the state-of-the-art on-body typing system.

The remainder of this paper is structured as follows. In Section 2, we first provide the related work in the context of this work. Then, Section 3 proposes a vibration propagation model. Section 4 presents the overview of ViType showing the design goals and challenges. Section 5 describes the three main modules of ViType. Section 6 explains the detailed implementation technique, followed by a comprehensive experimental evaluation of our system. We elaborate on the outcome of a user study in Section 7. Finally, Section 8 draws a limitation and future work in this paper and we conclude this paper in Section 9.

2 Related Work

Text Input for Wearable Devices. Speech recognition [1] is popular for text input. However, it is sensitive to noise levels in the environments and insecure for sensitive information such as password input due to its convenience to easily eavesdrop. For the same reason, it is also intrusive to the people around the user. Recent works by FingerIO [2] and LLAP [3] achieve millimeter-scale localization accuracy for

1. https://youtu.be/taLZLFyPB4M

fingertip tracking, which enables users to write letters on ubiquitous surfaces instead of tiny touch screens. However, writing letters is significantly slower compared to typing, which results in poor user experience. Some projects require a user to carry extra devices such as a ring, a pen or even a shoulder-mounted camera for text input [7], [8], [9], [10], [11]. However, they are uncomfortable and cumbersome. It is also unacceptable for users to implant sensors under the skin [12] for text input. Camera technique suffers great controversy in terms of privacy issues and requires line-of-sight sensing [7]. Although the infrared technique has achieved some simple human-computer interaction designs, it is still not suitable for text input because of power consumption and accuracy limitation. Refs. [13], [14] TapSkin [15] and iDial [16] utilize commodity smartwatch to detect the keystrokes via the acoustic microphones, which are sensitive to noise such as human voice. Skinput [4] collected keystrokes via 10 vibration sensors at a very high sampling rate (e.g., 55 kHz). In this paper, we only use one vibration sensor that works in a lower sampling rate (e.g., 600 Hz) in order to recognize keystrokes on the back of the hand.

Keystroke Localization. Toffee [17] utilized four piezoelectric vibration sensors to classify keystrokes tapping on the table with the mobile phone. SurfaceVibe [18] used four geophones to localize taps and swipes for supporting text input on the surfaces such as wood tables. However, these methods used the TDOA-based localization concept which requires multiple sensors, and hence these are not appropriate for the small size wearable wristbands. Refs. [17], [18] VibSense [19] and Ubik [6] also realize the typing objective which adopt portable and external paper keyboards for the mobile phone. However, we propose a virtual keyboard for the smart wristband to type on the back of the hand. Wikey [20] classifies the keystrokes on a real keyboard based on the fact that different keystrokes at different locations produce different multipath reflection of WiFi signal. Liu [21] utilizes the microphones of the mobile phone to detect the keystrokes on the physical keyboard to enable acoustic snoop of passwords. Wang [22] also used microphones of the mobile phone to recognize combined keystrokes based on the blind source separation mechanism.

Vibration based Smart Sensing. Viband [23] leverages accelerometers of a smartwatch to detect the vibration for classifying hand gestures, such as flicks, claps, scratches, taps and also can recognize grasped motor-powered objects. Serendipity [24] leveraged accelerometer and gyroscope to detect vibration for identifying five finger gestures, such as pinch, tap, rub, squeeze and wave. SpiPhone [25] also used an accelerometer to decipher keystrokes. Moreover, AGIS [26] used an accelerometer to recognize tools, such as a drill, grinder, rotary hammer based on their reduced vibration signal. VibID [27] leverages a vibration motor to generate vibration signals on the wrist and then utilizes accelerometers to detect the vibration response as user ID for the smart wristband. Vibwrite [28] also set a vibration motor on the door. When users touch the door, the vibration propagation scenarios are changed, and hence the change of vibration signal is used for user authentication. There are some works that adopt the geophones to detect the vibration signal caused by footsteps. FootprintID [29] utilized footstep induced vibration signal to identify occupants using an iterative transductive learning algorithm. [30] presented a method to monitor the traffic associated with multiple occupants by sensing the ambient structured vibration signal. The system achieves its traffic monitoring objective by acquiring signals from the structured vibration sensors and analyzing their features. Bales [31] used footstep induced vibration signal to identify gender information of the occupants. Jia [32], [33] used geophones to detect vibration to monitor heartbeats of people lying on the bed. To reduce the multipath effect of RF and acoustic signals, Floc [34] suggested a vibration localization method with a SWIM algorithm to detect the footsteps. Mirshekari [35] utilized a wavelet transform technique to localize the vibration signal of footsteps.

3 Vibration Propagation Model

When a finger taps on the opisthenar, two separate forms of vibration signal are produced, which are transverse and longitudinal waves. Transverse waves translate along the hand surface while longitudinal waves move into and out of the bone through soft tissues. Moreover, during the propagation of vibration from a tapped location to sensor, the signals suffer attenuation and the model can be stated as follows [4].

$$A(d) = A_0 e^{-\alpha \times d}, \tag{1}$$

where A_0 is the initial amplitude, d is the propagation distance and α is the attenuation coefficient. The attenuation coefficient quantifies the intensity of the signal attenuation resulted from different mediums. The relation in (1) further reveals that the amplitude of vibration signal is dominated by the propagation distance and attenuation coefficient. Under a fixed propagation distance, vibration signals with different frequencies will experience different attenuation. Whereas signals with the same frequency will also experience different attenuation when traversing through different materials and paths. Roughly speaking, higher frequencies propagate more readily through the bone compared to the soft tissue, and hence bone conduction carries energy over a larger distance unlike that with the soft tissue [3]. *Thus, the vibration is unique and diverse across tapping position.*

4 Overview of ViType

4.1 Design Goals and Challenges

We design ViType to meet the following goals and overcome the following challenges which are basically required to use this system in practice.

1) *User friendly:* It will result in terrible experience if users have to reset the input system each time they type. Such time overhead is not negligible and annoying if the usage duration is short. Therefore, ViType needs to make sure its temporal stability that each user has to launch the setup procedure only once.
2) *Availability:* The localization mechanism of ViType should also be designed to fit in different operating conditions. For instance, users may apply different tapping force and change their hand posture over time and even need to type while in the walking phase. Besides, ViType should have a strong ability

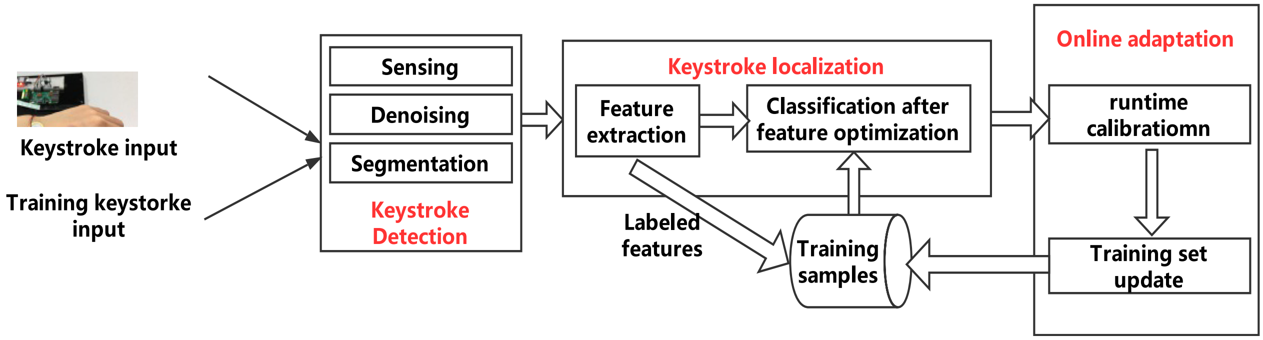

Fig. 3. Architecture of ViType.

of resisting acoustic noise caused from surrounding environments.

