EpiPad: Enabling Multi-Finger and Force-Sensitive Skin Interface for Smartwatches

YULIANG FU, North Carolina State University, USA
RAKSHITA RANGANATH, North Carolina State University, USA
HUINING LI, North Carolina State University, USA
CHENHAN XU*, North Carolina State University, USA

With the growing popularity of smart wrist-worn devices, their small touchscreen size fundamentally limits the expressiveness of user input. Although various interaction methods have been proposed to extend input capabilities, few provide a seamless transition from traditional touchscreens and touchpads. To address this challenge, we present EpiPad, which for the first time enables multi-finger and force-sensitive input for smartwatches through mmWave-enabled on-skin touch interfaces. Leveraging a compact wrist-mounted mmWave array, EpiPad transforms the forearm's skin surface into an extended input area that preserves familiar gestures such as tapping and sliding, while enriching them with multi-finger and force interactions common in commodity touchpads. To achieve stable force estimation, we develop a novel signal extraction method based on mmWave phase dynamics and implement a recognition pipeline composed of lightweight machine learning models. Experiments with 20 users show that, after a short one-time calibration of 3 minutes 54 seconds, EpiPad achieves a gesture recognition accuracy of 92.8%, force detection accuracy of 90.2%, and ranging precision of 10.2 mm on unseen users. A real-time evaluation validates the system's reliability for real-world deployment, and a follow-up survey further confirms that EpiPad offers a low learning cost, rich gesture diversity, and an input experience closely aligned with traditional touch interfaces.

CCS Concepts: • **Human-centered computing** → **Ubiquitous and mobile computing systems and tools**; *Mobile devices*; **Interaction devices**.

Additional Key Words and Phrases: mmWave Sensing, Smartwatch, On-skin Input, Touch Force Estimation

ACM Reference Format:
Yuliang Fu, Rakshita Ranganath, Huining Li, and Chenhan Xu. 2026. EpiPad: Enabling Multi-Finger and Force-Sensitive Skin Interface for Smartwatches. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 10, 2, Article 35 (June 2026), 35 pages. https://doi.org/10.1145/3810191

1 INTRODUCTION

As compact and lightweight wearable devices, smartwatches and other wrist-worn devices provide a ubiquitous and easily accessible interface for user interaction, and have gained huge popularity nowadays. However, with more and more functions and applications being integrated into smartwatches, one limitation becomes increasingly apparent: the size of the touchscreen is too small to allow the recognition of complex input commands (e.g., multi-finger input) required by these applications, which have already been supported by other common devices in daily life, such as smartphones and touchpads. This characteristic leads to a significant reduction

*Corresponding author.

Authors' Contact Information: Yuliang Fu, North Carolina State University, Raleigh, NC, USA, yfu34@ncsu.edu; Rakshita Ranganath, North Carolina State University, Raleigh, NC, USA, rrangan2@ncsu.edu; Huining Li, North Carolina State University, Raleigh, NC, USA, hli83@ncsu.edu; Chenhan Xu (corresponding author), North Carolina State University, Raleigh, NC, USA, cxu34@ncsu.edu.



This work is licensed under a Creative Commons Attribution 4.0 International License.
© 2026 Copyright held by the owner/author(s).
ACM 2474-9567/2026/6-ART35
https://doi.org/10.1145/3810191

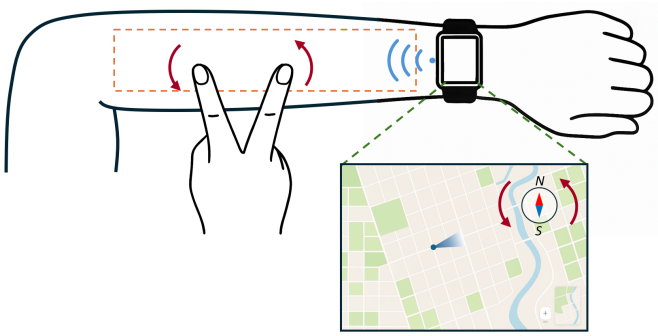


Fig. 1. EpiPad enables a similar input experience with large touchscreens and touchpads.

in user interactions as well as functionalities supported in smartwatch apps compared to their corresponding counterparts on smartphones [12].

The pressing need to bring more types of input for smartwatches has motivated researchers to develop off-screen interfaces using wrist-worn sensors. Among them, some researchers aim to transfer the input interface from a common touchscreen to other surfaces in the surrounding environment. Recent work achieved this with a multi-modality sensing method and succeeded in telling finger tapping positions from wrist vibrations [80]. However, the system requires external objects to generate detectable vibration upon tapping and to serve as a positional reference. This constrains the system's mobility to indoor environments, and weakens one of the dominating advantages of smart wrist-worn devices [59]. Moreover, with smartwatches commonly worn on the non-dominant hand [16], typing solely with the hand wearing them inevitably reduces user comfort and convenience.

Alternatively, an accessible interface to perform input commands is the user skin. With touching and sliding being the simplest and most universal inputs for touchscreens, previous works have focused on the on-skin detection of these activities to enrich the input for wrist-worn devices. Fingers on the hand come with 14 knuckles as natural landmarks, and provide inherent distinctions for finger tapping [7, 11, 27]. Meanwhile, the complex anatomy of the palm also provides clues to locate contact activities [10, 11, 84]. These works took important steps to build on-skin interfaces with an elevated user experience. However, the reliance on body landmarks limits the input locations to discrete predefined positions. Additionally, some research focuses on finger tracking on the forearm, providing an expanded interface with continuous positional input with one-dimensional infrared sensors [73] and electrical wave [90]. While these sensing methods are elegant solutions for simple one-finger tasks, they are not precise enough to handle interference brought by multi-finger contacts. As a result, while existing on-skin and wrist-based interfaces effectively expand single-finger input space, they remain limited in supporting complex, high-degree-of-freedom gestures such as screen rotation, zooming, and deep menu navigation, which are essential for intuitive mobile interactions [19]. Overall, there is still a significant misalignment between the operation experience on smartwatches and on other touch-based interfaces.

To bridge this gap, we present EpiPad, which is the first to enable multi-finger and force-sensitive input for smartwatches through mmWave-enabled on-skin touch interfaces. EpiPad employs a compact Single-Input Multiple-Output (SIMO) mmWave transceiver array that can be integrated along the side of a smartwatch or wristband. As shown in Fig. 1, by effectively turning the forearm into a full-function touchpad, EpiPad enables users to seamlessly transfer familiar interaction habits from large touchscreens and touchpads to wrist-worn devices, thereby extending input expressiveness without compromising usability.

The main challenges in realizing this system arise from two aspects. *First*, touch force estimation is difficult due to the subtle and continuous nature of skin deformation. Because minor skin deformation serves as the most consistent indicator of touch force, we utilize the mmWave phase signal, which offers sub-millimeter sensitivity, to infer force levels. However, in a practical input process, the arrival of fingers brings an irrelevant phase change consistent with the following skin deformations. This makes the clues directly associated with touch force mixed up with other processes in the phase values, bringing significant difficulty for separation. To overcome this, we conduct a detailed analysis of phase dynamics during touch and propose a novel feature extraction method that captures force-related patterns from the often-overlooked holding phase. *Second*, cross-user variability, including differences in multi-finger input habits and heavy-touch thresholds, limits the generalization of deep learning approaches. To address this, we design an input recognition pipeline that combines lightweight deep-learning models with data augmentation. Specifically, we introduce data processing approaches to remove user-dependent features and synthetic data generation strategies to diversify training samples and reduce bias toward specific users. The network is pre-trained on a cross-user dataset to learn invariant representations across skin properties and interaction styles, and subsequently fine-tuned with a small amount of user-specific data. This two-stage adaptation enables rapid personalization and maintains generalization. As a result, EpiPad can achieve stable performance on a new user after a one-time calibration of less than four minutes.

We built our prototype with a specially designed positioning of antennas, introducing an elevation and depression angle to better perceive on-skin activities. The system provides a 15×4 cm input area of 60 square centimeters on the forearm. Our input recognition task is broken down into three parts: force-decoupled gesture recognition, touch force level detection, and on-skin finger tracking. For system evaluation, we conducted experiments on 20 participants and conducted a user survey thereafter. Results show that with our proposed one-time calibration process, our system reaches a high accuracy on all three tasks above for unseen users. User feedback also recognizes our system as an intuitive input method with low learning cost.

Our contribution can be summarized as follows:

- For the first time, we enabled multi-finger and force-sensitive input for smartwatches through mmWave-enabled on-skin touch interfaces. With a compact wrist-worn mmWave sensor array, the system transfers prevalent input commands on large touch devices to a compact smartwatch, thereby providing users with a consistent input experience with other touch-enabled smart devices.
- We build an input recognition pipeline to sense different dimensions of user input. A novel signal processing method is deployed to extract force-related patterns from mmWave phase signals, and lightweight models are built to make the system adapt easily to new users.
- We evaluate our proposed system with a comprehensive study on 20 participants. Results show that our system has a superior performance for input recognition, with an accuracy of 92.8%, 90.2%, and 10.2 mm for gesture recognition, force detection, and finger tracking, respectively. A follow-up survey further confirms that our system provides both diversified input commands and low learning cost.

2 mmWAVE PRELIMINARIES

2.1 Object Ranging

Our system is built with mmWave sensors, which emit linear Frequency-Modulated Continuous-Wave (FMCW) signals, as shown in Fig. 2. The property of the FMCW signal allows it to achieve a large product of time and bandwidth, and further ensures both a high signal-to-noise ratio (SNR) and an accurate range resolution [41]. A single sweep in the frequency domain is called a chirp. For an object at a distance R, the transmitted chirp will be reflected by the object and travel back to the receiver in $\tau = 2R/c$, where c is the speed of light. The receiver mixes the transmitted signal and the reflected signal, resulting in an intermediate frequency (IF) signal being

sampled, represented by:

$$S_{IF}(t) = A_{IF} e^{j(2\pi S\tau t + 2\pi f_0 \tau)}, \tag{1}$$

where A_{IF} is the amplitude of the IF signal, f_0 is the starting frequency of the chirp, and $S = B/T_c$ is the slope of the chirp, which equals the quotient of its bandwidth B and duration T_c. From the equation above, we can derive the frequency and phase of the IF signal

$$f_{IF} = S\tau = \frac{2RB}{cT_c}, \qquad \phi_{IF} = 2\pi f_0 \tau = \frac{4\pi R}{\lambda}, \tag{2}$$

where $\lambda = c/f_0$ is the wavelength of the chirp at the starting frequency. With a discrete Fourier transform (DFT) performed, the frequency and phase of the IF signal can be revealed, and the distance of the target can be calculated as $R = cf_{IF}/2S$. The range resolution is further limited by the bandwidth of the radar, represented by:

$$R_{res} = \frac{c}{2B}. \tag{3}$$

Although the frequency of the IF signal plays a vital role in object detection, the phase signal is commonly omitted in traditional radar applications. The primary reason lies in the problem of phase wrapping. With the complex signal being periodic, the phase signal "wraps around" every half wavelength, corresponding to a phase change of 2π. Therefore, the calculated phase value does not provide an unambiguous indication of the object position. However, the short wavelength of mmWave radar makes the phase signal sensitive to position shifts at millimeter or sub-millimeter level. This is significantly smaller than the range resolution derived from the IF frequency, which typically varies from centimeter to meter level. This feature makes the relative change of phase capable of sensing minor deformations. Such a sensitivity to minor deformations makes it possible for the phase signal to capture subtle movements, including the vibration of the skin under touch force, which is exactly what our system needs to monitor.

2.2 Angle-of-arrival Calculation

The ranging method above allows us to estimate the distance of an object. However, with only one pair of antennas, angular positions cannot be reflected in the received signal. To calculate an object's angle-of-arrival (AoA), a receiving antenna array is required. As illustrated in Fig. 3, with two receivers placed with a spacing d, an object appearing at azimuth θ will have a difference in its signal propagation path towards the two receivers. With Eq. 2, we can calculate the AoA of the signal using the phase difference between the two antennas as:

$$\theta = \arcsin\left(\frac{\lambda(\phi_{Rx1} - \phi_{Rx2})}{2\pi d}\right), \tag{4}$$

where ϕ_{Rx1} and ϕ_{Rx2} represent the phase values collected by the two mmWave receivers, respectively. Such a method for AoA calculation allows us to know not only the object's distance from the sensor, but also its azimuth towards the sensor array. Moreover, with another receiver array expanding in orthogonal directions, the radar

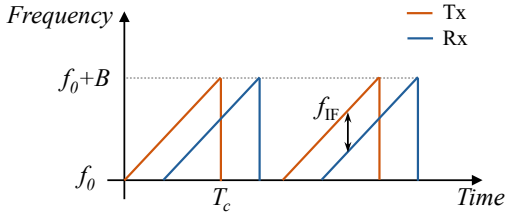


Fig. 2. FMCW chirps over time.

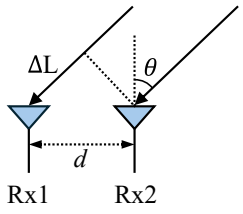


Fig. 3. Multiple mmWave receivers for AoA sensing.

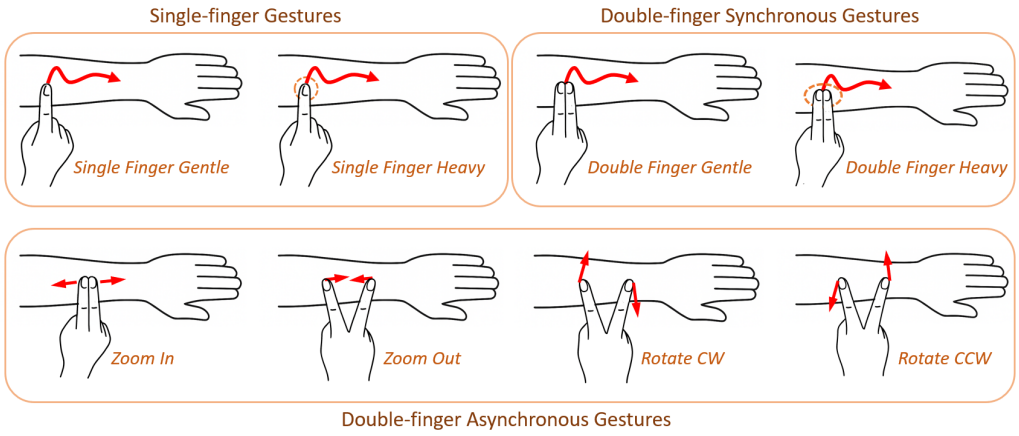


Fig. 4. Gesture set for EpiPad derived from commodity touchpads.

will be able to calculate both the azimuth and the elevation, forming a complete spherical coordinate of an object. Therefore, it is feasible for an mmWave sensor with multiple receiving antennas to perform 2-D ranging on a plane, or even 3-D ranging in space.

3 INTERACTIVE SYSTEM DESIGN

3.1 Design Considerations and Gesture Set

In developing our system, we prioritize three key design goals: enabling a diverse set of input gestures, providing users with high freedom of expression, and ensuring high prior familiarity to minimize learning effort. To achieve these objectives, we build our gesture set from the well-established interaction vocabulary of touchpads and smartphone interfaces, where tapping and sliding gestures are already deeply embedded in users' daily experience. Based on these basic single-finger gestures, we further enrich the gesture set by incorporating multi-finger inputs and force-based touch interactions, which have also gained high popularity among modern commodity touchpads. The inclusion of these dimensions allows for greater expressiveness and control while maintaining an intuitive user experience, thereby facilitating seamless adoption of the proposed on-skin interaction approach.

