VibRing: A Wearable Vibroacoustic Sensor for Single-Handed Gesture Recognition

BU LI, University of British Columbia, Canada
XINCHENG HUANG, University of British Columbia, Canada
ROBERT XIAO, University of British Columbia, Canada

Fig. 1. **(a) Example uncommon gestures** included in our gesture design set. Top: Gesture "PN_rub", where the index fingernail rubs (scratches) the thumb's pad; bottom: "NP_flick", where the thumbnail presses on the middle finger's pad and flicks away. **(b) VibRing's wearable platform** including an armband encasing the ESP32 Adafruit Feather chip and a thumb ring consisting of a DC bias circuit and a conditioned piezoelectric disk. **Demonstrated online applications** including **(c)** a music player, **(d)** a phone dialer, and **(e)** an AR painter.

Single-handed gestures offer rapid and intuitive interactions for input in interactive applications ranging from smartwatches and phones to augmented reality. Past research has explored using computer vision or inertial measurement units (IMUs) to sense such gestures, but these sensing modalities can be variously subject to occlusion, high power consumption, or sensitivity to random motion. In this work, we explore passively detecting the vibroacoustic signature of subtle single-handed gestures through a wearable piezoelectric sensor, providing a robust, low-power sensing modality. We present (1) a hand-gesture design framework encompassing a large set of subtle, rapid single-handed gestures which balance comfort and vibroacoustic distinguishability, (2) VibRing, a lightweight wireless hand-gesture sensing platform, leveraging a single finger-worn vibroacoustic sensor, and (3) a multifaceted system evaluation where we consider several aspects - general usability, tolerance to variance, user adaptability, and extended usage. Our results demonstrate that VibRing can support an 11-gesture set with a general accuracy of 94.2% and low-performance variance across multiple days (90.2% accuracy in cross-day validation). To support a new user, VibRing requires only 10 minutes of training data to achieve an accuracy of 92.7%. We also tested the extended use of VibRing in an office study where users performed periodic gesture inputs during typical office tasks with real-time classification, achieving a true-positive rate of 90.9%. Finally, to demonstrate the utility of VibRing, we present three examples of applications which benefit from our subtle gesture interactions.

CCS Concepts: • **Human-centered computing** → **Gestural input**.

Additional Key Words and Phrases: single-handed gesture design, gesture recognition, acoustic sensing

Authors' Contact Information: Bu Li, University of British Columbia, Department of Computer Science, Vancouver, British Columbia, Canada, buli@cs.ubc.ca; Xincheng Huang, University of British Columbia, Department of Computer Science, Vancouver, British Columbia, Canada, xchuang@cs.ubc.ca; Robert Xiao, University of British Columbia, Department of Computer Science, Vancouver, British Columbia, Canada, brx@cs.ubc.ca.

1 Introduction

Compared with other interactions, single-handed gestures are natural, convenient and eyes-free, leading to their use in many applications. Gesture sets must be designed with the sensing approach in mind, and the choice of sensor thus limits what gestures can be sensed. Sensing modalities explored for hand gesture recognition in past work include capacitive sensing [47, 48], inertial measurement units (IMUs) [11, 26, 28], proximity/IR [42, 50], electrical field [22, 45, 49, 59], vision [8, 57], mechanical [29, 30], active acoustics [35, 56] and other sensing techniques [16, 33]. However, we find that passive vibroacoustic sensing is relatively under-explored, as reviewed by Vatavu et al. [44]. Existing systems using this modality mainly include on-surface interactions and event detection [18].

In this work, we propose VibRing, a wearable gesture-sensing ring that demonstrates the viability of vibroacoustic sensing for sensing subtle single-handed gestures. We harness a sensitive surface-coupled vibroacoustic sensor – a conditioned piezoelectric (piezo) disk – to pick up gesture-induced vibrations propagated along a hand (Fig 2). Our sensing approach offers several advantages over other techniques: power efficiency, robustness, and support for fast and subtle interactions. First, piezos are purely passive devices that need no external power to generate a signal, allowing us to passively trigger on gestural input and substantially reducing power requirements. Second, piezos only perceive contact vibrations when coupled to the skin, making them resilient to acoustic noise. They are also not affected by occlusion or orientation, allowing the user to naturally perform gestures from any starting pose. Finally, while IMU sensors typically operate at low sampling rates, piezos can capture fine-grained features from rapid interactions using high-frequency sampling.

To acquire a set of distinguishable single-handed gestures, we propose a design framework that produces gestures with distinctive vibrations based on our sensing principle. We identify three regions of the finger – nail (N), pad (P), and knuckle (K) – as well as three actions – click (tap), flick and rub. We obtain gestures that produce distinctive vibroacoustic profiles by conducting different actions with different pairs of finger regions. Combinatorially, our design framework produces 27 gestures, of which 17 were identified as comfortable and subtle through a preliminary study. All gestures can be executed rapidly: the duration of each gesture ranges from 82 to 225 milliseconds (mean 140 ms, std 41 ms). The signal profiles for three representative gestures are shown in Fig. 3.

However, not all 17 gestures are usable in practice due to signal similarities. To improve recognition performance, we removed 6 gestures to produce a final set of 11 distinguishable gestures, which we evaluated in more detail: performance across multiple days, adaptability to new users, and usability in extended real-time usage. Finally, we developed 3 online applications highlighting different use cases.

To summarize, our contributions are as follows:

- A systematic gesture design approach that produces a large set of subtle and rapid single-handed gestures.
- A power-efficient system that harnesses the vibroacoustic profiles to classify gestures. The general classification accuracy of an 11-gesture set reaches 94.2%, and the cross-day accuracy reaches 90.2%.
- A user adaptability evaluation that shows that VibRing adapts well to new users with limited data (92.7% accuracy with 10 minutes of collected data).
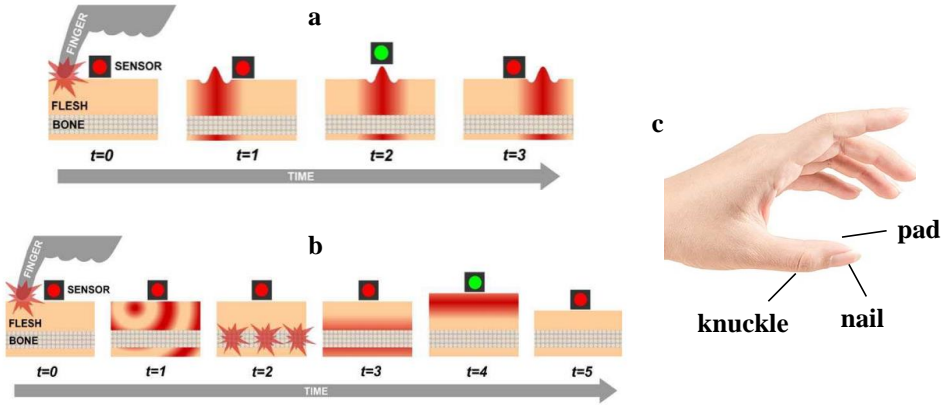
Fig. 2. **(a)** shows the surface wave propagation of vibroacoustic signal [14] while **(b)** shows the bone conduction propagation. In **(c)**, we show three main areas of the hand that have different vibroacoustic profiles.

- An extended usage evaluation in an office environment that demonstrate VibRing's potential to support robust inputs when the usage is interleaved with daily tasks.
- Three online applications showcasing potential use cases.

2 Related works

2.1 Micro-gestures

Hand gestures have received much attention within HCI, particularly due to their suitability for enhancing communication as sign languages [21, 43] and extending interactions as extra input channels as presented in TapSkin [54], AuraSense [62], Apple Watch[1], HoloLens[2], and many other systems. Hand gestures can be broadly classified into two categories: bimanual and single-handed gestures. While bimanual gestures involve hand-to-hand interactions (e.g., drawing [36, 46, 60], tapping [14, 53, 54] and rubbing [26]), single-handed gestures emphasize finger-to-finger interactions on one hand.

Subtle single-handed gestures are also known as *micro-gestures* [6, 7, 38, 39]. The rapid, eyes-free and low-engagement nature of micro-gestures makes them excellent input alternatives when users are occupied (e.g. exercising, driving or carrying objects) or when unobtrusive performance is preferred (e.g. in social settings). For instance, MyoSpring [30] sensed finger-flexion gestures that could be used to subtly respond to a phone call (reject or mute). Tomo [59] supported hand poses for navigating through messages, and Herath et al. [17] applied micro-roll gestures to map navigation. Boldu et al. [2] specifically enabled micro-gesture interactions in athletic activities like running. Works like FingerInput [39] and Transferable Microgestures [19] instead exemplified audio applications - controlling a music player or composing music pieces with a sequence of gestures. These works demonstrated the affordances and applications of micro-gestures to different contexts.

According to μglyph [6], micro-gestures can be further divided into free-hand and grasping gestures based on whether an object is held or not. For VibRing, we explore free-hand micro-gestures, designing specifically for fast interactions, robustness and long-term usage. We show that VibRing can receive inputs while in motion (music player application), support complex interfaces

---

[1]https://www.apple.com/ca/watch/
[2]https://www.microsoft.com/en-ca/hololens
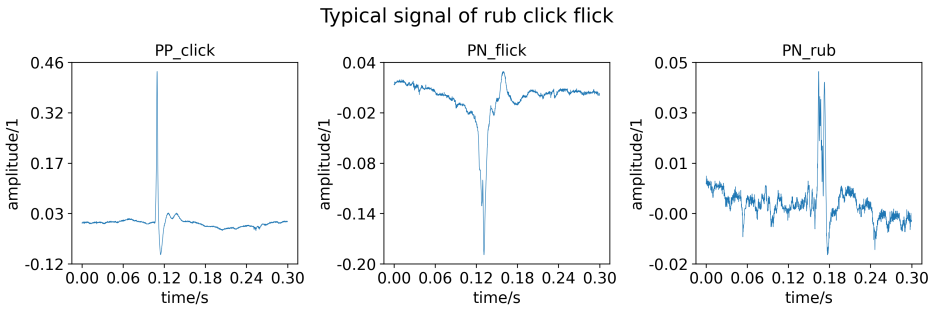
Typical signal of rub click flick

Fig. 3. In VibRing, we take advantage of three main action categories - click, flick and rub. Gestures within each category should share a similar signal profile. We demonstrate 3 typical signals from each category: from left to right, PP_click, PN_flick and PN_rub. The differences are visually obvious, suggesting the basic distinguishability of the gestures.

(phone dialer application), and synergize with existing gesture systems (painting application in HoloLens), demonstrating the flexibility of our approach.

2.2 Gesture Design for Subtle Interactions

Gesture design varies widely, depending on the context of use, sensing principle, and prevailing norms. For instance, gestures used when interacting with a phone are tapping [9], pressing, scrolling and zooming gestures [34], while in contrast, AR applications favour gestures such as pointing, gripping and tapping [5, 23].

To provide good performance, gesture design is typically coupled to the sensing principle. Capacitive sensing is sensitive to finger tapping and direction of movement. Works like [10] thus adopted a set which includes tapping and swiping gestures. Proximity sensing requires distance profiles, so zSense [50] and ThumbTrak [42] took advantage of swiping and touching gestures, which easily produce varying distance measurements. Active acoustic sensing picks up gesture features from the sound propagation path. Hence, FingerPing [56] designed the hand poses to explicitly form distinct paths from the speaker to the receiver. Vision requires unoccluded line-of-sight. Consequently, CyclopsRing [8] mounted a fish-eye camera between adjacent fingers to sense gestures happening in front of it. Apart from the general approaches above, more explicit designs for gestures were observed in other sensing platforms. In MyoSpring [30], the sensor perceived the pressure of the tendons around the wrist, so gestures were mainly finger flexions that move different tendons. Lu et al.'s system [32] collected surface electromyographic (EMG) signals and acceleration, and thus the proposed set of gestures included hand flexions that generate EMG signals and in-air motions that produce distinct acceleration profiles. Finally, Z-Ring[45] senses bioimpedance when a finger touches different surfaces, and thus adopts multiple-tapping and swiping hand gestures.

