

MouseRing: Always-available Touchpad Interaction with IMU Rings

Xiyuan Shen
shenxiyuan_864@163.com
Department of Science and
Technology, Tsinghua University
Beijing, China

Chen Liang
lliangchenc@163.com
Department of Science and
Technology, Tsinghua University
Beijing, China

Chun Yu
chunyu@tsinghua.edu.cn
Department of Science and
Technology, Tsinghua University
Beijing, China

Haozhan Chen
chz21@mails.tsinghua.edu.cn
Department of Science and
Technology, Tsinghua University
Beijing, China

Xutong Wang
wangxuto18@mails.tsinghua.edu.cn
Department of Science and
Technology, Tsinghua University
Beijing, China

Yuanchun Shi
shiyu@tsinghua.edu.cn
Department of Science and
Technology, Tsinghua University
Beijing, China



Figure 1: Left: MouseRing is an IMU ring-shaped device that facilitates continuous on-surface finger tracking. Middle: MouseRing is an always-available pointing technique in VR, AR, large-screen interactions, etc. Right: MouseRing supports both single-ring and dual-ring interaction on diverse surfaces.

ABSTRACT

Tracking fine-grained finger movements with IMUs for continuous 2D-cursor control poses significant challenges due to limited sensing capabilities. Our findings suggest that finger-motion patterns and the inherent structure of joints provide beneficial physical knowledge, which lead us to enhance motion perception accuracy by integrating physical priors into ML models. We propose MouseRing, a novel ring-shaped IMU device that enables continuous finger-sliding on unmodified physical surfaces like a touchpad. A motion dataset was created using infrared cameras, touchpads, and IMUs.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CHI '24, May 11–16, 2024, Honolulu, HI, USA

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 979-8-4007-0330-0/24/05...\$15.00
<https://doi.org/10.1145/3613904.3642225>

We then identified several useful physical constraints, such as joint co-planarity, rigid constraints, and velocity consistency. These principles help refine the finger-tracking predictions from an RNN model. By incorporating touch state detection as a cursor movement switch, we achieved precise cursor control. In a Fitts' Law study, MouseRing demonstrated input efficiency comparable to touchpads. In real-world applications, MouseRing ensured robust, efficient input and good usability across various surfaces and body postures.

CCS CONCEPTS

• Human-centered computing → Ubiquitous and mobile computing; Pointing devices.

KEYWORDS

Ring Interaction, IMU, Finger Tracking, Touch Interface, Input, Wearable, Pointing Technique

ACM Reference Format:

Xiyuan Shen, Chun Yu, Xutong Wang, Chen Liang, Haozhan Chen, and Yuanchun Shi. 2024. MouseRing: Always-available Touchpad Interaction with IMU Rings. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24), May 11–16, 2024, Honolulu, HI, USA*. ACM, New York, NY, USA, 19 pages. <https://doi.org/10.1145/3613904.3642225>

1 INTRODUCTION

Target selection is one of the most fundamental tasks in human-computer interaction. Traditional methods, such as the mouse and touchpad, have been widely adopted due to their intuitive design, efficiency, and precision. However, their physical attributes present certain constraints, particularly in mobile environments. The rise of ubiquitous computing devices, such as AR/VR and large-screen displays, has generated demands for always-available input solutions. While remote controllers[3, 43] and computer-vision-based finger-pointing techniques[44, 57] have found their niche, their complex setup and large computational power requirements restrict them from being always accessible.

We propose that wearable IMU rings could potentially serve as an always-available touch interface[4]. Our goal is to retain the efficiency and comfort characteristic of touchpad interactions while optimizing the device's form factor to enhance its portability and availability at all times. IMU rings are smaller and lower-powered compared to cameras[45] or electromagnetic sensors[9, 48], making them suitable for long-term daily use. However, precise finger tracking based on IMU rings[29, 62] is challenging due to limited information and noisy signals. While researchers have attempted physical mapping and machine learning methods to solve gesture classification[16, 24, 39, 41] and typing tasks[19, 20, 30], few prior works have demonstrated the use of IMU rings for high-precision 2D cursor control and target selection tasks.

We propose MouseRing, a ring-formed IMU device that can accurately track fingertip movement trajectories. By incorporating touch state detection as a cursor movement switch, MouseRing enables always-available touchpad interaction. In our work, we follow a data-driven research process. We first uncover some physical constraints as prior during finger-sliding based on data analysis. We then train ML models for fingertip velocity prediction. Finally, we integrate the knowledge of both for more stable and accurate tracking.

We construct a multimodal motion dataset of sliding fingers using OptiTrack, pressure touchpads, and IMUs. The dataset contains contact points, nail tips, and several key joints of the index fingers. Following the research approaches in computer vision[36, 37], we model the index finger as articulated objects in 3D space with interconnected joints. We propose several hypotheses that satisfy certain constraint relationships. By analyzing the data (analysis process detailed in Appendix A), we verify the existence of coplanarity, velocity correlation, rigid constraints, and other physical relationships between joint nodes.

We use an end-to-end RNN model to predict the instantaneous velocity of fingertips. We then quantify the degree of conformity between the predicted velocity and the physical constraints with confidence scores. The confidence score serves as a weight to correct and smooth the trajectory. Using the dataset for offline simulation, we compare the performance of different model settings, ring

numbers, and sliding modes. In a purely kinematics-based baseline model, the tracking error is significant. End-to-end models shows acceptable performance. By incorporating physical constraints, the results become more stable and accurate. Dual MouseRing can accurately select small targets, with a mean angular error θ_{error} of 6.6° , while a single MouseRing has a θ_{error} of 12.3° but is more lightweight.

We evaluate the input efficiency of MouseRing in ideal laboratory conditions and real-world situations. In the Fitts' Law Study, MouseRing achieves input efficiency close to laptop touchpads (MT = 658.5ms vs. 629.1ms). In a real-world large-screen interaction task, single and dual MouseRing can both achieve robustly and quickly 2D cursor control on surfaces of different hardness and flatness and in standing and sitting postures. Its speed is similar to that of mouse devices and significantly outperforms AirMouse devices, which share the same in-air interaction paradigm as visual hand tracking and remote controllers. Participants appreciated the naturalness of the interaction and found wearing MouseRing more comfortable than hand-held devices.

In summary, our main contributions include:

- We propose MouseRing, a ring-formed IMU device that tracks index finger sliding on unmodified physical surfaces and supports accurate and robust continuous pointing interactions.
- We model finger sliding and identify several physical priors between key joints of the index finger.
- We propose a precise and stable fingertip-tracking algorithm that incorporates physical knowledge into machine learning methods.

2 RELATED WORK**2.1 Always-available Pointing Technique**

The emergence of various always-available pointing techniques aims to strike a balance between convenience and performance to meet requirements in different scenarios, such as AR/VR and large displays. The most prevalent approach relies on cameras and CV algorithms to recognize finger-pointing[44, 57]. Finger-pointing in mid-air has already become the standard paradigm for AR/VR interaction[15]. To accommodate hands-free situations, head movement[13, 58] and eye-tracking[10, 23, 56, 61] for cursor control have also been proposed. Researchers also combined cameras with other sensors such as EMG[52], IMU[21], and touchpads[55] to improve input efficiency. Camera-based solutions offer high precision but require sensors deployed in the environment or on the headset. They also present privacy problems[22]. Consequently, researchers have begun exploring the use of other smart devices for cursor control tasks. The first category includes dedicated devices like laser pointers[25, 31] and remote controllers[3, 43], which offer good precision but are not easily portable for users.

The second category involves using daily-carry smart devices, such as smartphones[2, 5, 17, 26] and smartwatches[27, 28], to facilitate pointing input. Smart rings also fall into this category. Due to their small size and portability, smart rings can free users from handheld devices like controllers and environment-deployed sensors such as cameras, offering a universally applicable pointing technique. Additionally, smart rings stand out in certain scenarios compared to smartphones and smartwatches as they are always

Table 1: Prior Work on Ring-shaped devices for finger-tracking or target selection tasks

| Work | Sensor | Number | Position | Additional Device | Calibration | Interaction Paradigm | Support Finger Tracking | Support 2D cursor control |
|---------------------|-----------------------|--------|---|-------------------|-------------|----------------------|-------------------------|---------------------------|
| Finexus[9] | Electromagnet | 4 | Fingertip | Wristband | ✗ | In-air pointing | ✓ | ✗ |
| AuraRing[48] | Electromagnet | 1 | Proximal phalanx | Wristband | ✓ | In-air pointing | ✓ | ✗ |
| Yuki Kubo[32] | Pressure Board | 1 | Intermediate phalanx | - | ✗ | On-surface sliding | ✗ | ✓ |
| Magic Finger[59] | Camera | 1 | Fingerpad | - | ✗ | On-surface sliding | ✓ | ✗ |
| LightRing[29] | IMU + Infrared Sensor | 1 | Proximal phalanx | Infrared Sensor | ✓ | On-surface sliding | ✓ | ✓ |
| Mouse on a Ring[62] | IMU | 1 | Proximal phalanx | - | ✗ | In-air tilting | ✗ | ✓ |
| Anywhere Touch[47] | IMU | 1 | Fingertip | - | ✗ | On-surface sliding | ✓ | ✗ |
| MouseRing | IMU | 1/2 | Proximal phalanx (Intermediate phalanx) | - | ✗ | On-surface sliding | ✓ | ✓ |

worn and do not need to be adjusted or taken from pockets. At the same time, we aim for the ring device to maintain high input efficiency comparable to traditional pointing input methods like touchpads and mice. In summary, our research on MouseRing aims at delivering an always-available pointing interface with greater convenience while maintaining satisfactory usability.

2.2 Ring-based Interactions

The primary usability goal of MouseRing is to enable efficient and accurate input on diverse surfaces while ensuring comfort and convenience through its always-available nature. To summarize the previous work, we have listed prior works that utilize ring-shaped devices for finger-tracking or target selection tasks in Table 1. We have outlined them based on three usability aspects: the comfort and portability of the setup, the interaction paradigm, and the sensing capability.