According to the input semantics as well as the fingers' moving patterns, we divide double-finger input gestures into two categories: synchronous gestures and asynchronous gestures. Synchronous gestures, where the fingers involved perform identical movements, are commonly seen in tapping and sliding inputs. These gestures share similar use cases with traditional single-finger inputs, with the location of finger(s) combined to form a complete input command. Asynchronous gestures, where the fingers perform distinct movements individually, include the gestures of rotation clockwise (CW), rotation counterclockwise (CCW), zoom in, and zoom out. These gestures, in contrast, convey a complete input command once performed. In our gesture design, force touch is integrated specifically into single-finger and synchronous gestures, but not into asynchronous interactions. This decision is grounded in both ergonomic and functional considerations. Synchronous gestures are primarily simple linear or discrete actions, where additional force is easy to perform and naturally serves as an extra dimension to convey emphasis or mode switching. In contrast, asynchronous gestures hold fixed meanings that are hard to extend. With these gestures inherently involving complex finger coordination, introducing force variation would only increase cognitive and motor load. Therefore, limiting force touch from asynchronous gestures strikes a balance between interaction richness and intuitive usability.

To make the input activities detectable for sensors from the watch dial, two surfaces adjacent to the wrist: the back of the forearm and the back of the palm, become possible candidates for our system. Between them, we consider the back of the forearm as a more ideal contact surface for two reasons. On the one hand, the forearm provides a significantly larger area than the palm, facilitating accurate performance of multi-finger gestures. On the other hand, a thicker layer of subcutaneous tissue and muscle brings a more comfortable experience for heavy pressing activities. Based on these analyses, we propose our gesture set illustrated in Fig. 4.

3.2 System Overview

With the considerations above, we design a wrist-worn system to sense touch-based activities on the back of the forearm. The system integrates a compact mmWave sensing unit, a signal processing pipeline, and a machine learning–based gesture recognition framework. The mmWave module emits FMCW chirps and receives reflected signals that encode subtle surface deformations and finger movements. The collected raw data are converted into amplitude and phase representations, which are then processed to reveal features reflecting contact location and gesture types. These features are then fed into neural network models to recognize user input.

Our input recognition is implemented with a logic shown in Fig. 5. The system is in an idle state by default when being powered up. The detection of asynchronous is first performed due to their independent semantics. If no asynchronous gestures are identified, the signals are then examined for single-finger and synchronous gestures. If no touching activities are detected, the system holds an idle state and starts the next detection loop. Otherwise, if a touch with either a single or double finger is recognized, the detection of force will be triggered, and the number of fingers and force level are combined as the input command. With the touch remaining, the tracking of fingers is continuously performed, updating the positional input from the user. Simultaneously, the synchronous detection module is also active, and when a finger removal is detected, the finger tracking will be stopped, marking the end of a single-finger or synchronous input command.

To ensure a reliable input performance on all effective ranges for different users, we rely on our insights in both the hardware layout and the recognition algorithms, with relevant considerations further discussed in the following two sections.

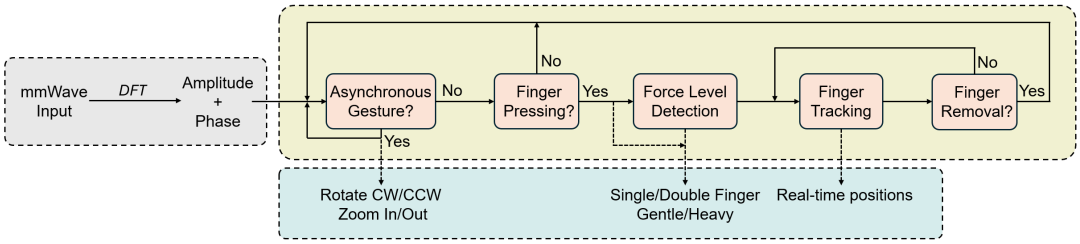


Fig. 5. EpiPad input recognition logic.

4 HARDWARE PROTOTYPING

4.1 Sensor Configurations

We implement a sensor system based on an Infineon BGT60TR13C chip, which features a 1-Tx 3-Rx setup [1]. With 3 receiving antennas forming an L shape, as shown in Fig. 6, the received signal contains diverse information from the azimuth and elevation directions. For our short-range sensing task, we utilize the full frequency band from 58 to 63.5 GHz, achieving an utmost range resolution of 2.73 cm. The FMCW chirps are sent with an interval of 1 ms, resulting in a 1 kHz slow-time sample rate.

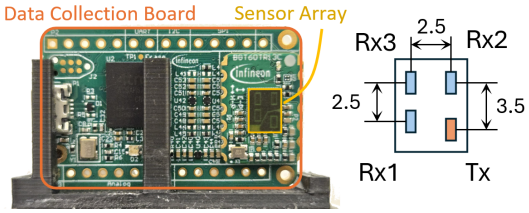


Fig. 6. Sensor transceiver layout (dimensions in mm).

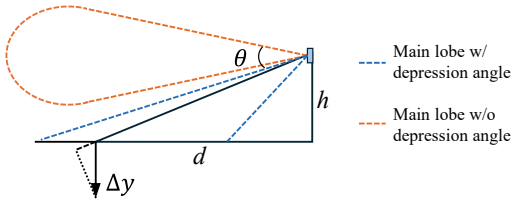


Fig. 7. Geometric diagram for sensor placement.

4.2 Transceiver Positioning

mmWave sensors rely on the change in wave propagation length to sense the displacement of objects. With the vertical deformation of skin being the critical marker for on-skin touch force, the radar will achieve its best performance of touch force estimation when the wave's propagation path is parallel to the direction of deformation, i.e., having a top-down view of the arm. However, such a requirement is impossible to satisfy, given that smartwatches are usually worn on the wrist and are close to the skin. In this setup, a significant challenge exists: with pressing activities bringing a minor vertical deformation on the skin, the response of mmWave signals will have minimal changes when the wave's direction of propagation is orthogonal to the direction of movement.

To address this problem, the angle between the signal propagation path and the skin surface needs to be increased. We achieve this by elevating the sensor array from the skin surface, as shown in Fig. 7. In this layout, the response of the phase signal to a vertical deformation is expressed as

$$\Delta\phi = \frac{2\pi}{\lambda} \cdot \frac{\Delta y}{\sqrt{1 + (\frac{d}{h})^2}}, \tag{5}$$

where $\Delta\phi$ is the change of phase signal, λ is the mmWave's wavelength, Δy is the vertical displacement, h is the height of antennas above the skin, and d is the horizontal distance between the place of deformation and the antennas. The equation shows that for any vertical deformation on the skin, a higher position of the transceiver array will always increase the response of phase signals. Therefore, we expect our system to be placed near the top of a smart wrist-worn device. Referring to the dimensions of the Apple Watch Ultra 2[1], which has a thickness of 14.4 mm, we place our system with the sensor array 14 mm above the skin.

The introduction of a vertical elevation, however, raises another problem. Although the transmitted wave from the sensor propagates in the form of a spherical wave, a typical mmWave antenna does not radiate energy uniformly in all directions; instead, its radiation pattern resembles a lobe-shaped distribution, with maximum gain along the boresight direction and gradually decreasing gain as the angle deviates from the center. As per our schematics, when the sensor views from a slightly elevated view from the side, the skin surface extends across a wide angular range relative to the antenna. As a result, regions aligned with the antenna's main lobe are captured with high signal strength, whereas areas toward the periphery are exposed to lower intensity. This inherent characteristic of antenna radiation patterns introduces non-uniformity in our sensing system, and activities performed on segments with low radiation intensity may have low effects on amplitude and phase, even though the trajectory of movement corresponds to the signal propagation path.

To mitigate such an effect, we introduce a depression angle to the placement of the transceiver array. As shown in Fig. 7, a depression angle moves the center of the main lobe onto the skin and increases the overall signal intensity on the skin. However, the change of the depression angle affects differently on different parts of the

[1] https://www.apple.com/am/apple-watch-ultra-2/specs/

skin. With a depression angle θ, the centerline of main lobe falls on the skin surface at a distance $x_{ML} = h/\tan\theta$. As the angle θ becomes larger, the change of intensity at a fixed position x_i on skin is non-monotonic, with the intensity increasing when $x_i > x_{ML}$, and decreasing when $x_i < x_{ML}$.

Although the radiation pattern can be calculated for an ideal antenna, factors including mutual coupling and packing make the real-world patterns deviate significantly from the theoretical values, and their characterization relies highly on experiments [1]. Based on the analysis above, we design a preliminary experiment to find the optimal depression angle for our system. An ideal placement should be optimized on two aspects: the signal's sensitivity to on-skin objects, and its response to minor skin deformations. Thus, we design a two-step preliminary study. We mount the radar on a wristband worn by the user, with a set of holders providing a fixed 14 mm height, and depression angles from 0° to 20° with a 4° interval. A calibrated capacitive force sensor is attached to the user's fingertip to monitor the touch force in real time. We then ask the user to (1) place his finger gently on the skin, and (2) apply a force of 1 N on the skin. With the typical length of human forearm being 21 to 27 cm [30], we set the maximum effective range of our system to 18 cm to ensure a high usability across populations. Therefore, for all six depression angles, the process is repeated at positions of 3 cm and 18 cm from the radar for 10 times each, representing the closest and farthest effective range for our proposed system. Specifically, we focus on two metrics as the performance indicator: The change of amplitude in process (1), and the change of phase in process (2), which correspond to the above-mentioned aspects.

Results plotted in Fig. 8 show that during the change of depression angles, the closer boundary of the effective range always shows a more significant response than the farther boundary. This allows us to focus on the farther boundary only when picking the best depression angle. For the farther boundary at 18 cm, both metrics achieve the best performance at $\theta = 8°$. Therefore, such a placement is adopted for the rest of our study. To reliably maintain such a positioning, we mount the mmWave sensor on the wrist with a customized holder and double-sided strip, with the hardware prototype shown in Fig. 9.

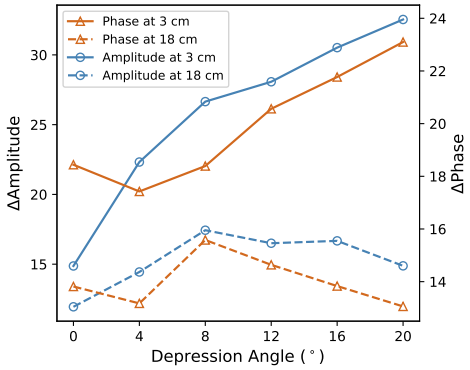


Fig. 8. Amplitude and phase at different angles.

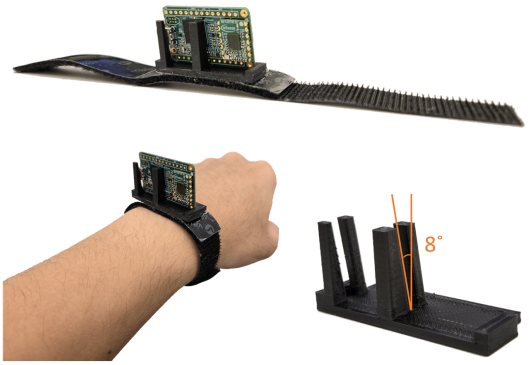


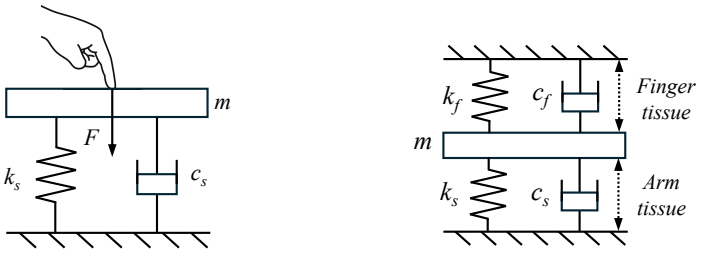
Fig. 9. EpiPad hardware prototype.

5 EpiPad SENSING SCHEMES

5.1 Touch Force Modeling and Feature Extraction

The generation of fingertip forces is a subtle process involving minor deformations, primarily captured in the phase signal of mmWave sensors. With phase signals being highly sensitive to noise and irrelevant movements, it is important to understand the force generation process and extract the most relevant signals for force level prediction.

According to the Kelvin–Voigt material model, the mechanical model of on-skin vertical contact can be simplified as a combination of an elastic spring and a viscous dashpot connected in parallel, as shown in Fig. 10a. The contact surface can be abstracted as a rigid object with mass m, supported by a spring of constant k and a damper of damping coefficient c, both with negligible mass. Previous works have analyzed the movement characteristics of the system under the assumption that the finger-skin contact is a transient process, and further utilized the vibrations after the finger has left the skin surface [7, 80]. However, such an analysis cannot be generalized to our system. With on-skin input gestures easily lasting for seconds, the transient contact assumption is no longer valid. Moreover, an intolerable input delay would also occur if the post-contact response were used for our recognition.



(a) Traditional model. (b) Proposed model for the holding stage.

Fig. 10. Physical model of on-skin contact.

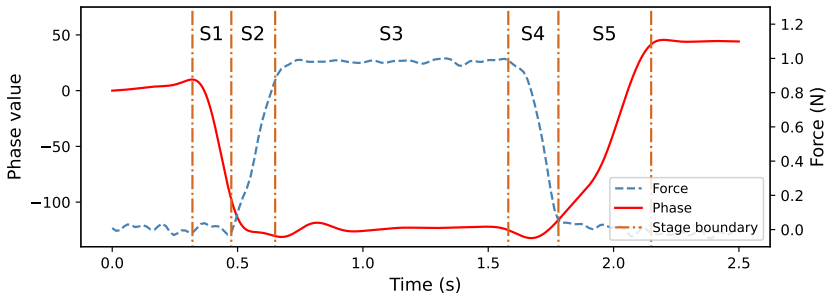


Fig. 11. Touch force and phase response during the on-skin touching process of S1: Finger arrival in air, S2: Finger pressing down on the skin, S3: Finger holding on the skin, S4: Finger removing from the skin, S5: Finger departure in air. We exploit patterns in S3 for our force level detection.

To understand the physical process where the finger establishes, remains, and releases contact with the skin, we investigate the received phase signal in conjunction with the applied force collected by the force sensor. As illustrated in Fig. 11, the on-skin touch process is divided into five stages based on the temporal patterns of these two signals. During stages 1 and 5, the phase signal exhibits a large range of variation. However, the force sensor reading remains near zero, indicating that the finger is not in contact with the skin. This phenomenon suggests that the most substantial phase changes during the touching process are brought by the in-air motion of the hand, which is not directly related to the applied touch force. Stages 2 and 4, characterized by rapid changes in

the force signal, represent the moments of initial contact and final release, respectively. Although these stages involve significant variations in the applied force, the phase signal follows a pattern similar to that observed in the preceding and succeeding in-air stages. Consequently, although these two stages are most directly associated with the on-skin forces, it is extremely challenging to isolate their phase signal without external references and further utilize them for force-related predictions.