In VibRing, we also designed a gesture set tailored to the VibRing sensing approach. Our set considers the different vibroacoustic profiles of different finger regions (i.e. pad, nail and knuckle, also examined in TapSense [13]) and actions (i.e. click, flick and rub) to create distinctive signals. Additionally, we explicitly chose rapid gestures which can be performed in less than half a second. This property of fast interaction aligns with the property of subtlety, affording reduced social obtrusiveness and higher input rates. As vibroacoustic sensors can easily provide high sampling rates, those fast interactions can be captured with less complexity and power consumption.

2.3 Passive Vibroacoustic Sensing

Passive vibroacoustic sensing harnesses a surface-contacting sensor to "hear" vibrations propagated through either the surface or body of the object. Events can then be classified by the characteristics of the received acoustic profile. Unlike optical methods, passive vibroacoustic sensing does not depend on lighting conditions. Furthermore, vibroacoustic sensing supports high sample rates (>10 kHz) with ease, whereas a typical higher-end commercial IMU like STMicroelectronics LSM9DS1 [41] only supports ~500 Hz. In academic research, although systems like ViBand [26] demonstrate higher-speed IMU sensing (~4 kHz), it is not straightforward nor low-power to achieve. Moreover, vibroacoustic sensing naturally rejects airborne vibrations (such as speech and background sounds) and lends itself to simple hardware design, reducing the complexity of the finger-worn device. Thus, passive vibroacoustic sensing has found use in detecting on-surface interactions [4, 12, 13, 31]. To name a recent effort, SAWSense [18] successfully repurposed a voice pick-up unit [40] (VPU) to detect events that emit surface waves, enabling it to differentiate 16 on-desk events and 6 hand-to-desk gestures accurately.

Extending the technique to hand gesture recognition, FingerSound [55] leveraged the vibroacoustic profiles of drawing gestures to reject noise and segment each gesture instance. Skinput [14] introduced a system that perceives taps on different areas of an arm through received vibrations. Other vibroacoustic systems like [61] classified a small gesture set (5 classes) using vibroacoustic profiles. However, none of these have addressed the viability of using pure vibroacoustic profiles to enable a rich gesture set. Our work fills this gap by successfully classifying 11 subtle gestures, with accuracy reaching 94.2% among 15 participants. We also performed a thorough evaluation encompassing cross-day, adaptability, and extended usage.

2.4 Long-Term Real-World Use

Although many gesture recognition systems have demonstrated great success in experimental setups, their performance in real-world scenarios remains questionable. A key challenge is determining how well they can tolerate noise from daily activities. To address this, some systems implement wake-up mechanisms to distinguish intentional input from noise. For example, FingerPing [56] suggested that users can touch three phalanges to activate the system. Other work [1] also noted using a series of gestures as a sign of input intent. CyclopsRing [8] stated that a hand drawing session can be initiated by a simpler gesture - bending the thumb. In addition to wake-up design, some systems consider modeling the noise and rejecting it. For instance, Z-Ring [45] collected data from participants' interaction with smartphones, wallets and desks to define a "null" gesture which represents all irrelevant conduct. However, without formal evaluation, it remains unclear whether these approaches would be practical in real life.

To showcase the system's noise resilience, efforts have now been put into conducting more thorough evaluations. SenseIR [33] realized that arm movement would influence gesture sensing in practice, and thus examined gestures under different arm poses. FingerSound [55] evaluated usage under "noisy" conditions, where participants gestured while walking. However, SenseIR did not examine performance during dynamic arm motions. FingerSound evaluated motions, but only in the case of walking, and the experimenter implicitly controlled the participants' walking speed.

For VibRing, we evaluated the proposed noise-resistance mechanism in a more realistic scenario. We first developed a sleep-and-wake-up mechanism to help VibRing explicitly separate daily tasks and gesture input sessions. We then evaluated how well VibRing perceived input initiations and rejected noise in an office setup through a 5-hour experiment and 1-hour user study with 5 participants. The study setup was more free-form, and various daily tasks were included, such as

	Thumb Knuckle									Thumb Pad									Thumb Nail								
	KK-rub	KK-flick	KK-click	KP-rub	KP-flick	KP-click	KN-rub	KN-flick	KN-click	PK-rub	PK-flick	PK-click	PP-rub	PP-flick	PP-click	PN-rub	PN-flick	PN-click	NK-rub	NK-flick	NK-click	NP-rub	NP-flick	NP-click	NN-rub	NN-flick	NN-click
TLX	4.4	5	4	1.8	3.2	1.6	4.2	2.6	2.2	2	3	1.6	1.2	2.4	1	2	1.4	2.2	4.6	4	4.2	2	2.8	2.2	3	4.2	3.6
OBS	3	3.4	2.2	1	4.6	1	2.2	1	1.4	1	1	1	1	1.8	1	1.8	1	1	1.4	1	1	1	1	1	2.6	3	1.8
FREQ	1	1	1	1.4	1	1.4	1.2	1.6	1.2	2.2	2	1.2	3.6	3.6	4	2	3.6	2.2	1.2	1	1	1.6	1.4	2	1	1	1
Yellow picked																									√		√

Fig. 4. All 27 gestures underwent the acceptance evaluation. From 1 to 7 in TLX, the score represents most accepting to least accepting; from 1 to 7 in OBS, the score represents perfect to incorrect performance; from 1 to 7 in FREQ, the score represents least frequently used to most frequently used. After analysis, the red gestures are immediately eliminated because of high demand and discomfort, while green gestures are directly adopted.

texting, keyboard typing, walking, jumping jacks, etc. The results show that VibRing can work for a long time with very few false positives and false negatives.

3 Gesture Design

One major gesture design criterion was obtaining clear and distinct vibroacoustic signals. As seen in [13], fingernails [20, 27], knuckles and finger pads can produce many hand interactions featuring distinct vibroacoustic profiles. We envisioned that this characteristic could also be extended to single-handed gestures, where we conduct different actions with different pairs of finger regions to produce unique vibration profiles. We thus crossed the pairs of finger regions with a set of actions - click (tap), flick and rub. Those actions are rapid and hence promote more subtle interactions. We chose to place the sensor onto the thumb because it is the most dextrous finger; additionally, as it is the shortest finger, it yields a shorter propagation path and thus less signal attenuation.

After establishing the gesture design heuristics and the sensing location, we proposed the design framework. Similar to other systematic gesture invention processes [39, 52], we defined gestures by a combination of three factors: the component of the thumb (pad, knuckle, or nail), the component of the other finger, and the action (rub, click or flick). In total, this produces 3x3x3 = 27 gestures. We abbreviate each gesture by concatenating the thumb component (P, K or N), the other finger's component, and the action. Thus, for instance, clicking the index finger pad to the thumb's pad (pinching) is called PP_click in our study.

However, not all gestures produced are comfortable and easy to perform. Prior work [7] pointed out that gesture design processes that fail to consult end users can lead to worse acceptance. Hence, we conducted a brief acceptance study with 5 participants (3 male and 2 female) to identify usable gestures. They learned and practiced the 27 gestures while we observed and noted how well they performed. Then, they scored each gesture with an adapted NASA TLX form [15], where we added an extra dimension called "frequency" to measure how often users might perform the gesture in their daily lives. Each TLX category was scored on a 7-point Likert scale, with lower scores indicating less demand, more acceptance or less frequent use. For the observation sheets, we coded the gesture performance on a 1-7 scale, with 1 meaning perfect performance and 7 meaning unable to perform accurately at all.

The final results are shown in Fig. 4. By taking the maximum out of all categories in the TLX form, we obtained the approximated load level. We then averaged these maximums across all participants to compute the final estimated load of a gesture ("TLX" row). We separately averaged the frequency and observation scores to produce the "FREQ" and the "OBS" rows. The results identified three categories of gestures, marked with different colors - green, red and yellow. The red gestures ("TLX" score equal to or higher than 4) were immediately eliminated due to high demand, while the green gestures ("TLX" score lower than 3) were accepted for further study because of high acceptance.
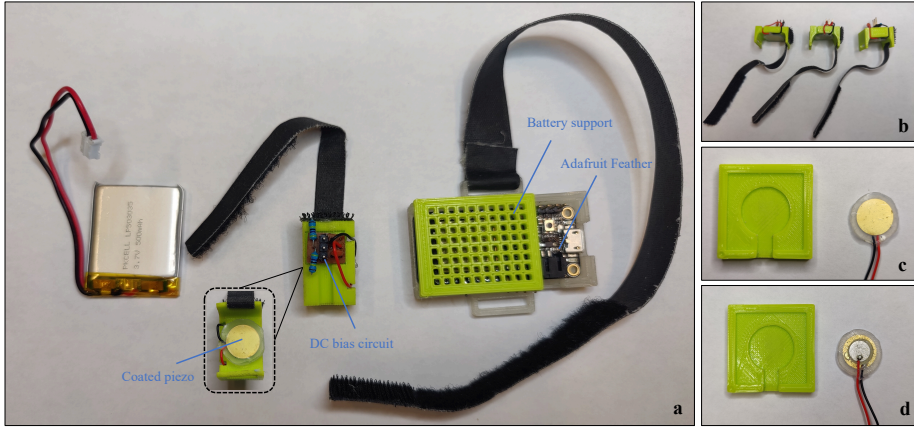
Fig. 5. We present the detailed view of the VibRing prototype in **(a)**. The composition of the platform is discussed in the hardware section. **(b)**: 3 different sizes of thumb rings (from left to right: large, medium, small). **(c)**: mould for the silicone rubber coating. **(d)**: the rubber-coated piezo disk.

Yellow gestures lie between the two groups above and were manually inspected; we eliminated KP_flick because it had a high "OBS" score and PK_flick because it failed in the segmentation test.

Hence, a total of 17 gestures (15 green gestures plus 2 yellow gestures) were considered for subsequent evaluation: KP_click, NP_rub, KN_flick, PK_rub, PK_click, KP_rub NN_click, PP_rub, PP_flick, PN_click, NN_rub, KN_click, PP_click, PN_rub, PN_flick, NP_flick, and NP_click. For analysis purposes, we also collected a "background" gesture in studies, in which users move their fingers slowly and freely without finger-to-finger interactions. This gesture captures the usual background noise arising from daily interactions, which helps eliminate false positives. We also extracted a noise threshold from this gesture and used it for gesture instance segmentation. Samples of this "background" gesture can be seen in the supplemental video.