Finexus[9] and AuraRing[48] utilize multiple electromagnetic sensors to position the fingertip relative to the wrist nodes, thereby supporting in-air pointing for target selection. However, this necessitates wearing additional wristbands or watches and at least four sensors for positioning based on their relative distances. Finexus and AuraRing utilize an in-air pointing interaction paradigm, where users control the cursor by adjusting the orientation of their wrist or fingers. In contrast, MouseRing offers an interaction mode similar to traditional mouse devices, where users slide their fingers on flat surfaces. Both paradigms are easy to use, but prolonged in-air interaction may quickly lead to fatigue. Yuki Kubo[32] has implemented a touchpad-like interaction by installing a small pressure plate on the side of the ring. Magic Finger[59] achieves mouse-like functionality by using a camera on the finger pad to detect relative movement between the finger and the surface. While these interactions are highly intuitive, placing sensors on the finger pad or touchpads on the side of the finger significantly compromises comfort, impacting everyday activities.

IMU rings offer the advantage of being lighter, featuring a low-power, small-size setup, ensuring users can comfortably wear and use smart rings for extended periods. However, their sensing capabilities are somewhat limited. Despite considerable research implementing various gesture-based interactions[11, 16, 24, 39, 41] and Bayesian inference-based typing inputs[20, 30, 38] through IMU rings, there have been few efforts to track the fingertip for continuous 2D cursor control directly. LightRing[29] first proposed using

an IMU to estimate lateral finger movements and an infrared sensor to perceive finger bending to estimate forward and backward movements. However, this approximation lacks precision and requires a complex calibration phase. Mouse on a Ring[62] does not directly track finger movement but controls cursor movement through an airmouse mechanism, using the tilt and acceleration changes of the IMU ring in the air. AnywhereTouch[47] uses the attitude angle of the fingertip IMU, calculates the speed of each finger joint through inverse kinematics, and achieves a 93% accuracy rate in uni-stroke three-classification tasks. This approach has been inspiring for our research approach to physics-informed machine learning. However, the finger-tracking results obtained directly from inverse kinematics are limited and do not support target selection tasks. Placing an IMU on the fingertip can also interfere with daily activities. Our work has systematically studied and modeled fingertip motion behavior, enhancing the fingertip tracking precision against prior work through the use of physics-informed machine learning. This has allowed us to realize the concept of precise touchpad interaction everywhere. Furthermore, the requirement of being always available also brings forth the usability demands of comfort and convenience. MouseRing meets these criteria by being lightweight, calibration-free, and robust to different surfaces, thereby ensuring a positive wearing and using experience in real-world scenarios.

2.3 Hand Modelling

In Human-Computer Interaction (HCI) and Computer Vision (CV), human hands are often modeled as articulated objects in three-dimensional space with interconnected joints[12, 49]. Kuch *et al.*[33] and Lee *et al.*[36, 37] have proposed simplified hand skeleton models with 26 and 27 degrees of freedom (DOF) that systematically describe the movements of individual joints. Many studies on gesture recognition[1, 50, 51, 54] followed their works and simplified models for specific tasks to achieve state-of-the-art performances. For example, Ahmad *et al.*[1] achieved 30 FPS gesture recognition using a 19-DOF simplified model. Spurr *et al.*[54] integrated hand constraints to achieve weakly-supervised gesture recognition.

We believe incorporating physical knowledge into the pipeline of sensing finger-sliding behaviors in MouseRing can also benefit tracking accuracy. However, several challenges exist. Firstly, the pressure between fingertips and surfaces can cause forced bending of finger joints and ligament deformation[34], rendering Lee's constraints[36] for relaxed hands no longer valid. Secondly, the

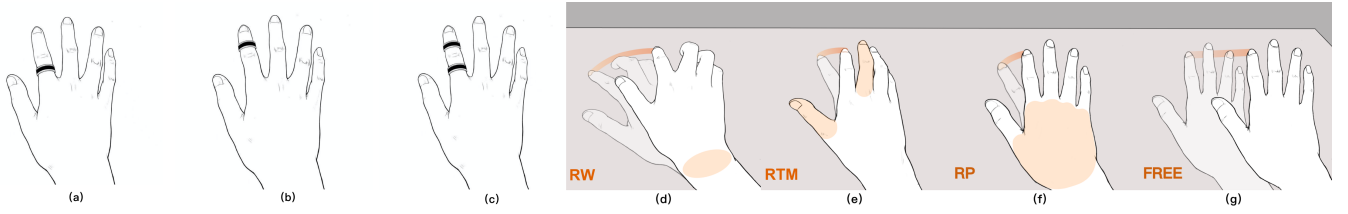


Figure 2: Different wearing configurations and finger-sliding modes. (a) Single ring on the proximal phalanx. (b) Single ring on the intermediate phalanx. (c) Double rings on both phalanxes. (d) Rested Wrist. (e) Rested Thumb and Middle finger. (f) Rested Palm. (g) Hand freely suspended in the air.

impact of skin deformation must be considered when we observe finger movements using an IMU ring fixed on the skin. Lastly, finger sliding only represents a small subspace of hand movement, potentially resulting in additional beneficial physical motion relationships between joints.

Our study builds upon previously proposed 3D hand models[37] and focuses on finger-sliding input tasks. We employ data-driven research methods to propose new motion relationships and disprove invalid physical constraints. The physical knowledge can provide assistance in developing stable and accurate MouseRing fingertip tracking algorithms.

3 INTERACTION DESIGN SPACE

The input action of MouseRing inherits the standard touchpad input for users. Users can control the cursor and turn the physical surface into a virtual touchpad by touching and sliding their index fingers on the surface. The sliding action is very straightforward for users because the physical contact position between the finger and the surface maps well to the cursor on the virtual interface. However, for IMU sensing, different ring numbers, wearing positions, and the movement modes of the entire hand during finger sliding may significantly impact fingertip tracking accuracy. Different ways of wearing rings and finger-sliding patterns create a tradeoff between the naturalness of wearing&interacting and the algorithm’s tracking performance. While fewer rings and freer sliding give users a more comfortable input experience, they can also lead to poorer accuracy and bring users a worse sense of control. The fluency of input and the sense of control in the interaction process make up the overall user experience of MouseRing.

Hence, we believe exploring possible ring positions and finger-sliding patterns is necessary. Evaluating their impact on tracking performance in subsequent research will enable us to choose the most natural wearing and interaction method that meets the accuracy requirements.

3.1 Ring Number and Position

MouseRing has three ring-wearing configurations, including double-ring and two single-ring configurations. Under the double-ring configuration (Fig.2(c)), users wear two rings with 6-axis IMU sensors on their index finger’s intermediate phalanx and proximal phalanx. Under the single-ring configuration (Fig.2(a)-(b)), users wear a ring on either phalanx.

For the ring position, previous research[18] has shown that users prefer to use their index finger for touch input. Therefore, we chose

to place rings on the index finger. Although placing the IMU sensor on the distal phalanx provides the richest information due to its proximity to the fingertip[53], it would significantly impact users’ daily activities. Therefore, we only consider the phalanxes, which are further back in position.

Regarding the number, placing more rings on one finger may help better estimate the angle between finger bones. We conducted a pilot study and interviewed 12 users to determine the maximum number of rings they could tolerate for an extended period. All users agreed that one ring was acceptable, and most users (9 in 12) considered that two rings were acceptable despite having a minor impact on daily activities. Most users rejected more rings due to interference with daily activities, comfort, and aesthetics. Given that the pilot study is preliminary and solely based on intuition, we will further evaluate the comfort level of wearing varying numbers of rings under real-world conditions.

3.2 Finger-sliding Mode

IMU rings predict finger movements by observing the local accelerations of the skeletal region where they are worn and establishing a relationship with the corresponding fingertip movements. Therefore, the prediction accuracy is greatly influenced by the chosen finger-sliding mode. It’s obvious that the level of hand restriction presents a trade-off between accuracy and comfort. Performing finger sliding when the entire hand is suspended in the air without any constraints (Fig.2(g)) is quite effortless and free for users. However, at this point, IMU-based tracking is impractical in principle, because the uniform motion of the entire hand cannot be perceived by accelerometers or gyroscopes. Prior work has explored finger-sliding under the fixation of the middle finger and thumb (LightRing[29]) or the entire hand (Anywhere Touch[47]). In our work, we propose three different finger-sliding modes (Fig.2(d)-(f)) with varying degrees of restriction, and systematically compare their algorithm tracking accuracy in subsequent research, aiming to find an interaction method that satisfies both accurate tracking and user-friendliness.

- **Rested Wrist mode (RW):** Users keep their wrist rested on the surface during each finger-sliding stroke. Each finger can bend and move freely. The entire hand can freely rotate around the wrist. RW restricts the translation of the palm, but the rotation of the palm and the translation of each finger are still free.
- **Rested Thumb & Middle Finger mode (RTM):** Apart from the carpus, users need to keep their middle finger and thumb

resting on the surface during each finger-sliding stroke. RTM restricts the translation and large-scale rotation of the palm. Due to the flexibility of the joints, small-scale rotation still exists, and the movement of the index finger is unrestricted.

- **Rested Palm mode (RP):** Users need to place their palm rested on the surface during each finger-sliding stroke. Under RP, both the translation and rotation of the palm are fixed, and only the movement of the index finger is unrestricted.

4 DATA COLLECTION

In this section, we collected multi-channel motion data of key joints on the index finger and the touchpoint during sliding. We used IMU sensors, an Optitrack optical tracking system, and a pressure touchpad for data collection. We had three main motivations for collecting motion data. First, we planned to explore the physical motion model of index finger sliding, including the physical relationships and motion constraints between joints. Secondly, we aimed to design and implement the fingertip-tracking algorithm of MouseRing through data-driven approaches. Lastly, we also segmented data for touch state detection between fingers and surfaces.