The observations above drive us to focus on stage 3, where the on-skin touch force remains stable. We mark the start of this stage with the force's first arrival at a stable level. With no change in external excitation in this stage, we first look into the kinematic properties inherited from the previous stage 2. With the change of force being rapid, we simplify the initial contact stage into movements under a constant external force F_0. Representing the vertical displacement as x, we have the equation of motion as

$$F_0 = m\frac{d^2x(t)}{dt^2} + c_s\frac{dx(t)}{dt} + k_s x(t), \tag{6}$$

where m is the equivalent mass of the contacted skin, c_s and k_s are the damping coefficient and spring constant of the skin, respectively. To homogenize the differential equation above, we switch to the coordinate of the new equilibrium state, where $k_s x_{eq} = F_0$, and make the substitution $y(t) = x(t) - x_{eq}$. The homogeneous equation for y is derived as

$$m\ddot{y} + c_s\dot{y} + k_s y = 0. \tag{7}$$

For simplicity of expression, define the oscillation parameters of the skin ζ_s (damping ratio), ω_{ns} (natural frequency), and ω_{ds} (damped frequency) as

$$\zeta_s = \frac{c_s}{2\sqrt{k_s m}}, \quad \omega_{ns} = \sqrt{\frac{k_s}{m}}, \quad \omega_{ds} = \omega_{ns}\sqrt{(1-\zeta_s^2)}. \tag{8}$$

With initial conditions $y(0) = -x_{eq}$, $\dot{y}(0) = 0$, in the underdamping conditions ($0 \le \zeta_s < 1$), the solution to Eq. 7 is given as

$$y(t) = -x_{eq}e^{-\alpha t}(\cos(\omega_{ds}t) + \beta\sin(\omega_{ds}t)), \tag{9}$$

where $\alpha = \zeta_s\omega_{ns}$, and $\beta = \alpha/\omega_{ds}$. For the first time the system reaches the equilibrium state, we seek the smallest solution $t_1 \in (0, \pi)$, satisfying

$$\cos(\omega_{ds}t_1) + \beta\sin(\omega_{ds}t_1) = 0. \tag{10}$$

The equation above gives the time t_1 when stage 2 comes to an end and stage 3 starts. With the position x_{eq} fixed for the start of stage 3, the velocity becomes the most important property that has a long-lasting influence on the following movements. Therefore, we calculate the velocity at t_1, which is also the initial velocity of stage 3:

$$v_1 = \dot{y}(t_1) = \frac{F_0}{k_s}\omega_{ns}\exp\left(-\beta\cdot(\pi - \arctan(\frac{1}{\beta}))\right) \propto F_0. \tag{11}$$

With all other parameters being the intrinsic properties of the skin and underlying tissues, Eq. 11 shows that the starting velocity of stage 3 is proportional to the applied force F_0. Such a characterization establishes a physical linkage between the touch force and the kinetic behavior during the sustained contact stage. Based on such a finding, we further investigate the motion characteristics of stage 3. While the finger and skin remain in contact, similar to the previous analysis, the magnitude of movement in this stage necessitates a reconsideration of the mechanical model. In stage 2, the observed motion of the skin results primarily from the deformation of the skin and subcutaneous layers, while deformation within the fingertip itself is minimal. In contrast, the smaller vibration amplitude in stage 3 leads to comparable dynamic responses in both the fingertip and the skin. To capture this behavior, we propose an updated mechanical model, as illustrated in Fig. 10b, where the fingertip

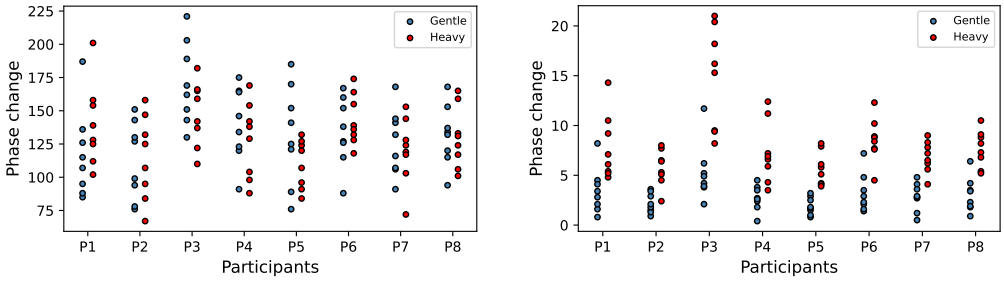
tissues are represented by an additional spring–damper pair connected to the existing system. By defining the spring constant and damping coefficient of the fingertip as k_f and c_f, the equation of motion is given as

$$m\ddot{y'} + c_c\dot{y'} + k_c y' = 0, \tag{12}$$

where $k_c = k_s + k_f$, $c_c = c_s + c_f$, standing for the combined constant and coefficient of skin and fingertip, respectively. With initial conditions $y'(t_1) = 0$ and $\dot{y'}(t_1) = v_1$, we have the solution as

$$y'(t) = \frac{v_1}{\omega_{dc}} \exp(-\zeta_c \omega_{nc}(t - t_1)) \sin(\omega_{dc}(t - t_1)). \tag{13}$$

The combined oscillation parameters ζ_c, ω_{nc} and ω_{dc} are calculated similar to Eq. 8. This implies that although stage 3 has no correspondence to any major body movements, it does perform the motion of an underdamped oscillation, with the amplitude v_1/ω_{dc} directly influenced by the applied force.



(a) Complete touch process.

(b) Holding stage only.

Fig. 12. Range of phase changes under different touch force levels.

To validate our model, we look into the change of phase when different forces are used for touching. Given that body conditions, including skin stiffness and fat/muscle compositions, commonly lead to different skin mechanical properties, we recruited eight participants with different body shapes and exercise habits, with body mass index (BMI) ranging from 18.3 to 30.3. The participants performed gentle and heavy touch eight times each on their left forearm at a distance of 9 cm from the radar. With the holding stage 3 manually identified, the ranges of phase change during the whole pressing process and in the holding stage are each calculated. In Fig. 12, we can tell that within the same user, while the phase range in the whole process has minimal distinction between different force levels, the range of phase vibration in stage 3 generates a more stable identifier for the force applied.

This observation validates our proposed physical model and provides us with practical insights for mmWave signal preprocessing. A straightforward approach to isolate force-related information would be to extract the segment corresponding to stage 3 from the entire pressing sequence. However, this operation requires prior knowledge of the equilibrium phase value, which is difficult to determine in real time. To address this limitation, we seek an alternative indicator that can be derived directly from the observed signal. In our sensing context, the optimization of a cropped window aims to achieve two objectives: first, to minimize the overall trend component, distinguishing it from the in-air and initial pressing stages; and second, to maximize the signal variance, thereby preserving the oscillatory features that characterize dynamics. This balance allows the retained segment to emphasize the vibration-related information while avoiding the later holding stage, where oscillation has become minimal due to damping. According to these two targets, we propose our own indicator to detect the optimal

Algorithm 1 Window Extraction for Force Prediction

Require: mmWave amplitude $A(r,t)$ phase $\phi(r,t)$ over time t and range bins r, sample interval t_s
Ensure: Cropped time window W for fingertip force prediction

1: **if** new pressing detected in window $[t_0, t_1]$ **then**
2: $ref_bin \leftarrow find_max_variance_bin(A[t_0, t_1])$ // Phase data preprocessing
3: $\phi_r(t) \leftarrow \phi[ref_bin]$
4: $\phi_{fr}(t) \leftarrow lowpass_filter(\phi_r)$
5: $t_{start} \leftarrow find_max_time(|\dot{\phi}_{fr}|)$
6: $t \leftarrow t_{start}$
7: **while** $t < t_1$ **do**
8: $\phi_W \leftarrow \phi_{fr}[t : t + WINDOW_LENGTH]$ // Indicator calculation based on Eq. 14
9: $trend \leftarrow \phi_W[-1] - \phi_W[0]$
10: $range \leftarrow Max(\phi_W) - Min(\phi_W)$
11: $Indicator[t] \leftarrow Abs(trend/range)$
12: **if** $t > t_{start} + 2t_s$ **then**
13: **if** $Indicator[t] > Indicator[t - t_s] < Indicator[t - 2t_s]$ **then**
14: **return** $[t, t + WINDOW_LENGTH]$ // Window selection
15: **end if**
16: **end if**
17: $t \leftarrow t + t_s$
18: **end while**
19: **return** $[t_1, t_1 + WINDOW_LENGTH]$
20: **end if**

window location. For the phase signal in a cropped time window ϕ_W, the indicator I is calculated as

$$I = \left| \frac{\phi_W[-1] - \phi_W[0]}{Max(\phi_W) - Min(\phi_W)} \right|. \tag{14}$$

Such an indicator can be interpreted as the absolute value of a min-max normalized trend, falling in the range of $[0, 1]$. A favored time window containing the damped oscillation period will achieve a low value approximating 0 on this indicator, whereas windows containing a rapid fall or invariant signal will yield a value near 1. **Algorithm 1** explains how the indicator is used to complete the cropping process. The algorithm starts after a pressing activity is detected within a larger time window. As shown in lines 2 to 5, a preprocessing is first performed on the phase data. To decide on a single range bin as the reference for cropping, the variance of amplitude along time is calculated. The range bin with the largest variance represents where the majority of finger reflection is distributed, and the phase signal from it is further used as the reference. To reduce the influence of outlier readings, a low-pass Butterworth filter is applied to the phase signal. Before the start of window selection, it is important to note that phase fluctuations before the pressing process can possibly introduce local minima on the indicator and further lead to a wrong window being selected. Therefore, to prevent this from happening, we start the window selection process after the input activity has been started. This is achieved by calculating the change rate of the phase. The maximum absolute value in the change rate corresponds to the fastest movement of fingers in the touch process, which typically appears in mid-air. Therefore, we can reliably assume that the holding stage will not start before this point, and that the change of phase values is dominated by finger movements and skin deformations thereafter.

After such a start point is determined, we apply fixed-length sliding windows to the phase signal, and the indicator I is calculated for the windows at each position, corresponding to algorithm lines 7 to 10. Line 11 and below further explain the window selection process. According to the previous analysis, the indicator of our preferred window should yield a local minimum over time. With possible interference already removed, the first window with a local minimum is adopted. Fig. 13 shows a sample window cropped with our extraction algorithm. It demonstrates that our method is successful in isolating the oscillation signal from the in-air and initial pressing stages, and increases its presence after a normalization transformation.

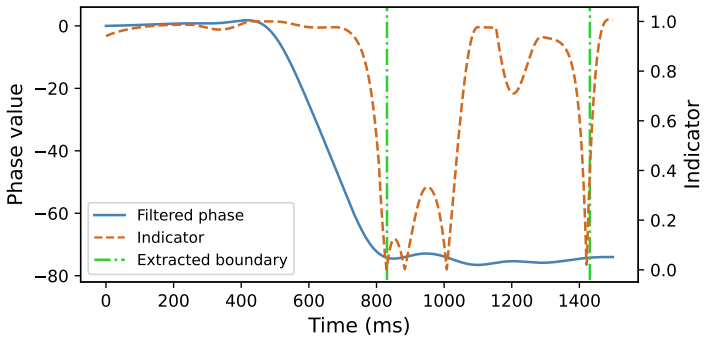


Fig. 13. Feature extraction example with 600 ms window length.

5.2 Multi-finger Sensing Principles

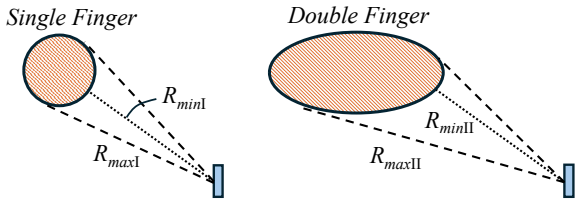


Fig. 14. Single- and double-finger reflection diagram.

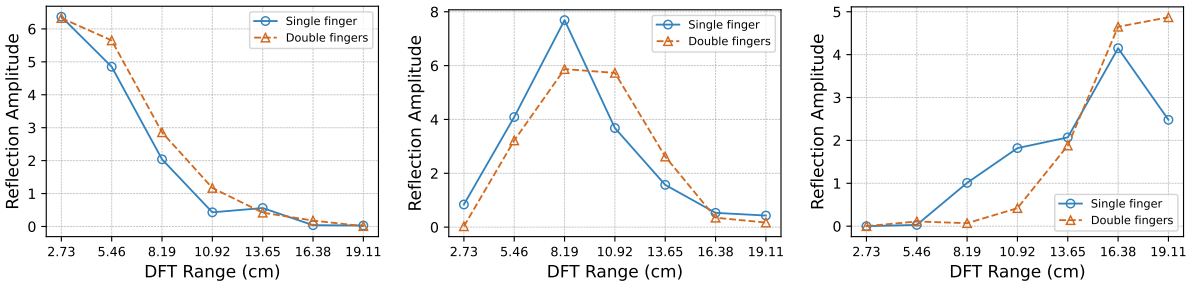
Common on-skin multi-finger input gestures, including sliding and tapping, are typically performed with the fingers held close together and oriented sideways relative to the wrist and elbow. With our sensor system mounted on the wrist, this configuration introduces a sensing challenge. While mmWave possesses a penetration ability, the depth is limited to the sub-millimeter range for epidermis and dermis layers [2]. Consequently, when the fingers are positioned such that the mmWave sensor views them entirely from the side, the system will receive no reflected waves from the farther finger, making multi-finger gestures barely detectable.

We address this issue with our multi-antenna setup, which provides an intrinsic spatial diversity. With receiving antennas placed at different locations, it is ensured that at least one receiving antenna has an oblique line of sight to the fingers. However, the mere reception of a reflected signal does not guarantee that individual fingers can be distinguished. mmWave's ability to resolve multiple reflectors is fundamentally constrained by its range resolution, which depends solely on the signal bandwidth. Working on a 5.5 GHz bandwidth, our system reaches a

resolution of 2.73 cm, which is superior among mmWave sensors. Nevertheless, during synchronous gestures, the fingers often move with only millimeter-level separations. This spacing is substantially smaller than the sensor's resolution limit, making it exceedingly difficult for conventional object detection algorithms to distinguish the presence of multiple fingers.

Such a physical limit makes us change our perspective on the presence of multiple fingers. From the view of mmWave signals, the introduction of additional fingers does not bring additional targets to recognize; instead, reflections of the two fingers are mixed, leading to them being detected as a single object with a different shape. For simplicity, we approximate a single finger as a cylinder, and the combination of two fingers can be modeled as an elliptical cylinder with the same width, and the schematics for one and two-finger sensing can be illustrated as Fig. 14.

The change of the equivalent shape brings two differences for mmWave sensors. First, the cross-sectional area increases when additional fingers are involved, leading to a larger amplitude of the received peak. Second, the span of the object in space is enlarged, corresponding to a wider peak across different range bins. With the natural shape of the forearm having a relatively large aspect ratio, the angle-of-arrival is typically small. In such cases, the change in cross-sectional area is relatively small, with noise and random variations possibly suppressing the difference. We plot a range-reflection plot with single and double fingers at three different positions on skin in Fig. 15, and the energy distribution confirms our analysis that the spatial expansion of the reflection peak serves as a more stable criterion for multi-finger detection.



(a) R=3 cm. (b) R=10.5 cm. (c) R=18 cm.

Fig. 15. Range-reflection plot for single- and double-finger on skin.

5.3 Input Recognition Pipeline

Our prediction framework consists of three complementary tasks: (1) force-decoupled gesture recognition, (2) force levels estimation, and (3) two-dimensional finger tracking. Each task is characterized by distinct signal features. Gesture recognition, independent of applied force, primarily relies on the temporal patterns of reflected amplitude signals. Force level prediction, in contrast, is governed by the temporal variations of the phase signal and requires appropriate window selection in advance to isolate force-related segments. Finger tracking depends less on temporal dynamics and more on the transient spatial distribution of signal energy across range bins and radar receiving channels. Given the distinct signal patterns, different signal processing procedures and machine learning models are deployed to complete the three tasks.

5.3.1 Signal Preprocessing. The raw IF signal is composed of reflections of different objects at the corresponding distances. Among them, the static body parts and background objects generate significant responses irrelevant to our interaction activities, and increase the influence of the environment and cross-user distinctions. Therefore,

to extract the patterns of the input activities, we deploy a two-fold approach on the calculated amplitude. First, with most interference from the background environment coming from farther distances, we drop the range bins where a legal input gesture will not be performed. With our bin width being 2.73 cm and the effective range being 3 to 18 cm, we keep only range bins 1 to 7 for further processing and prediction input. Second, with most irrelevant body parts being relatively static and contributing a near-invariant reflection amplitude, we remove them by using the first 10 ms of data when the device is first powered as a reference signal, and deduct the mean value of amplitude from the following signals.

The phase signal, however, has a different nature of changes. Being highly sensitive to subtle deformations, the slight but continuous change of limb positions can easily cause a drift in the phase signals, and eventually bring a noticeable deviation from the initial values. Therefore, on top of the removal of irrelevant range bins, for each time window used for prediction, we align the starting value of phase from each receiving antenna and range bin to $[0, 2\pi)$ and unwrap the following phase values in accordance with these starting values. With the continuous drift being minor compared to phase shifts caused by movements, this approach avoids the use of inaccurate absolute values but keeps the two patterns we expect the phase signal to provide: the changing trend within a time window, corresponding to the skin deformations, and the relative values across antennas, which reflect the angular distribution of the fingers.

5.3.2 Data Preparation. The preprocessed amplitude and phase are combined differently for different prediction tasks. Among the force-decoupled gestures, single-finger, synchronous, and asynchronous double-finger gestures all involve macro movements of the hand, and consequently have their patterns in the reflected amplitude. However, the emphasis of recognition is different. For the first two tasks, we focus on the detection of finger(s)' arrival to and departure from the skin, which lasts for a relatively short time and may be repeatedly performed in a short period. Asynchronous gestures, in contrast, take a significantly longer time to complete. Therefore, we use two time frames simultaneously: a 1-second time frame consisting of the amplitude for accurate and timely detection of single and synchronous gestures, and a 3-second time frame for a stable recognition of asynchronous gestures.