4 Implementation

4.1 Hardware

Sensor Selection. There are several options to sense vibration: contact microphone, piezo, and inertial measurement units (IMUs). Although well-engineered contact microphones such as voice pick-up units (VPUs) offer a flat response over a wide band (i.e., the human hearing range), they require dedicated post-processing circuits, making them less passive than piezos. Hence, choosing piezos is aimed at trading performance for low power consumption, benefiting the subsequent long-term design. As another alternative, IMUs, a widely used approach, have a relatively low sampling rate (e.g., the STMicroelectronics LSM9DS1 supports up to ~500 Hz), whereas piezos can easily achieve much higher sampling rates (over 10 kHz). In our design, we emphasize rapid gestures that finish within fractions of a second. Accordingly, our signal analysis window is short, and only a high-sample-rate sensor can capture enough features. To illustrate the sampling differences between piezo and IMU when sensing quick events, we conducted a comparison experiment, as detailed in section 4.2.
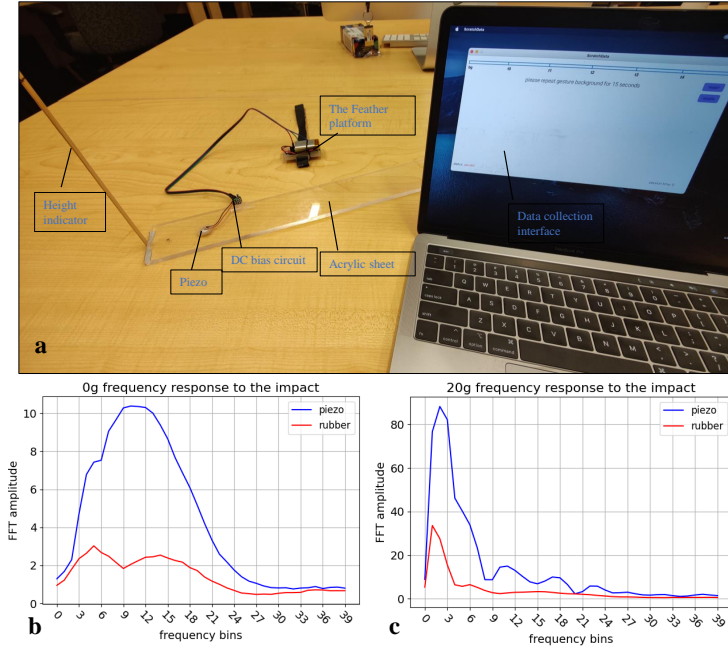
Fig. 6. **(a)** The setup of the ball-drop experiment. We drop a ping-pong ball onto the acrylic sheet, while the sensor attached to the sheet senses a pulse-like signal. **(b, c)** presents the frequency response of the piezo under different weighting configurations. **(b)** shows that both the coated piezo and pure piezo are more sensitive to mid-frequency bins, yet the coated piezo has a unique "pit" at the center. On the contrary, in **(c)**, we observe that the two curves have more similar contours, and that the energy concentrates more in the lower bins.

Overview. VibRing, pictured in Fig. 5, consists of a finger-worn ring connected via wires to an arm-mounted sensing board. The armband comprises several parts - the ESP32 Adafruit Feather chip, a 3D printed holding case with a velcro strap, and a 3.7 V 500 mAh LiPo battery. The armband communicates wirelessly (i.e. WiFi) with a remote server (a laptop), which provides signal processing and machine learning features. The finger ring is a 3D-printed device, which we designed in three different sizes to fit different fingers (Fig. 5b). The 12 mm piezo disk is inlaid into a slot on the bottom and covered with silicone rubber. It is then connected to a small DC-bias circuit, which brings the signal into the range expected by ESP32's ADC and then to the arm-worn sensing board.

The silicone rubber layer serves several purposes: it cushions the skin from the metal disk, provides vibroacoustic coupling between the skin and the sensor, and increases friction to reduce motion-induced noise. We use a 3D-printed mould (Fig. 5c) to shape Ecoflex-10 rubber into a small disk of a controlled thickness (0.8 mm). However, one downside of this layer is that it also attenuates desirable signal vibrations, and thus, we performed an experiment (Section 4.2) to characterize this dampening effect.

4.2 Experiments

VibRing is designed to sense rapid single-handed gestures, as short as 80 ms in duration. To validate the selection of sensor and other core parameters, we conducted a set of experiments. For design convenience, the gesture signals are represented by pulse-like impacts that are generated by
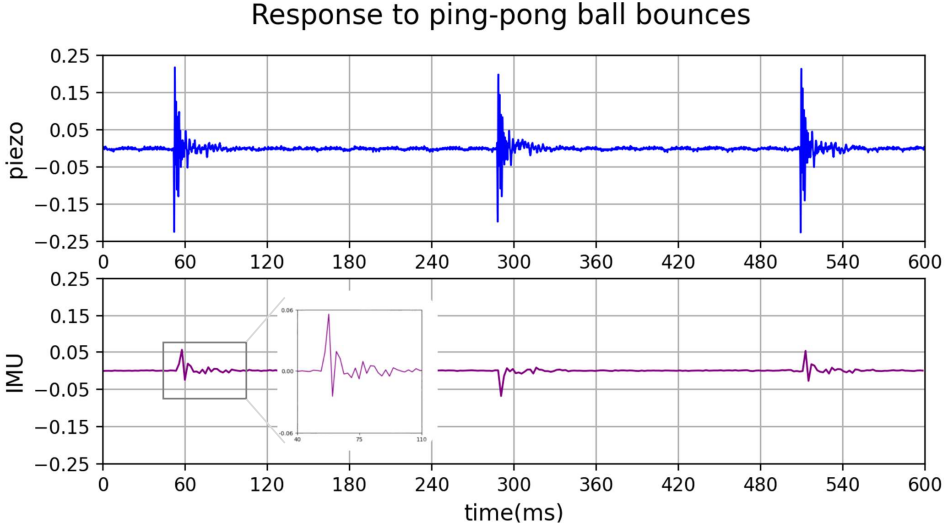
Response to ping-pong ball bounces

Fig. 7. In this experiment, piezo presents rich features (e.g., signal peaks and valleys) and a consistent response to 3 similar bounces, yet the IMU fails.

dropping a ping-pong ball onto an acrylic sheet (shown in Figure 6a). The ball's impact is short (dampens in 100 ms) and possesses rich features (e.g., peaks and valleys), similar to our gestures. Therefore, if a design can perceive the ball impact, it should give a good response to the actual gestures as well.

Sensor Selection Experiment. In this experiment, we compared piezo and IMU's differences in sensing the ball impact. We attached a 12 mm diameter piezo and an Arduino Nano 33 BLE chip, equipped with an STMicroelectronics LSM9DS1 IMU, on the acrylic sheet. We aligned the two sensors at the edge and put the same weights (10 g) on them. We then dropped the ball from a height of 100 mm onto a point 40 mm away from the two sensors. The piezo was connected to ESP32 and sampled at 8 KHz. The IMU was sampled at the maximum practical rate of 468 Hz, similar to the max rate (500 Hz) reported by SAWSense [18] for this IMU. The response to 3 ping-pong ball bounces is presented in Fig. 7. We observe that piezo's responses are consistent and capture rich information (e.g., the fading fluctuations), whereas the IMU responds differently to each bounce, rendering rough bounce images. Thereby, we conclude that the piezo is more suitable than IMU for sensing quick events, such as rapid gestures.

Optimization Experiment. Despite piezo's advantage, we found three factors that can impact VibRing's system sensitivity: the DC bias circuit, rubber layer, and contact condition. The DC bias circuit adjusts the signals so that they lie within the ADC's measurement range, but it also constantly drains power and introduces a high pass filter that influences sensitivity. After examination, we used 4.6 MΩ resistors coupled with a 1 nF capacitor to minimize constant power dissipation and set an appropriate time constant to decide the filter frequency.

While different contact conditions (i.e., pressure between the sensor and thumb skin) change VibRing's sensitivity, the rubber layer also impacts the perception of vibrational signals. To measure those effects, we carried out another ball-drop experiment, where we compared ball impact responses in different setups – with or without the rubber layer – and emulated two distinct contact conditions by placing either no additional weight or a 20 gram weight on top of the sensor. The results are shown in Fig. 6 b and c. Though the rubber layer dampens signals in all frequencies,

it is still acceptable because the high sensitivity of the DC circuits can compensate for the loss. Regarding contact conditions, with 0 grams, the main frequency response is concentrated in the middle bands, yet the silicone rubber's curve presents a unique "dip". This difference is not observed in the 20-gram setup, where the two curves follow each other well, and the main responses appear in lower bands. Since our gestures require better sensitivity in lower bands, we will need to maintain enough pressure between the sensor and the thumb skin by appropriately tightening the sensor's strap.

4.3 Software

We developed firmware using the Arduino SDK to drive the ESP32. A laptop runs a Processing script to communicate with it via WiFi, visualizing and saving the sensor data in real-time. After collecting data, a Python project is responsible for data analysis, including filtering, instance segmentation, feature extraction, and classification.

4.3.1 Data Collection. We observed that the acoustic profile for all gestures possesses rich components under 3 kHz, a fact that lines up with previous works [13, 18], where the authors studied acoustic frequency profiles of nails, pads and knuckles while interacting with daily objects and surfaces. Hence, we set the sample rate to 8 kHz, enough to cover frequencies under 3 kHz. After the Processing script connects to ESP32, the program first collects participants' demographic information. Data collection then consists of a number of sessions. In each session, gestures are presented in a random order. For each gesture, users are shown a text prompt and pre-recorded video describing the gesture, then prompted to perform the gesture repeatedly, 15 times over approximately 15 seconds. The 15-second recording is saved in an audio file and post-hoc segmented into individual gestures for analysis. In the actual study, to mitigate learning effects, we discarded the data from the initial sessions.

4.3.2 Post-processing and Segmentation. We developed a Python script to process all gesture signals. The raw signals have rich low-frequency components, which interfere with instance segmentation but can be good sources for extracting features. Therefore, during segmentation, we applied a 30 Hz high pass filter to sharpen transitions. We then used the extracted threshold from the "background" gesture to perform energy-based instance segmentation [55]. This process detects rising and falling edge pairs in the short-time Root Mean Square Energy (RMSE) representation and outputs the center of each pair as each gesture instance's location. We extracted a fixed-length window of 2400 points (~300 ms) centered on this location from the raw, unfiltered signal as the gesture instance. Fig. 8a, b present the segmentation plots of some gestures. To train bigger models, we also considered data augmentation. Fig. 8c shows three augmented samples generated by shifting one signal instance by different temporal offsets. This enlarges the data set three-fold while helping the classifier be more resilient to temporal variations in segmentation.

4.3.3 Feature Extraction and Classification. We evaluated two machine learning approaches: Support Vector Machine (SVM) and Convolutional Neural Network (CNN). We selected SVM to represent traditional ML methods because it has been widely used in past work and provides good efficiency. We opted to compare it against a larger, more complex CNN model that aims for higher accuracy but requires more computational power.

We started with a set of time- and frequency-domain features inspired by ViBand [26]. We considered statistics for the time domain – sum, mean, and standard deviation. For the frequency domain, we chose two different temporal resolutions to capture different points on the time-frequency resolution tradeoff. Specifically, we selected 11 windows for FFT: 1 across the entire 2400-point signal, 3 overlapping sliding windows of 1200 points and 7 windows of 600 points.
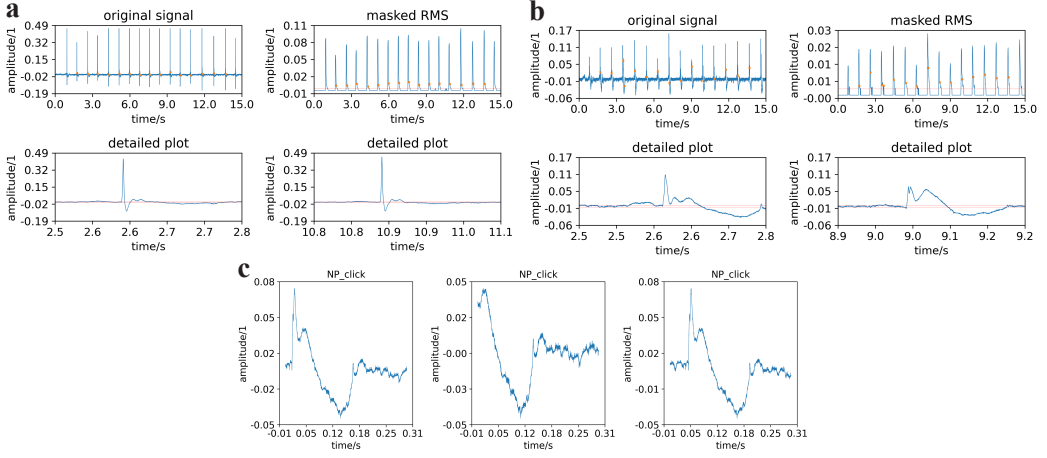
Fig. 8. Segmentation results of some example gestures. The red dashed lines show the threshold that we extract from the "background" gesture, which is used to zero out the baseline noise. **(a, b)** present the segmentation of gestures "PP_click" and "NP_click". **(c)** shows the data augmentation we applied, where the sliding window creates 3 copies of one signal.