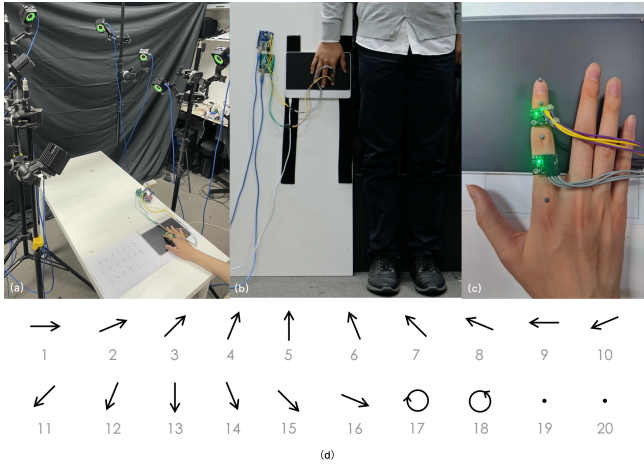


Figure 3: Apparatus for data collection. (a)The arrangement of the Optitrack camera array. **The participant is inputting on a horizontal touchpad. (b)** The participant interacting on a vertical touchpad. **(c)** The index finger with two IMUs attached and four retroreflective spheres on three joint points and the nail tip. **(d)** Finger-sliding stroke set.

4.1 Apparatus

The experimental apparatus is shown in Fig.3. Participants wore two IMU rings on their index finger’s intermediate and proximal phalange. Participants were instructed to wear the IMU sensors on the dorsal side of their fingers. In addition, four 2mm-diameter infrared reflective spheres were attached to the metacarpophalangeal joint (MCP), proximal interphalangeal joint (PIP), distal interphalangeal joint (DIP), and nail tip of the index finger. The four markers were pre-labeled in the Optitrack system, allowing us to identify which joint each marker corresponds to. Participants were asked

to slide their index fingers on a pressure touchpad, ensuring constant contact between the fingertip and the touchpad. The touchpad recorded the position of the contact point between the fingertip and the surface. During the experiment, we sampled the IMU’s acceleration and angular acceleration data, the position of the four key points of the finger tracked by Optitrack, and the pressure array data from the touchpad.

Each IMU sensor was a 9-axis accelerometer MPU9250. We recorded 6-axis acceleration and angular velocity data from the sensor and simultaneously logged the IMU’s attitude angles calculated in real time by the attitude estimation algorithm (section 6.2). We only used the six-axis data of acceleration and angular acceleration because we discovered that the uneven magnetic field variations in indoor spaces resulted in higher accuracy in 6-axis attitude estimation compared to the 9-axis (mentioned in section 6.2). The IMU frame rate was 200Hz. Each IMU was fixed on an adjustable iron ring and connected to an Arduino UNO via DuPont wires. During data collection, wired data transmission ensured high data quality. However, in real-world user experiments (section 9), we utilized a wireless transmission MouseRing prototype to ensure a more realistic user experience. The touchpad was the Morph Sensal touchpad, which can sense the pressure peak point of the touch and return the pressure force and peak point coordinates. The touchpad frame rate was 80Hz. We used eight OptiTrack Prime 13 motion capture cameras to capture the three-dimensional coordinates of the index finger’s joints and the nail tip. The camera array was placed 1-2m from the finger to ensure high-precision capture. The frame rate of OptiTrack Motive was 200Hz.

4.2 Participants

We recruited 12 participants (5 females, aged 20-26, $M = 23.6$) from the university campus. All participants were right-handed and used their right index fingers to input. The average length of index fingers was respectively 75.7mm($std=1.60$) and 79.2 millimeters ($std=2.79$) for females and males, which were close to the existing literature[40]. We concluded that our participants’ finger lengths can represent the majority of users.

4.3 Design and Procedure

Data collection included $2postures * 3modes = 6sessions$. Participants were required to perform finger-sliding input on a horizontally positioned touchpad while sitting and a vertically positioned touchpad while standing. For each touchpad orientation, participants were required to perform data collection sessions using the three finger-sliding modes: Rested Wrist mode, Rested Thumb & Middle Finger mode, and Rested Palm mode. During each session, participants completed 20 different one-stroke finger-sliding movements.

Fig.3(c) shows the 20 strokes. The 1st-16th one-stroke movements are straight lines in various directions. We divided 360° into 16 equal parts, and the angle between the i^{th} line and the positive x-axis was $22.5(i - 1)^\circ$. These data were used to simulate the user moving the cursor in various directions. The 17th and 18th one-stroke movements are clockwise/counter-clockwise circles, which simulate the user making small-radius turns and slight adjustments. For the 19th and 20th one-stroke movements, the participant’s index

finger remained stationary, and these data were used as negative examples for ML learning. We encouraged participants to perform input with various initial postures to cover the entire posture space in our dataset.

In each session, assisted by paper-printed one-stroke images, the participants controlled their index fingers to slide 10 times along their imaginary direction. Before starting each one-stroke movement, participants needed to tap the touchpad twice quickly. These double taps were only used to align the data from different sensors. It is worth noting that the stroke directions on paper were only provided as a reference for the participants. During algorithm optimization, we used the actual trajectory captured by the pressure sensors and cameras as the ground truth. Each participant completed a total of $2(\text{postures}) \times 3(\text{modes}) \times 20(\text{strokes}) \times 10(\text{times}) = 1200$ finger-sliding strokes. They were allowed to rest for 5 minutes between sessions. The entire experiment lasted approximately 100 minutes.

4.4 Data Pre-processing

We first unified the data from all channels to 200Hz using linear interpolation. Due to the time delay in transmitting OptiTrack data, we developed a graphical interface to manually align the OptiTrack signal with the other two channels. The double taps made by the participants during the experiment left two peaks in the z-axis data from the accelerators and the OptiTrack. Two annotators independently adjusted the time deviation to align the signals, and the labeled time deviation differences between both annotators were less than or equal to 2 frames ($\leq 0.01s$). Each segment of aligned data has a recording duration of approximately 10 minutes (around 2000 frames). Each data segment includes about 200 instances of finger-sliding data, where the finger performs uni-stroke gestures on the touchpad.

4.5 Data Segmentation

4.5.1 finger-sliding data segmentation. We extracted data segments for finger tracking for each finger-sliding movement during the uni-stroke gestures. The rising and falling edges of the pressure touchpad data were utilized to segment each stroke automatically. Additionally, the script automatically filtered out data that was too long (over 5 seconds) or too short (less than 0.2 seconds), which accidental touches or quick taps could cause. Ultimately, we obtained approximately 14,000 finger-sliding data segments, equalizing around 7,000 segments in the horizontal and vertical planes. The average length of each data segment is 0.7 seconds (141 frames). Each data frame includes information on the three-dimensional coordinates of four key joints, accelerations, attitude angles, angular accelerations from two IMUs, and the two-dimensional coordinates of the fingertip on the pressure pad.

4.5.2 touch events segmentation. In addition to tracking the fingertip on the surface, our system also requires real-time detection of the user's fingertip contact state with the surface. Therefore, we segmented four types of touch events, namely touch-down, touch-up, touching, and in-air, which are useful for touch state detection. For touch-down events, we identified the rising edge of the pressure data on the touchpad at frame t and extracted the data from frames $[t-9, t]$ as touch-down data. Similarly, we identified the falling edge of the pressure data at frame t for touch-up events and extracted

the data from frames $[t, t+9]$ as touch-up data. Additionally, we randomly selected time windows of 10 frames, where the finger was in complete contact with the pressure touchpad, as touching data. We set time windows of 10 frames where the finger was completely in the air as in-air data. As a result, we obtained 28,000 clipped touch event data samples in total. Each event accounted for one-fourth of the total dataset.

5 UNCOVERING PHYSICAL KNOWLEDGE OF FINGER-SLIDING

Although two 6-axis IMU sensors can provide rich motion information, it is still far from sufficient to fully reconstruct the index finger's motion. In this section, we analyzed the collected finger-sliding data segments following the research process of "observation-hypothesize-analyze-verify/falsify." We induced several motion laws of key joint movement, which would be applied as physical knowledge for fingertip tracking.

5.1 Modeling Index Finger Movement

We simplified the physical model of the various bones and joints of the index finger using a kinematic chain (Fig.4(a)), referencing modeling methods from computer vision[36, 37] and surgical medicine[34]. The three joints of the index finger (DIP, PIP, MP) connect the three bones of the finger (distal phalanx, intermediate phalanx, proximal phalanx) and the carpal bones of the hand. In addition, we also introduced the tip of the fingernail, as well as the contact point between the fingertip and the surface, because they are also crucial in finger sliding. The nail tip and DIP jointly model the vector corresponding to the distal phalanx, whose displacement is strongly correlated with the displacement of the contact point. The tactile sensation brought by the contact point and surface is the most intuitive way for users to perceive finger sliding.

5.2 Constraints on Joint Motion during Finger Sliding

Based on the aforementioned modeling, we propose six hypotheses regarding motion constraints, which are based on existing literature[37] and observations. To validate or falsify these hypotheses, we employed hypothesis testing on statistical measures and visualized the results in Figure 4(b)-(j). We found that several assertions regarding finger joint constraints in the natural state may not hold when the finger is in a tense state, as the joints and ligaments can be passively pulled. Conversely, due to the small-scale movements of the joints involved in finger sliding, complex hand mechanical movements can be approximated by simpler models in local motion spaces. We summarize the conclusions here, while the detailed data analysis is presented in Appendix A.