Force level recognition, as discussed previously, requires a cropping based on the input recognition results. Although primary clues of force levels are represented in phase signals, amplitude channels act as important references for the importance of different range bins. Thus, both amplitude and phase channels are used for the detection of force levels, with the influence of window length further discussed in the evaluation sections. Finally, the finger tracking is achieved by estimating the transient position of the finger(s) within a small time period. With the signals' development over time being unimportant, we take the amplitude and phase channels and calculate their average along time in a small 200 ms time window. This preserves the signal's energy distribution across the space and increases the system's robustness to transient error values.

To avoid model overfitting and expand the scale of our data, we implement two data augmentation techniques. (1) Amplitude jittering. The samples used for training are multiplied by random factors ranging from 0.95 to 1.05. This approach helps the model adapt to different hand sizes and finger diameters, as well as increase its robustness to noise. (2) Random time shift. The training samples involving time windows are shifted along the time axis with a time -200 to 200 ms, with the vacant period filled with Gaussian noise centered at the ending or starting value. This prevents the model from fitting to the time when the gesture is performed in a time frame.

5.3.3 Deep Learning Models. An important consideration in our deep model design is the ability to support few-shot learning. Unlike contact-based sensing methods, mmWave sensors capture reflections from the entire space above the arm, making the received signals sensitive to variations in hand postures. As is previously illustrated in Fig. 12, different users exhibit distinct thresholds for heavy-touch activities. In addition, users vary in touch behavior, such as whether they use the fingertip or finger pad, their hand orientation, finger-skin contact angle, and rotation axis. These inter-user differences introduce overlap between gesture categories and

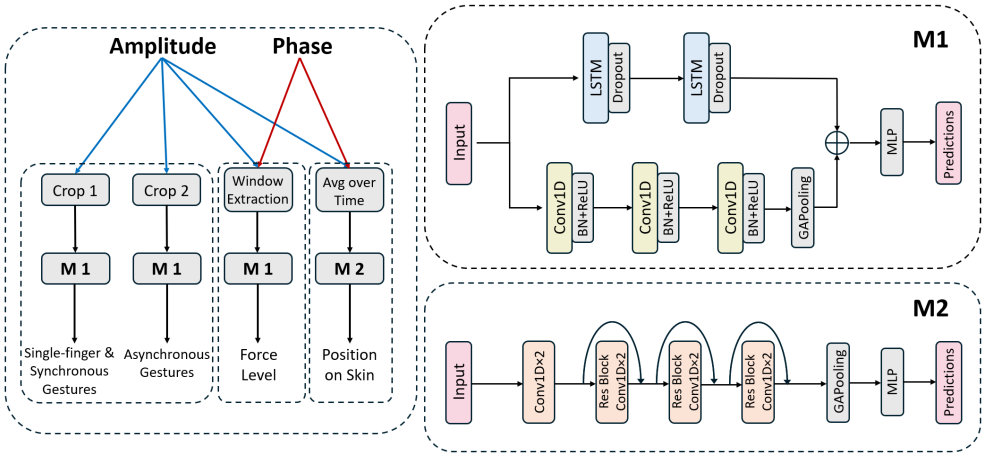


Fig. 16. EpiPad model structure and prediction workflow.

significantly constrain the generalizability of a universal cross-user model. To address this challenge, we aim to design models capable of adapting effectively with only a small amount of user-specific data. This requirement precludes the use of transformer-based architectures, which typically demand large datasets to achieve stable convergence. Instead, we adopt lightweight RNN- and CNN-based models with a limited number of layers and parameters, which enable efficient fine-tuning for individual users while maintaining a high adaptability.

According to the different feature dependencies for our tasks, we deploy and train separate deep learning models for each task to ensure optimal feature extraction and task-specific performance, with the structures shown in Fig. 16. For signals related to temporal patterns, we adopt a hybrid model for classification. The signal patterns are extracted through a parallel LSTM-FCN structure with two LSTM layers with hidden sizes of 128 and 64, and three one-dimensional convolutional layers. The two parallel branches are capable of handling different types of signal patterns, with LSTM possessing the ability to memorize long-term temporal patterns, and convolution layers excelling in handling local information. The outputs of the two individual branches are concatenated and passed to a multilayer perceptron (MLP) composed of fully connected layers, which maps the features extracted to the prediction values of gestures or force. With the hybrid model perceiving both the temporal changes and local patterns, such a structure has shown an elevated performance for time series classification tasks [34].

For finger tracking tasks, we aim to predict the transient position utilizing a short time period, where the finger's position is approximated as static. Although traditional algorithms can determine the positions of a detected object, we still utilized a machine learning approach for two reasons. First, due to the difference in size, the majority of reflection during input will come from the palm, instead of the finger(s) on the skin, making the contact position cannot be directly calculated. Second, DFT-based algorithms cannot distinguish a spacing smaller than the sensor's range resolution, yet there can still be clues in the energy distribution across range bins and antennas. Therefore, with temporal information eliminated in this process, we abandon the RNN layers in the model and instead apply a residual CNN to form a deeper but still lightweight network to detect minor clues in energy distribution and phase variation.

6 EXPERIMENT SETUP

6.1 Participants

To evaluate the performance of our proposed system on various subjects, we conducted a user study approved by our Institutional Review Board (IRB). We recruited 20 participants, including 14 males and 6 females, with ages 22 to 35 (mean=25.4, SD=2.9). The participants vary in height (157 to 184 cm, mean=172 cm, SD=7.5 cm), weight (46 to 91 kg, mean=68.6 kg, SD=12.4 kg), and wrist circumference (14.3 to 17.3 cm, mean=15.7 cm, SD=1.19 cm), ensuring a diversity in body dimensions as well as the properties of skin and subcutaneous tissue. For their past experience with smartwatches and smart wristbands, 9 of the participants report that they wear them very often, 5 report that they used to wear them but not often, and 6 report that they have rarely or never worn them. Each participant spent approximately 1.5 hours on the study and received a 20 USD gift card as compensation.

6.2 Data Collection Procedures

During our data collection process, the participants were seated in front of a table. With our proposed gestures simple enough to be performed easily with either hand, all participants were instructed to use their left forearm as the input surface and perform input gestures with their right hand. To do this comfortably, they were asked to rest their left forearm in a natural position on the table, with the angle between the upper arm and the forearm being approximately 90 degrees.

For each study, the participants were asked to wear our customized wristband with mmWave sensors comfortably on their wrist, with the mmWave transceivers facing the forearm. Our data collection board is connected to a laptop with an Intel Core i7-13620H CPU via USB for power, signal emission control, and real-time raw data transmission. To provide a simple reference of the locations, we used numbered stickers to label an 11×5 grid on the back of the forearm, with the grid size being 15×10 mm. The researcher gave verbal instructions across all data collection sessions, and participants were given at least 1 practice session before the collection started.

The collection process can be divided into two parts: (1) single-finger and double-finger synchronous gestures, and (2) double-finger asynchronous gestures. For the first part, participants were asked to complete four different pressing activities: single-finger gentle pressing, single-finger hard pressing, double-finger gentle pressing, and double-finger hard pressing. Data collection of this part was made up of eight sessions, each of which consisted of 55 six-second cycles of press-hold-remove-rest. Each touching method was performed 2 times on all 55 locations, with all the methods and locations randomized throughout the eight sessions. For the second part, participants were asked to perform four gestures, including rotating clockwise/counterclockwise and zooming in/out with the index and middle fingers. The gestures were repeated at a 3-second interval, with participants free to choose the position within the grid space. A session in this part contained 15 repetitions of each of the four asynchronous gestures, organized in a random order. With four sessions performed, each participant completed 60 repetitions of the four asynchronous gestures. After all 12 data collection sessions, the participants completed a brief survey on their experience with our proposed gestures. Following such a procedure, a total of 56 minutes of data were collected from each participant, resulting in 18.7 hours of data from all 20 participants. In the following data review process, 1.7 minutes of data, where the participants failed to conduct the correct activities, were excluded from our dataset.

6.3 Training Settings

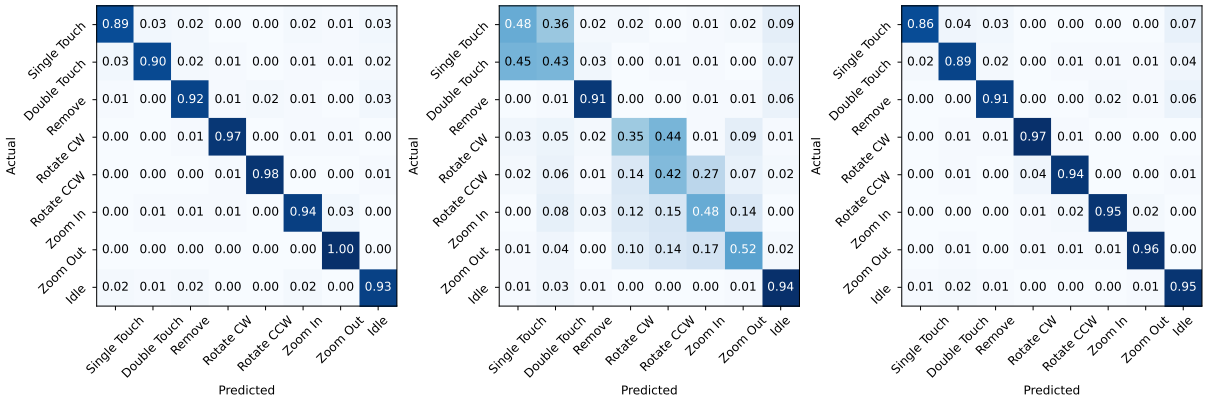
To comprehensively evaluate the performance of our system and its transferability to unseen users, we adopt three different training schemes. (1) Random Splitting. The data collected for all 20 participants is randomly split on an 80-20 ratio for training and testing. (2) Leave-one-out (LOO). One of the participants is excluded from the training dataset, with the trained model evaluated on the unseen user. The process is repeated for all participants. (3) User-specific calibration. Based on the LOO models, a fine-tune process is performed using a small dataset

from the unseen users. For all classification models, cross-entropy loss (CE) is used for model training, and for the 2-D tracking model, mean squared error loss (MSE) is adopted. All trainings are completed on an NVIDIA A6000 GPU with 48 GB of memory. We report the evaluation results of our three complementary tasks identified in Sec. 5.3.

7 EVALUATION

7.1 Force-decoupled Gesture Recognition

The force-decoupled gestures in our system include single-finger gestures, double-finger synchronous gestures, and double-finger asynchronous gestures. Figure 17a and 17b present the system's normalized prediction results on a randomly split dataset and under leave-one-out user evaluation, respectively. While the model achieves an average accuracy of 94.1% on the entire dataset, its performance drops drastically to 56.7% when evaluated on unseen users. As indicated by the confusion matrices, this reduction in accuracy arises primarily from misclassifications related to the number of fingers and confusions within asynchronous gestures. These confusions



(a) Random split. (b) Leave-one-out validation. (c) Calibrated.

Fig. 17. Confusion matrices for force-decoupled gesture recognition.

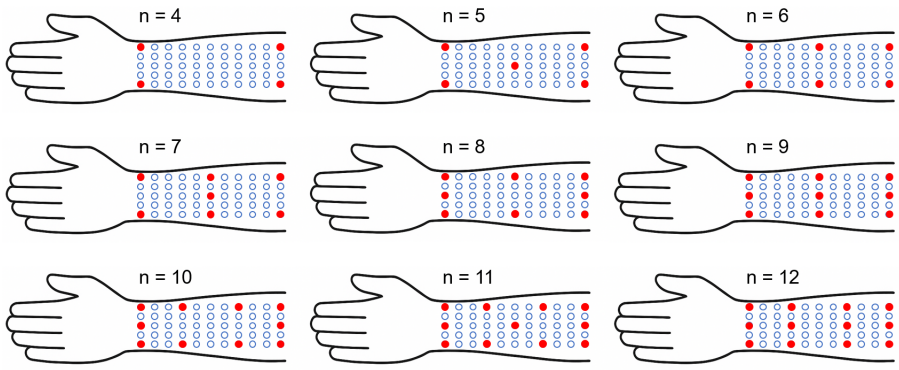
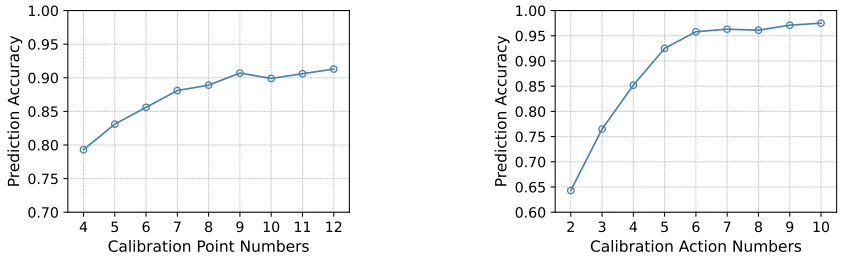


Fig. 18. Calibration points distribution (Selected points highlighted in red).



(a) Single-finger and synchronous gestures.

(b) Asynchronous gestures.

Fig. 19. Change of calibration performance with user-specific data amount.

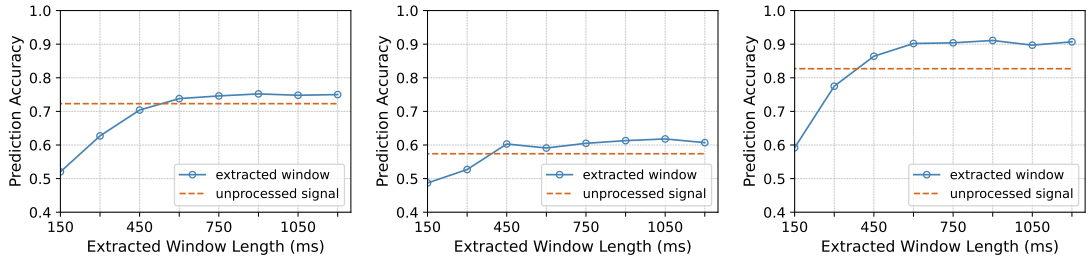
come from several sources of inter-user variability. In finger number recognition, differences in finger thickness, separation distance, and angular relation during double-finger gestures introduce substantial mismatches between the pre-trained model and new users. For asynchronous gestures, variations in the way of the fingers' movements can markedly alter the IF signal patterns. For example, different users perform rotation inputs with distinct rotation axes. In such cases, the distance of the detected object may even change reversely depending on individual user habits, further complicating cross-user generalization. Such a performance degradation verifies our assumption that a user-specific calibration is required.

To address the inter-user variability observed, we develop a calibration process that adapts the model to individual user habits. To ensure a balanced dataset for single and synchronous gesture calibration, we select an equal number of pressing samples from our four categories: single-finger gentle, single-finger heavy, double-finger gentle, and double-finger heavy. For ease of implementation, the calibration process follows a predefined spatial distribution for the selected touch points, as shown in Fig. 18. This design serves two key purposes. First, the points are positioned along the center lines and edges of the effective sensing range, as well as at their midpoints and quarter points, reducing the difficulty of accurate positioning for users in real applications. Second, this configuration provides comprehensive spatial coverage across the input area, ensuring that the calibration data include sufficient references for edge conditions and boundary behaviors. For the calibration of asynchronous gestures, we select the same number of samples from each gesture.

We fine-tune the pre-trained leave-one-out models on the selected datasets of the unseen users. The change in average prediction accuracy versus the number of selected points and gestures is illustrated in Fig. 19. The figure shows that the accuracy of both models rises significantly after the fine-tuning process and reaches an accuracy near the random-splitting prediction. We observe that the accuracy of the two models shows only a minor increase after 9 points and 6 samples per gesture, respectively. This corresponds to 3 minutes 54 seconds of time in our data collection procedures. The combined gesture recognition result is presented in Fig. 17c, reaching a 92.8 % overall accuracy across all users and gestures. Such a calibration approach reduces the performance degradation for new users to 1.3%.