3) *Fine-grained:* ViType utilizes the relatively wide area of the opisthenar as the interactive interface for users to type in. However, the space between two keys is only around 2 cm. In order to achieve a centimeter-scale keystroke localization, we have to realize a localization mechanism that can recognize keystrokes with high accuracy. In Skinput [3], the accurate localization on forearm is realized using 10 sensors of an armband at a very high sampling rate such as 55 kHz. However, ViType needs to achieve accurate localization on opisthenar using only one sensor at a low sampling rate such as 600 Hz.

4) *Deviation-tolerant:* Marking keyboard layout on the opisthenar with a pen is inconvenient and the marks on the hand tend to be erased, which will result in poor user experience. One of the schemes to perform a virtual keyboard on a particular interactive surface is projection [13], which can visualize the layout of keyboard. However, it is impractical to project a virtual keyboard on the opisthenar as smart wristband is energy limited and extra employment of hardware is needed. Hence, ViType has to attain its localization with high accuracy in the circumstances that users have no visual keyboard layout and assistant marker to keep tapping on the same key position over time. In addition, similar to the deviation of keystrokes, the shift of a smart wristband over time also needs to be taken into consideration.

4.2 System Architecture

The architecture of ViType consists of three major components in order to build a robust and self-contained keystroke localization system for smart wristband. The functionalities of these components are described in the following.

1) *Keystroke detection:* ViType employs the piezoelectric ceramic sensor to convert the vibration signals into recordable electrical signals which are then denoised using a filter and segmented by a double threshold-based mechanism.

2) *Keystroke localization:* Once the vibration signals are received and detected, ViType utilizes a keystroke localization algorithm to extract the unique vibration feature (i.e., power spectrum density) in the frequency domain and fuses it with amplitude signals as inputs to a trained classifier after a fine-grained feature optimization for the purpose of localization.

3) *Runtime calibration and adaptation:* ViType takes the advantage of user's on-screen feedback to correct accidental classification errors and this process is called calibration. Furthermore, it adopts a runtime adaptation algorithm to update and enlarge the training set to maintain the high accuracy of classification.

Fig. 3 describe the work-flow of ViType system. In the initial training stage, the vibration signals of keystrokes are sensed, denoised and segmented by the detection mechanism. Afterwards, the localization algorithm extracts the feature and and conducts a fine-grained weighted feature optimization. When a new keystroke is detected, the localization algorithm finds the best match in the trained model and then output the resulting number on the screen along with candidate keys which can be calibrated by the user. The feedback of calibration is transmitted to the runtime adaptation algorithm to update the training set.

5 ViType

5.1 Keystroke Detection

5.1.1 Sensing

There are inertial measurement units (IMU) in the COTS wearable wristbands which are able to detect vibration. However, they are engineered for very different applications such as motion tracking rather than measuring acoustic signals propagated through the human body. Consequently, they are unfit in many crucial ways [3]. Piezoelectric ceramic sensor uses the piezoelectric effect to measure the vibration intensity by converting it to an electrical charge. In a piezoelectric ceramic device, mechanical stress, instead of externally applied voltage, causes the charge separation in the individual atoms of a material. Thus, the vibration caused by finger taps is able to be converted to an electrical charge. Fig. 2 shows a sample piezoelectric ceramic sensor whose external diameter is 20 mm and thickness is only 0.4 mm (FT-20T- 6.5A1). The small form factor of the sensor makes it easy to be embedded to a smart wristband as a chassis.

5.1.2 Denoising

Unlike a microphone-based acoustic system, ViType is capable of resisting environmental noise using the vibration signal. Therefore, ViType has a low level of noise naturally. At first, we used a 20 Hz Butterworth high pass filter to remove

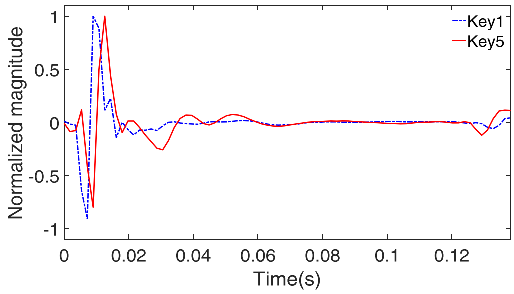

Fig. 4. Normalized amplitude of key 1 and 5.

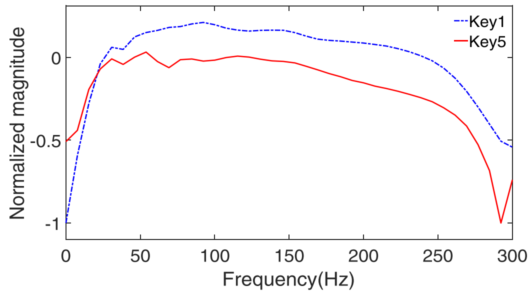

Fig. 5. Power spectral density of key 1 and 5.

the low frequency noise caused by the direct current component and the human mobility (less than 5 Hz) [27]. Second, we set the cut-off frequency to 300 Hz of a Butterworth low pass filter since the vibration signal tapped on the opisthenar is realized in low frequencies (less than 200 Hz) domain base on Fourier transform analysis. Then, the noise in the higher frequency domain can be removed as well.

5.1.3 Segmentation

We use energy-based double threshold approach to detect the start point of a keystroke [19]. The lower threshold is $\mu + \sigma$ and the higher one is $\mu + 3\sigma$, where μ and σ are the mean and standard deviation of energy obtained from collected signals, respectively. The lower one is very sensitive to the variation of signal and can easily be broken, whereas the higher one will not. Exceeding lower threshold level is not necessary to detect the start point as there might be some noise whose energy is higher than it. Only when the high threshold is overpassed, the low threshold can be considered as the start point of the signal to be detected. In terms of the end point, we set it at 0.1 s after the start point as the duration of a keystroke signal is usually around it.

5.2 Keystroke Localization

5.2.1 Feature Selection

The attenuation model of vibration signals, as stated by the relation in (1), provides us a hint of using raw data of amplitude as location signature and Fig. 4 shows the distinguishable vibration signals collected from key "1" and key "5". Furthermore, the signals also carry diminishing energy at different frequencies over different distances, and thus we investigate the profile pattern of key "1" and key "5" in frequency domain. Specifically, we choose PSD of the collected vibration signals, which reveals the power distribution in different frequency. If k_i is the received vibrations signals, then the PSD can be defined as

$$PSD_i = 10\log_{10}\frac{(abs(FFT(k_i)))^2}{f_s \times n}, \tag{2}$$

where $FFT(\cdot)$ is the fast Fourier transform operation, f_s is the sampling rate, and n is the number of samples of received signal k_i. Fig. 5 shows that the PSD features of two keys have different profile, which exhibits distinct values across frequencies. This gives us another justification for using PSD to locate the keystrokes. Therefore, we decide to extract amplitude and PSD from raw signal and fuse them together as the inputs to estimate the classification model in ViType system.

Note that the profile of each key shows the distinction but we only present the profiles of 2 keys for the brevity of image expression.

5.2.2 Classification Algorithm

ViType classifies keystrokes based on a pattern classification scheme that matches the fusion of amplitude and PSD features of user-typed keystrokes with those instances in the training set.

Basic Classification. After feature selection, ViType runs a nearest-neighbor based pattern matching algorithm that compares the extracted features with those in the training set. The training key with minimum distance is declared as the current keystroke and output to the user interface. In the simplest form, we use euclidean distance as the metric of comparison.