- **Conclusion 1:** The squeezing between the plane and the index finger leads to $\theta_{DIP} \neq \frac{2}{3}\theta_{PIP}$, where θ_{DIP} is defined as the angle between the phalanxes connected by the DIP joints. θ_{PIP} is the angle between the phalanxes connected by the PIP joint. (Fig.4(b)-(c))
- **Conclusion 2:** The three phalanxes of the index finger are in the same plane. (Fig.4(d))

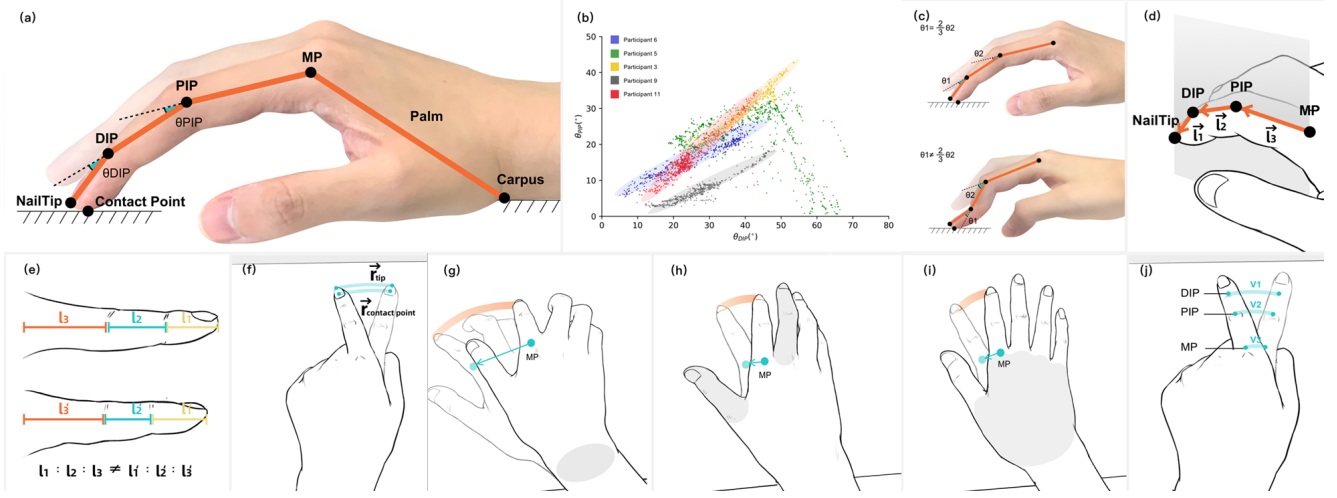


Figure 4: (a) Physical model of the index finger. In the kinematic chain, the nail tip, three joints of the index finger (DIP, PIP, MP), and carpus are articulated. (b) The relationship between θ_{DIP} and θ_{PIP} . We selected 5 representative participants from the initial 12. For P3 and P11, the angular relationship remained consistent. For P6 and P9, the proportionality coefficient was no longer $\frac{2}{3}$. For P5, the relationship no longer stands due to the squeezing between the finger and the plane. (c) The force exerted by the surface on the distal phalanx broke the proportionality relationship between θ_{DIP} and θ_{PIP} . (d) The distal phalanx, intermediate phalanx, and proximal phalanx are in the same plane. (e) The ratio of the phalanxes' lengths varies significantly between individuals. (f) The displacements of the fingertip and the contact point are equal. (g)–(i) MP's displacement cannot be ignored during finger-sliding. (j) There is a strong correlation between the projected velocities of the fingertip, DIP, and PIP.

- **Conclusion 3:** Each person's skeletal length is fixed during index finger sliding, but the ratio of the phalanxes' lengths varies significantly between individuals. (Fig.4(e))
- **Conclusion 4:** The displacement of the fingertip projection on the physical surface is approximately equal to the displacement of the contact point. (Fig.4(f))
- **Conclusion 5:** Among all three sliding modes (RW, RTM, RP), the displacement of MP cannot be ignored. (Fig.4(g)–(i))
- **Conclusion 6:** There is a strong correlation between the projected velocities of the fingertip, DIP, and PIP on the physical surface. (Fig.4(j), more quantitative analysis in Appendix A)

6 MOUSERING ALGORITHM

This section introduces the MouseRing algorithm, which aims at achieving precise and stable fingertip motion tracking. Our algorithm consists of four key processes, which are also the main technical contributions of this paper: (1) **High-precision IMU attitude estimation for smart ring interactions**, (2) **Fingertip velocity prediction based on RNN models**, (3) **Velocity correction using physical constraints**, and (4) **Robust touch-state detection**.

6.1 Overview

The overall goal of MouseRing is to achieve precise fingertip tracking through IMU sensing. We employ an intuitive approach that predicts the real-time velocity of the fingertip, accumulates these velocities, and updates the sliding trajectory in real-time. As shown in Fig.5, we introduce the algorithm pipeline. We use the orientation of the IMUs worn on the intermediate and proximal phalanx

of the index finger to represent the spatial orientation of the two finger phalanxes. We estimate each bone's attitude by processing the continuously read acceleration and angular acceleration data streams. Since the accuracy of the attitude is crucial for calculating the ML model and physical constraints, we have carefully optimized the algorithm for hand interaction in attitude estimation. Next, we train an RNN-based model to learn from features such as finger skeleton attitudes and ring accelerations and to predict fingertip velocity. However, black-box probabilistic models suffer from unstable predictions and poor interpretability. Therefore, we establish several physical constraints based on the attitudes and velocity, judge the degree of compliance between the predicted instantaneous velocities and physical constraints, and correct the velocity. In addition, we also implement touch state detection and cursor smoothing to achieve a complete mouse-like target selection interaction experience.

6.2 Attitude Estimation

We placed an OptiTrack marker on an IMU sensor under rotational motion to collect ground truth data. We employed direct integration, standard 6-axis complementary filtering, and 9-axis complementary filtering methods to estimate the attitude, resulting in average errors of 6.63° , 3.58° , and 9.25° . Direct integration of angular accelerations led to significant attitude drift due to random environmental noise and sensor system bias. On the other hand, indoor environments exhibit pronounced and non-uniform variations in magnetic fields, with the orientation of the magnetic vector deviating up to 40° within a $1m \times 1m$ area. This significant deviation severely affects the accuracy of magnetometer data. Therefore, considering the

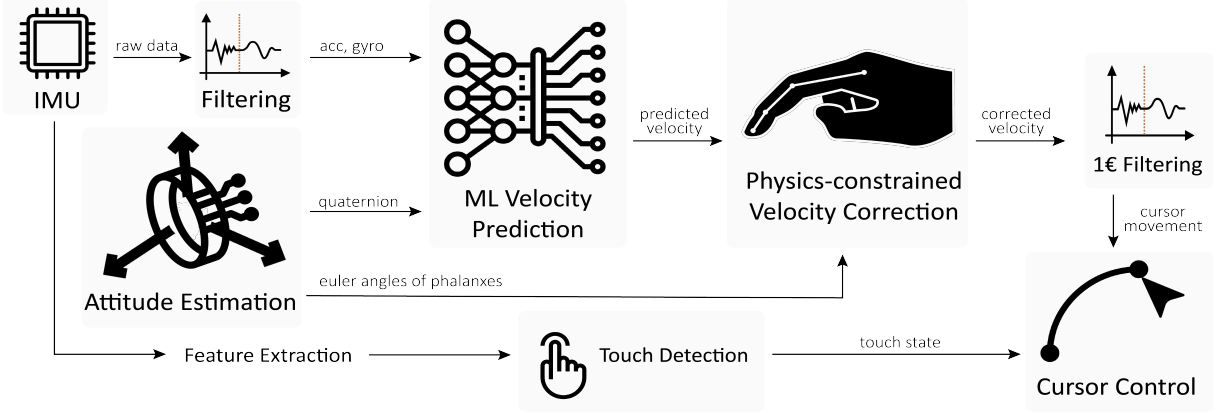


Figure 5: System overview. We predict the real-time velocity through an RNN model. Then, we correct the velocity with physical constraints to achieve accurate and stable tracking.

aforementioned challenges, we optimized our system within the framework of 6-axis attitude estimation.

We find that as a musculoskeletal movement mechanism, the movement of the index finger is smooth. There is no high-frequency component in its acceleration and angular acceleration. Thus, we recommend using a 1Hz-5Hz Butterworth bandpass filter to filter the acceleration signal and a 1Hz-10Hz Butterworth bandpass filter for the angular acceleration signal. After removing high-frequency noise, we compare the measured acceleration with the ground truth obtained by differentiating 3D positions. The error is reduced by over 20%.

Secondly, we apply a passive complementary filter method to calculate the attitude of the ring. Similar algorithms[14] were initially used for attitude estimation of large-scale, high-speed objects such as aircraft. We redesign the controller parameters for small-scale finger movements. We use the Mahony algorithm[14] as the framework. Its complementary filter algorithm can be regarded as a second-order control system with the characteristic polynomial $S^2 + K_p S + K_i$. K_p and K_i can be expressed as $K_i = \omega^2$, $K_p = 2\zeta\omega$, where ω and ζ represent the cutoff frequency and damping coefficient. In this control system, the second-order control system has the best response when the damping ratio $\zeta = 0.707$ [60]. When ω is around 1 rad/s, the finger IMU's attitude estimation has a faster response speed and more accurate results. Too small ω causes an attitude drift, while too large ω results in a significant fluctuation. Therefore, we set the parameters as $\omega=1.5\text{rad/s}$ and $\zeta=0.707$. We use Euler angles (roll, pitch, and yaw) to measure the average error of the algorithm. The mean errors are 0.55° , 3.12° , and 1.65° .

6.3 Machine Learning for Speed Prediction

We design an RNN model to predict the index finger's real-time velocities of the four key joint points (NailTip, DIP, PIP, and MP). We use the Nail Tip instead of the Contact Point of the fingertip as the optimization target (Conclusion 4 in Section 5.2). The model includes predictions of the MP joint for all input modes because its displacement cannot be ignored (Conclusion 5 in Section 5.2).

We select the filtered acceleration, the filtered angular acceleration, and the attitude obtained from the Mahony algorithm as

the input features. We use quaternions Q as input features for the IMU attitude, which provides better prediction results than Euler angles and direction vectors. Instead of frame-to-frame prediction, we use the signals within a 20-frame (0.1s) time window. Thus, the model can learn both the current motion state and the recent motion trends. Depending on the number of rings, either or both of the ring information are utilized.

The model's output predicts the real-time velocities of the index fingers' four key points (Nail Tip, DIP, PIP, and MP). The projected velocity of the Nail Tip represents the user's input. The velocity predictions of other key points can help correct the velocity of the Nail Tip in physical constraints. We average the velocities of the last five frames to smooth the displacement jumps caused by Optitrack cameras.