We applied Hann windowing and transformed them with the FFT. In addition to mean, standard deviation, min, max, and sum of the FFT signal, we also considered peak information that includes the first 5 peak values and their frequency index in each FFT. Then, we divided each FFT profile into 12 bins with logarithmic spacing, subsequently computing the band ratios of each pair of bins. In total, we had 3 time-domain features, $(5 + 10) \times 11 = 165$ statistical spectral features, and $C(12, 2) \times 11 = 726$ band ratios, producing an 894-dimension feature vector for each signal. For SVM classification, we chose an RBF-kernel SVM and used hyperparameters $\gamma = 0.001$ and $C = 10$, tuned by pilot study data collected from the authors.

There are two concerns about the mentioned feature extraction: SVM is not good at processing high dimensional data, and we have relatively few time-domain features, omitting beneficial time-domain information; for example, local phase information can be used to differentiate between a rising and a falling peak. To address these problems, we built a CNN as shown in Fig. 9. This new model not only handles high-dimensional data but can also extract additional time-domain features. As a simple example, the convolutional filter $[-1, 1]$ extracts the first-order derivative of the 1D sequence, and could conceivably be learned by the CNN. Similarly, we expect the training process of CNN to form more useful convolutional filters to encode and abstract more time-domain information. After 3 convolutional layers, the model produces a flattened 1D activation with 896 dimensions. These are concatenated to the previously mentioned spectral features and processed through a final linear layer for classification.

4.3.4 Development for Long-term Use. The minimal, wireless and skin-friendly design has the potential for long-term usage. However, such a setup would still be limited by the capacity of our battery. We use a 500 mAh battery, comparable with the batteries in commercial smartwatches. Although the passive piezoelectric sensor has negligible power consumption compared with other active sensing modalities, a continuously running system still drains the battery after around 2 hours due to ESP32's constant sampling and WiFi connection. To improve battery life, we developed a sleep-and-wake-up mechanism, which passively detects certain gestures and then wakes up the main processor of ESP32 to sample the subsequent signal and reject false-positive events. Such a
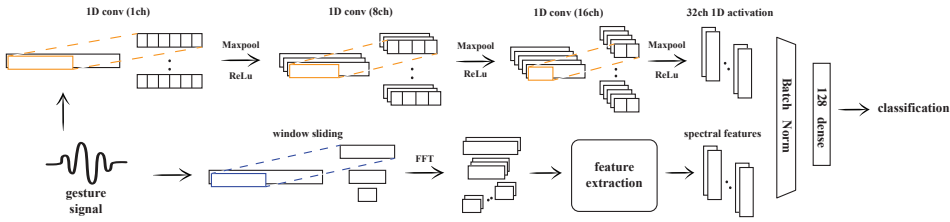
Fig. 9. Architecture of the VibRing CNN. The gesture signal passes through 3 convolutional layers, ultimately providing 32 channels of 1D activation. The gesture signal is also processed by spectral feature extraction, producing features that address different window sizes. Finally, the two groups of features are flattened, concatenated and normalized before they enter the decider (120-neuron dense layer).

process ensures that ESP32 is normally in "deep sleep" and consumes nearly zero power but can be woke up by pulse-like voltage signals inherently generated by piezo disks upon performing certain click gestures.

Specifically, we employed two consecutive PP_click events, where the first click wakes ESP32 up, and the second is sampled and classified by the remote server to either initiate input sessions or reject false positives. If a valid second click is detected, a 10-second input session will be initiated. Any input during the period upon detection will renew the session to allow successive inputs. If no valid second click or no input is presented in 10 seconds, the server will instruct the ESP32 to sleep.

The second PP_click gesture is crucial for reducing false positives. However, there's a chance that VibRing could miss it, as ESP32 takes time to reconnect to the remote server after waking up. To tackle this, we added a parallel task to buffer the incoming gesture signal to RAM during the WiFi reconnection process (which typically takes around 800 ms). Once connected to the server, the buffered signal is sent out, and the buffer is cleared. Subsequent data is then directly sent out through the WiFi, reducing the need for further buffering. On the server side, we employed a dynamic window to ensure the capture of complete gesture instances. The 400 ms analysis window is extended by another adjacent window if an incomplete gesture is detected at the window's edge. In this way, the shortest interval between successive segmented gestures is 400 ms, while in practice, the shortest interval recorded was 450 ms. Once the instance is segmented, it will be sent for feature extraction and classification as described previously.

5 Evaluation

All studies were cleared by the institution's Office of Research Ethics before commencing the research.

5.1 General Evaluation

To evaluate the 17-gesture set, we recruited 15 participants (7 female; aged 19 to 30; 2 had polished nails; 1 left-handed and 14 right-handed) to participate in the data-collection study. We measured the circumference of each participant's proximal thumb phalanx (mean: 61.0 mm, std: 5.0 mm, range: 53.5-69.1 mm) and selected an appropriate ring size accordingly (5 large, 8 medium, 2 small). The study was split across two days: one hour on one day and half an hour on the other, separated by 24 to 48 hours, depending on participant schedules. This was done to ensure that data was less temporally correlated. This structure also helps us understand how forgetfulness affects users' performances on some uncommon gestures. Each participant was paid 24 C$ for their participation.

On the first day, we gave general instructions, helped participants find a suitable ring size and decided on proper wearing tightness. We controlled the tightness by measuring how much of the
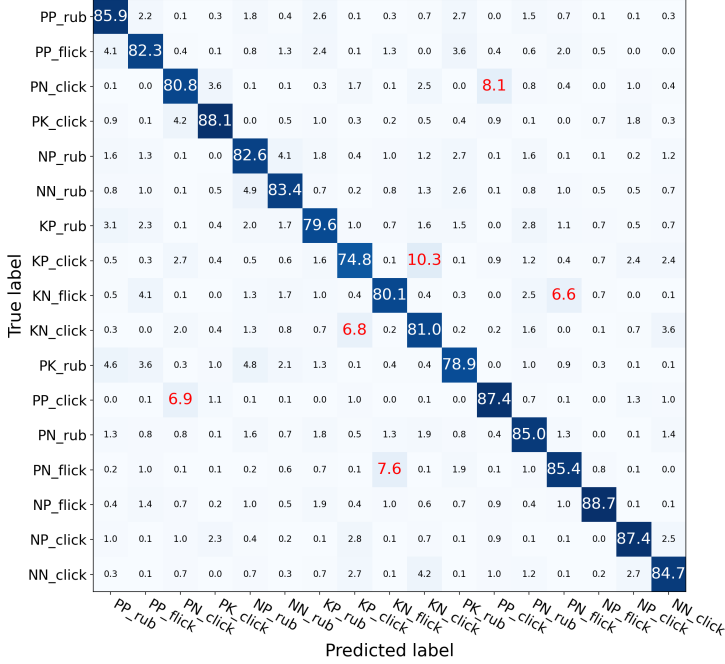
17-Gesture Confusion Matrix

Fig. 10. The 17 gestures' general confusion matrix is presented. The red numbers show the main confusion encountered by a gesture. Apparently, VibRing cannot distinguish all 17 gestures due to signal similarities, and thus we eliminated some gestures to achieve better performance.

velcro band was left with a millimetre-precision ruler after the thumb ring was secured, which also provided a measurement of the thumb size. This ensured a consistent sensor response across all sessions, as noted in our prior optimization experiment. The study consists of a series of data-collection sessions. Each session consists of 18 gestures – 17 actual gestures and a "background" gesture, with each gesture performed 15 times. Upon completing each session, we asked the participants to take off the ring and relax their hands before the next session started. On the first day, we collected 6 sessions, and discarded the first 2 to mitigate learning effects, resulting in a total of 60 instances per gesture. On the second day, we collected 3 sessions, discarding the first to again mitigate learning effects, for a total of 30 instances per gesture. Participants used the same ring and tightness setting as the previous day, so any inconsistencies in the signal were expected to come from variances in gesture performance rather than hardware differences.

We considered three aspects when evaluating VibRing: cross-session accuracy, cross-day accuracy, and user adaptability. The first one demonstrates general accuracy, but the second one focuses on how well participants perform the gestures across time. Both are based on per-user analysis. To understand how well VibRing adapts to new populations, the third aspect aims to find the least amount of data that we need to collect from a new user to recognize their gestures accurately.

5.1.1 Preliminary Analysis. Theoretically, all gestures should have distinct vibroacoustic profiles. However, the signal difference could be too subtle to detect by piezo. For example, PN_click (clicking index fingernail tip to thumb's pad) and PP_click (clicking index finger pad to thumb's pad) sound
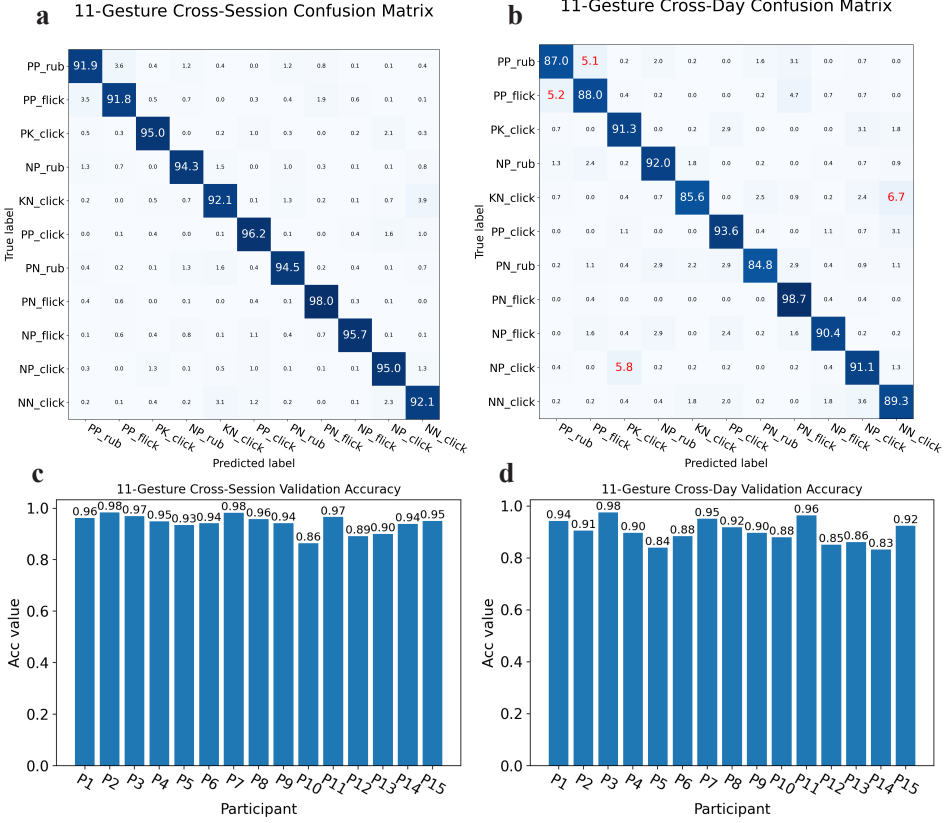
Fig. 11. **(a)** shows the final 11-gesture confusion matrix with the CNN model; the average accuracy is 94.2%. **(c)** delivers the result from each participant's perspective. To measure how well VibRing tolerates gesture-performing variances, a cross-day evaluation was conducted, and the results are presented in **(b, d)**. VibRing still keeps a relatively high accuracy of over 85% for most participants (13 out of 15). On average, however, the accuracy is 90.2%, demonstrating VibRing's good tolerance to variance.

slightly different to the human ear; however, their responses from the piezo are very similar. We understand that this ambiguity could limit how many gestures VibRing can distinguish.