In our previous work [37], we utilize an artificial neural network(ANN) to automatically optimize features and classify keystrokes in high accuracy. However, considering the temporal stability, we need to update keystrokes when usage (Section 5.3 will discuss the runtime calibration and adaptation scheme of ViType) which asks for retraining model in 0.6 seconds. Due to the time latency for training, our previous work [37] design to retrain the model when several accumulated error keys occur. In this way, the real time of the system has decreased which can not calibrate the error key immediately. To solve this problem, we proposed nearest-neighbor which does not have a training process and enable ViType to calibrate and update keys in time when each an error key happens. In order to keep as high accuracy as using ANN, we designed a fine-grained feature optimization to improve accuracy.

Feature Optimization. We observe that both the features with small distances between samples from the same class and the features with large distances between samples from the different classes represent the location profiles well. Therefore, ViType utilizes ReliefF algorithm [38] to optimize features.

The algorithm randomly selects a sample R from the training set D, and then finds the k nearest neighbor samples from the samples of the same class of R, called near Hit, and finds the k nearest neighbor samples M from the samples of different classes of R, called Near Miss. ReliefF update the weight of each feature according to the following rules: If the distance between R and Near Hit on a feature is less than the distance between R and Near Miss, it indicates that the feature is beneficial to distinguish the nearest neighbors of the same class and different classes; then increase the weight of the feature. By contrast, if the distance between R and Near

Hit is greater than the distance between R and Near Miss, indicating that the feature has a negative effect on distinguishing the nearest neighbors of the same type and different classes, the weight of the feature is reduced. The above process is repeated for a number of times, and finally, the average weight of each feature is obtained. The running time of the Relief algorithm increases linearly with the increase of the sample number of samples and the number of original features, thus running efficiently.

With ReliefF algorithm, ViType extracts the fine-grained vibration features with weights, and the resultant distance becomes $\sqrt{(w_i X_i - w_i Y_i)^2}$. X_i and Y_i represent the ith feature of the tapping vibration of different human body positions, and W_i is the weight of this feature. According to the multiplicative distribution ratio, we obtain $w_i\sqrt{(X_i - Y_i)^2}$. That means the distance of each feature is scaled according to the weight of the feature.

In the application scenario of ViType, when a user wears our wristband for the first time, a short initial phase (tapping each key 20 times within 3 minutes) is needed for building a samples set for matching. However, the initial sample set will be updated automatically each time users type with ViType system.

5.3 Runtime Calibration and Adaptation

ViType designs a runtime calibration and adaptation system to adapt itself to the deviation of wristband and keystrokes, and keeps the training data as new as possible to achieve better localization accuracy. As for calibration, for each keystroke, besides the output which is given by the classification algorithm, ViType also displays other top 2 candidate keys. A user can click any candidate key if it is the actual intended key when the algorithm gives a wrong output on the touch screen. If there is no intended key contained in the candidate list, the user will require to turn to the built-in on-screen keyboard.

In terms of practical usage, there are three cases: (1) a user does not select the candidate key and ViType will deem the localization output as correct, (2) a user selects any candidate key, which means that the system gives a wrong output and ViType maps the current input signal with candidate key rather than the wrong output, (3) a user taps the "Delete" button and it is not necessarily a hint for localization error as it may be the user's own input error.

Therefore, for adaptation, we have designed a special scheme to update the training set. For case 1, the input sample will not be added into the training set. For case 2, the input sample will be added into the training set with the label of selected candidate key. At the same time, the nearest neighborhood of the input sample will be deleted since it leads to the output error. Note that the oldest samples leave the corresponding queue if the training set sizes reach the maximum of 70 for each key.

6 Implementation & Evaluation

Implementation. In this prototype, we implemented ViType using a piezoelectric ceramic sensor and an amplifier connected to a Raspberry Pi controller via an Analog to Digital Converter (ADC). The Body vibrations are collected via BCM2835 Library with C. Then, we transmit them to a conventional desktop computer by a PL2303 USB To Transistor-transistor logic (TTL) Converter Adapter Module. It is implemented via WiringPi Library with C. As for the keystroke recognition, the signal denoising, keystroke detection and classification algorithm are implemented in Matlab toolbox.

Experimental Setup. We recruited 30 participants (20 of them are male) who are in the age range of [19], [20], [21], [22], [23], [24] and stand for the crowd that are most likely to use our system. Besides, their body mass indexes (BMIs) range from 17.26 (lean) to 29.38 (obese). Note that all the experiments involving human subjects conformed to the relevant regulations of our university.

The evaluation experiments are launched in a conventional office environment. At the beginning of the experiments, the instructor marked the location of each key using a marker. This is because the participants may not realize the fact that tapping a bony area produces more stable vibration signal compared to the case when fleshy area of the back of one's hand is tapped [3]. Moreover, participants are given a 10-minute warm-up period to become familiar with our system before the experiments. In all experiments, we adopt the following default setting unless explicitly specified. The participants are instructed to tap 30 times on each key in an orderly fashion (270 examples for each person, 8,100 data points in total). For example, we ask the participants to tap on key 1 for 30 times, then key 2 for 30 times, and so on. Note that the calibration and adaptation scheme is turned on for the experiments in Section 6.3 only. Some parts of the experimental setup in Sections 6.2 and 6.3 are different and will be introduced in corresponding section.

6.1 Accuracy of ViType

In this section, we first verify the suitability of the features we used by comparing three other feature subsets. Then, we validate the accuracy of ViType in terms of the keystrokes detection and keystrokes classification. Afterwards, we evaluate the accuracy comparing with the existed work Skinput [3]. We end this section with the discussion about the impact of training sets of different sizes on the keystrokes recognition accuracy.

6.1.1 Effect of Feature Subset

We have got two potential features for classification in Section 4.2, and hence we obtain three different feature subsets, which are (1) only amplitude, (2) only PSD, (3) fusion of amplitude and PSD. In this experiment, we investigate the classification accuracy with respect to different features mentioned above. As shown in Fig. 6, the feature set (3) obtains the highest average accuracy at 95 percent, followed by (1) at 92.9 percent and (2) at 87.7 percent, respectively. The amplitude of each keystroke reflects the attenuation information in time domain while the PSD reflects it in the frequency domain. Consequently, the fusion of these two features complements each other and improves the classification performance. Note that we also tried to add Mel-Frequency Cepstral Coefficients (MFCC) into the feature set. However, it showed no improvement in accuracy (95 percent). We believe that it is because our raw data has too short length in time domain, such as 50 samples. Considering the latency problem, we did not adopt mfcc as features.

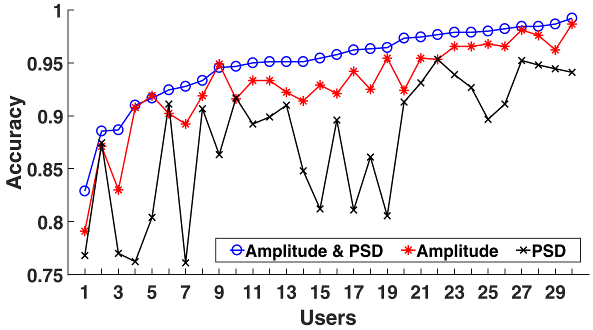

Fig. 6. Classification accuracy of three feature subsets.

6.1.2 Baseline Detection and Localization

We perform the experiment of keystroke localization following the setting discussed above. After collecting the data, we obtain the results with 0 percent mis-detection and 0 percent false-alarm. In terms of localization, Fig. 7 plots the resulting confusion matrix of localization accuracy for 30 participants, showing that the average classification accuracy is 95 percent.