7.2 Force Level Detection

We evaluate our system's ability to identify pressing activities performed at different force levels, as well as the effectiveness of our proposed feature extraction algorithm. Fig. 20 shows the force detection accuracy using the uncropped window and the cropped window of different lengths with our proposed processing algorithm. Figures 20a and 20b correspond to the performance in random splitting and the average in leave-one-out validation. We can see that the performances in random splitting and leave-one-out conditions are hardly satisfying, with the



(a) Random split. (b) Leave-one-out validation. (c) Calibrated.

Fig. 20. Change of force recognition accuracy with feature extraction window length.

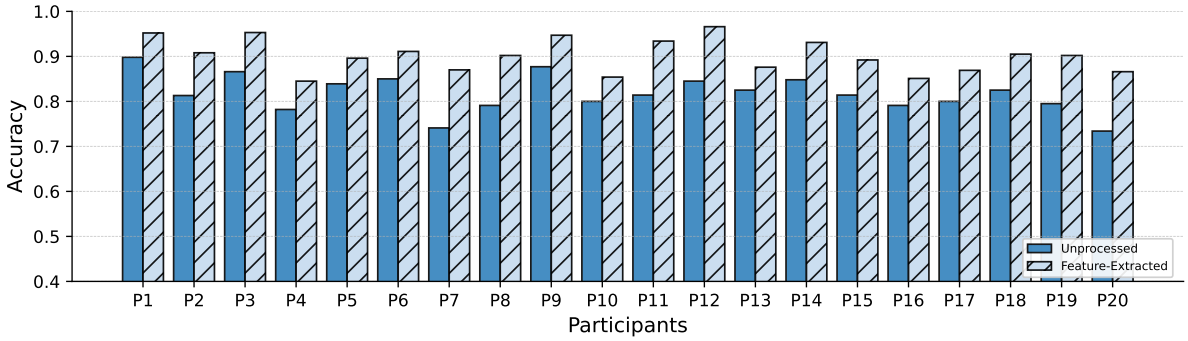


Fig. 21. Increase in force recognition accuracy across users.

accuracy with uncropped samples being 72.3% and 57.4%, and the maximum accuracy with processed time frames being 75.2% and 61.8%, for the two conditions, respectively. Such a low recognition performance can be caused by two reasons. First, with heavy and gentle being subjective criteria, different users may adopt different thresholds to distinguish the force levels in a touch, resulting in contacts with a similar force being labeled differently for different users. In addition, different users possess different physical properties for their forearm skin, reflected as a diversity in the oscillation parameters in Eq. 13. These factors prevent the accurate detection of force levels without user-specific information.

Therefore, we investigate whether a calibration process can bring enough information for the model to work for a specific user. Using the selected samples for single and synchronous gesture recognition, the accuracy with uncropped samples rises to 82.7%, and the maximum accuracy with cropped samples reaches 91.1%. With the accuracy change against time window length shown in Fig. 20c, we adopt a window size of 600 ms to ensure a high input quality and maintain a low recognition delay. Under such a setup, the accuracy of force recognition becomes 90.2%, bringing a relative increase of 9.1% from the uncropped samples. Fig. 21 shows the accuracy of force detection accuracy with cropped and uncropped samples for each participant, and proves that our proposed feature extraction method has robust effectiveness across users.

With the body's mechanical response of force being relevant to the physical properties of body tissues, we study the relevance of force detection accuracy to the users' body shape. Body mass index (BMI), calculated as $\text{BMI} = weight(kg)/height(m)^2$, provides an estimation of the amount of body tissue and partly reflects important body conditions, including skin stiffness and muscle tensions. We plot the accuracy of each user with their

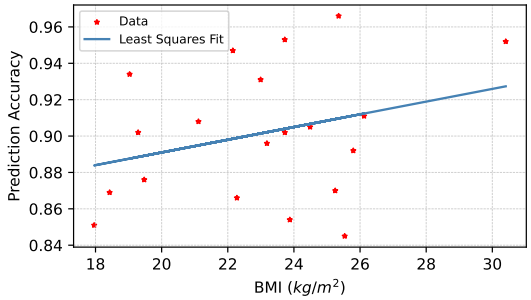
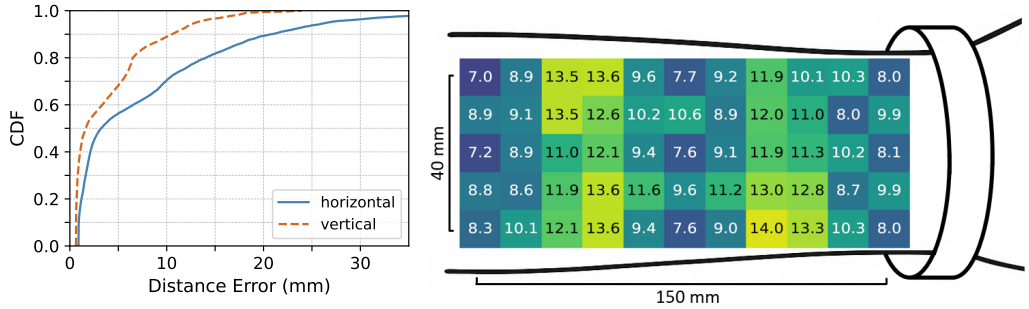


Fig. 22. Force recognition accuracy shows no correlation with BMI.



(a) Cumulative distribution function.

(b) MDE at different positions on the arm.

Fig. 23. Finger tracking performance with calibration.

BMI in Fig. 22. In a correlation test, the two variables yield a Pearson coefficient $r = 0.298$, and a p-value $p = 0.201 > 0.05$. The result shows that our method adapts successfully to participants with different body shapes, with the prediction accuracy having no significant variance towards BMI.

7.3 Finger Tracking

We assess the model's performance on two-dimensional finger tracking during single-finger and synchronous gestures. To quantify the error, we calculate the mean distance error (MDE) of the predictions, which is calculated as

$$MDE = \frac{1}{N}\sum_{i=1}^{N} \| \mathbf{r}_i - \hat{\mathbf{r}}_i \|_2 \ . \tag{15}$$

On the complete 20-user dataset, the system reaches an MDE of 8.8 mm and SD=7.6 mm. On leave-one-out validation of unseen users, the error becomes 16.2 mm and SD=16.5 mm. This degradation is mainly caused by the different landscapes of the forearms brought about by the different diameters of the limbs. With the calibration process, the tracking performance is recovered to 10.2 mm with SD=8.7 mm. The cumulative distribution functions (CDF) on both directions are illustrated in Fig. 23a, showing that the majority of predictions fall in a low-error range.

To explore how the tracking precision changes with the input location, we calculate the average MDE across users on all 55 grid points in our 11×5 grid, and illustrate them as a heatmap in Fig. 23b. Among all the grid

points, the method reaches a minimum error of 7.0 mm and a maximum error of 14.0 mm. Generally, positions close to the calibration points achieve an overall high precision, whereas the regions far from them demonstrate a slightly worse performance.

7.4 Live Interaction Study

Given EpiPad's satisfactory performance in offline evaluations, we implemented a real-time interactive prototype to further examine the system's behavior under practical input conditions. Real-time signal processing and gesture recognition were performed on a laptop with an Intel i7-13620H CPU. To represent the scenario of embedded system deployment, we used four CPU cores for model inference (no CUDA acceleration), with 2 GB RAM allocated. To approximate the visual constraints of commercial smartwatches, we used a 2-inch LCD display with a resolution of 240 × 320 pixels (pixel pitch: 0.128 mm). The experiment setup is shown in Fig. 24.

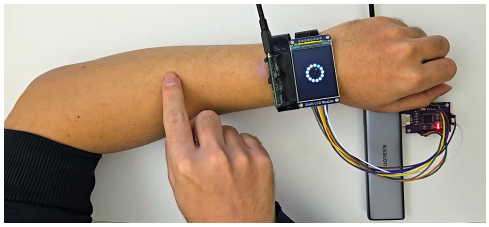
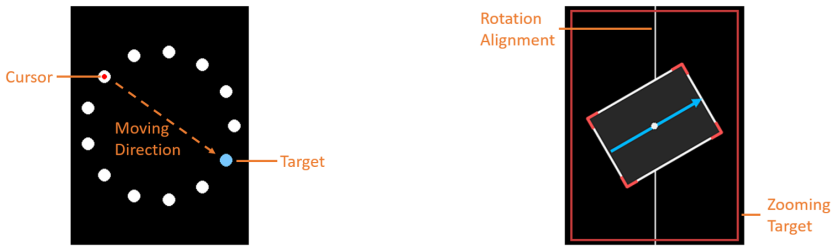


Fig. 24. Real-time live interaction study setup.



(a) Reciprocal tapping ($W = 20, D = 200$). (b) Rotate and zoom (scale = 200%, angle = 60°).

Fig. 25. User interfaces for real-time input tasks.

7.4.1 Task Design. The study comprises two tasks that together exercise all dimensions of EpiPad's input recognition pipeline. In the *first task*, the display presents 13 positions marked with white circles evenly distributed along a circular layout, as shown in Fig. 25a. Participants use *single-finger gentle touch* to move a red cursor from a starting position to a target highlighted in light blue. The target is always positioned diametrically opposite the starting position, with its distance D equal to the diameter of the circular arrangement. Upon reaching the target, participants perform a *double-finger heavy touch* to select it, after which the selected position becomes the new start position for the cursor. Target width W and moving distance D are manipulated as independent variables. This reciprocal tapping task [64] ensures an extensive coverage of task difficulty, with smaller target size and longer moving distance leading to larger difficulty, and vice versa. This setup offers two key advantages. First, it closely mirrors a common pattern in everyday input, i.e., repeated move-and-select actions, thereby ensuring

the task reflects realistic interaction demands. Second, by systematically varying W and D, it directly supports rigorous, quantitative efficiency analysis using Fitts' Law [48], enabling clear comparisons across conditions and input techniques. In the *second task*, we covered multi-finger asynchronous gestures. Participants used *double-finger asynchronous inputs* to rotate and zoom the display to match the prompted targets, as shown in Fig. 25b. The participants were allowed a ±10% error for zooming and ±5° error for rotation.

7.4.2 Comparison Methods. To better understand the efficiency of EpiPad relative to existing approaches, two additional input methods were included for comparison in the first task. The first is a *tilt cursor*, which has been widely proposed as a competitive option for precise target selection on smartwatches [25, 37, 66]. The system was built using a WitMotion WT901 sensor[2], where participants tilted their wrist to steer the cursor, with cursor speed ranging from 0 to 320 pixels per second—the same range as EpiPad. The second method simulated a *commercial smartwatch touchscreen*, where participants performed conventional finger dragging to move the cursor and tapping to select the target directly on the display surface. Both comparison methods used the same W and D combinations as EpiPad.

7.4.3 Participants and Procedure. We recruited 12 participants for this study. Each session followed a fixed order of three phases for each input method. First, participants completed the *calibration phase*: they went through EpiPad's proposed calibration process (Section 7.1) before using EpiPad, while no calibration was required for the tilt cursor and touchscreen methods. Second, participants entered an *open-ended practice phase* in which they could spend unlimited time familiarizing themselves with the current input method and the task interfaces before any formal data collection began. Third, participants completed the *formal task phase*: each participant completed two repetitions of all task configurations for each input method. To cover a wide range of task difficulties, we tested nine configurations with $W = 10/20/40$ pixels and $D = 50/100/200$ pixels in the first task. For the second task (EpiPad only), we tested eight conditions combining zoom scales of 50% and 200% with rotation angles of ±30° and ±60°. The movement time (MT) was automatically recorded for the first task, with the first target in each configuration dropped for behavioral consistency.

7.4.4 Results. **How well does EpiPad recognize gestures?** Considering the inevitable noise and outliers in real-time systems, an input was considered valid only when at least two of the most recent three predictions agreed on the same outcome, which improved system stability. The recognition of force was enhanced similarly by applying a ±20 ms shift on the extracted window with 450 ms length. Under this criterion, the system achieved a recall of 91.6% ($TP/(TP + FN)$) for input recognition in the reciprocal selection task, and a recall of 95.2% for zooming and rotation gestures. Across all sessions, the observed false activation rate was 1.8%, showing a high reliability of the recognition pipeline.

How much latency do users experience? To quantify system latency from user action to recognition, we recorded interaction sessions using a webcam operating at 120 fps connected to the processing laptop. The moment when the fingertip contacts the forearm was marked as the start of user input. For force-decoupled gesture recognition, the system exhibited an average end-to-end delay of 90 ms, which falls within the commonly accepted input latency range of 50–200 ms for commercial interactive devices [53]. Force recognition incurred a longer delay of 470 ms due to the temporal window required by the feature extraction process. However, this delay has a limited impact on user experience, as users naturally dwell longer during heavy-touch interactions, with 84% of all heavy touches lasting longer than this delay in our offline data collection. Overall, these results indicate that the proposed system achieves responsive and reliable performance suitable for real-time deployment.

How efficient is EpiPad compared to other input methods? To better understand the efficiency and overhead of our proposed system, we adopt Fitts' Law as a quantitative metric in our 2D reciprocal tapping task. For the computation of the Fitts' Law, we adopted the effective target width characterized by the distribution

[2]https://witmotion-sensor.com/products/accelerometer-module-tilt-sensor-mpu9250-imu-for-arduino

of user selection positions to reflect the actual utilization of target sizes, which is affected by user habits [47]. The effective target width is calculated as $W_e = 4.133 \times SD_x$, where SD_x is the standard deviation of the selection coordinates along the axis of approach to the target. We further computed the index of difficulty $ID_e = \log_2(D/W_e + 1)$, throughput $TP = ID_e/MT$, and the fitting parameters for the Fitts' law $MT = a + b \cdot ID_e$. We focus on six indicators that are commonly used: Movement time MT measures how long users take to move the cursor from one target to another, directly reflecting the overall speed of the input system. Error rate ER measures how often users miss the target, indicating how accurately and controllably the input system supports precise pointing. Throughput TP combines speed and accuracy into a single measure, representing the overall efficiency of the input system in performing pointing tasks. Intercept a captures the fixed overhead independent of task difficulty. Slope b indicates how strongly movement time increases as tasks become more difficult, with a smaller b implying better scalability and control. Lastly, the coefficient of determination R^2 indicates how well user performance follows Fitts' Law, with higher values suggesting stable, predictable, and well-behaved input behavior. Among all these indicators, previous research shows that a can be unstable for real-world experiments, while b and TP are relatively robust and serve as the primary indicators in Fitts' Law studies [24, 82]. Results from our experiment are summarized in Table 1.

Table 1. Fitts' Law calculation and fitting results.

Input Method	MT (s) ↓	ER (%) ↓	TP (bit/s) ↑	a (ms/bit) ↓	b (ms/bit) ↓	R^2 ↑
Touchscreen	3.26 ± 2.13	21.7	1.23 ± 0.59	−34 ± 1012	1037 ± 370	0.37 ± 0.21
Tilt Cursor	3.27 ± 0.61	7.4	1.06 ± 0.27	147 ± 301	960 ± 148	0.87 ± 0.06
EpiPad	2.82 ± 0.81	6.5	1.29 ± 0.26	382 ± 165	708 ± 116	0.81 ± 0.09

From the indicators, we observe that the conventional touchscreen interface yields performance characteristics that deviate markedly from typical Fitts' Law behavior, including abnormally large variance on most indicators, a substantially high error rate of 21.7%, and a low goodness of fit (R^2=0.37). This outcome arises primarily because, when targets become extremely small, the fingertip fully occludes the target area, resulting in prolonged fine-grained adjustments during the final selection phase. Such behavior has been shown to violate the assumptions of Fitts' Law in extra-small target acquisition on touchscreens [3]. In our study, target sizes range from 1.28 mm to 5.12 mm. Although this range is smaller than that typically used in standard Fitts' Law evaluations, it is both necessary and representative for smartwatch interaction scenarios. For example, a typical on-screen 26-key keyboard on commercial smartwatches allocates a key width of approximately 3 mm. These results suggest that while traditional touchscreen input remains intuitive, it is inherently limited for precise target acquisition and fails to provide an effective interface for smartwatches.