Thus, in this step, we aim to analyze the distinguishability of 17 gestures. We adopted a leave-one-session-out cross-validation (LOSO-CV) method, where we kept one session's data out as a validation set and used the rest as a training set; results were averaged across all six folds and all participants. We chose SVM as the classifier because it provides a good baseline and is easy to train, thus accelerating the experiment iteration on different gesture subsets. The general accuracy of the full 17-gesture set was 83.3%, with detailed results presented in Fig. 10. Finally, we eliminated 6 gestures that have the poorest performance to achieve a preliminary general accuracy of 90.1% over 11 gestures, which we would improve upon later.

5.1.2   Cross-Session Validation. The baseline presented above shows the feasibility of distinguishing 11 gestures. However, we want to improve the performance further to enable real-time application. We designed a multi-layered 1D convolutional network, illustrated in Fig. 9. Because the convolutional and spectral features are produced in different processes, they may have different scales that
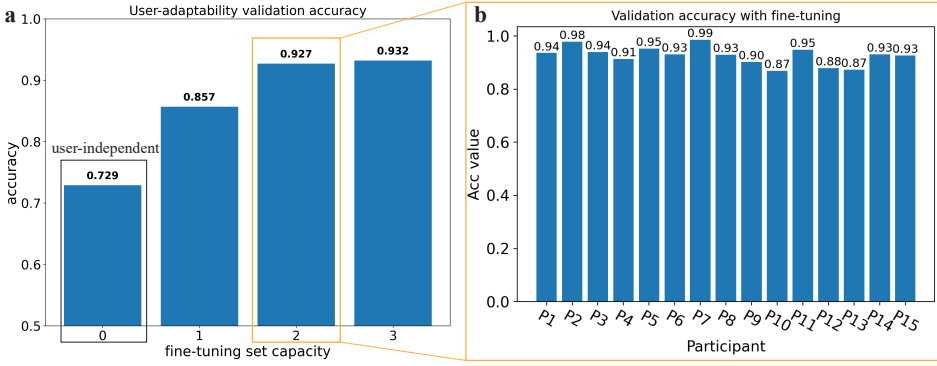
Fig. 12. VibRing's performance on user-independent analysis is not satisfactory (72.9% shown in **(a)**), which is anticipated from our studies, where we observed that participants performed the gestures differently due to various hand dimensions. However, this performance variance could be captured with some data collection. Taking the transfer learning concept, we demonstrate that only 2 sessions (about 10 minutes) of data are needed to improve the accuracy to 92.7%, as presented in **(b)**.

hinder the network's training. Therefore, we added a batch normalization layer before sending the concatenated features to classification. In evaluation, we ran a 6-fold LOSO-CV with the augmented data set. With an Adam optimizer, a cross-entropy loss function, batch size 200 and a loss cutoff at 5e-5, the validation accuracy improved to 94.2%. We attribute this improvement to the additional convolutional features which extract local phase information. The final 11-gesture confusion matrix across all participants is shown in Fig. 11a, and the accuracies for different participants are shown in Fig. 11c.

5.1.3   Cross-Day Validation. Given that the 11-gesture subset has uncommon gestures, we also want to measure how well participants can remember them and how well they can perform consistently across days. If gesture performance has only a minor variance that the model can tolerate, cross-day validation results should match those in cross-session. We employed the CNN model in this evaluation.

We split the dataset by day: the four sessions from the first day were used for training data, and the two sessions from the second day were used for testing. The confusion matrix of cross-day evaluation is presented in Fig. 11b. The accuracy across different participants are shown in the histogram Fig. 11d. Overall, the accuracy is 90.2%. Most participants still kept a relatively high accuracy above 85%. However, there are accuracy drops for some participants in cross-day validation, suggesting that gesture recall differs among participants. Some participants are more consistent in their gesture performances across days, but others tend to forget how they performed the gestures before. To summarize, VibRing can generally tolerate the variance of gestures across days. More practice and more training data to cover the performance variation should help improve the accuracy even further.

5.1.4   User Adaptability Validation. The two per-user analyses have proven the feasibility of input with VibRing gestures. However, the data needed is significant: on average, finishing one gesture in a session took about 25 seconds, and 12 gestures were considered. While we arranged 9 sessions (data from only 6 of them were used) on two days, users hence spent $12 \times 25 \times 9 \div 60 = 45$ minutes before VibRing built their specific classification models. To minimize the per-user training effort, we want to determine the minimal amount of data needed for VibRing to adapt to new populations.

Table 1. A list of general tasks in an office setup

tasks	potential hand events	tasks	potential hand events
text on phones	click, scroll on screen	read books	pinch, rub
type on a keyboard	tap on keys	walk around	swing, rub on pants
drink water	touch, grasp	sanitize hands	impact, intense rub
write	gentle rub to writing pads	door knob interaction	impact on knob
squats	hand press on laps	bimanual interactions	rub, clap and squeeze
jumping jacks	intense swing	take off and put on VibRing	ring touch

Table 2. Task schedule in the 1-hour office setup study

tasks	tasks (continued)
input session	take off and put on
texting on a phone (5 mins)	input session
input session	bimanual interactions (2 mins)
keyboard typing (5 mins)	input session
input session	jumping jacks (5 mins)
keyboard typing (5 mins)	input session
input session	writing (5 mins)
walking around (5 mins)	input session

Optimistically, the best approach is to have a user-independent model that works without new data. However, this approach is not successful for VibRing. We ran a 15-fold leave-one-participant-out validation, where we trained the model with 14 users' data and validated it with the remaining user's data. The average accuracy is unsatisfactory (for SVM 62.2% and CNN 72.9%). We anticipated this result since our observations from the study were that gesture conduct varied across users, and the signal varied across different hands.

Inspired by applications of transfer learning [37, 58], we realized that providing some data from new users might significantly improve the performance. We expected that pre-training could find generalized convolutional filters that apply to new populations, and fine-tuning could decide on the intermediate statistics in the last layers, which are more user-specific. Hence, we first trained our model with data from 14 users until we reached a training loss where, heuristically, we believed the convolutional filters were formed. Then, we froze the filters and fed some new user data to tune the last dense layers' weights. Finally, the remaining data was used to validate the tuned model. The batch sizes were adjusted to 2000 and 100 for pre-training and fine-tuning, respectively. The training cutoff occurs when the loss goes below 1e-2, and the tuning stops when the loss goes below 1e-5. In terms of computation time, the pre-training takes minutes to finish (but only needs to be done once), and the per-user fine-tuning finishes in seconds.

The final results are presented in Fig. 12. The performance increases drastically with 1 or 2 sessions of training data and saturates when adding more. Specifically, when 2 sessions of new data are considered, the validation result reaches 92.7%. In this way, we reduce the data needed from 6 sessions for building each user-specific model to 2 sessions for fine-tuning a pre-built generic model. In actual practice, if 1 extra session is also arranged to mitigate the learning effects, VibRing will need 3 sessions to adapt to new populations, which only takes around 15 minutes to complete.

5.2 Extended Usage Evaluation

To evaluate our system's performance in a more natural setting, we tested VibRing in an office environment. This evaluation has two parts: an elongated experiment carried out by an author over 5 hours, and a 1-hour user study by 5 participants. The setup interleaves daily tasks with the usage

Table 3. Results of long-term usage

	FT	FP	FN	TP	accuracy
5-hour experiment	69	2	2	18	N/A
1-hour study (P4)	10	0	2	8	87.5%
1-hour study (P9)	14	0	1	8	87.1%
1-hour study (P11)	55	5	0	8	92.5%
1-hour study (P12)	17	0	1	8	92.5%
1-hour study (P15)	4	0	0	8	90.0%

of VibRing gestures. We expected that VibRing could reject noise generated by vibrations during tasks while accepting the users' explicit input.

We defined 4 metrics – False Trigger (FT), False Positive (FP), False Negative (FN) and True Positive (TP) – to measure VibRing's ability to reject noise and detect inputs. FT denotes how often a noise wakes up VibRing, yet does not pass the initiating gesture check. FP shows how often a noise unexpectedly initiates an input session. FN denotes how often VibRing fails to detect the input request. Finally, TP shows how often VibRing successfully detects that request. Both FT and FP result in the system waking up, but in an FT, the system wakes up only long enough to receive and process the initiating gesture, whereas with an FP, the system wakes up and falsely classifies noise as an initiating gesture, thus starting an input session. Thus, we consider FT to be a minor failure, compared with FP.

In the extended usage test, the first author performed representative tasks as enumerated in Table 1; the detailed schedule may be found in the Supplementary Material. The experiment was accomplished in a medium-sized (11 by 7.5 meters) lab within the WiFi reception range. The final results are presented in the first row of Table 3. 69 FT events were observed, but only 2 FP events occurred. Both FP events were associated with a hand-clapping task (see supplemental material). Meanwhile, 2 FNs were observed, indicating cases where the experimenter's first attempt to wake up the device failed; in both cases, a second attempt was successful. Regarding power consumption, one fully charged ~3.7 V 500 mAh LiPo battery lasted for the full five hours, consuming approximately 20% of the capacity. The input rate was 4 sessions per hour, each consisting of up to 35 seconds of gesture input.

In the subsequent user study, we chose typical tasks based on our observation from the 5-hour experiment. Each study consisted of 8 VibRing input sessions in a 1-hour time window, as illustrated in Table 2. We invited 5 participants, all of whom participated in the data-collection study, to wear VibRing while completing tasks according to the schedule. Unlike a fully controlled experiment, no restrictions were imposed on this study, so all participants had a chance to explore VibRing using the time between tasks. For each input session, we prompted the participants to wake up VibRing and perform two repetitions of 4 randomly chosen gestures. Thus, in total $4 \times 2 \times 8 = 64$ gestures were classified in real time using per-user models. We chose not to provide feedback on the gesture classification to users, preventing them from adapting their performance based on the results; this allowed us to probe the worst-case performance of our system.