6.1.3 Comparison with Skinput

Here, we compare the performance of ViType to the state-of-the-art approach Skinput, in which signals are collected from 10 piezo films of an armband at a very high sampling rate. Whereas, ViType uses a sensor with small size for making it easier and more cost effective to be embedded on smart wristbands. Moreover, ViType samples at an order-of-magnitude lower rates that makes it more efficient to run on resource limited smart wristbands. In this experiment, we input the extracted features adopted in Skinput of the same raw data into a SVM classifier (used in Skinput as well). Fig. 8 demonstrates how ViType outperforms Skinput. Comparing the red line with the blue line, we observe that Skinput obtains an average accuracy of 61.6 percent, while ViType can obtain an average accuracy of 95 percent. In other words, ViType can obtain an approximate 1.54 times higher average classification accuracy compared to Skinput. Furthermore, ViType has a relatively steady accuracy among different users while the accuracy of Skinput has a large difference among people and the standard deviation of accuracy are 0.036 and 0.124 for ViType and Skinput, respectively. We note that the worst case of ViType outperforms the best case of Skinput.

6.1.4 Impact of Training Set Size

Intuitively, the classification accuracy of our system can be enhanced by enlarging the size of the training set. This is due to the fact that it is rather difficult for ViType users to tap exactly on the same point for each key without any deviation. To verify this hypothesis, 8 participants are asked to tap 80 times per key in order to produce a training set. We calculate the accuracy with respect to different size of training set (from 5 to 70) for both ViType and Skinput and plot Fig. 9.

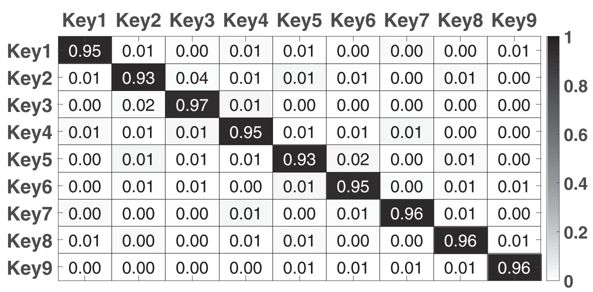

Fig. 7. Confusion matrix of 9 keys using PSD.

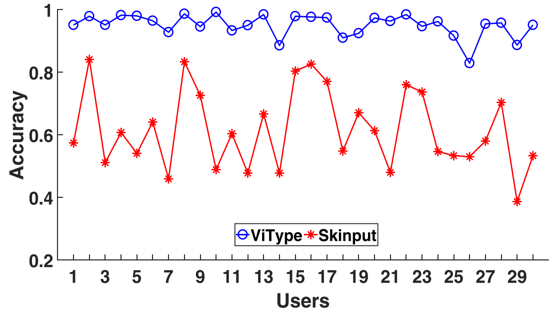

Fig. 8. Classification accuracy comparison between ViType and Skinput.

We can observe evidently that the classification accuracy rises upward monotonically with the increasing size of the training set for both ViType and Skinput. However, the accuracy of ViType increases faster than that of Skinput when the size of training samples rises from 5 to 20 (at about 89 and 95.8 percent with weighted features, respectively), while the accuracy of Skinput is below 60 percent in this range. This implies that ViType has a better user experience as we may ask a user to tap 20 times for each key only to initialize the training sets, the duration of which is within 3 minutes. Also, the weighted feature shows better accuracy. ViType has the calibration and adaptation scheme. Hence, as a user types every day, its training set grows continuously (e.g., even 70 for each key), which means that the accuracy of ViType can approach nearly 98 percent (see Section 6.3 for more details).

6.2 Robustness of ViType

In this section, we focus on the robustness of the system and employ our system under several different conditions. Note that we did not turn on runtime calibration and adaptation feature of ViType in these experiments. The issues, in which we are concerned, are stated as follows.

- Position setting
- Positional variation of wristbands
- Positional deviation of tap
- Difference tap force
- Mobility
- System limit

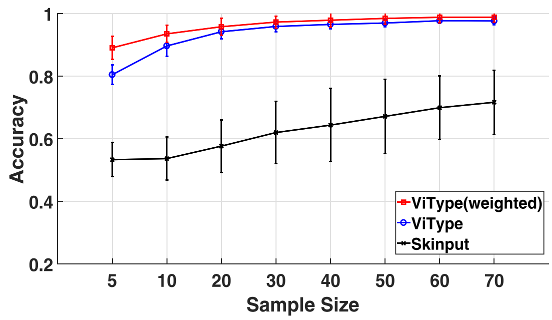

Fig. 9. Impact of initial training set size for ViType and Skinput.

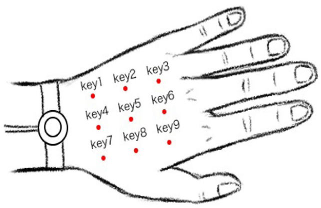

Fig. 10. A keypad on opisthenar.

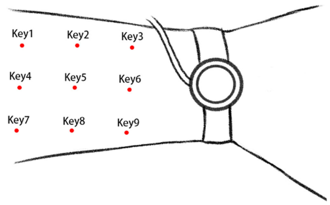

Fig. 11. A keypad on forearm.

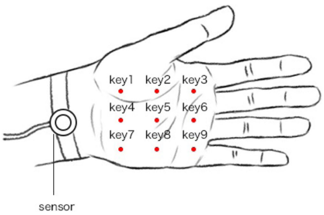

Fig. 12. A keypad on the palm.

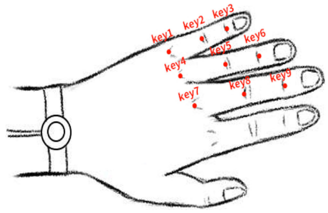

Fig. 13. A keypad on knuckles.

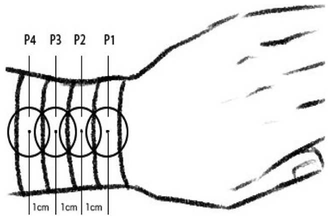

Fig. 14. Positional variation of wristband.

6.2.1 Position Setting

We have considered four potential positions of body for setting the T9 keyboard, including opisthenar, forearm, palm and the knuckles which are illustrated in Figs. 10, 11, 12, and 13 respectively. We believe that these four positions are of particular interest when it comes to interface design. We choose these positions for the following reasons. First, when people tap on the opisthenar, vibration propagates through tissue and bone. The difference of skin's damp and bone structure make the responding signal variant between different key pressing events. This makes each key more distinctive which is supposed to be beneficial to classification. Second, in terms of spatial characteristics, forearm is compelling since it is relatively large and flat which makes it an ideal input interface. Third, for palm, we take a peculiar group who enjoy wearing watches' face on the inside side of their wrists as Fig. 12 showed and investigate the classification accuracy in this setting. At last, when it comes to the realistic scenarios, it is difficult for new users to tap on fixed location without markers on the users opisthenar (or other positions). Fortunately, knuckles are natural landmarks on hand, requiring no projection or drawing. Thus, we map the keypad to knuckles as shown in Fig. 13.

In this experiment, we draw a keypad of 9 keys on the aforementioned four locations and ask 8 of our participants to tap on every key for 30 times. We take 8640 responses $(4 \text{ locations} \times 9 \text{ keys} \times 8 \text{ users} \times 30 \text{ times})$. The average accuracy of different positions have slight variation, and all have high accuracy. The position where our system obtains highest average classification accuracy (at 95.2 percent) is opisthenar compared with that of forearm (at 94.2 percent), palm (at 90.5 percent) and knuckles (at 94.5 percent). We reckon that the fleshier characteristic of palm lead to inferior propagation quality of vibration signals, that results in lower localization accuracy.