Our model consists of a single-layer LSTM with a hidden state size of 32 and two linear layers, followed by RELU as the activation function before each linear layer. Let y be the velocity vector of the key joints. We design our loss function as:

$$L = 0.2 * (1 - \text{Cos_Similarity}(y_{pred}, y_{true})) + 0.8 * \text{MSE}(y_{pred}, y_{true})$$

to increase the weight of the accuracy of velocity direction prediction relative to the accuracy of velocity magnitude prediction because, in target selection tasks, people are more sensitive to inconsistent directions. We implement the above model in Python based on the PyTorch framework, with a batch size of 32, and train the model to the best using cross-validation.

6.4 Physics-constrained velocity correction

The predicted velocities of key points from the ML model are independent. The independency leads to inconsistency in velocities among key points. The predictions between consecutive frames are also independent, which leads to instability in the predicted velocity.

The attitude of the skeletons is a slowly changing and relatively stable quantity. Using it to establish physical constraint relationships can help establish connections between predicted velocities of different joints among frames. Our idea of velocity correction is to quantify the degree of conformity between the current predicted velocity and the physical constraints with a confidence score. The

confidence score serves as a weight to update the correction value of the current velocity. The current velocity combines with the historical one to achieve smoother and more stable velocity prediction. We find the following physics constraints to be effective.

Directional consistency: Confidence score C_1 characterizes the consistency of the velocity directions of different joints (Conclusion 6 in 5.2.6). If the consistency is greater than a threshold, the confidence score of the predicted value will be reduced.

$$\begin{cases} \alpha_1 = \arccos\langle \hat{\mathbf{v}}_{\text{nailtip}} \| \mathbf{A}, \hat{\mathbf{v}}_{\text{DIP}_n} \| \mathbf{A} \rangle \\ \alpha_2 = \arccos\langle \hat{\mathbf{v}}_{\text{DIP}_n} \| \mathbf{A}, \hat{\mathbf{v}}_{\text{PIP}_n} \| \mathbf{A} \rangle \\ \alpha_3 = \arccos\langle \hat{\mathbf{v}}_{\text{nailtip}} \| \mathbf{A}, \hat{\mathbf{v}}_{\text{PIP}_n} \| \mathbf{A} \rangle \\ C_1 = \frac{1}{3} \sum_{i=1}^3 \mathbb{I}(\alpha_i \leq \alpha_{i_{\text{thres}}}) + \frac{1}{3} \sum_{i=1}^3 \mathbb{I}(\alpha_i > \alpha_{i_{\text{thres}}}) \cos^2(\alpha_i) \end{cases}$$

$\langle \mathbf{v}_1, \mathbf{v}_2 \rangle$ represents the dot product. Hat denotes the unit vector. \mathbf{A} is the given horizontal/vertical plane where the index finger is sliding. $\hat{\mathbf{v}}_{\text{Joint}_n} \| \mathbf{A}$ is the projection of the velocity vector on plane \mathbf{A} . The values of $\alpha_{i_{\text{thres}}}$ are also referenced from 5.2.6. $\alpha_{1_{\text{thres}}} = 13^\circ$ $\alpha_{2_{\text{thres}}} = 15^\circ$ $\alpha_{3_{\text{thres}}} = 30^\circ$

Co-planarity: Confidence score C_2 characterizes co-planarity. The four key points (NailTip, DIP, PIP, and MP) are always on the same plane (Conclusion 2 in 5.2.2). Therefore, their instantaneous normal velocity vectors relative to their common plane are also coplanar. We have:

$$C_2 = (\hat{\mathbf{v}}_{\text{DIP}_n} - \hat{\mathbf{v}}_{\text{PIP}_n}) \times (\hat{\mathbf{v}}_{\text{PIP}_n} - \hat{\mathbf{v}}_{\text{MP}_n}) \cdot (\hat{\mathbf{v}}_{\text{nailtip}_n} - \hat{\mathbf{v}}_{\text{DIP}_n})$$

Hat denotes the unit vector. The subscript “n” represents the component of the vector in the normal direction.

Length consistency: Confidence score C_3 characterizes length consistency. The points on the same bone have equal instantaneous radial velocities due to the rigid body constraint (Conclusion 3 in 5.2.3).

$$C_3 = \min\left(\frac{\mathbf{v}_{\text{MP}} \| \mathbf{L}_1}{2\mathbf{v}_{\text{DIP}} \| \mathbf{L}_1}, \frac{\mathbf{v}_{\text{DIP}} \| \mathbf{L}_1}{2\mathbf{v}_{\text{MP}} \| \mathbf{L}_1}\right) + \min\left(\frac{\mathbf{v}_{\text{DIP}} \| \mathbf{L}_2}{2\mathbf{v}_{\text{PIP}} \| \mathbf{L}_2}, \frac{\mathbf{v}_{\text{PIP}} \| \mathbf{L}_2}{2\mathbf{v}_{\text{DIP}} \| \mathbf{L}_2}\right)$$

\mathbf{L}_1 and \mathbf{L}_2 are the directional vectors of the intermediate and proximal phalanx. They are calculated from the attitude angles of the key points in the attitude estimation.

The overall confidence score, denoted as C , is calculated as the product of individual constraint confidence scores C_i . As different constraint equations have varying refinement effects on the accuracy of velocity estimation, we introduce an exponent parameter for each C_i , which is explored by traversing all possible values to achieve the global optimum for velocity prediction. The final confidence score and velocity correction formula are as follows. $\tilde{\mathbf{v}}^t$ is the model's predicted value at time t , and \mathbf{v}^t is the correction value.

$$\begin{cases} C = C_1^3 \cdot C_2^2 \cdot C_3 \\ \tilde{\mathbf{v}}_{\text{nailtip}}^t = C \cdot \mathbf{v}_{\text{nailtip}}^t + (1 - C) \cdot \tilde{\mathbf{v}}_{\text{nailtip}}^{t-1} \end{cases}$$

6.5 Touch State Detection

To enable tracking the sliding of the finger on the physical surface, the system needs to detect in real-time whether the user's index finger is in contact with the surface to exclude the case where the finger is hovering in the air. We refer to previous works[18], which achieved a 99% accuracy in touch-down event detection. For each axis of the 6-axis IMU, we computed the maximum, minimum, mean, skewness, and kurtosis within a 10-frame time window. These

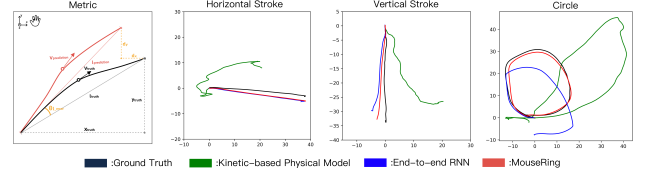


Figure 6: The evaluation metrics and visualization of trajectory prediction using different model settings.

features, totaling 30 dimensions, were used to classify the data from each time window into four categories: touch-down, touch-up, in-air, and touching, using an SVM model.

We then employed a state machine to detect the touch state in real-time. When the touch state is false, if the touch detector detects a touch-down event or continuous touching for five consecutive time windows (with 80% overlap between adjacent windows), the touch state transitions to true. Conversely, when the touch state is true, if the touch detector detects a touch-up event or continuous in-air state for five consecutive time windows, the touch state transitions to false. We conducted tests on our dataset. 95.5% of finger-sliding interactions were accurately identified in their entirety. When participants' fingers continuously interacted during the user experiment, only 11.8 seconds per hour of fingertip in-air state were mistakenly recognized as touch state. The algorithm demonstrated robust performance as a switch for touch input.

6.6 Scaling the Velocities in Different Directions

For right-handed users, sliding towards the lower left and upper right is more effortless than sliding towards the upper left or lower right because the natural rotation of the right hand around the wrist causes the fingers to move in these two directions. To address this issue, we amplify the speed amplitude of the more difficult sliding directions. With this optimization, users have a similar subjective sliding experience when moving the cursor in all directions.

6.7 Cursor Filtering

We further implement the 1Efilter[8] for the corrected velocity, which performs well in smoothing mouse input. The filter has two parameters: the minimum cutoff frequency $f_{C_{min}}$ and the speed coefficient β . Reducing the minimum cutoff frequency will reduce slow speed jitter, while increasing the speed coefficient will reduce speed lag. We select $f_{C_{min}} = 0.004$ and $\beta = 0.08$ to filter the corrected mouse displacement in the model calculation.

7 SIMULATION WITH OFFLINE DATA

In this section, we use offline data from the finger-sliding dataset to simulate the algorithm's performance under different settings and primarily evaluate its effectiveness. We address three research questions in the following subsections:

RQ1: Does fixing some parts of the hand improve prediction accuracy in RTM and RP modes?

RQ2: What level of accuracy can be achieved under single-ring and double-ring configurations?

RQ3: How is MouseRing's performance compared to RNN and finger-kinematics-based methods?

7.1 Simulation Set-up

We utilized data from all 12 participants, trained the model using the leave-one-out cross-validation method, and predicted the velocity within each time window for each user during the uni-stroke process of finger-sliding. By integrating the velocities, we simulated the predicted fingertip sliding trajectory. This section compared the predicted trajectories against the ground truth trajectories from Optitrack using the following five metrics in Figure 6:

- θ_{lerror} : The angle between the real fingertip displacement l_{truth} and the predicted displacement $l_{prediction}$. It represents the accuracy of tracking the direction of fingertip motion over a period of time.
- θ_{verror} : The angle between the real instantaneous fingertip velocity v_{truth} and the predicted velocity $v_{prediction}$. It represents the accuracy and stability of tracking the direction of fingertip motion.
- l_{error} : The relative error between the fingertip displacement l_{truth} and the predicted displacement $l_{prediction}$.
- x_{error}, y_{error} : We define the left-right movement of the fingertip as along the x-axis and forward-backward (horizontal plane)/up-down (vertical plane) movement of the fingertip as along the y-axis. The absolute errors in the x and y directions between the real trajectory and the predicted trajectory are denoted as dx and dy. x_{error} and y_{error} are the relative errors of dx and dy with respect to l_{truth} .

Table 2 presents the simulation results obtained under different datasets, ring configurations, and model settings. For the MouseRing algorithm, we separately trained the models using three different finger-sliding modes' data. For simulation groups without annotated datasets, we report the average error of the three action modes (RW, RTM, RP). We also compared the effect of ring position and quantity on tracking accuracy by training the models using single-ring and dual-ring data.