First, we observed participants' exploration of VibRing. For instance, P12 interacted with the door knob using the hand that wore VibRing, intentionally probing the system out of curiosity. P9 picked up a water bottle and moved a chair while walking around. None of these actions caused FPs, but some led to FTs, which were already counted in the final result. Specifically, we report the 4 numbers with classification accuracy in Table 3. We observed a significant amount of FTs induced mostly by jumping jacks, keyboard typing and bimanual interactions (e.g., hand stretching and hand-to-hand rubbing); however, they were rejected ultimately and did not cause much power
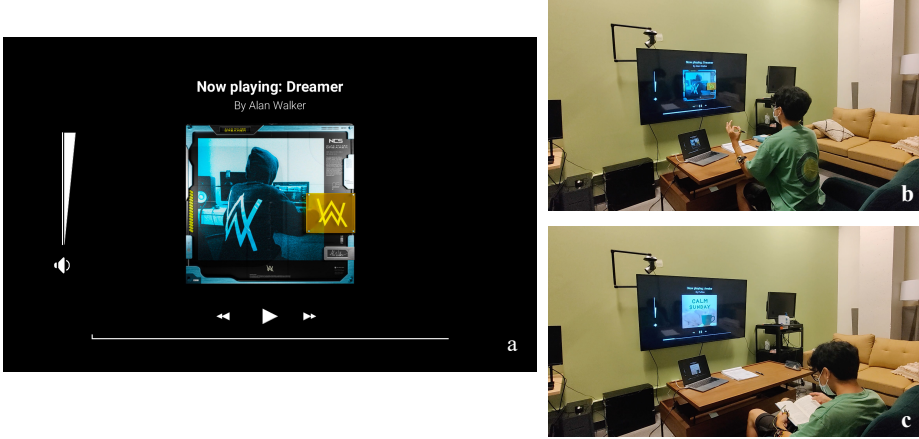
Fig. 13. This figure presents our first application - a music player. In this scenario, with 7 gestures deployed, we demonstrate VibRing's capability of more unambiguous input. The user can wear the system and complete some daily tasks like reading and writing without suffering from false triggering.

consumption. We found almost no FPs except for P12, where VibRing falsely initiated input sessions when she was typing on the keyboard. To address the problem, a future design could substitute the second PP_click with a more uncommon gesture like NP_flick. Additionally, we also observed some FNs when VibRing missed the second PP_click gesture due to a hardware limitation of the ESP32, which caused WiFi to stall temporarily. For real-time classification, we achieved an average accuracy of 89.9%, lower than the general accuracy of 94.2%, which is reasonable due to gesture recall since all participants completed this study at least a week after the data collection.

6 Application Scenarios

To showcase the potential of VibRing as a gestural input system, we developed three gesture-controlled applications: a music player, a phone dialer, and an AR painting app. Please also see the Video Figure for demonstrations of the applications in action.

Music Player: Inputs Alongside Daily Activities. People use their hands to engage in different daily activities all the time. Therefore, applying a hand-worn gesture sensor to daily use could result in frequent mis-triggering induced by noise. VibRing's capability of noise rejection enables unambiguous inputs alongside other daily activities. To showcase this capability, we developed a simple music player controlled by VibRing. A user can play, pause, adjust the volume, fast forward/backward and switch between songs using 7 VibRing hand gestures. In Fig. 13, we show that the user can engage in daily tasks (e.g., handwriting, reading) like they are used to while unambiguously controlling the music player. Operation is eyes-free and can work under various hand orientations and body postures.

Phone Dialer: Extending the Interactivity of Small Screens with Gestures. Small touch screens (e.g., smart watches and phones) could impede smart devices' interactivity due to the fat-thumb problem [3]. In contrast, hand gestures have a higher degree of freedom and can thus extend the interactivity of devices with small-sized screens [16, 51]. To showcase the expressive input space of VibRing, we demonstrate VibRing's capability of operating a phone dialer (Fig. 14; simulated
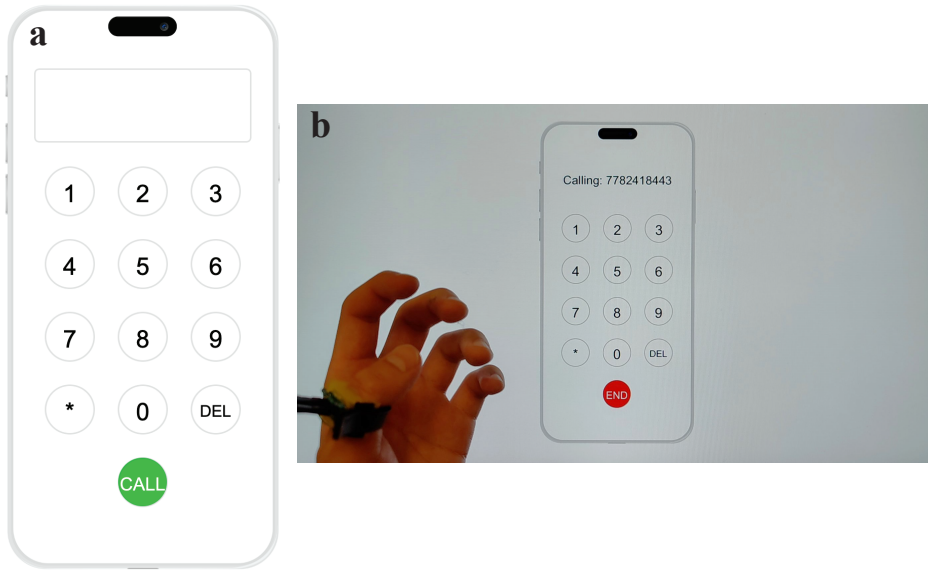
Fig. 14. In this figure, we demonstrate another application - phone dialer, where 11 gestures are deployed and connected to the ten-digit buttons and 1 functional button - CALL. This showcases the capability of the full gesture set to emulate a more complicated interface.

with Processing.java[3]). By mapping each button on the dialer to one of the 11 VibRing gestures, we implemented the 0-9 digit inputs and a function button - CALL.

AR painter: Line-of-sight Independent AR Gesture Control. Hand gestures are a major input channel for head-mounted augmented reality. However, contemporary head-mounted AR headsets use image-based gesture sensing, which is subject to occlusions and the camera's field of view. VibRing's hand gesture-sensing capability can provide AR applications with gesture control independent of line-of-sight. Here we demonstrate an application scenario where VibRing gestures work alongside regular AR gestures. We developed an AR painting application with Microsoft HoloLens[4]. In Fig. 15, we show that the user can use a regular AR pinching gesture to draw virtual strokes in their physical environment with the right hand while using VibRing gestures to switch colours, erase a previous stroke, and clear all paintings with their left hand. This design provides a more seamless collaboration of both hands and opens a potential space of richer bimanual interactions.

7 Discussion

Long Term Usage. VibRing uses a piezo that is purely passive – the sensor generates detectable signals without external power. Though the functions running on the ESP32 still need a battery, the sensor's passivity benefits the whole system's low-power design. For instance, we showed that VibRing could harness the unamplified raw signal to not only classify gestures but also wake up the system. Accordingly, VibRing can decide whether to start input based on the first gesture classification, thereby reducing the number of false positives and the overall system power consumption. In the examination, VibRing only consumed 20% capacity of a 3.7 V 500 mAh battery

---

[3]https://processing.org/
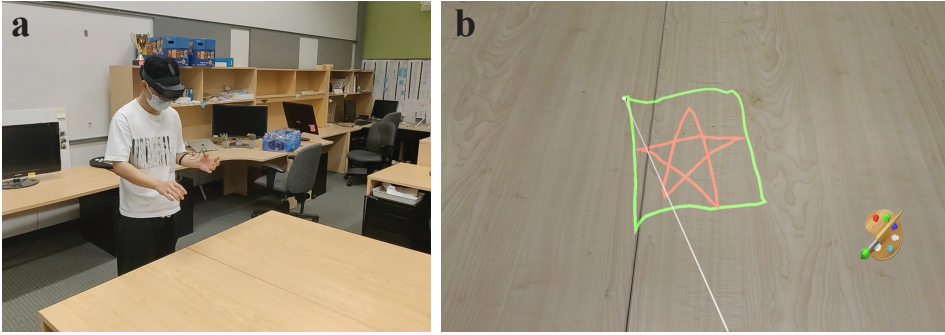[4]https://www.microsoft.com/en-ca/hololens

Fig. 15. In this figure, an AR painter is demonstrated to show how seamlessly we can combine the HoloLens hand gesture (right hand) with VibRing's gesture (left hand) to perform a painting task.

over the course of 5 hours. If we keep the current design, one can theoretically use VibRing for about 2 hours with a 50 mAh battery (similar capacity to AirPods Pro 2) and for over 20 hours with a 500 mAh power bank (similar capacity to AirPods charging case). If optimized with Bluetooth Low Energy (BLE), which is less power-hungry than WiFi, the battery life would be even longer. In a nutshell, passive sensing helps VibRing reduce power consumption and makes it more suitable for long-term usage.

Robustness. We highlighted VibRing's robustness to varying environments. Unlike microphones, VibRing rejects any airborne noise like speech, preserving privacy and providing a less noisy input. It also features input-on-the-go, as presented in the music player application, where VibRing takes inputs when a user sits, stands and walks with the arm lifting or naturally hanging down. Such robustness is hard to achieve with IMUs and magnetic sensing because random hand orientation and motions may confuse the system. Besides, without dependence on line-of-sight, VibRing can easily augment current vision-based gesture systems, providing sensing when the fingers are out of view. Based on the results of our office study, VibRing accepted input requests with a 90.9% accuracy, rejected most unintentional conduct and offered 89.9% accuracy for real-time gesture classification. Therefore, we believe that VibRing has proven its ability to tackle environmental noise and has successfully shown use in more realistic environments.

Rapid Interaction. VibRing enables rapid micro-gestures, thanks to piezo's easy access to high-frequency sampling. As is shown in the experiment 4.2, normal IMUs cannot support consistent responses to quick events due to their limited sample rates. Though academic research [26] demonstrated a high-sample-rate IMU, which should perform similarly to piezo, it fell short by consuming too much power. Cameras can also get high sampling rates; however, they raise privacy concerns and are subject to occlusion. Hence, piezos are more suitable for detecting fast and subtle gestures. With such a property, VibRing enables gestures that have a duration ranging from 82-225 ms, subsequently affording rapid sequences of gestures separated by as little as 450 ms. By comparison, FingerSound [55] uses a two-second window to capture a single drawing gesture; this gives VibRing a four-fold improvement in effective input rate. Besides, VibRing's quick response to wake-up (around 800 ms) also facilitates rapid interaction. We attribute that to two factors - the quick wake-up of ESP32 and the low-latency reconnection to the remote server of WiFi. To summarize, the current design of VibRing supports rapid and subtle interactions, and the proposed application (e.g., the phone dialer) thus has a higher input rate and lower latency.

Form Factor. We acknowledge that VibRing's current design, consisting of two components – a ring and a wristband – is not compact. However, unlike a product, VibRing is aimed at demonstrating

the viability of applying vibroacoustic sensing to single-handed gesture recognition. Therefore, we accepted the prototype limitations and argued that they could be resolved by dedicated engineering. Specifically, our bias circuit used cylindric resistors and ceramic capacitors, whereas applying SMD electronics can afford a smaller ring design. A smaller BLE chip (e.g., Apple H2[5]) can rid VibRing of the wristband, which is mainly for accommodating the ESP32. With a ~50mA battery, all hardware could now be integrated into a ring. To minimize the system volume further, the 12 mm-diameter piezo could be replaced by other smaller options like Knowles BU-21771 contact microphone (~7.92mm by 5.59mm) [25] or a Sonion voice pick-up unit (~2.65mm by 3.5mm) [40]. Those options might also outperform piezo for signal acquisition because they trade power for flatter responses across a wide spectral range (i.e. human hearing range). VibRing still adopted the piezo because it is passive and more suitable for showcasing long-term usage. Finally, VibRing requires a remote server to run the signal processing and classification pipeline. For a more portable design, the whole pipeline with a CNN can be easily moved to a mobile device like a smartphone. If a pipeline with a SVM is deployed instead, we could integrate VibRing entirely onto the ring and remove the server.