To summarize, among all the potential interface of opisthenar, forearm, palm, and the knuckles, the accuracy of opisthenar is the highest. And we regard opisthenar as our choice not only for its high classification accuracy but also the perspective of user preference.

6.2.2 Positional Variation of Wristbands

The key insight of ViType is the distinctive vibration signal produced by tapping on different locations on the opisthenar which requires to fix the position of the wristband. However, there is quite a common scenario that the wristbands user wearing shifts from original location to others over the time of usage. In this experiment, we assume that the original position of the sensor is point P1. We then have points P2, P3 and P4 by shifting 1 cm towards the elbow gradually (Fig. 14). We change the location of the sensor from P1 to P4 and ask 8 participants to consecutively tap on each key for 30 times. In Fig. 16, the X-axis is the format of "training data-test data". For example, "1-2" indicates that we train the classifier with 20 samples collected on P1 and test the system with 10 samples collected on P2. Particularly, "12-12" means that we train the classifier with 20 samples from P1 and P2, respectively (40 training samples in total), and test the system with the remaining 10 samples from P1 and P2, respectively. We have 2 observations in the context of this experiment: (i) using the samples collected from the same location for the test purpose (e.g., 1-1) can achieve much higher accuracy comparing with 1-2, 1-3, 1-4 cases. (ii) initializing the system from the samples collected from different points (e.g., 12-12, 14-14) can mitigate the impact of wristband displacement. Consequently, it provides us a hint of designing a runtime adaptation scheme to update the training set (see Section 6.3 for the details).

6.2.3 Positional Deviation of Tapping

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 8 participants

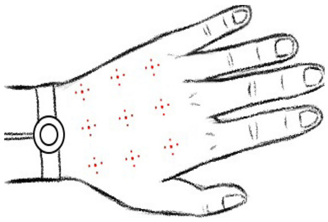

Fig. 15. Positional variation of taps.

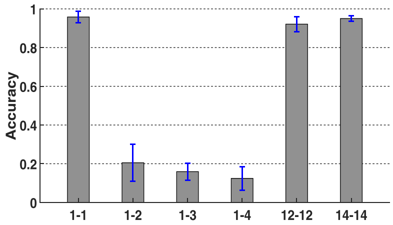

Fig. 16. Accuracy of positional variation of wristband. Error bars show std.

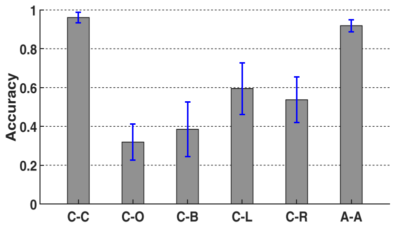

Fig. 17. Accuracy of positional variation of taps. Error bars show std.

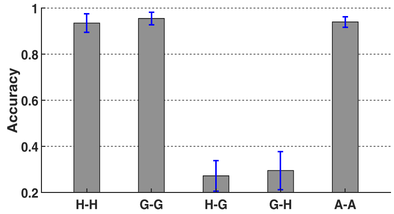

Fig. 18. Impact of different force taps. Error bars show std.

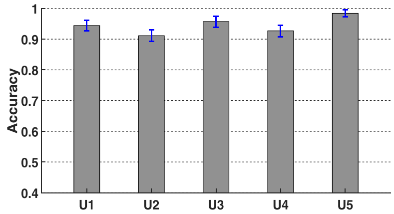

Fig. 19. Impact of tapping when walking. Error bars show std.

to consecutively tap on each key for 30 times as well as the deviation key which are illustrated in Fig. 15 (with 0.5 cm interval from the center of key in four directions, namely over (O), below (B), left (L) and right (R)). In Fig. 17, similar to Fig. 16, the X-axis is also the format of "training data-test data". Particularly, "A-A" means that we train the classifier via 100 samples collected from every point (20 from each location) and test the system with the rest of 50 samples (10 from each location). The histogram shows that when the training set and the test set are from different locations, the localization accuracy suffers a great drop to less than 60 percent. However, the accuracy of the "A-A" case recovers to around 91.9 percent, which means that ViType has the resilience to tapping deviation when the training set is associated with the tapping deviation.

6.2.4 Force of Tap

The resultant vibration signal may be different when users apply different tapping force, which results in localization errors. To examine the impact of tapping force, we conduct this experiment in which we ask 8 participants to tap on each key 30 times both gently and heavily, that results in 4,320 responses $(9 \text{ keys} \times 8 \text{ users} \times 30 \text{ times} \times 2 \text{ ways})$. Note that "H" indicates that a user taps heavily while "G" indicates that a user taps gently. Fig. 18 shows the resulting graphics. Similar to the previous sections, the X-axis stands for "training force-testing force". From the figure, we discover that the classification accuracy drops to below 30 percent when the testing tap force is different from the training tap force (i.e., G-H and H-G). A different tap force incurs different characteristics of signals, which leads to lower classification accuracy. Moreover, while initializing our classifier with both "heavy" and "gentle" data, the accuracy recovers to the same level with the accuracy of using the same force (H-H and G-G).

In reality, users may apply different tap force to make the training set more approximate to the results in "A-A" model, which may contain all kinds of tap force in the test phase.

6.2.5 Mobility

If users have other physical movements while they type on ViType, it will probably result in noise interference and cause a higher detection error rate. This happens regularly in our daily life. For example, we may need to send an important message or chat with someone when we are in the walking phase. Practically, we cannot avoid noise interference due to this movement. To investigate how mobility impacts the classification accuracy, we conduct the following experiment to study the accuracy of our system while walking and typing simultaneously. In this experiment, we ask five of our participants to tap on nine keys 30 times each respectively when walking to test the system. Note that the training data are collected in static status and the test samples are collected when users are walking. Fig. 19 plots the results of this experiment, which shows the individual accuracy of every participant. Compared with participants sitting in an office, ViType still obtains a high accuracy (94.4 percent on average). The reason of high accuracy is that the noise caused by human mobility is at low frequency (less than 5 Hz) [27] and we remove it via a 20 Hz Butterworth high pass filter. We did not study how our system performs when users are jogging since it is too dangerous to type in this scenario and we strongly advise the users to avoid typing when jogging.

6.2.6 System Limits

In Vitype, we devise a virtual keyboard on the hand back with a pad size of 3×3. We might be curious about what will happen when enlarging the pad size to 4×4 or even 5×5. Can Vitype still provide a fine-grained keystroke recognition? Thus, we launch this experiment to investigate how much degradation regarding to accuracy will our systems suffer. In this experiment, we mark out the keyboard on the hand back and set the key array size to be 3×3, 4×4, and 5×5 respectively. And the corresponding interval

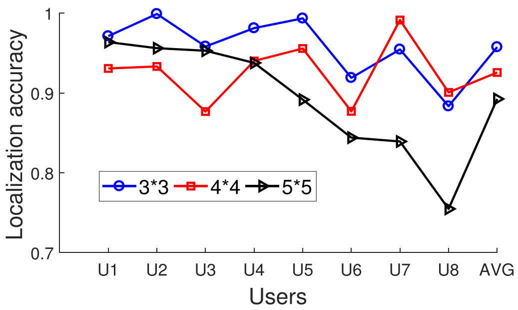

Fig. 20. The localization accuracy with respect to different key array size.

between each key is strictly controlled to be 2.5 cm (3×3) , 2 cm(4×4) and 1.5 cm(5×5) respectively. We ask 8 participants to tap 30 times on each key to generate the data set (270 + 480 + 750 = 1500 samples for each person). Fig. 20 shows that our system achieves the average localization accuracy of 95.7, 92.6 and 89.2 percent when setting the key array size to be 3×3, 4×4 and 5×5 respectively. This indicates that our system can still provide a fine-grained keystroke recognition when enlarging the pad size to 4×4. However, the accuracy drops below 90 percent when enlarging the pad size to 5×5.