As a baseline algorithm, we replicated the kinetic-based model from AnywhereTouch[47], an existing work that supports finger tracking based on finger-worn IMUs. It uses the change in pitch angle and the relation formula $\theta_{DIP} = \frac{2}{3}\theta_{PIP}$ to estimate forward and backward displacement. It predicts the left and right movement by mapping changes in yaw angle to the displacement. Furthermore, we conducted a simple ablation study where we removed the physical-constrained correction from the MouseRing algorithm and evaluated the performance of the end-to-end RNN. Additionally, during the training process of the RNN model, we selectively masked the components of each axis of the 6-axis IMU to assess the utility of each axis.

7.2 Finger-sliding Modes

We separately trained models for the three sliding modes (Rested Wrist, Rested Thumb & Middle Finger, and Rested Palm). We initially expected that the RP and RTM modes, which have stronger hand constraints and fewer degrees of freedom, would be more accurately predicted. However, the simulation results only partially met our expectations.

Considering both the wearing & input experience and fingertip tracking accuracy, RW (Rested Wrist) is the best input mode for overall interaction. Among the three modes, the RP mode has the

Table 2: Performance of the model under different settings.

| | θ_{lerror} | θ_{verror} | l_{error} | x_{error} | y_{error} |
|-------------------------------------|-------------------|-------------------|----------------|---------------|----------------|
| Dual ring (RW dataset) | 7.51° | 15.66° | 14.70mm | 7.16mm | 12.84mm |
| Dual ring (RTM dataset) | 7.36° | 14.51° | 13.97mm | 6.78mm | 12.21mm |
| Dual ring (RP dataset) | 5.34° | 13.97° | 13.60mm | 6.39mm | 12.00mm |
| Proximal single ring | 12.33° | 22.56° | 14.89mm | 7.87mm | 12.64mm |
| Intermediate single ring | 13.23° | 28.04° | 24.01mm | 8.91mm | 22.30mm |
| Dual ring | 6.61° | 14.53° | 14.08mm | 6.80mm | 12.33mm |
| Dual ring (kinetic-based model) | 36.64° | 32.38° | 50.27mm | 29.43mm | 40.75mm |
| Dual ring (end-to-end RNN) | 8.75° | 32.80° | 13.94mm | 6.44mm | 12.36mm |
| Dual ring (RNN, a_x removed) | 12.10° | 34.22° | 14.69mm | 8.01mm | 12.31mm |
| Dual ring (RNN, a_y removed) | 8.69° | 32.17° | 14.09mm | 6.82mm | 12.33mm |
| Dual ring (RNN, a_z removed) | 9.48° | 33.49° | 16.47mm | 7.68mm | 14.57mm |
| Dual ring (RNN, ω_x removed) | 17.01° | 32.76° | 28.11mm | 6.97mm | 27.24mm |
| Dual ring (RNN, ω_y removed) | 8.74° | 32.92° | 13.91mm | 6.88mm | 12.09mm |
| Dual ring (RNN, ω_z removed) | 15.13° | 35.75° | 20.39mm | 11.70mm | 16.70mm |

smallest θ_{lerror} of 5.34°, while RW and RTM have similar θ_{lerror} of 7.51° and 7.36°. On the one hand, fixing the entire palm on the surface does make the prediction more accurate. However, the improvement in accuracy is less significant compared to the loss of interaction comfort. On the other hand, while placing the thumb and index finger on the surface causes less loss of interaction comfort, it does not provide much help to the model's prediction. We conclude that the RW mode allows for natural input and maintains accuracy similar to the other two input modes.

7.3 Ring Number and Position

The accuracy of the single-ring configurations is lower than the double-ring configuration, with displacement angle errors of 13.23° and 12.33°, respectively, compared to the 6.61° of the double-ring sensor configuration. For the proximal phalanx ring, y_{error} increased significantly from 12% to 22.3%, resulting in a l_{error} of 24.01%. This can be attributed to the fact that, due to its longer distance from the fingertip, the proximal ring exhibits smaller variations in orientation when the fingertip moves forward or backward. Thus, the information from the IMU is not sufficient to predict the y-direction displacement.

In conclusion, for the single-ring configuration, it is more suitable to wear the ring on the intermediate phalanx. Despite the decreased accuracy, we argue the accuracy is enough for non-fine-grained target selection. In daily-life scenarios, the single-ring configurations are more comfortable to use due to their lighter wear.

7.4 Machine Learning vs. Kinematic-based Modelling

The kinematic-based method has a significant systematic bias in the prediction direction for two reasons. Firstly, the $\theta_{DIP} = \frac{2}{3}\theta_{PIP}$ relationship does not hold when the finger slides, leading to a highly inaccurate forward and backward displacement prediction. Secondly, differences in the wearing position of the ring, finger length, and finger sliding habits among different users can result in significant prediction errors during angle mapping. RNN model can learn the sliding action mode well, achieving a θ_{lerror} of 8.75° and a l_{error} of 13.94%. However, its θ_{verror} reaches 32.8°, indicating

that the instantaneous speed is very inaccurate on occasion. After incorporating the physical constraints into the MouseRing algorithm, the physical constraints reject inferior speed prediction. The θ_{error} is significantly reduced to 14.53° . The θ_{error} also benefits from more stable velocity prediction. The general results demonstrate that physics-based knowledge can assist machine learning, making finger-sliding predictions more stable and accurate.

By comparing models with and without removing 6-axis features, we also analyzed the useful information provided by each axis of the IMU sensor. a_x and ω_z contribute significantly to the prediction of displacement in the x-direction. We attribute this to the integration of x-axis acceleration related to x-axis displacement. Also, the z-axis angular velocity strongly correlates with fingertip movement when the palm rotates around the wrist. After removing these features from the model, the error in x-axis displacement increased from 6.80% to 8.01% and 11.70%.

Surprisingly, for the prediction of y-axis displacement, the acceleration in the y-direction is not the most important. Instead, a_z , ω_x , and ω_z contribute more. These features determine the angle between the index finger phalanges and their respective postures. This indicates that the y-axis displacement of the fingertip is mainly influenced by finger bending and changes in hand posture rather than the translational motion of the index finger. Even though removing information from other axes did not significantly increase the errors in the x or y-directions, θ_{error} still increased significantly. All 6-axis data is helpful for accurately predicting the velocity.

7.5 Summary

For RQ1, we find that the prediction accuracy of different motion modes is similar, thus rejecting the previous hypothesis. RW (Rested Wrist) is an input mode that balances free interactive motion and good accuracy. For RQ2, while the double-ring configuration ($\theta_{error}=6.61^\circ$) can achieve higher prediction accuracy, the single-ring configuration still has considerable prediction performance ($\theta_{error}=12.33^\circ$). It potentially supports simple cursor control tasks in mobile scenarios. For RQ3, a pure RNN can learn the pattern of finger motion well, while physical knowledge connects the estimated finger attitude with velocity predictions and refines the prediction results. The combination of the two can achieve high-precision and stable trajectory prediction.

8 LAB ENVIRONMENT FITTS' LAW STUDY

We conducted two studies to evaluate the MouseRing device. The first study was a Fitts' Law experiment conducted in a controlled laboratory environment, aiming at assessing the input efficiency of MouseRing under ideal conditions. The second study was conducted in a real-world large-screen interaction scenario, allowing us to evaluate the usability and robustness of MouseRing in practical settings.

In the Fitts' law study, we compared MouseRing with two baseline input methods commonly used for cursor control in target selection tasks: laptop touchpads and air mice commonly used for controlling presentation slides remotely. We recorded the mean selection time and plotted a graph to depict the time-difficulty relationship according to Fitts' Law. We answer the following questions: **RQ4: How does the input efficiency of MouseRing compare to the baseline?**

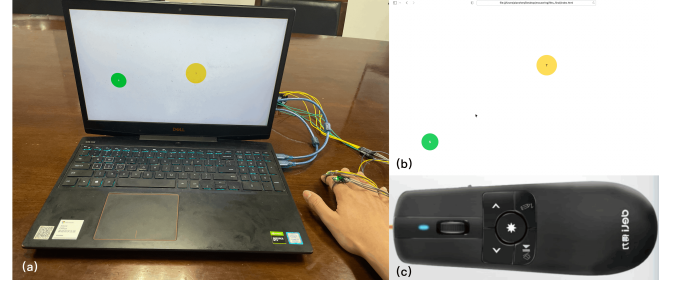


Figure 7: (a) The setup for Fitts' Law study. (b) The GUI of Fitts'-like target selections. (c) Deli2803 flying mouse.

8.1 Input Methods

We compared three input methods. In addition to MouseRing, we chose two target selection methods based on the mouse selection paradigm as baselines. For MouseRing, we tested both double-ring and single-ring setups.

- **TouchPad:** Participants use a laptop's touchpad, the golden standard for controlling cursor movement for target selection.
- **AirMouse:** Air Mouse, also known as a gyroscopic remote controller, enables anywhere-available cursor control. Participants hold the air mouse in their hands and move it in the air. The gyroscope senses the movement and maps it to the cursor's movement.
- **MouseRing (double-ring):** Users wear two rings, sit in front of the screen, and use their index fingers to slide on the desktop to control the cursor.
- **MouseRing (single-ring):** Participants wear a ring on the intermediate phalanx, sit in front of the screen, and use their index fingers to input.

8.2 Apparatus

We ran the Fitts Law study's JavaScript program on a Dell G3 laptop. The laptop's touchpad was used as the TouchPad baseline. We used the Deli 2803 flying mouse as the AirMouse input device. For MouseRing, participants wore the same prototype device as in the data collection section. We removed the display-to-control ratio of the Windows system by reverse-engineering[7]. In this way, we eliminated the potential impact of the display-to-control ratio on different input methods.

8.3 Participants

We recruited 12 participants (6 females, aged 20 to 25, $M = 24.0$) from the campus. All participants were right-handed. All participants were very familiar with TouchPad input. Four participants had previous experience with Air Mouse. None had used MouseRing before.