Limitations. The current incarnation of VibRing has some limitations that will be addressed in future work. Although 17 gestures were produced by gesture design, piezos only enabled 11 of them due to substantially similar signals. By using a VPU, we can potentially expand the usable gesture set due to its better signal reception. We also acknowledge that the development of long-term usage was not ideal. Detecting the end of an input session was implicit (i.e., 10-second inactivity), and false positives (FPs) were reported during some tasks. To improve those, a future system could define a ending gesture similar to the wake-up and implement a "Null" gesture [45] to further reduce the FP rates. Regarding evaluation, the accuracy drop in cross-day validation and real-time classification suggests that uncommon gestures can cause performance variations. To address this, the general study should be spread out over more days to account for variance. Besides, although we recruited participants with different hand sizes, handedness, nail conditions, and genders, they were all young adults (aged 19 to 30). To validate this technology for other age groups, a broader user study should be conducted.

We also found that most participants had difficulty recalling a gesture from its name (notation). Although the usage of systematic notations is common in gesture-inventing papers [6, 24, 52], a system of "common names" could be used to help users remember gestures more effectively. Finally, the extended usage study still happened under the supervision of an experimenter, with only 5 participants for an hour each. For a better demonstration, future work can distribute VibRing to more users, letting the usage happen in real life, and analyzing in-the-wild usage reports.

8 Conclusion

In this work, we propose VibRing, a wireless vibroacoustic wearable platform that can distinguish subtle single-handed gestures. The system can recognize an 11-gesture set derived from a systematic, iterative design approach. For general accuracy, VibRing applies a per-user model and presents an accuracy of 94.2%. To evaluate how users' gesture-performing variances affect VibRing's robustness, the cross-day validation shows a 90.2% accuracy, demonstrating a good tolerance. To alleviate the data collection burden, we also show that VibRing adapts well (92.7%) to new users with only 10 minutes of data. We also carried out a 5-hour experiment and a 1-hour office study with 5 participants, where VibRing's usage is interleaved with other ordinary tasks; the low FP and FN numbers highlight the robustness of VibRing in a realistic operating context. Finally, to demonstrate use cases, we developed 3 applications - a music player, a phone dialer and a VR painting tool.

---

[5]https://www.apple.com/ca/airpods-pro/

Acknowledgments

References

[1] Vincent Becker, Linus Fessler, and Gábor Sörös. 2019. GestEar: combining audio and motion sensing for gesture recognition on smartwatches. In *Proceedings of the 23rd International Symposium on Wearable Computers (ISWC '19)*. Association for Computing Machinery, New York, NY, USA, 10–19. doi:10.1145/3341163.3347735

[2] Roger Boldu, Alexandru Dancu, Denys J.C. Matthies, Pablo Gallego Cascón, Shanaka Ransir, and Suranga Nanayakkara. 2018. Thumb-In-Motion: Evaluating Thumb-to-Ring Microgestures for Athletic Activity. In *Proceedings of the Symposium on Spatial User Interaction (SUI '18)*. Association for Computing Machinery, New York, NY, USA, 150–157. doi:10.1145/3267782.3267796

[3] Sebastian Boring, David Ledo, Xiang 'Anthony' Chen, Nicolai Marquardt, Anthony Tang, and Saul Greenberg. 2012. The Fat Thumb: Using the Thumb's Contact Size for Single-Handed Mobile Interaction. In *Proceedings of the 14th International Conference on Human-Computer Interaction with Mobile Devices and Services* (San Francisco, California, USA) *(MobileHCI '12)*. Association for Computing Machinery, New York, NY, USA, 39–48. doi:10.1145/2371574.2371582

[4] Andreas Braun, Stefan Krepp, and Arjan Kuijper. 2015. Acoustic tracking of hand activities on surfaces. In *Proceedings of the 2nd international Workshop on Sensor-based Activity Recognition and Interaction (iWOAR '15)*. Association for Computing Machinery, New York, NY, USA, 1–5. doi:10.1145/2790044.2790052

[5] Volkert Buchmann, Stephen Violich, Mark Billinghurst, and Andy Cockburn. 2004. FingARtips: gesture based direct manipulation in Augmented Reality. In *Proceedings of the 2nd international conference on Computer graphics and interactive techniques in Australasia and South East Asia (GRAPHITE '04)*. Association for Computing Machinery, New York, NY, USA, 212–221. doi:10.1145/988834.988871

[6] Adrien Chaffangeon Caillet, Alix Goguey, and Laurence Nigay. 2023. μGlyph: a Microgesture Notation. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. ACM, Hamburg Germany, 1–28. doi:10.1145/3544548.3580693

[7] Edwin Chan, Teddy Seyed, Wolfgang Stuerzlinger, Xing-Dong Yang, and Frank Maurer. 2016. User Elicitation on Single-hand Microgestures. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. Association for Computing Machinery, New York, NY, USA, 3403–3414. doi:10.1145/2858036.2858589

[8] Liwei Chan, Yi-Ling Chen, Chi-Hao Hsieh, Rong-Hao Liang, and Bing-Yu Chen. 2015. CyclopsRing: Enabling Whole-Hand and Context-Aware Interactions Through a Fisheye Ring. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. Association for Computing Machinery, New York, NY, USA, 549–556. doi:10.1145/2807442.2807450

[9] Manoel Farhad and I. Scott MacKenzie. 2018. Evaluating Tap-and-Drag: A Single-Handed Zooming Method. In *Human-Computer Interaction. Interaction Technologies (Lecture Notes in Computer Science)*, Masaaki Kurosu (Ed.). Springer International Publishing, Cham, 233–246. doi:10.1007/978-3-319-91250-9_18

[10] Sarthak Ghosh, Hyeong Cheol Kim, Yang Cao, Arne Wessels, Simon T. Perrault, and Shengdong Zhao. 2016. Ringteraction: Coordinated Thumb-index Interaction Using a Ring. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16)*. Association for Computing Machinery, New York, NY, USA, 2640–2647. doi:10.1145/2851581.2892371

[11] Yizheng Gu, Chun Yu, Zhipeng Li, Weiqi Li, Shuchang Xu, Xiaoying Wei, and Yuanchun Shi. 2019. Accurate and Low-Latency Sensing of Touch Contact on Any Surface with Finger-Worn IMU Sensor. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19)*. Association for Computing Machinery, New York, NY, USA, 1059–1070. doi:10.1145/3332165.3347947

[12] Chris Harrison and Scott E. Hudson. 2008. Scratch input: creating large, inexpensive, unpowered and mobile finger input surfaces. In *Proceedings of the 21st annual ACM symposium on User interface software and technology - UIST '08*. ACM Press, Monterey, CA, USA, 205. doi:10.1145/1449715.1449747

[13] Chris Harrison, Julia Schwarz, and Scott E. Hudson. 2011. TapSense: enhancing finger interaction on touch surfaces. In *Proceedings of the 24th annual ACM symposium on User interface software and technology (UIST '11)*. Association for Computing Machinery, New York, NY, USA, 627–636. doi:10.1145/2047196.2047279

[14] Chris Harrison, Desney Tan, and Dan Morris. 2010. Skinput: appropriating the body as an input surface. In *Proceedings of the 28th international conference on Human factors in computing systems - CHI '10*. ACM Press, Atlanta, Georgia, USA, 453. doi:10.1145/1753326.1753394

[15] Sandra G. Hart and Lowell E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In *Advances in Psychology*, Peter A. Hancock and Najmedin Meshkati (Eds.). Human Mental Workload, Vol. 52. North-Holland, 139–183. doi:10.1016/S0166-4115(08)62386-9

[16] Eiji Hayashi, Jaime Lien, Nicholas Gillian, Leonardo Giusti, Dave Weber, Jin Yamanaka, Lauren Bedal, and Ivan Poupyrev. 2021. RadarNet: Efficient Gesture Recognition Technique Utilizing a Miniature Radar Sensor. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 5, 14 pages. doi:10.1145/3411764.3445367

[17] Anuradha Herath, Bradley Rey, Sandra Bardot, Sawyer Rempel, Lucas Audette, Huizhe Zheng, Jun Li, Kevin Fan, Da-Yuan Huang, Wei Li, and Pourang Irani. 2022. Expanding Touch Interaction Capabilities for Smart-rings: An Exploration of Continual Slide and Microroll Gestures. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems (CHI EA '22)*. Association for Computing Machinery, New York, NY, USA, 1–7. doi:10.1145/3491101.3519714

[18] Yasha Iravantchi, Yi Zhao, Kenrick Kin, and Alanson P. Sample. 2023. SAWSense: Using Surface Acoustic Waves for Surface-bound Event Recognition. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–18. doi:10.1145/3544548.3580991

[19] Nikhita Joshi, Parastoo Abtahi, Raj Sodhi, Nitzan Bartov, Jackson Rushing, Christopher Collins, Daniel Vogel, and Michael Glueck. 2023. Transferable Microgestures Across Hand Posture and Location Constraints: Leveraging the Middle, Ring, and Pinky Fingers. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*. Association for Computing Machinery, New York, NY, USA, 1–17. doi:10.1145/3586183.3606713

[20] Hsin-Liu (Cindy) Kao, Artem Dementyev, Joseph A. Paradiso, and Chris Schmandt. 2015. NailO: Fingernails as an Input Surface. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. Association for Computing Machinery, New York, NY, USA, 3015–3018. doi:10.1145/2702123.2702572

[21] Adam Kendon. 1994. Do Gestures Communicate? A Review. *Research on Language and Social Interaction* 27, 3 (July 1994), 175–200. doi:10.1207/s15327973rlsi2703_2 Publisher: Routledge.

[22] Wolf Kienzle, Eric Whitmire, Chris Rittaler, and Hrvoje Benko. 2021. ElectroRing: Subtle Pinch and Touch Detection with a Ring. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–12. doi:10.1145/3411764.3445094

[23] Daehwa Kim, Keunwoo Park, and Geehyuk Lee. 2021. AtaTouch: Robust Finger Pinch Detection for a VR Controller Using RF Return Loss. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (CHI '21)*. Association for Computing Machinery, New York, NY, USA, 1–9. doi:10.1145/3411764.3445442

[24] Kenrick Kin, Björn Hartmann, Tony DeRose, and Maneesh Agrawala. 2012. Proton: multitouch gestures as regular expressions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. Association for Computing Machinery, New York, NY, USA, 2885–2894. doi:10.1145/2207676.2208694

[25] Knowles. 2024. Contact Microphons. https://www.knowles.com/subdepartment/dpt-sensors/subdpt-accelerometers

[26] Gierad Laput, Robert Xiao, and Chris Harrison. 2016. ViBand: High-Fidelity Bio-Acoustic Sensing Using Commodity Smartwatch Accelerometers. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. Association for Computing Machinery, New York, NY, USA, 321–333. doi:10.1145/2984511.2984582

[27] DoYoung Lee, SooHwan Lee, and Ian Oakley. 2020. Nailz: Sensing Hand Input with Touch Sensitive Nails. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. doi:10.1145/3313831.3376778

[28] Chen Liang, Chun Yu, Yue Qin, Yuntao Wang, and Yuanchun Shi. 2021. DualRing: Enabling Subtle and Expressive Hand Interaction with Dual IMU Rings. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 3 (Sept. 2021), 115:1–115:27. doi:10.1145/3478114

[29] Jhe-Wei Lin, Chiuan Wang, Yi Yao Huang, Kuan-Ting Chou, Hsuan-Yu Chen, Wei-Luan Tseng, and Mike Y. Chen. 2015. BackHand: Sensing Hand Gestures via Back of the Hand. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. Association for Computing Machinery, New York, NY, USA, 557–564. doi:10.1145/2807442.2807462

[30] Stephen Shiao-ru Lin, Nisal Menuka Gamage, Kithmini Herath, and Anusha Withana. 2022. MyoSpring: 3D Printing Mechanomyographic Sensors for Subtle Finger Gesture Recognition. In *Sixteenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '22)*. Association for Computing Machinery, New York, NY, USA, 1–13. doi:10.1145/3490149.3501321

[31] Pedro Lopes, Ricardo Jota, and Joaquim A. Jorge. 2011. Augmenting touch interaction through acoustic sensing. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces (ITS '11)*. Association for Computing Machinery, New York, NY, USA, 53–56. doi:10.1145/2076354.2076364

[32] Zhiyuan Lu, Xiang Chen, Qiang Li, Xu Zhang, and Ping Zhou. 2014. A Hand Gesture Recognition Framework and Wearable Gesture-Based Interaction Prototype for Mobile Devices. *IEEE Transactions on Human-Machine Systems* 44, 2 (April 2014), 293–299. doi:10.1109/THMS.2014.2302794 Conference Name: IEEE Transactions on Human-Machine

Systems.