With input operations decoupled from the screen, tilt cursor and EpiPad fit well (R^2>0.8) to the Fitts' Law despite the small size of the targets. However, the tilt cursor requires the user to perform precise body control, which brings a larger overhead than sliding and tapping. With other indicators at a similar level, EpiPad has a more gentle slope (b=708<960), leading to a higher overall throughput (TP=1.29>1.06), which serves as the most comprehensive indicator of the input efficiency. Paired t-tests confirm the statistical difference on slope ($p < 0.05$) and throughput ($p < 0.05$), thereby suggesting that EpiPad demonstrates a higher overall efficiency and better scalability with task difficulty. Generally, such results from Fitts' Law evaluation show that EpiPad provides a more efficient input method on smartwatches compared to conventional touchscreens and tilt cursors.

7.5 Robustness Analysis

7.5.1 Calibration Consistency Across Days and Device Remounting. Remounting of devices is a common scenario in real-world, long-term use and commonly limits the performance of wearable sensing techniques [28, 52]. To

evaluate the robustness of our proposed system and calibration procedure under such conditions, we conducted a follow-up study focusing on the performance impact of remounting. 10 participants from the original user study were asked to repeat the same set of gestures on two additional days. To limit the extra data collection time, the additional sessions contain performing the single-finger and synchronous gestures once on a 6×3 grid, which covers the same area as the complete grid but doubles the interval, and performing each asynchronous gesture 10 times. To minimize learning effects and unintentional optimization of gesture execution, the additional data collection sessions spanned three months from our initial experiment, and the order of gestures was randomized within each added session. The calibrated models from the initial data collection were used for the evaluation of the new dataset.

On the dataset collected on different days with device remounting, the calibrated models reach an average performance of 91.1% accuracy for force-decoupled gesture recognition, 88.9% accuracy for force level detection, and an MDE=11.3 mm for finger tracking. Despite the overall performance showing a minor degradation due to the slight positional shift of the sensor and the variations in user habits, performance on all three dimensions maintains a comparable level with the previously reported same-day calibration performance. The results show that our one-time-per-user calibration protocol maintains a robust performance over time, and is sufficient for long-term practical deployment. With such a calibration process integrated into the device setup process, the proposed system can provide a reliable interface for precise input and multi-finger control, and further facilitate applications like texting and map reading.

7.5.2 Finger Combinations for Multi-finger Input. Among double-finger input gestures, two finger combinations are most commonly used in practice: thumb–index and index–middle. While our primary experiments focus on the index–middle combination, we would like to examine whether the proposed system can generalize to the thumb–index configuration. To this end, we conducted a supplementary study with 10 participants drawn from the original user study. In this study, participants performed the same set of double-finger gestures using the thumb–index combination, following the same experimental protocol as in the main study but with half the number of repetitions.

We first evaluated the system using models calibrated on the previous index–middle finger combination. Under this setting, performance degraded noticeably, achieving a gesture recognition accuracy of 70.3%, a force recognition accuracy of 58.6%, and a position-tracking MDE of 16.0 mm. Such a degradation highlights the differences in radar backscatter patterns for the two combinations. To address the problem, we subsequently redo the calibration process by fine-tuning the pretrained models with data collected with the new finger combination. Such an approach resulted in improved averages of 89.4%, 83.8%, and 12.6 mm for the three metrics, respectively. Furthermore, we tried using the thumb–index dataset for both the pretraining and calibration process, which further improved performance to 93.2%, 87.9%, and 11.5 mm. These results indicate that although the models trained on the index-middle combination cannot directly transfer to other finger pairs, the proposed system can achieve satisfactory performance with our proposed pretrain-calibration pipeline using combination-specific data. This demonstrates the flexibility and strong generalizability of our approach to different user habits and interaction styles.

7.5.3 Performance Under Motion. Our system is primarily designed for precise controls that need an augmented interface in addition to the touchscreen. Although these inputs are typically performed in a near-static condition, we still conduct a study to evaluate the system's performance under motion. We asked five participants to complete the simplified data procedure described in Section 7.5.1. During the performance of the gestures, the participants walked slowly with their forearms held horizontally in front of the body. With the calibrated model using previous data, the system reaches an accuracy of 90.6% on force-decoupled gesture recognition, 88.4% on force level detection, and 13.1 mm MDE for finger tracking. We notice that while the finger tracking performance degrades slightly due to the unstable positioning of sensors and inaccurate execution of touching gestures, other

recognition dimensions achieve similar performance under the static condition. Overall, the system shows a reasonably good resistance to the interference in motion.

7.5.4 Impact of Clothes. To investigate how clothing affects the performance of the sensor system, we conducted a supplementary study with two participants. Each participant performed the same additional sessions as in the remounting experiment, first wearing a skin-tight garment and then a long-sleeve jacket, as illustrated in Fig. 26. Both garments had an approximate thickness of 1 mm. With the skin-tight garment, the calibrated models achieve reasonably good accuracies of 88.4% and 89.9% for gesture recognition and force detection, respectively, and a finger-tracking precision of MDE = 11.2 mm with SD = 9.8 mm. However, when the participants wore the jacket, the accuracies for the two tasks dropped to 72.3% and 63.2%, and the tracking performance degraded to MDE = 18.5 mm with SD = 17.7 mm.

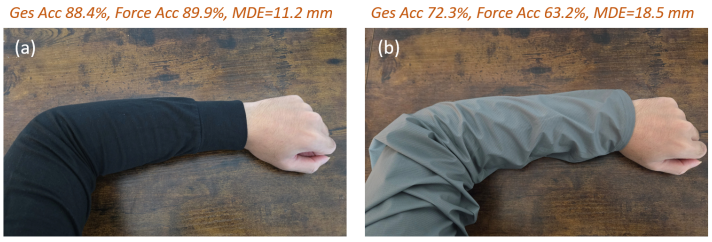


Fig. 26. System performance on two types of clothes (a) skin-tight garment (b) loose jacket.

This contrast highlights the significant impact of clothing tightness on system performance. The skin-tight garment largely follows the natural motion of the skin, introducing only minor interference in the mmWave IF signals. In contrast, the loose-fitting jacket hovers above the skin and produces intermediate reflections from its fabric surface, which moves independently through twisting and folding, thereby distorting the received mmWave signals.

7.6 Usability Study

To better understand the user experience with our proposed gesture set, especially the influence of the additional finger and the extra force, we conducted a post-study usability survey. We divided the gestures into four categories according to the fingers involved and the force applied. The participants were asked to rate each of the four categories on their prior familiarity ("I was already familiar with gestures in this category prior to this experiment"), physical comfort ("I feel physically comfortable when performing gestures in this category"), difficulty and efficiency ("It is easy and fast to perform gestures in this category"), and social acceptance ("I will *not* feel uncomfortable or embarrassed to perform gestures in this category in public").

The response from the participants, on a 7-point Likert scale, is shown in Fig. 27. The result shows that single-finger gentle and double-finger gentle inputs get a high rating on all four aspects, with single-finger gentle gestures receiving the best response as the most basic form of input nowadays. Single-finger heavy gestures fall slightly short on the physical comfort of performing compared with the previous two categories, but the rating is still high and demonstrates high usability. Double-finger heavy gestures are rated as high on their social acceptance, but provide a slightly degraded experience of prior familiarity, physical comfort, and the difficulty and efficiency of performing. This is intuitive because, although force and multi-finger gestures have been widely adopted in modern touchpads, introducing the two variations at the same time still brings more burden for the users. Nevertheless, with the participants still giving overall positive ratings on the three aspects, such a category can still be the candidate for less common commands, like the calling of a deep menu. In conclusion,

these findings from the user survey validate our assumption that with multi-finger and force-sensitive inputs, an on-skin touch-based interface can provide an input experience that enables abundant commands and meanwhile maintaining a low learning cost.

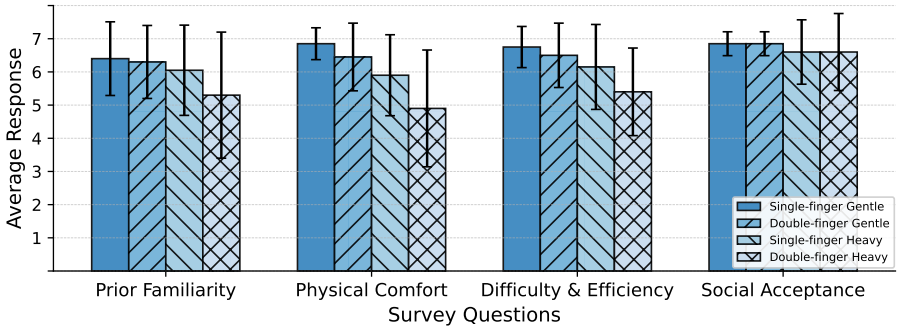


Fig. 27. Usability study on the proposed gestures.

Apart from the users' feedback on the input gestures, we would like to further investigate the overall user experience on the deployed input system. Therefore, we conducted a survey based on the standard System Usability Scale (SUS) [5] following the real-time user study. The SUS survey consists of ten statements, with users providing ratings on a 5-point Likert scale, ranging from strongly disagree (1) to strongly agree (5). The statements include SUS01: I think that I would like to use this system frequently. SUS02: I found the system unnecessarily complex. SUS03: I thought the system was easy to use. SUS04: I think that I would need the support of a technical person to be able to use this system. SUS05: I found the various functions in this system were well integrated. SUS06: I thought there was too much inconsistency in this system. SUS07: I would imagine that most people would learn to use this system very quickly. SUS08: I found the system very cumbersome to use. SUS09: I felt very confident using the system. SUS10: I needed to learn a lot of things before I could get going with this system. Responses from the 12 involved participants are illustrated in Fig. 28.

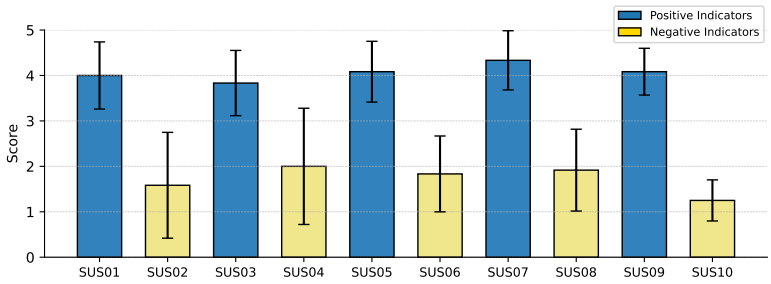


Fig. 28. System usability scale (SUS) score from the real-time evaluation.

We calculate the SUS score with the formula

$$SUS = 2.5(20 + \Sigma(\text{SUS01}, \text{SUS03}, \text{SUS05}, \text{SUS07}, \text{SUS09}) - \Sigma(\text{SUS02}, \text{SUS04}, \text{SUS06}, \text{SUS08}, \text{SUS10})).$$

Our real-time input system gets an SUS score of 79. With the industry average for SUS being 68 [61], our system shows a usability higher than most industrial approaches, and falls in a high 85% to 89% percentile range in the SUS benchmarks [40]. Altogether, our usability study results show that EpiPad is built on a gesture set well aligned with the user preferences, and provides an input system with high effectiveness and user satisfaction.

7.7 Power Consumption

To evaluate the overhead of our sensing system, we measured the power consumption with a USB tester in 1-minute sessions. With the board's MCU powered on and transceivers not working, the system consumes an average power of 578 mW. And with transceivers working in our proposed condition, the power rises to 672 mW. This indicates a 94 mW power consumed by the working transceivers and data transmissions. With major components on smartwatches like the screen consuming 200 to 300 mW, Such a power consumption is appropriate as an embedded sensing technique.

8 RELATED WORK

8.1 Touch Force Estimation

Traditional approaches for fingertip force measurement typically involve engineering on the contact surface. For wearable computing applications, common methods include using customized gloves with embedded force-sensitive textiles [46, 92], or attaching force sensors to the surface where the contact may occur [68, 70, 85]. However, these additional contact-based sensors significantly interfere with the input experience and are impractical to deploy for a pervasive on-skin input scenario. Therefore, researchers explore the possibility of indirect measurement of fingertip forces with other sensors. Deformations of the fingertip directly reflect the force applied, yet existing work utilizing mmWave is sensitive to noise brought by surface deformations, and becomes limited for on-skin activities [20]. Electromyogram (EMG), as a modality directly related to muscle activities, has been widely explored for the prediction of forces. With multi-channel EMG previously proven to correlate well with fingertip forces [14, 49, 50, 91], recent research shows that single-channel EMG collected from the wrist can also indicate grip forces when augmented with IMU [75]. In addition, blood volume changes monitored by PPG sensors [6, 17, 18] and muscle action noises [58] have also been used for the prediction of fingertip forces. While these sensors can be easily wearable, they perceive the force applied using the wearing hand, and cannot be used when the adjacent skin is used only as an input surface.

Instead of studying the body performing the touch, a limited number of studies focus on the contact surface for the estimation of force. The deformation brought by pressing has a response on the internal IMU [22] and Barometer [57, 67], and can be further utilized for the prediction of touch forces. However, these approaches require the touch to be performed on the rigid surface of a device, and cannot be transferred to on-skin pressing activities. Recently, HaloTouch [72] proposes a remote sensing technique with cameras. However, it relies on the detection of the shift in fingertip-surface distance, which will be unstable on human soft tissues. EgoTouch [51] provides a solution for on-skin touch force detection, yet it requires a camera from a top-down view, which cannot be satisfied solely with a smartwatch. How on-skin touch force can be remotely estimated remains an open question.

8.2 mmWave for Human Sensing

Millimeter-wave (mmWave) radar has emerged as a privacy-preserving and contactless sensing modality that is robust to environmental factors such as lighting conditions. These properties make it a promising approach for human-related sensing tasks. Prior research has predominantly focused on scenarios involving relatively long sensing ranges, where human activities induce distinctive patterns in range profiles and range–Doppler representations. Such patterns have enabled effective recognition of hand poses [36, 42, 44], facial expressions and landmarks [31, 33, 62, 76], mid-air and body gestures [26, 29, 45, 79], and even whole-body reconstruction [63, 78]. In these contexts, reflections are distributed across multiple range bins, allowing rich spatial–temporal features to be extracted using advanced signal processing and machine learning techniques.

However, when the scope of activity goes smaller, most informative reflections become concentrated within a small number of range bins. Consequently, this reduces the effectiveness of feature extraction methods that rely

on broad spatial distributions. In such cases, phase signals within individual range bins offer superior sensitivity to subtle surface deformations, enabling the detection of fine-grained activities such as eye motions [88] and vital signals [71, 81, 83]. These works demonstrate the capability of mmWave radar to capture subtle on-body motions, but they typically target passive monitoring scenarios and are susceptible to interference from unrelated body movements encountered in daily activities.

To mitigate such interference, prior approaches often exploit the periodic nature of vital signals to separate them from background motion [13, 23]. While effective for physiological monitoring, this strategy becomes less applicable when large-scale movements coexist with non-periodic subtle deformations. As a result, although mmWave radar is inherently sensitive to both large-scale and fine motions, jointly extracting and interpreting these signals from the same target remains challenging. This limitation has constrained the use of mmWave sensing for fine-grained, force-sensitive, and multi-contact input on the body, and motivated us to explore mmWave sensing systems that can robustly capture complex on-skin interactions.

8.3 Augmented Input for Wrist-worn Devices

The limitation of screen size on smart wrist-worn devices has motivated researchers to develop augmented input methods. Early research explored mechanical modifications to power more operations on the watch dial [74]. However, the extra structure is hard to accommodate in the already compact hardware setup. Additionally, an ultrasonic-based approach brings the ability to tell which finger is being used to press on the screen [35], and customization of the watch bezel allows multiple fingers to be recognized on the bezel [60]. Yet these trials do not get rid of the limitation of the screen size, and consequently make the input experience somewhat cumbersome. Instead, more studies have focused on sensing techniques to remove the need for on-device interactions. Among them, a number of researchers have designed input protocols utilizing hand gestures of the wearing hand. Such recognitions have been proven feasible with photoplethysmogram (PPG) sensors [66, 89], inertial measurement unit (IMU) [8, 69, 93], acoustic signals [32, 77, 86], force sensors [15], and electromagnetic induction [56]. While the flexibility of hands provides a large gesture set for interactions, research has also shown that not all self-defined gestures are friendly to users in terms of memorability, comfort, and social acceptance [4, 21].