6.3 Runtime Calibration and Adaptation

While using ViType in practice, it is difficult for users to keep the position of the wristband and keystroke unchanged while the usage period of time. It is quite unfriendly for users if they need to reinitialize the system every time they use ViType. ViType has proved its robustness and indicates that we can alleviate the deterioration of accuracy by increasing the training samples collected in different conditions. However, we still want to achieve better accuracy and provide better user experience in reality. Hence, we design the calibration and adaptation system. In the following two experiments, we turn on the runtime calibration and adaptation scheme.

6.3.1 Resilience to Displacement

In the first experiment, we count the localization accuracy averaged over the last 50 keystrokes with respect to the displacement of wristband and tap position in different levels. The resulting impact on accuracy is shown in Fig. 21. In terms of the displacement of the wristband, the localization accuracy drops to around 70 percent if the wristband is moved from the original position no matter with a displacement of 1 cm, 2 cm or 3 cm. As for the displacement of tap position, in the case with the smallest displacement at 0.5 cm, the accuracy shows no significant degradation. Whereas in the case with larger displacement of 1 cm and 1.5 cm, the accuracy drops to about 76 and 72 percent, respectively. Particularly, when the displacement of wristband and tap position occur at the same time, the accuracy drops to around 70 percent. However, ViType's calibration and adaptation scheme can mitigate the impact of these wristband and tap position displacement and recover the accuracy to above 95 percent after a few tens of inputs.

6.3.2 Temporal Stability

In order to judge this metric, we conducted experiments 5 times over the interval of 1 hour, 1 day, 2 days, 1 week and 1 month. In each time, we tap from key "1" to key "9" for 100 rounds (900 keystrokes in total) while considering the average localization accuracy of the past 50 keystrokes. We observe that as the size of the training samples is enlarged, the localization accuracy remains stable at around 98 percent each time. This indicates that ViType is temporally stable over the time.

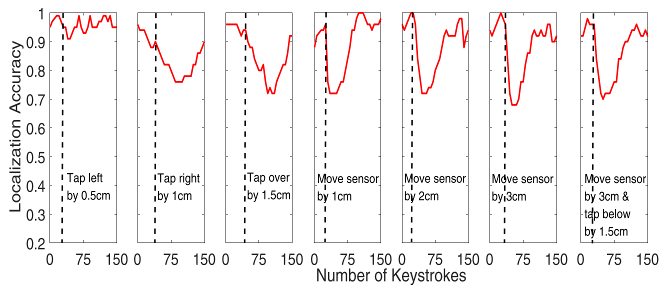

Fig. 21. Runtime calibration and adaptation scheme helps ViType to restore its high accuracy after deviation of watch and keystrokes (dotted lines denote the moment of displacement occurrence).

6.4 Cost

First, users only need to input 20×9 training instances at the beginning (all 30 subjects finished tapping within 3 minutes of our evaluation). Moreover, we measure the latency between each tap and ViType outputs (i.e., the localization result) on the screen. The results show that the classification latency is around 0.2 s with training set size of 20 using KNN, which is well below the human response time. [39] Therefore, there is no lagging effect when users use ViType. Noted that it requires a period of 0.6 s for retraining when users update the training set in our previous work [37] whereas no training time with KNN here. Furthermore, ViType with a low sampling rate (e.g., 600 Hz) is significantly more efficient to run on energy-limited smart wristbands compared with state-of-art approach Skinput. When it comes to the case of hardware expense, since we only employ one piezoelectric ceramic (at 0.15 dollar) and one amplifier (at 0.45 dollar), it is not expensive for a manufacturer to embed ViType on a smart watch.

7 User Study

To investigate the usability of ViType in real time, we conduct a number-entry field trial with ViType system in both static and moving status. For the purpose of comparison, concurrently, we pursue the same experiments with a COTS smart watch.

Experimental Setup. In this experiment, we recruit seven participants (5 females and 2 males) and compare the performance of ViType with a COTS smart watch (Huawei Watch2). Note that we turn the runtime calibration and adaptation scheme on in the following experiments. Considering user experience, we do not mark 9 points in participants' opisthenar for training or testing phase as we know that ViType has the resilience to positional deviation of taps (Section 6.3). However, we ask participants to try their best to fix the location of 9-keys and tap on the same location for each key to get better accuracy of ViType. To train the ViType system, participants are instructed to tap each key 20 times. To measure the input accuracy and input speed, participants are required to enroll in two sessions of testing: (1) using

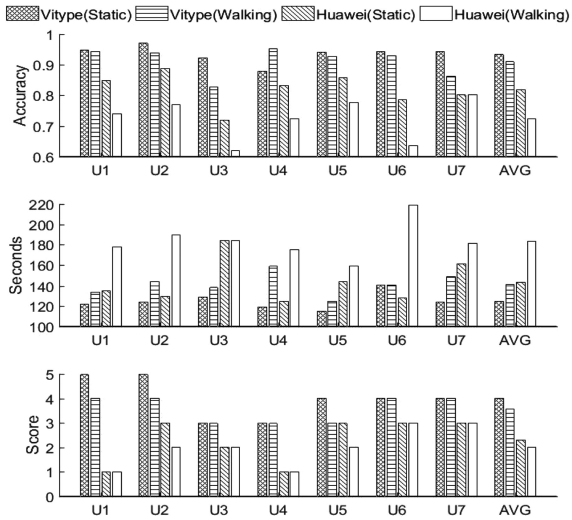

Fig. 22. Comparison of input accuracy, speed and user experience between ViType and HUAWEI Watch2.

Huawei Watch2 on-screen keyboard. (2) using ViType without key location marked, each is involved in typing 90 random numbers that are uniformly selected from 1-9. In each session, participants are asked to do the experiments in static and moving status respectively. They are instructed to correct their erroneous input and we count out the completion time when they finish typing all 90 random numbers. In the end, a questionnaire is filled in by each participant to evaluate user experience and give marks to both ViType and Huawei Watch2 (from 0 at the bottom to 5 at the top).

Localization Accuracy. Fig. 22 shows the result of static and moving status in terms of accuracy for two different types of input methods. The accuracy of ViType is 93.6 percent which is better than the accuracy (at 81.9 percent) of Huawei Watch2 due to the small size of the touch screen experienced by the participants. To consider the mobility problem in the user study, we can see that the result is similar with the one when people are sitting in an office (static status). This is because that the noise caused by human mobility is at low frequency (less than 5 Hz) [27] and we remove it via a 20 Hz Butterworth high pass filter. Furthermore, as Fig. 22 shows, there are two points that are worth to be noted: (1) user 3 has the poorest accuracy on Huawei Watch2 which is only 72 percent (this is because he has fatter fingers and it is difficult for him to hit the key location on the tiny watch screen), (2) during the user study, we observed that participants who have more practice on ViType achieve higher accuracy. Note that user 2 has the highest accuracy (at 98 percent) and is most familiar with the ViType system. We believe that users can improve the accuracy further if they become more and more familiar with ViType in terms of usage.