[33] Jess McIntosh, Asier Marzo, and Mike Fraser. 2017. SensIR: Detecting Hand Gestures with a Wearable Bracelet using Infrared Transmission and Reflection. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17)*. Association for Computing Machinery, New York, NY, USA, 593–597. doi:10.1145/3126594. 3126604

[34] Takashi Miyaki and Jun Rekimoto. 2009. GraspZoom: zooming and scrolling control model for single-handed mobile interaction. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '09)*. Association for Computing Machinery, New York, NY, USA, 1–4. doi:10.1145/1613858.1613872

[35] Adiyan Mujibiya, Xiang Cao, Desney S. Tan, Dan Morris, Shwetak N. Patel, and Jun Rekimoto. 2013. The sound of touch: on-body touch and gesture sensing based on transdermal ultrasound propagation. In *Proceedings of the 2013 ACM international conference on Interactive tabletops and surfaces (ITS '13)*. Association for Computing Machinery, New York, NY, USA, 189–198. doi:10.1145/2512349.2512821

[36] Uran Oh and Leah Findlater. 2014. Design of and subjective response to on-body input for people with visual impairments. In *Proceedings of the 16th international ACM SIGACCESS conference on Computers & accessibility (ASSETS '14)*. Association for Computing Machinery, New York, NY, USA, 115–122. doi:10.1145/2661334.2661376

[37] Sinno Jialin Pan and Qiang Yang. 2010. A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering* 22, 10 (Oct. 2010), 1345–1359. doi:10.1109/TKDE.2009.191 Conference Name: IEEE Transactions on Knowledge and Data Engineering.

[38] Adwait Sharma, Joan Sol Roo, and Jürgen Steimle. 2019. Grasping Microgestures: Eliciting Single-hand Microgestures for Handheld Objects. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. doi:10.1145/3290605.3300632

[39] Mohamed Soliman, Franziska Mueller, Lena Hegemann, Joan Sol Roo, Christian Theobalt, and Jürgen Steimle. 2018. FingerInput: Capturing Expressive Single-Hand Thumb-to-Finger Microgestures. In *Proceedings of the 2018 ACM International Conference on Interactive Surfaces and Spaces (ISS '18)*. Association for Computing Machinery, New York, NY, USA, 177–187. doi:10.1145/3279778.3279799

[40] Sonion. 2024. Sonion • Improving people's quality of life. https://www.sonion.com/hearing/bone-conduction-sensors-and-actuators/vpu-voice-pick-up-sensor/

[41] STMelectronics. 2024. LSM9DS1 - 9-axis iNEMO inertial module (IMU): 3D magnetometer, 3D accelerometer, 3D gyroscope with I2C and SPI - STMicroelectronics. https://www.st.com/en/mems-and-sensors/lsm9ds1.html

[42] Wei Sun, Franklin Mingzhe Li, Congshu Huang, Zhenyu Lei, Benjamin Steeper, Songyun Tao, Feng Tian, and Cheng Zhang. 2021. ThumbTrak: Recognizing Micro-finger Poses Using a Ring with Proximity Sensing. In *Proceedings of the 23rd International Conference on Mobile Human-Computer Interaction (MobileHCI '21)*. Association for Computing Machinery, New York, NY, USA, 1–9. doi:10.1145/3447526.3472060

[43] Clayton Valli and Ceil Lucas. 2000. *Linguistics of American Sign Language: An Introduction*. Gallaudet University Press. Google-Books-ID: mfS3GlTLAUMC.

[44] Radu-Daniel Vatavu and Laura-Bianca Bilius. 2021. GestuRING: A Web-based Tool for Designing Gesture Input with Rings, Ring-Like, and Ring-Ready Devices. In *The 34th Annual ACM Symposium on User Interface Software and Technology (UIST '21)*. Association for Computing Machinery, New York, NY, USA, 710–723. doi:10.1145/3472749.3474780

[45] Anandghan Waghmare, Youssef Ben Taleb, Ishan Chatterjee, Arjun Narendra, and Shwetak Patel. 2023. Z-Ring: Single-Point Bio-Impedance Sensing for Gesture, Touch, Object and User Recognition. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–18. doi:10.1145/3544548.3581422

[46] Cheng-Yao Wang, Min-Chieh Hsiu, Po-Tsung Chiu, Chiao-Hui Chang, Liwei Chan, Bing-Yu Chen, and Mike Y. Chen. 2015. PalmGesture: Using Palms as Gesture Interfaces for Eyes-free Input. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '15)*. Association for Computing Machinery, New York, NY, USA, 217–226. doi:10.1145/2785830.2785885

[47] Tiantong Wang, Yunbiao Zhao, and Qining Wang. 2021. Flexible Non-contact Capacitive Sensing for Hand Gesture Recognition. In *Intelligent Robotics and Applications (Lecture Notes in Computer Science)*, Xin-Jun Liu, Zhenguo Nie, Jingjun Yu, Fugui Xie, and Rui Song (Eds.). Springer International Publishing, Cham, 611–621. doi:10.1007/978-3-030-89095-7_58

[48] Mathias Wilhelm, Daniel Krakowczyk, and Sahin Albayrak. 2020. PeriSense: Ring-Based Multi-Finger Gesture Interaction Utilizing Capacitive Proximity Sensing. *Sensors* 20, 14 (Jan. 2020), 3990. doi:10.3390/s20143990 Number: 14 Publisher: Multidisciplinary Digital Publishing Institute.

[49] Mathias Wilhelm, Daniel Krakowczyk, Frank Trollmann, and Sahin Albayrak. 2015. eRing: multiple finger gesture recognition with one ring using an electric field. In *Proceedings of the 2nd international Workshop on Sensor-based Activity Recognition and Interaction (iWOAR '15)*. Association for Computing Machinery, New York, NY, USA, 1–6. doi:10.1145/2790044.2790047

[50] Anusha Withana, Roshan Peiris, Nipuna Samarasekara, and Suranga Nanayakkara. 2015. zSense: Enabling Shallow Depth Gesture Recognition for Greater Input Expressivity on Smart Wearables. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. Association for Computing Machinery, New York, NY, USA, 3661–3670. doi:10.1145/2702123.2702371

[51] Xuhai Xu, Jun Gong, Carolina Brum, Lilian Liang, Bongsoo Suh, Shivam Kumar Gupta, Yash Agarwal, Laurence Lindsey, Runchang Kang, Behrooz Shahsavari, Tu Nguyen, Heriberto Nieto, Scott E Hudson, Charlie Maalouf, Jax Seyed Mousavi, and Gierad Laput. 2022. Enabling Hand Gesture Customization on Wrist-Worn Devices. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 496, 19 pages. doi:10.1145/3491102.3501904

[52] Xuhai Xu, Haitian Shi, Xin Yi, WenJia Liu, Yukang Yan, Yuanchun Shi, Alex Mariakakis, Jennifer Mankoff, and Anind K. Dey. 2020. EarBuddy: Enabling On-Face Interaction via Wireless Earbuds. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–14. doi:10.1145/3313831.3376836

[53] Kaoru Yamagishi, Lei Jing, and Zixue Cheng. 2014. A system for controlling personal computers by hand gestures using a wireless sensor device. In *2014 IEEE International Symposium on Independent Computing (ISIC)*. 1–7. doi:10.1109/INDCOMP.2014.7011759

[54] Cheng Zhang, AbdelKareem Bedri, Gabriel Reyes, Bailey Bercik, Omer T. Inan, Thad E. Starner, and Gregory D. Abowd. 2016. TapSkin: Recognizing On-Skin Input for Smartwatches. In *Proceedings of the 2016 ACM International Conference on Interactive Surfaces and Spaces (ISS '16)*. Association for Computing Machinery, New York, NY, USA, 13–22. doi:10.1145/2992154.2992187

[55] Cheng Zhang, Anandghan Waghmare, Pranav Kundra, Yiming Pu, Scott Gilliland, Thomas Ploetz, Thad E. Starner, Omer T. Inan, and Gregory D. Abowd. 2017. FingerSound: Recognizing unistroke thumb gestures using a ring. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (Sept. 2017), 120:1–120:19. doi:10.1145/3130985

[56] Cheng Zhang, Qiuyue Xue, Anandghan Waghmare, Ruichen Meng, Sumeet Jain, Yizeng Han, Xinyu Li, Kenneth Cunefare, Thomas Ploetz, Thad Starner, Omer Inan, and Gregory D. Abowd. 2018. FingerPing: Recognizing Fine-grained Hand Poses using Active Acoustic On-body Sensing. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–10. doi:10.1145/3173574.3174011

[57] Qian Zhang, Yetong Cao, Huijie Chen, Fan Li, Song Yang, Yu Wang, Zheng Yang, and Yunhao Liu. 2020. airFinger: Micro Finger Gesture Recognition via NIR Light Sensing for Smart Devices. In *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*. 552–562. doi:10.1109/ICDCS47774.2020.00073 ISSN: 2575-8411.

[58] Ruidong Zhang, Ke Li, Yihong Hao, Yufan Wang, Zhengnan Lai, François Guimbretière, and Cheng Zhang. 2023. EchoSpeech: Continuous Silent Speech Recognition on Minimally-obtrusive Eyewear Powered by Acoustic Sensing. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23)*. Association for Computing Machinery, New York, NY, USA, 1–18. doi:10.1145/3544548.3580801

[59] Yang Zhang and Chris Harrison. 2015. Tomo: Wearable, Low-Cost Electrical Impedance Tomography for Hand Gesture Recognition. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. ACM, Charlotte NC USA, 167–173. doi:10.1145/2807442.2807480

[60] Yang Zhang, Junhan Zhou, Gierad Laput, and Chris Harrison. 2016. SkinTrack: Using the Body as an Electrical Waveguide for Continuous Finger Tracking on the Skin. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. Association for Computing Machinery, New York, NY, USA, 1491–1503. doi:10.1145/2858036.2858082

[61] Bing Zhou, Matias Aiskovich, and Sinem Guven. 2021. Acoustic Sensing-based Hand Gesture Detection for Wearable Device Interaction. doi:10.48550/arXiv.2112.05986 arXiv:2112.05986 [cs].

[62] Junhan Zhou, Yang Zhang, Gierad Laput, and Chris Harrison. 2016. AuraSense: Enabling Expressive Around-Smartwatch Interactions with Electric Field Sensing. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. Association for Computing Machinery, New York, NY, USA, 81–86. doi:10.1145/2984511.2984568