To address this issue, researchers look into contact-based interactions performed with the other hand, which provide a higher prior familiarity and have human skin as an accessible input surface. In such a scenario, challenges lie in the diversity of input gestures. Early research used optical solutions to detect on-skin touch activities directly adjacent to the watch [38, 54], yet the number of input commands remains constrained. To support a larger interface, a popular approach is to leverage the body landmarks, including finger knuckles and the complex anatomy of the hand [7, 9, 11, 84, 87], which generate differentiated signals when tapping at different positions. Nevertheless, with only tapping activities supported at certain positions on the hand, the freedom of operation is still limited, with complex input commands needing to be encoded. RadarHand combines the natural distinction of hand knuckles and the flexibility of self-defined gestures [27], but in return still possesses a learning cost that cannot be ignored.

A higher level of freedom for on-body input involves finger tracking. LumiWatch utilized visible light projection and vision-based depth sensing [73], achieving the ability of sliding direction recognition. Alternatively, SkinTrack used the electrical signal generated by a high-frequency power source worn on the finger [90], which has the ability for continuous position tracking yet requires additional devices that cannot be embedded into smartwatches. Therefore, an accurate approach for on-skin finger tracking with only wrist-worn sensors is still in need.

Our research gains inspiration from the above-mentioned works, and strives to provide a user-friendly interface with diverse gestures. We compare our work with existing on-skin input methods for smartwatches in Table 2. With the ability of multi-finger and force-sensitive sensing, our work allows users to seamlessly transfer their input habits on trackpads to small-screen wearables like smartwatches.

Table 2. On-skin input methods for smartwatches.

Related Works	Modality	Privacy	Familiarity	Continuous Tracking	Multi-Finger	Force Touch
WatchSense [65]	Vision	×	✓	✓	✓	×
Skinput [28], PUB [43]	Acoustic	✓	✓	×	×	×
Taprint [7], ViType [9], IDial [87]	IMU	✓	✓	×	×	×
ViWatch [11], ViBand [39]	IMU	✓	×	×	×	×
TapSkin [84]	IMU + Acoustic	✓	✓	×	×	×
SkinTrack [90]	Electrical Wave	✓	✓	✓	×	×
LumiWatch [73], SenSkin [55]	Infrared	×	✓	✓	×	×
RadarHand [27]	mmWave	✓	×	×	×	×
EpiPad	**mmWave**	✓	✓	✓	✓	✓

9 CONCLUSION AND FUTURE WORK

In this paper, we present EpiPad, which is the first to enable multi-finger and force-sensitive input for smartwatches through mmWave-enabled on-skin touch interfaces. Through our proposed feature extraction strategy and the use of compact deep learning models, we demonstrated that mmWave signals can effectively recognize single- and multi-finger gestures on the forearm, while also estimating touch force levels. The system exhibits strong cross-user adaptability, requiring only a short one-time calibration to achieve robust performance for unseen users. Results from our user survey further indicate that EpiPad delivers an interaction experience comparable to traditional touchpads, offering intuitive, diverse, and low-effort control. With its compact single-input multiple-output (SIMO) antenna array design, the system can be seamlessly integrated into smartwatch bezels, showing great promise for practical deployment in everyday wearable interfaces.

Looking forward, several directions can further advance EpiPad. In terms of the system design, the current mmWave transceiver placement, although compatible with the dimensions of popular off-the-shelf products, requires a 14 mm elevation from the skin. To accommodate smartwatches with thinner cases, beam-forming techniques on customized hardware with multiple transmitters could increase the sensing SNR, thereby achieving a higher recognition performance at a lower transceiver position. Besides, our experiment has not exhaustively tested the system under all body gestures. Although we do not observe noticeable performance degradation brought by natural body movements thanks to our preprocessing procedures, more validation will help verify its usability under different body conditions. Another promising avenue is to explore the adaptability of mmWave sensing under different textile obstructions. Incorporating advanced noise suppression and feature extraction techniques could possibly enhance robustness against clothing covering the sensing area.

The user experience of EpiPad can also be improved and further assessed in the future. As a one-time process, our calibration procedures may be combined with the new device setup process. Potential design of interaction games can also be explored to deliver the calibration gestures in a recreational way. Moreover, long-term user studies will help assess the system's sustained usability and social acceptance across diverse real-world contexts, paving the way toward practical, on-skin mmWave-based interaction.

References

[1] Infineon Technologies AG. 2023. *XENSIV™ BGT60TR13C 60 GHz radar sensor datasheet.* https://www.infineon.com/assets/row/public/documents/24/49/infineon-bgt60tr13c-datasheet-en.pdf

[2] S Alekseev, A.A. Radzievsky, Mahendra Logani, and M Ziskin. 2008. Millimeter wave dosimetry of human skin. *Bioelectromagnetics* 29 (01 2008), 65–70. doi:10.1002/bem.20363

[3] Xiaojun Bi, Yang Li, and Shumin Zhai. 2013. FFitts law: modeling finger touch with fitts' law. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 1363–1372. doi:10.1145/2470654.2466180

[4] Idil Bostan, Oğuz Turan Buruk, Mert Canat, Mustafa Ozan Tezcan, Celalettin Yurdakul, Tilbe Göksun, and Oğuzhan Özcan. 2017. Hands as a Controller: User Preferences for Hand Specific On-Skin Gestures. In *Proceedings of the 2017 Conference on Designing Interactive Systems* (Edinburgh, United Kingdom) *(DIS '17)*. Association for Computing Machinery, New York, NY, USA, 1123–1134.

[5] John Brooke. 1996. SUS-A quick and dirty usability scale. *Usability evaluation in industry* 189, 194 (1996), 4–7.

[6] Thisum Buddhika, Haimo Zhang, Samantha W. T. Chan, Vipula Dissanayake, Suranga Nanayakkara, and Roger Zimmermann. 2019. fSense: Unlocking the Dimension of Force for Gestural Interactions using Smartwatch PPG Sensor. In *Proceedings of the 10th Augmented Human International Conference 2019* (Reims, France) *(AH2019)*. Association for Computing Machinery, New York, NY, USA, Article 11, 5 pages. doi:10.1145/3311823.3311839

[7] Wenqiang Chen, Lin Chen, Yandao Huang, Xinyu Zhang, Lu Wang, Rukhsana Ruby, and Kaishun Wu. 2019. Taprint: Secure Text Input for Commodity Smart Wristbands. In *The 25th Annual International Conference on Mobile Computing and Networking* (Los Cabos, Mexico) *(MobiCom '19)*. Association for Computing Machinery, New York, NY, USA, Article 17, 16 pages. doi:10.1145/3300061.3300124

[8] Wenqiang Chen, Lin Chen, Meiyi Ma, Farshid Salemi Parizi, Shwetak Patel, and John Stankovic. 2021. ViFin: Harness Passive Vibration to Continuous Micro Finger Writing with a Commodity Smartwatch. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 1, Article 45 (March 2021), 25 pages.

[9] Wenqiang Chen, Maoning Guan, Yandao Huang, Lu Wang, Rukhsana Ruby, Wen Hu, and Kaishun Wu. 2018. ViType: A Cost Efficient On-Body Typing System through Vibration. In *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. 1–9. doi:10.1109/SAHCN.2018.8397098

[10] Wenqiang Chen, Maoning Guan, Yandao Huang, Lu Wang, Rukhsana Ruby, Wen Hu, and Kaishun Wu. 2020. A Low Latency On-Body Typing System through Single Vibration Sensor. *IEEE Transactions on Mobile Computing* 19, 11 (2020), 2520–2532. doi:10.1109/TMC.2019.2928549

[11] Wenqiang Chen, Ziqi Wang, Pengrui Quan, Zhencan Peng, Shupei Lin, Mani Srivastava, Wojciech Matusik, and John Stankovic. 2023. Robust Finger Interactions with COTS Smartwatches via Unsupervised Siamese Adaptation. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 25, 14 pages. doi:10.1145/3586183.3606794

[12] Xiao Chen, Wanli Chen, Kui Liu, Chunyang Chen, and Li Li. 2021. A comparative study of smartphone and smartwatch apps. In *Proceedings of the 36th Annual ACM Symposium on Applied Computing* (Virtual Event, Republic of Korea) *(SAC '21)*. Association for Computing Machinery, New York, NY, USA, 1484–1493. doi:10.1145/3412841.3442023

[13] Zhe Chen, Tianyue Zheng, Chao Cai, and Jun Luo. 2021. MoVi-Fi: motion-robust vital signs waveform recovery via deep interpreted RF sensing. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking* (New Orleans, Louisiana) *(MobiCom '21)*. Association for Computing Machinery, New York, NY, USA, 392–405. doi:10.1145/3447993.3483251

[14] Younggeol Cho and Pyungkang Kim. 2022. Real-Time Finger Force Estimation Robust to a Perturbation of Electrode Placement for Prosthetic Hand Control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 30 (2022), 1233–1243. doi:10.1109/TNSRE.2022.3171394

[15] Artem Dementyev and Joseph A. Paradiso. 2014. WristFlex: low-power gesture input with wrist-worn pressure sensors. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14)*. Association for Computing Machinery, New York, NY, USA, 161–166. doi:10.1145/2642918.2647396

[16] David Dobbelstein, Gabriel Haas, and Enrico Rukzio. 2017. The effects of mobility, encumbrance, and (non-)dominant hand on interaction with smartwatches. In *Proceedings of the 2017 ACM International Symposium on Wearable Computers* (Maui, Hawaii) *(ISWC '17)*. Association for Computing Machinery, New York, NY, USA, 90–93. doi:10.1145/3123021.3123033

[17] Kangkang Dong, Yao Chu, Xuanchen Tian, Tianqi Fang, Xiongying Ye, Xiaohao Wang, and Fei Tang. 2023. Wearable Photoelectric Fingertip Force Sensing System Based on Blood Volume Changes without Sensory Interference. *ACS applied materials & interfaces* 15 (07 2023). doi:10.1021/acsami.3c06469

[18] Kangkang Dong, Wentai Deng, Yuanyuan Liu, Tianqi Fang, Huaiyi Chen, and Fei Tang. 2024. Design and modeling of a fingertip force sensing system without tactile interference. In *Fourth International Conference on Sensors and Information Technology (ICSI 2024)*, Lijun Wu and Zhongpan Qiu (Eds.), Vol. 13107. International Society for Optics and Photonics, SPIE, 131071N. doi:10.1117/12.3029118

[19] Guilherme Konopatzki Filho, Guilherme Corredato Guerino, and Natasha M. C. Valentim. 2023. Usability and User Experience of Multi-touch Systems: A Systematic Mapping Study and Benchmark. *Journal on Interactive Systems* 14, 1 (Jul. 2023), 292–316.

[20] Yuliang Fu, Zhi Zhang, Rakshita Ranganath, Zhizhen Li, Yuchen Liu, Ning Sui, Huining Li, and Chenhan Xu. 2025. TouchWave: Exploring mmWave-Based Non-Contact Fingertip-Force Sensing in Activities of Daily Living. In *2025 IEEE 21st International Conference on Body Sensor Networks (BSN)*. 1–4. doi:10.1109/BSN66969.2025.11337366

[21] Hyunjae Gil, Hongmin Kim, and Ian Oakley. 2018. Fingers and Angles: Exploring the Comfort of Touch Input on Smartwatches. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 4, Article 164 (Dec. 2018), 21 pages. doi:10.1145/3287042

[22] Mayank Goel, Jacob Wobbrock, and Shwetak Patel. 2012. GripSense: using built-in sensors to detect hand posture and pressure on commodity mobile phones. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. Association for Computing Machinery, New York, NY, USA, 545–554.

[23] Jian Gong, Xinyu Zhang, Kaixin Lin, Ju Ren, Yaoxue Zhang, and Wenxun Qiu. 2021. RF Vital Sign Sensing under Free Body Movement. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 3, Article 101 (Sept. 2021), 22 pages. doi:10.1145/3478090

[24] Julien Gori and Olivier Rioul. 2020. A feedback information-theoretic transmission scheme (FITTS) for modeling trajectory variability in aimed movements. *Biological Cybernetics* 114, 6 (2020), 621–641.

[25] Anhong Guo and Tim Paek. 2016. Exploring tilt for no-touch, wrist-only interactions on smartwatches. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services* (Florence, Italy) *(MobileHCI '16)*. Association for Computing Machinery, New York, NY, USA, 17–28. doi:10.1145/2935334.2935345

[26] Yifan Guo, Zhu Wang, Qian Qin, Yangqian Lei, Qiwen Gan, Zhuo Sun, Chao Chen, Bin Guo, and Zhiwen Yu. 2025. mmPencil: Toward Writing-Style-Independent In-Air Handwriting Recognition via mmWave Radar and Large Vision-Language Model. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 9, 3, Article 81 (Sept. 2025), 30 pages. doi:10.1145/3749504

[27] Ryo Hajika, Tamil Selvan Gunasekaran, Chloe Dolma Si Ying Haigh, Yun Suen Pai, Eiji Hayashi, Jaime Lien, Danielle Lottridge, and Mark Billinghurst. 2024. RadarHand: A Wrist-Worn Radar for On-Skin Touch-Based Proprioceptive Gestures. *ACM Trans. Comput.-Hum. Interact.* 31, 2, Article 17 (Jan. 2024), 36 pages. doi:10.1145/3617365

[28] Chris Harrison, Desney Tan, and Dan Morris. 2010. Skinput: appropriating the body as an input surface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA) *(CHI '10)*. Association for Computing Machinery, New York, NY, USA, 453–462. doi:10.1145/1753326.1753394

[29] 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

[30] Eunah Hong, Dai-Soon Kwak, and In-Beom Kim. 2021. Morphological symmetry of the radius and ulna—can contralateral forearm bones utilize as a reliable template for the opposite side? *PLoS One* 16, 10 (2021), e0258232.

[31] Naveed Imran, Jian Zhang, Zheng Yang, and Jehad Ali. 2025. mm-FERP: An effective method for human personality prediction via mm-wave radar using facial sensing. *Information Processing & Management* 62, 1 (2025), 103919. doi:10.1016/j.ipm.2024.103919

[32] Yasha Iravantchi, Mayank Goel, and Chris Harrison. 2019. BeamBand: Hand Gesture Sensing with Ultrasonic Beamforming. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–10. doi:10.1145/3290605.3300245

[33] Chenxu Jiang, Sihan Yu, Jingjing Fu, Chun-Chih Lin, Huadi Zhu, Xiaolong Ma, Ming Li, and Linke Guo. 2024. Behaviors Speak More: Achieving User Authentication Leveraging Facial Activities via mmWave Sensing. In *Proceedings of the 22nd ACM Conference on Embedded Networked Sensor Systems* (Hangzhou, China) *(SenSys '24)*. Association for Computing Machinery, New York, NY, USA, 169–183. doi:10.1145/3666025.3699330

[34] Fazle Karim, Somshubra Majumdar, Houshang Darabi, and Shun Chen. 2018. LSTM Fully Convolutional Networks for Time Series Classification. *IEEE Access* 6 (2018), 1662–1669. doi:10.1109/ACCESS.2017.2779939

[35] Jiwan Kim and Ian Oakley. 2022. SonarID: Using Sonar to Identify Fingers on a Smartwatch. 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 287, 10 pages. doi:10.1145/3491102.3501935

[36] Hao Kong, Haoxin Lyu, Jiadi Yu, Linghe Kong, Junlin Yang, Yanzhi Ren, Hongbo Liu, and Yi-Chao Chen. 2024. mmHand: 3D Hand Pose Estimation Leveraging mmWave Signals. In *2024 IEEE 44th International Conference on Distributed Computing Systems (ICDCS)*. 1062–1073. doi:10.1109/ICDCS60910.2024.00102

[37] Hiroki Kurosawa, Daisuke Sakamoto, and Tetsuo Ono. 2018. MyoTilt: a target selection method for smartwatches using the tilting operation and electromyography. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services* (Barcelona, Spain) *(MobileHCI '18)*. Association for Computing Machinery, New York, NY, USA, Article 43, 11 pages. doi:10.1145/3229434.3229457

[38] Gierad Laput, Robert Xiao, Xiang 'Anthony' Chen, Scott E. Hudson, and Chris Harrison. 2014. Skin buttons: cheap, small, low-powered and clickable fixed-icon laser projectors. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14)*. Association for Computing Machinery, New York, NY, USA, 389–394. doi:10.1145/2642918.2647356

[39] 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* (Tokyo, Japan) *(UIST '16)*. Association for Computing Machinery, New York, NY, USA, 321–333. doi:10.1145/2984511.2984582

[40] James R Lewis and Jeff Sauro. 2018. Item benchmarks for the system usability scale. *Journal of Usability Studies* 13, 3 (2018), 158–167.