Input Speed. With respect to input speed, Fig. 22 illustrates that ViType outperforms the Huawei Watch2. Participants spent 124.8 seconds to type 90 random numbers with ViType, but 161 seconds with Huawei Watch2 on average. This is consistent with the limitations as we mentioned above. Participants waste much time on aiming at the tiny keypads of the watch screen and correcting their erroneous input back and forth because of the tiny touch screen. While ViType addresses the limitation of the interface with small size by shifting a virtual keyboard onto the opisthenar or in other words scaling up the layout of keyboard. In conclusion, ViType accomplishes its ultimate mission of improving the input speed of smart watches.

User Experience. We employ a piezoelectric ceramic vibration sensor to detect the keystroke signal, which usually requires the sensor to be well-contacted with users skin to attain good quality data. Therefore, we ask every participant to grade their feeling about the tightness and comfort degree by launching two Likert-scale questions: participants respond an average tightness degree of 2.4 (1 = loose, 5 = tight) and an average comfort degree of 4.4 (1 = uncomfortable, 5 = comfortable). We also ask them how many points would they respectively give to these two types of input methods overall. Fig. 22 is a histogram that shows the degree of satisfaction from this experiment. We can see that ViType (4 points) outperforms the Huawei Watch2 (2.3 points) in terms of this metric. Moreover, while asking about the acceptance of the 3-minute period of initial training, all of our subjects provided positive responses. In our view, the small size of watch screens is troublesome for the users when they type, while ViType enables users to type on the opisthenar which makes them more comfortable.

8 Limitation and Future Work

Vitype used a piezoelectric ceramic sensor to detect the tap-induced vibration, however, the commodity smart wristbands do not have this kind of sensors. To make our system be more convenient, we will try to use off the shelf sensors in the commodity wristbands to detect the vibration, such as inertial measurement unit. Besides, we will try to connect the smart wristbands with other smart wearable devices with Bluetooth such as AR/VR glasses, thus providing input systems for other smart devices. Note that users can type on the hand ”landmarks”, such as finger knuckles. In this way, users could type without seeing the hand. In addition, we will try to recognize more finger motions through vibration, such as left, right, up and down, to interact with devices. Finally, the unique vibration profile may provide identity characteristics. We will try to do research on the authentication for wearable devices base on the body vibration.

9 Conclusion

This paper presents a novel text input system for wristbands assuming the back of one’s hand as a virtual keyboard. Body vibration is detected by a small sensor embedded to the wristbands at a lower sampling rate, and then classified after a fine-grained features extraction. ViType achieves high keystrokes recognition accuracy and is also robust under several realistic text input conditions such as tapping with a different force, typing when walking and so on. The result demonstrates that ViType is more accurate and robust with more training samples collected under different conditions and the update of a training set can be done by the calibration and adaptation feature of ViType.

Acknowledgments

This research was supported in part by the China NSFC Grant (61872248, 61872246, U1736207), Guangdong Natural Science Foundation 2017A030312008, Fok Ying-Tong Education

Foundation for Young Teachers in the Higher Education Institutions of China(GrantNo.161064), Guangdong Special Support ProgramShenzhen Science and Technology Foundation JCYJ20170817095418831 and GDUPS(2015). This work is partially supported by the project "PCL Future Regional Network Facilities for Large-scale Experiments and Applications (PCL2018KP001)". This work is partially supported by Tianjin Key Laboratory of Advanced Networking (TANK), School of Computer Science and Technology, Tianjin University, Tianjin China, 300350.

REFERENCES

[1] V. Lakshmipathy, C. Schmandt, and N. Marmasse, "TalkBack: A conversational answering machine," in *Proc. 16th Annu. ACM Symp. User Interface Softw. Technol.*, Nov. 2003, pp. 41–50.

[2] R. Nandakumar, V. Iyer, D. Tan, and S. Gollakota, "FingerIO: Using active sonar for fine-grained finger tracking," in *Proc. CHI Conf. Human Factors Comput. Syst.*, May 2016, pp. 1515–1525.

[3] W. Wang, A. X. Liu, and K. Sun, "Device-free gesture tracking using acoustic signals," in *Proc. 22nd Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2016, pp. 82–94.

[4] C. Harrison, D. Tan, and D. Morris, "Skinput: Appropriating the body as an input surface," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, Apr. 2010, pp. 453–462.

[5] A. Abdullah and E. F. Sichani, "Experimental study of attenuation coefficient of ultrasonic waves in concrete and plaster," *Int. J. Adv. Manuf. Technol.*, vol. 44, no. 5/6, pp. 421–427, 2009.

[6] J. Wang, K. Zhao, X. Zhang, and C. Peng, "Ubiquitous keyboard for small mobile devices: Harnessing multipath fading for fine-grained keystroke localization," in *Proc. 12th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, Jun. 2014, pp. 14–27.

[7] C. Harrison, H. Benko, and A. D. Wilson, "OmniTouch: Wearable multitouch interaction everywhere," in *Proc. 24th Annu. ACM Symp. User Interface Softw. Technol.*, Oct. 2011, pp. 441–450.

[8] M. Ogata, Y. Sugiura, H. Osawa, and M. Imai, "iRing: Intelligent ring using infrared reflection," in *Proc. 25th Annu. ACM Symp. User Interface Softw. Technol.*, Oct. 2012, pp. 131–136.

[9] S. Nirjon, J. Gummeson, D. Gelb, and K. H. Kim, "TypingRing: A wearable ring platform for text input," in *Proc. 13th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, May 2015, pp. 227–239.

[10] W. Kienzle and K. Hinckley, "LightRing: Always-available 2D input on any surface," in *Proc. 27th Annu. ACM Symp. User Interface Softw. Technol.*, Oct. 2014, pp. 157–160.

[11] M. Prtorius, A. Scherzinger, and K. Hinrichs, "SkInteract: An on-body interaction system based on skin-texture recognition," in *Proc. IFIP Conf. Human-Comput. Interaction*, Sep. 2015, pp. 425–432.

[12] T. Sekitani, M. Kaltenbrunner, T. Yokota, and T. Someya, "Imperceptible electronic skin," in *Proc. SID Symp. Digest Tech. Papers*, Jun. 2014, vol. 45, pp. 122–125.

[13] 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. 27th Annu. ACM Symp. User Interface Softw. Technol.*, Oct. 2014, pp. 389–394.

[14] S. Nirjon, J. Gummeson, D. Gelb, and K. H. Kim, "TypingRing: A wearable ring platform for text input," in *Proc. 13th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, May 2015, pp. 227–239.

[15] 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 Int. Conf. Interactive Surfaces Spaces*, Nov. 2016, pp. 13–22.

[16] 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," *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 2, Mar. 2018, Art. no. 55.

[17] R. Xiao, G. Lew, J. Marsanico, D. Hariharan, S. Hudson, and C. Harrison, "Toffee: Enabling ad hoc, around-device interaction with acoustic time-of-arrival correlation," in *Proc. 16th Int. Conf. Human-Comput. Interaction Mobile Devices Serv*., Sep. 2014, pp. 67–76.

[18] S. Pan, C. G. Ramirez, M. Mirshekari, J. Fagert, A. J. Chung, C. C. Hu, J. P. Shen, H. Y. Noh, and P. Zhang, "SurfaceVibe: Vibration-based tap and swipe tracking on ubiquitous surfaces," in *Proc. 16th ACM/IEEE Int. Conf. Inf. Process. Sensor Netw.*, Apr. 2017, pp. 197–208.

[19] J. Liu, Y. Chen, M. Gruteser, and Y. Wang, "VibSense: Sensing touches on ubiquitous surfaces through vibration," in *Proc. 14th Annu. IEEE Int. Conf. Sens. Commun. Netw.*, Jun. 2017, pp. 1–9.