8.4 Design and Procedure

The experiment was based on a Fitts' Law target selection GUI (Fig.7(b)). Each time, two yellow buttons, S (Start) and E (End), appeared on the screen. The buttons were randomly generated with diameters ranging from 12mm to 30mm and distances ranging

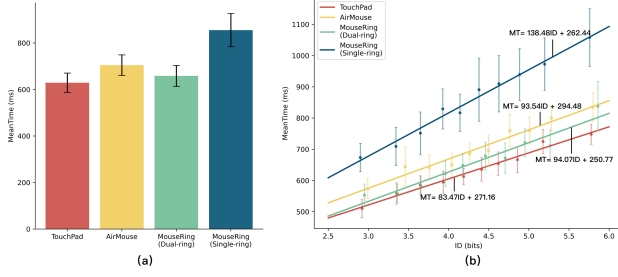


Figure 8: (a) The mean selection time of different input methods. (b) The linear relationship between MeanTime and ID.

from 90mm to 400mm. Participants first moved the cursor to the Start button. After 1 second, the Start button turned green. They then moved the cursor to the End button. After a 250 ms pause, the End button turned green, indicating the completion of a target selection. The time interval between the color changes of the two buttons was recorded as the selection time.

Each participant had to complete the task under four different input methods or configurations. Participants had to complete 5 rounds * 10 times/round = 50 selections for each input method. Before using each input method, participants could practice freely until they felt they had mastered the input method. Each participant performed the tasks in a different order to eliminate the effects of fatigue and learning.

8.5 Results

We used linear regression in the Fitts' Law study to fit the relationship between the average selection time and index of difficulty ($ID = \log_2 \frac{2D}{W}$). We ran one-way RM-ANOVA and Friedman tests for different input methods to test the significance of differences between the average time.

The Friedman test revealed significant differences in the mean selection time between different input methods ($\chi^2 = 26.51, p < 0.001$). The input efficiency of dual MouseRing ($MT = 658.1ms, STD = 45.1ms$) was only slightly slower than TouchPad ($MT = 629.1ms, STD = 41.5ms$). In contrast, AirMouse and single MouseRing were significantly slower than TouchPad ($F = 5.9, 12.2, 15.0, p < 0.05$). We found that the double-ring configuration of MouseRing was a fast, anywhere-available input method compared to AirMouse.

Furthermore, we fitted the relationship between MeanTime and ID, where the slope of the line represents the cursor movement rate, and the intercept reflects the target-locking speed (Fig.8(b)). TouchPad ($k = 83.47$) had the fastest cursor movement speed, followed closely by MouseRing ($k = 94.07$) and AirMouse ($k = 93.54$). The model's accuracy decreased in the single-ring configuration and decreased cursor movement speed. AirMouse had the largest intercept because the participant's suspended hand was prone to shaking, making target locking difficult.

For RQ4, we found that the MouseRing with a double-ring configuration can achieve comparable input speed (629 ms vs. 658 ms) to the TouchPad and is faster than the anywhere-available baseline (AirMouse). With a more lightweight wearing experience,

as a tradeoff, the MouseRing in the single-ring configuration exhibits approximately 20% higher completion times compared to the touchpad. Participants reported that target selection becomes challenging when the targets are small. Therefore, in real-world tasks, it is necessary to investigate how much MouseRing can support fine-grained target selection tasks in the single-ring configuration.

9 REAL-WORLD SCENARIO EVALUATION

The Fitts' Law study conducted in a lab setting demonstrated that MouseRing achieves input efficiency comparable to a touchpad in controlled environments. Our final study was conducted in a large-screen real-world interaction scenario. We evaluated the usability and robustness of MouseRing technology in real-world applications across various physical surfaces and different body postures. We also investigated the limit of sensing precision under single-ring and dual-ring configurations. We addressed the following research questions:

RQ5: How does MouseRing's input efficiency vary with different softness, hardness, and flatness levels of input surfaces, as well as different user body postures during the interaction?

RQ6: To what extent can MouseRing support fine-grained target selection tasks in the single-ring and dual-ring configurations?

RQ7: Does MouseRing provide better comfort for wearing/carrying and achieve a workload better than the baselines?

9.1 Setup

MouseRing has the potential advantage of providing always-available interaction, so we asked participants to complete target selection tasks in both standing and sitting postures on different surfaces. We evaluated the usability of MouseRing in both dual-ring and single-ring configurations on four different surfaces. The desktop, sofa, wall, and thigh surfaces cover different plane orientations, hardness, and flatness levels. The desktop and wall are hard and flat surfaces, while the sofa and thigh are soft and uneven. Participants interacted with the desktop and sofa (horizontal surfaces) while sitting and with the wall and thigh (vertical surfaces) while standing.

We chose the mouse and AirMouse as baselines, with the mouse replacing the touchpad used in the lab-condition study, as it would be difficult for participants to constantly hold a touchpad while standing. Participants used the mouse on a desktop and wall surface when sitting and standing, respectively. We conducted 12 within-subject studies (2 Mouse + 2 AirMouse + 4 Dual-Ring + 4 Single-Ring), with the input method, posture, input surface, and ring number as factors.

9.2 Apparatus

We used the Samsung UA65JU5900JXXZ as the large-screen device. The screen size is 65 inches, with participants inputting from a distance of 3-5m. The user experiment script and GUI were run on a laptop and projected onto the screen via an HDMI cable. We used the Logitech M186 as the mouse device and the Deli 2803 flying mouse as the AirMouse input device.

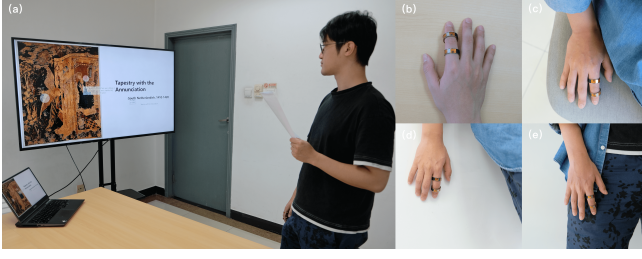


Figure 9: (a) The setup for the large-screen interaction user study. (b)-(e) The wireless version of the MouseRing prototype. We tested its performance on different surfaces: desk(b), sofa(c), wall(d), and thigh(e).

We integrated the Bluetooth and IMU sensor modules into a small wireless ring to provide a more realistic wearing and usage experience closer to real-life scenarios (Figure 9). This ring has the same sensing modules as the previous prototype device and sends data to the computer via Bluetooth at 200 Hz. A more detailed design of the wireless ring is presented in Appendix B.

9.3 Participants

We recruited 12 participants from the campus (7 females, aged 18 to 25, $M=22.4$). All participants were right-handed. All participants were very familiar with the mouse device. None had previous experience with either the AirMouse or MouseRing before.

9.4 Design & Procedure

We designed a user experiment for large-screen device interaction. Participants played the role of a museum guide and used the large-screen device to introduce exhibits to visitors. Participants were required to sequentially move the mouse and select buttons on the large screen, controlling the detailed descriptions of the exhibits to pop up individually and then read them aloud. Afterward, the participants clicked the page-turning button and continued introducing the following exhibit. Each page contained five buttons that could trigger events. The size and position of the buttons on the page were designed in advance to cover different sizes and distances. We provided the participants with a script to guide them on the order of clicking buttons and reading text. Screenshots of the interactive pages are presented in Figure 13 in Appendix C.

Before the experiment began, participants had 5 minutes to learn and familiarize themselves with controlling the cursor using the AirMouse and MouseRing. Each participant needed to complete one round of the experiment under 12 sets of settings (2 Mouse + 2 AirMouse + 4 Dual-MouseRing + 4 Single-MouseRing). Under each setting, participants completed 10 pages * 5 times/page = 50 selections. We recorded the time taken for each target selection, as well as the distance to and size of the target. Each participant completed the tasks in a different order to eliminate the effects of fatigue and learning. After the experiment, participants filled out a subjective questionnaire and briefly talked about their feelings. The experiment lasted approximately two hours.

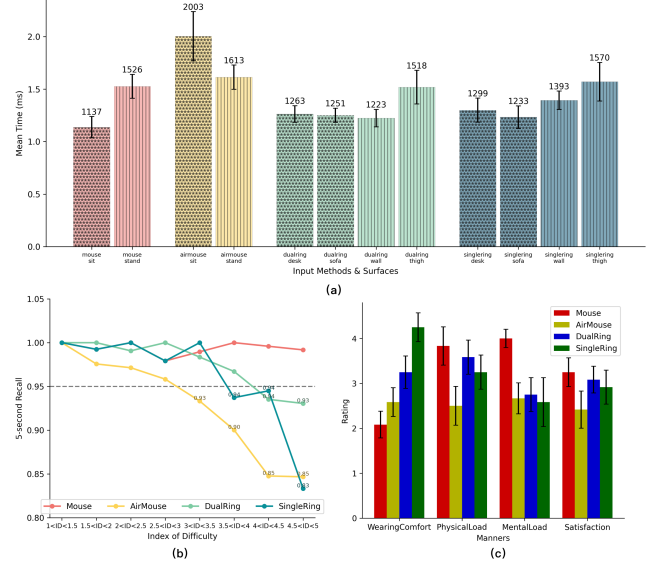


Figure 10: (a) The mean selection time of different input methods, ring numbers, surfaces, and body postures. (b) The 5-second recall of different methods when ID(index of difficulty) increases. (c) The subjective ratings for different input methods.

9.5 Results

We ran one-way RM-ANOVA and Friedman tests for different input settings to test the significance of differences between the average time. The significance of subjective ratings was tested using the Mann-Whitney-Wilcoxon rank test.

9.5.1 Input Method. Participants were able to utilize MouseRing effectively for target selection in real-world contexts. Although a mouse remains the fastest input method in a seated posture, the speed advantage over MouseRing is insignificant. On the other hand, in both dual-ring and single-ring configurations, MouseRing significantly outperformed the mouse device while standing across three planes (desk, sofa, wall) for dual-ring ($p < .01$, $F_{1,22} = 9.9, 11.9, 12.7$) and two planes (desk, sofa) for single-ring ($p < .01$, $F_{1,22} = 5.5, 9.7$). Moreover, MouseRing surpassed the AirMouse in terms of input efficiency on three out of four tested planes (desk, sofa, wall) ($p < .01$, $F_{1,22} = 24.5, 26.1, 26.9$).