[41] Xinrong Li, Xiaodong Wang, Qing Yang, and Song Fu. 2021. Signal processing for TDM MIMO FMCW millimeter-wave radar sensors. *IEEE Access* 9 (2021), 167959–167971.

[42] Jaime Lien, Nicholas Gillian, M. Emre Karagozler, Patrick Amihood, Carsten Schwesig, Erik Olson, Hakim Raja, and Ivan Poupyrev. 2016. Soli: ubiquitous gesture sensing with millimeter wave radar. *ACM Trans. Graph.* 35, 4, Article 142 (July 2016), 19 pages. doi:10.1145/2897824.2925953

[43] Shu-Yang Lin, Chao-Huai Su, Kai-Yin Cheng, Rong-Hao Liang, Tzu-Hao Kuo, and Bing-Yu Chen. 2011. Pub - point upon body: exploring eyes-free interaction and methods on an arm. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (Santa Barbara, California, USA) *(UIST '11)*. Association for Computing Machinery, New York, NY, USA, 481–488. doi:10.1145/2047196.2047259

[44] Changjiang Liu, Yuanhao Li, Dongyang Ao, and Haiyan Tian. 2019. Spectrum-Based Hand Gesture Recognition Using Millimeter-Wave Radar Parameter Measurements. *IEEE Access* 7 (2019), 79147–79158. doi:10.1109/ACCESS.2019.2923122

[45] Haipeng Liu, Yuheng Wang, Anfu Zhou, Hanyue He, Wei Wang, Kunpeng Wang, Peilin Pan, Yixuan Lu, Liang Liu, and Huadong Ma. 2020. Real-time Arm Gesture Recognition in Smart Home Scenarios via Millimeter Wave Sensing. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 4, Article 140 (Dec. 2020), 28 pages. doi:10.1145/3432235

[46] Yiyue Luo, Yunzhu Li, Pratyusha Sharma, Wan Shou, Kui Wu, Michael Foshey, Beichen Li, Tomás Palacios, Antonio Torralba, and Wojciech Matusik. 2021. Learning human–environment interactions using conformal tactile textiles. *Nature Electronics* 4 (2021), 193 – 201.

[47] I Scott MacKenzie. 1992. Fitts' law as a research and design tool in human-computer interaction. *Human-computer interaction* 7, 1 (1992), 91–139.

[48] I. Scott MacKenzie and William Buxton. 1992. Extending Fitts' law to two-dimensional tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Monterey, California, USA) *(CHI '92)*. Association for Computing Machinery, New York, NY, USA, 219–226. doi:10.1145/142750.142794

[49] He Mao, Yue Zheng, Chenfei Ma, Kai Wu, Patrick Li, and Peng Fang. 2023. Simultaneous estimation of grip force and wrist angles by surface electromyography and acceleration signals. *Biomedical Signal Processing and Control* 79 (01 2023), 104088.

[50] Long Meng and Xiaogang Hu. 2024. Unsupervised neural decoding for concurrent and continuous multi-finger force prediction. *Computers in Biology and Medicine* 173 (2024), 108384.

[51] Vimal Mollyn and Chris Harrison. 2024. EgoTouch: On-Body Touch Input Using AR/VR Headset Cameras. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology* (Pittsburgh, PA, USA) *(UIST '24)*. Association for Computing Machinery, New York, NY, USA, Article 69, 11 pages. doi:10.1145/3654777.3676455

[52] 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* (St. Andrews, Scotland, United Kingdom) *(ITS '13)*. Association for Computing Machinery, New York, NY, USA, 189–198. doi:10.1145/2512349.2512821

[53] Albert Ng, Julian Lepinski, Daniel Wigdor, Steven Sanders, and Paul Dietz. 2012. Designing for low-latency direct-touch input. In *Proceedings of the 25th Annual ACM Symposium on User Interface Software and Technology* (Cambridge, Massachusetts, USA) *(UIST '12)*. Association for Computing Machinery, New York, NY, USA, 453–464. doi:10.1145/2380116.2380174

[54] Masa Ogata and Michita Imai. 2015. SkinWatch: skin gesture interaction for smart watch. In *Proceedings of the 6th Augmented Human International Conference* (Singapore, Singapore) *(AH '15)*. Association for Computing Machinery, New York, NY, USA, 21–24. doi:10.1145/2735711.2735830

[55] Masa Ogata, Yuta Sugiura, Yasutoshi Makino, Masahiko Inami, and Michita Imai. 2013. SenSkin: adapting skin as a soft interface. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (St. Andrews, Scotland, United Kingdom) *(UIST '13)*. Association for Computing Machinery, New York, NY, USA, 539–544. doi:10.1145/2501988.2502039

[56] Farshid Salemi Parizi, Eric Whitmire, and Shwetak Patel. 2020. AuraRing: Precise Electromagnetic Finger Tracking. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 4, Article 150 (Sept. 2020), 28 pages. doi:10.1145/3369831

[57] Philip Quinn. 2019. Estimating Touch Force with Barometric Pressure Sensors. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–7. doi:10.1145/3290605.3300919

[58] Rakshita Ranganath, Zhi Zhang, Yuliang Fu, Jinyuan Jia, Ying Chen, Huining Li, and Chenhan Xu. 2025. Hand-Grip Strength Estimation through Bioacoustic Sensing. In *2025 IEEE 21st International Conference on Body Sensor Networks (BSN)*. 1–4. doi:10.1109/BSN66969.2025.11337954

[59] Reza Rawassizadeh, Blaine A Price, and Marian Petre. 2014. Wearables: Has the age of smartwatches finally arrived? *Commun. ACM* 58, 1 (2014), 45–47.

[60] Marvin Reuter, Ali Ünal, Jan Felipe Kolodziejski Ribeiro, David Petersen, and Matthias Böhmer. 2025. MultiBezel: Adding Multi-Touch to a Smartwatch Bezel to Control Music. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI EA '25)*. Association for Computing Machinery, New York, NY, USA, Article 408, 6 pages.

[61] Jeff Sauro and James R Lewis. 2016. *Quantifying the user experience: Practical statistics for user research.* Morgan Kaufmann.

[62] Biyun Sheng, Shuqi Sun, Hui Cai, Chen Dai, and Fu Xiao. 2025. From Coarse to Fine: Fast and Effortless Facial Landmark Detection via mmWave Signals. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 9, 4, Article 208 (Dec. 2025), 25 pages. doi:10.1145/3770656

[63] Ruili Shi, Shuai Wang, Luoyu Mei, Xuehan Zhang, Zhao-Dong Xu, and Shuai Wang. 2026. mmSeg: Leveraging mmWave Radar for Fine-grained Human Semantic Segmentation. *ACM Trans. Internet Things* 7, 1, Article 10 (Feb. 2026), 32 pages. doi:10.1145/3786769

[64] R William Soukoreff and I Scott MacKenzie. 2004. Towards a standard for pointing device evaluation, perspectives on 27 years of Fitts' law research in HCI. *International journal of human-computer studies* 61, 6 (2004), 751–789.

[65] Srinath Sridhar, Anders Markussen, Antti Oulasvirta, Christian Theobalt, and Sebastian Boring. 2017. WatchSense: On- and Above-Skin Input Sensing through a Wearable Depth Sensor. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 3891–3902. doi:10.1145/3025453.3026005

[66] Ke Sun, Yuntao Wang, Chun Yu, Yukang Yan, Hongyi Wen, and Yuanchun Shi. 2017. Float: One-Handed and Touch-Free Target Selection on Smartwatches. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 692–704. doi:10.1145/3025453.3026027

[67] Ryosuke Takada, Wei Lin, Toshiyuki Ando, Buntarou Shizuki, and Shin Takahashi. 2017. A Technique for Touch Force Sensing using a Waterproof Device's Built-in Barometer. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI EA '17)*. Association for Computing Machinery, New York, NY, USA, 2140–2146.

[68] Martin Weigel, Aditya Shekhar Nittala, Alex Olwal, and Jürgen Steimle. 2017. SkinMarks: Enabling Interactions on Body Landmarks Using Conformal Skin Electronics. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 3095–3105.

[69] Hongyi Wen, Julian Ramos Rojas, and Anind K. Dey. 2016. Serendipity: Finger Gesture Recognition using an Off-the-Shelf Smartwatch. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 3847–3851. doi:10.1145/2858036.2858466

[70] Pui Chung Wong, Kening Zhu, and Hongbo Fu. 2018. FingerT9: Leveraging Thumb-to-finger Interaction for Same-side-hand Text Entry on Smartwatches. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–10.

[71] Yingxiao Wu, Haocheng Ni, Changlin Mao, Jianping Han, and Wenyao Xu. 2023. Non-intrusive Human Vital Sign Detection Using mmWave Sensing Technologies: A Review. *ACM Trans. Sen. Netw.* 20, 1, Article 16 (Nov. 2023), 36 pages. doi:10.1145/3627161

[72] Ziyi Xia, Xincheng Huang, Sidney S Fels, and Robert Xiao. 2025. HaloTouch: Using IR Multi-Path Interference to Support Touch Interactions with General Surfaces. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 548, 17 pages. doi:10.1145/3706598.3714179

[73] Robert Xiao, Teng Cao, Ning Guo, Jun Zhuo, Yang Zhang, and Chris Harrison. 2018. LumiWatch: On-Arm Projected Graphics and Touch Input. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–11. doi:10.1145/3173574.3173669

[74] Robert Xiao, Gierad Laput, and Chris Harrison. 2014. Expanding the input expressivity of smartwatches with mechanical pan, twist, tilt and click. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 193–196.

[75] Yingjing Xiao, Zhichao Huang, Junbin Ren, Yuting Bai, Haichuan Song, Zhanpeng Jin, and Yang Gao. 2025. Wrist2Finger: Sensing Fingertip Force for Force-Aware Hand Interaction with a Ring-Watch Wearable. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology* (Busan, Republic of Korea) *(UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 24, 16 pages. doi:10.1145/3746059.3747767

[76] Jiahong Xie, Hao Kong, Jiadi Yu, Yingying Chen, Linghe Kong, Yanmin Zhu, and Feilong Tang. 2023. mm3DFace: Nonintrusive 3D Facial Reconstruction Leveraging mmWave Signals. In *Proceedings of the 21st Annual International Conference on Mobile Systems, Applications and Services* (Helsinki, Finland) *(MobiSys '23)*. Association for Computing Machinery, New York, NY, USA, 462–474. doi:10.1145/3581791.3596839

[77] Chenhan Xu, Bing Zhou, Gurunandan Krishnan, and Shree Nayar. 2023. AO-Finger: Hands-free Fine-grained Finger Gesture Recognition via Acoustic-Optic Sensor Fusing. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 306, 14 pages. doi:10.1145/3544548.3581264

[78] Hongfei Xue, Yan Ju, Chenglin Miao, Yijiang Wang, Shiyang Wang, Aidong Zhang, and Lu Su. 2021. mmMesh: towards 3D real-time dynamic human mesh construction using millimeter-wave. In *Proceedings of the 19th Annual International Conference on Mobile Systems, Applications, and Services* (Virtual Event, Wisconsin) *(MobiSys '21)*. Association for Computing Machinery, New York, NY, USA, 269–282. doi:10.1145/3458864.3467679

[79] Baiju Yan, Peng Wang, Lidong Du, Xianxiang Chen, Zhen Fang, and Yirong Wu. 2023. mmGesture: Semi-supervised gesture recognition system using mmWave radar. *Expert Systems with Applications* 213 (2023), 119042.

[80] Jialiang Yan, Siyao Cheng, Yang Zhao, and Jie Liu. 2025. TapWristband: A Wearable Keypad System Based on Wrist Vibration Sensing. *IEEE Transactions on Mobile Computing* 24, 4 (2025), 2949–2966. doi:10.1109/TMC.2024.3503417

[81] Zhicheng Yang, Parth H. Pathak, Yunze Zeng, Xixi Liran, and Prasant Mohapatra. 2016. Monitoring vital signs using millimeter wave. In *Proceedings of the 17th ACM International Symposium on Mobile Ad Hoc Networking and Computing* (Paderborn, Germany) *(MobiHoc '16)*. Association for Computing Machinery, New York, NY, USA, 211–220. doi:10.1145/2942358.2942381

[82] Shumin Zhai, Jing Kong, and Xiangshi Ren. 2004. Speed–accuracy tradeoff in Fitts' law tasks—on the equivalency of actual and nominal pointing precision. *International Journal of Human-Computer Studies* 61, 6 (2004), 823–856.

[83] Bo Zhang, Boyu Jiang, Rong Zheng, Xiaoping Zhang, Jun Li, and Qiang Xu. 2023. Pi-ViMo: Physiology-inspired Robust Vital Sign Monitoring using mmWave Radars. *ACM Trans. Internet Things* 4, 2, Article 15 (May 2023), 27 pages. doi:10.1145/3589347

[84] 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* (Niagara Falls, Ontario, Canada) *(ISS '16)*. Association for Computing Machinery, New York, NY, USA, 13–22. doi:10.1145/2992154.2992187

[85] Chenyang Zhang, Tiffany S Ma, John Andrews, Eric J Gonzalez, Mar Gonzalez-Franco, and Yalong Yang. 2025. ForcePinch: Force-Responsive Spatial Interaction for Tracking Speed Control in XR. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology* (Busan, Republic of Korea) *(UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 27, 16 pages. doi:10.1145/3746059.3747694

[86] 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. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 3, Article 120 (Sept. 2017), 19 pages.

[87] Maotian Zhang, Qian Dai, Panlong Yang, Jie Xiong, Chang Tian, and Chaocan Xiang. 2018. iDial: Enabling a Virtual Dial Plate on the Hand Back for Around-Device Interaction. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 1, Article 55 (March 2018), 20 pages.

[88] Shichen Zhang, Qijun Wang, Kunzhe Song, Qiben Yan, and Huacheng Zeng. 2025. RadEye: Tracking Eye Motion Using FMCW Radar. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 821, 13 pages. doi:10.1145/3706598.3713775

[89] Yu Zhang, Tao Gu, Chu Luo, Vassilis Kostakos, and Aruna Seneviratne. 2018. FinDroidHR: Smartwatch Gesture Input with Optical Heartrate Monitor. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 1, Article 56 (March 2018), 42 pages. doi:10.1145/3191788

[90] 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* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 1491–1503. doi:10.1145/2858036.2858082

[91] Bofang Zheng, Yixin Li, Guanghua Xu, Gang Wang, and Yang Zheng. 2024. Prediction of Dexterous Finger Forces With Forearm Rotation Using Motoneuron Discharges. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 32 (2024), 1994–2004. doi:10.1109/TNSRE.2024.3402545

[92] Guoyang Zhou, Ming-Lun Lu, and Denny Yu. 2023. Investigating gripping force during lifting tasks using a pressure sensing glove system. *Applied Ergonomics* 107 (2023), 103917. doi:10.1016/j.apergo.2022.103917

[93] Peide Zhu, Hao Zhou, Shumin Cao, Panlong Yang, and Shuangshuang Xue. 2018. Control with Gestures: A Hand Gesture Recognition System Using Off-the-Shelf Smartwatch. In *2018 4th International Conference on Big Data Computing and Communications (BIGCOM)*. 72–77. doi:10.1109/BIGCOM.2018.00018