[20] K. Ali, A. X. Liu, W. Wang, and M. Shahzad, "Keystroke recognition using WiFi signals," in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2015, pp. 90–102.

[21] J. Liu, Y. Wang, G. Kar, Y. Chen, J. Yang, and M. Gruteser, "Snooping keystrokes with mm-level audio ranging on a single phone," in *Proc. 21st Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2015, pp. 142–154.

[22] J. Wang, R. Ruby, L. Wang, and K. Wu, "Accurate combined keystrokes detection using acoustic signals," in *Proc. 12th Int. Conf. Mobile Ad-Hoc Sensor Netw.*, Dec. 2016, pp. 9–14.

[23] G. Laput, R. Xiao, and C. Harrison, "ViBand: High-fidelity bio-acoustic sensing using commodity smartwatch accelerometers," in *Proc. 29th Annu. Symp. User Interface Softw. Technol.*, Oct. 2016, pp. 321–333.

[24] H. Wen, J. Ramos Rojas, and A. K. Dey, "Serendipity: Finger gesture recognition using an off-the-shelf smartwatch," in *Proc. CHI Conf. Human Factors Comput. Syst.*, May 2016, pp. 3847–3851.

[25] P. Marquardt, A. Verma, H. Carter, and P. Traynor, "(sp)iPhone: Decoding vibrations from nearby keyboards using mobile phone accelerometers," in *Proc. 18th ACM Conf. Comput. Commun. Secur.*, Oct. 2011, pp. 551–562.

[26] 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 Int. Workshop Sensor-Based Activity Recognit. Interaction*, Jun. 2016, Art. no. 8.

[27] L. Yang, W. Wang, and Q. Zhang, "VibID: User identification through bio-vibrometry," in *Proc. 15th ACM/IEEE Int. Conf. Inf. Process. Sensor Netw.*, Apr. 2016, pp. 1–12.

[28] J. Liu, C. Wang, Y. Chen, and N. Saxena, "VibWrite: Towards finger-input authentication on ubiquitous surfaces via physical vibration," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Nov. 2017, pp. 73–87.

[29] S. Pan, T. Yu, M. Mishakeri, J. Fagert, A. Bonde, O. J. Mengshoel, H. Noh, and P. Zhang, "FootprintID: Indoor pedestrian identification through ambient structural vibration sensing," *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 1, Sep. 2017, Art. no. 89.

[30] S. Pan, M. Mishakeri, P. Zhang, and H. Noh, "Occupant traffic estimation through structural vibration sensing," *SPIE Smart Structures Materials+ Nondestructive Eval. Health Monitoring*, vol. 9803, 2016, Art. no. 980306.

[31] D. Bales, P. Tarazaga, M. Kasarda, and D. Batra, "Gender classification using under floor vibration measurements," in *Proc. Dyn. Coupled Structures*, vol. 4, 2016, pp. 377–383.

[32] Z. Jia, A. Bonde, S. Li, C. Xu, J. Wang, Y. Zhang, R. E. Howard, and P. Zhang, "Monitoring a person's heart rate and respiratory rate on a shared bed using geophones," in *Proc. 15th ACM Conf. Embedded Netw. Sensor Syst.*, Nov. 2017, Art. no. 6.

[33] Z. Jia, M. Alaziz, X. Chi, R. E. Howard, Y. Zhang, P. Zhang, W. Trappe, A. Sivasubramaniam, and N. An, "HB-phone: A bed-mounted geophone-based heartbeat monitoring system," in *Proc. 15th ACM/IEEE Int. Conf. Inf. Process. Sensor Netw.*, Apr. 2016, pp. 275–276.

[34] W. Chen, M. Guan, L. Wang, R. Ruby, and K. Wu, "FLoc: Device-free passive indoor localization in complex environments," in *Proc. IEEE Int. Conf. Commun.*, Jul. 2017, pp. 1–6.

[35] M. Mirshekari, S. Pan, P. Zhang, and H. Y. Noh, "Characterizing wave propagation to improve indoor step-level person localization using floor vibration," *Sensors Smart Structures Technol. Civil Mechanical Aerosp. Syst.*, vol. 9803, Apr. 2016, available: https://doi.org/10.1117/12.2222136.

[36] W. Chen, Y. Lian, L. Wang, R. Ruby, W. Hu, and K. Wu, "Demo: Virtual keyboard for wearable wristbands," in *Proc. 15th ACM Conf. Embedded Netw. Sensor Syst.*, Nov. 2017, pp. 44–45.

[37] 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. 15th Annu. IEEE Int. Conf. Sens. Commun. Netw.*, Jun. 2018, pp. 1–9.

[38] I. Kononenko, "Overcoming the myopia of inductive learning algorithms with RELIEFF," *Appl. Intell.*, vol. 7, no. 1, pp. 39–55, 1997.

[39] W. Chen, L. Chen, Y. Huang, X. Zhang, L. Wang, R. Ruby, and K. Wu, "Taprint: Secure Text Input for Commodity Smart Wristbands," to appear in *Proc. 25th Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2019.

Wenqiang Chen received the master degree in College of Computer Science and Software Engineering from Shenzhen University in 2019. His research interests include smart sensing and mobile computing. He published papers in primer conferences, such as ACM Mobicom, etc. He won the best paper awards in IEEE SECON 2018. He is the inventor of 8 Chinese pending patent (2 are issued).

Maoning Guan received the BS degree in the College of Information Engineering from Shenzhen University, Shenzhen, China, in 2018. He is currently working toward the MS degree in the College of Computer Science and Software Engineering, Shenzhen University. His research interest includes Smart Sensing and Mobile Computing.

Yandao Huang is currently a third-year undergraduate in College of Computer Science and Software Enginerring, Shenzhen University. His research interest lies in mobile computing, Human-Computer Interaction (HCI) and Internet of Things(IoT).

Lu Wang is currently an assistant professor in College of Computer Science and Software Engineering, Shenzhen University, China. Her research interests focus on wireless communications and mobile computing.

Rukhsana Ruby received her Masters degree from University of Victoria, Canada on 2009, and PhD degree from The University of British Columbia, Canada on 2015. From the broader aspect, her resource interests include the management and optimization of next generation wireless networks. She has authored nearly 50 technical papers of well-recognized journals and conferences. She has served as the lead guest editor for the special issue on NOMA techniques under EURASIP JWCN on 2017, and currently is serving as an associate editor to this journal. Besides, she has also been serving as a technical program committee member in various reputed conferences.

Wen Hu is a faculty member at School of Computer Science and Engineering, the University of New South Wales (UNSW). Much of his research career has focused on the novel applications, low-power communications, security and compressive sensing in sensor network systems and Internet of Things (IoT). Hu published regularly in the top rated sensor network and mobile computing venues such as ACM/IEEE IPSN, ACM SenSys, ACM transactions on Sensor Networks (TOSN),Proceedings of the IEEE, and Ad-hoc Networks.

Kaishun Wu received his Ph.D. degree in computer science and engineering from HKUST in 2011. After that, he worked as a research assistant professor at HKUST. In 2013, he joined SZU as a distinguished professor. He has co-authored 2 books and published over 100 high quality research papers in international leading journals and primer conferences, like IEEE TMC, IEEE TPDS, ACM MobiCom, IEEE INFOCOM. He is the inventor of 6 US and over 90 Chinese pending patents. He received 2012 Hong Kong Young Scientist Award, 2014 Hong Kong ICT Awards: Best Innovation and 2014 IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award. He is an IET Fellow.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/csdl.