9.5.2 Body Posture. The participant's posture significantly influenced the effectiveness of two baseline input methods ($p < .05$, $F_{1,22} = 17.8, 6.1$). Participants reported fatigue when using a mouse while standing or moving and encountered difficulty accurately manipulating the AirMouse in a seated position with limited body mobility. Contrastingly, no efficiency disparity was observed between MouseRing interactions on a desk (seated) and a wall (standing). These findings suggest that MouseRing is optimally suited for interactions in mobile contexts due to its always-available nature.

9.5.3 Interacting Surface. The desk, being a horizontal and rigid surface, mimics an interaction environment akin to a laboratory

setting. Input speeds on vertical hard planes (wall) or horizontal soft planes (sofa) were comparable to those on a desk. We concluded that MouseRing demonstrated robustness and rapid response in supporting inputs on surfaces with varying orientations and hardness levels. However, interaction on the thigh was significantly slower than on the desk ($p < .05$, $F_{1,22} = 5.7$), attributable to the irregularities caused by clothing wrinkles and the inherent curvature of leg muscles, which rendered the surface uneven during movement, thereby affecting fingertip tracking accuracy and decelerating selection speed.

9.5.4 Ring Number. The dual-ring configuration exhibits a slightly faster average time compared to the single-ring configuration, but the difference is not significant. To ascertain the precision limits of MouseRing in fine-grained target selection, we computed the proportion of successful button clicks within a five-second window (5-second-recall) across different difficulty levels ($ID = \log_2 \frac{D}{W}$). Following Fitts' law, while increased difficulty leads to protracted selection time, selections exceeding 5 seconds suggest that the participant had to make secondary cursor adjustments during that selection.

The 5-second recall for a mouse remained stable at over 98%. The dual-ring and single-ring configurations of MouseRing maintained accuracy rates of 97% and 100%, respectively, when the difficulty was less than 4 and 3.5, but these rates declined precipitously after that. For smooth selection, we recommend that the angle between the selection target and the cursor be less than 3.81° for dual-ring and 5.54° for single-ring (corresponding to $ID=4, 3.5$).

9.5.5 Subjective Ratings. MouseRing exhibits superior comfort for prolonged use. The single-ring configuration was significantly more comfortable than the dual-ring configuration ($p < .05$, $Z = -2.14$). Both configurations outperformed the mouse and AirMouse in comfort ($p < .05$, $Z = -2.63, -1.54, -3.21, -3.03$), as participants did not need to grasp any object in their hand during the extended user experiments. The physical load associated with MouseRing interaction was similar to that of a mouse and significantly lower than that of the AirMouse ($p < .05$, $X = -2.65, -2.13$). The mental load of using MouseRing and AirMouse was higher than using a mouse, although the difference was insignificant. AirMouse and MouseRing experienced tracking errors, resulting in minor discrepancies between the actual cursor movement and the participant's anticipated movement. Lack of prior experience with these devices could also increase mental load. Finally, MouseRing achieved a satisfaction level comparable to that of a mouse. Participants perceived MouseRing as a more natural and satisfactory input method ($p < .05$, $Z = -1.99$) than AirMouse.

9.5.6 Subjective Feedback. The participants found that MouseRing allowed them to input more comfortably. Participant 7 said, "My hand does not need to reach towards the middle of the desk, but can input at the edge." Several participants looked forward to using MouseRing to control devices that were further away and to control the cursor in more relaxed postures, such as lying down or sitting back in a chair. Additionally, some participants also mentioned that MouseRing could be applied in scenarios like stage performances and presentations that require discreet and subtle interactions. Participants 2, 6, and 7 all felt that the physical surface interaction

provided a more grounded feeling than AirMouse. They could stop sliding at any time by lifting their fingertips. In contrast, AirMouse required the hand to be suspended in the air for a long time, did not support hovering during cursor movement, and might cause sustained fatigue. It's also worth noting that participants' experiences with MouseRing evolved over time. Participant 3 said, "I need more practice time to get faster." We analyzed the selection time for the first ten and last ten selections and found that the average speed of the participants increased by 4%, confirming the existence of a learning effect.

10 APPLICATION

We have identified three application scenarios for MouseRing and implemented a range of interactions to highlight its always availability.

10.1 AR/VR Input

In scenarios involving visual occlusion and mobile interactions, MouseRing facilitates user manipulation of graphical interfaces in AR/VR environments. Unlike hand-held controllers, MouseRing is considerably smaller and can be worn daily. Additionally, MouseRing does not rely on HMDs to be equipped with cameras, offering a low-power sensing solution that can potentially reduce the weight and size of future HMDs. We have implemented two VR applications: a voice and video calling application (Fig 11(a)) and a video player (Fig 11(b)). These applications utilize touchpad interactions supported by MouseRing to enable convenient cursor control in VR for button selections.

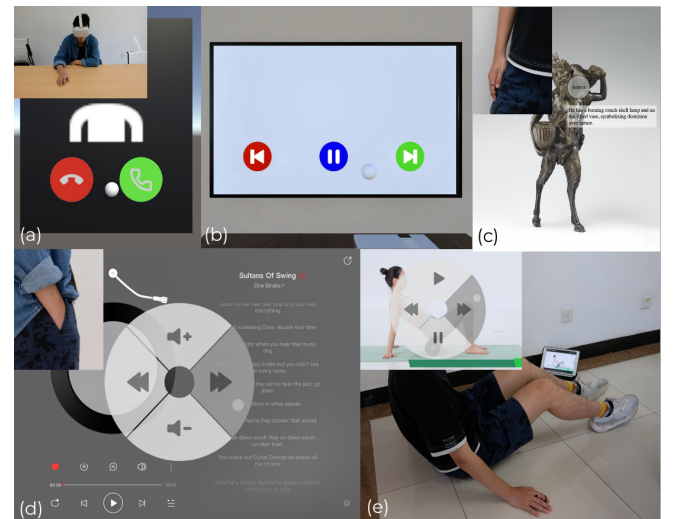


Figure 11: (a) MR video voice calling (b) VR video player. (c) Slider control on the thigh. (d) In-pocket subtle input for player volume adjustment. (e) Input through FaST Slider during yoga.

10.2 Large screen Displays

In conference and smart home scenarios, MouseRing can serve as a substitute for remote controllers and air mice to control projectors, smart TVs, and large-screen displays efficiently. We have implemented a series of cursor interactions and shortcut commands based on continuous mouse control (Fig 11(c)). Speakers can utilize MouseRing to highlight key points in presentation slides and switch content, even while standing or walking away from the screen.

10.3 Mobile and Sports Scenarios

In mobile or sports scenarios, we designed FaST Sliders[42], enabling MouseRing to support sending shortcut commands by sliding in different directions on any surface. During sports, users are constrained in body posture and input capabilities. In commuting scenarios, MouseRing facilitates subtle and rapid input. MouseRing imposes minimal physical effort demands and liberates users from needing additional devices. Based on MouseRing's FaST Sliders interaction, we have implemented the control of a music player and remote control functionality for a fitness instructional video app on a tablet (Fig 11(d),(e)).

11 DISCUSSION

11.1 Stronger Sensing Capability with Physically-informed Models

Utilizing IMU sensors for body tracking is challenging, not solely due to sensor noise leading to inaccurate pose estimation but also because sparse pose information is inadequate to recover the full spectrum of body movements. In our work, additional prior information can benefit finger tracking by providing helpful information gain in hand kinetics for ML models. We contend that the inherent knowledge within the physical structure and movement patterns of the human body can be modeled as priors, assisting in achieving robust perception from weak sensor signals. Similar body constraint modeling has already been employed in full-body pose estimation studies. We hope our approach can inspire more work in HCI to develop novel motion and behavior recognition techniques.

11.2 Towards Personalized Online Calibration

Although there are many common physical laws governing the movement of index fingers, the differences in finger length ratios and movement styles among individuals are difficult to avoid. Setting hyperparameters in the model to represent these individual differences and dynamically learning these parameters during user use can effectively improve the recognition accuracy of MouseRing. One possible approach is to fit the predicted fingertip trajectory of the model and the line between the initial cursor position and the user's following selected target (ground truth of trajectory). Compared to a series of calibrations in advance, online personalized calibration is iterated in the background without occupying additional user attention and achieves better results with the larger amount of online data.

11.3 Sensing Ability

Due to the interference of indoor magnetic fields, we utilized signals from a 6-axis IMU sensor and abandoned potentially helpful information from the magnetometer. This approach was feasible for

our research, as each user experiment lasted only about 10 minutes. After that, the initial attitude of the IMU was recalibrated to eliminate the impact of attitude angle drift. For long-term continuous use, the MouseRing algorithm also needs to be improved. Magnetometer information can complement accelerometer information in a clean outdoor space. For indoor circumstances, prior estimation of spatial magnetic fields could enhance the sensing capability of MouseRing. In the experiment, signals were collected at a frequency of 200 Hz. This frequency is sufficient for fingertip motion tracking tasks. However, a higher sampling frequency[35] may improve the accuracy of touch-down and touch-up event detection, as these events utilize the frequency domain characteristics of the signal. Nevertheless, there is a trade-off between the ring device's sensing capability and power consumption.

11.4 Long-term Wearing & Remounting

Although a single-ring setup can support simple input interactions, higher precision and accuracy of MouseRing sensing require wearing two rings. The long-term wearing of a single ring on the intermediate phalanx may cause slight discomfort. One possible solution is to wear the intermediate phalanx ring on the base of the other fingers for improved comfort during daily wear. The MouseRing requires users to position the IMU sensor on the backside of the finger when wearing the ring. Still, we did not explicitly require precise angle alignment during data collection or user evaluations. Participants also did not report any noticeable impact on accuracy due to the remounting process, which assures position adjustments while using MouseRing. The wearing status can be recognized by calculating the relative posture between the rings. Various dual-ring wearing methods among different fingers can potentially provide independent and richer input methods.

12 CONCLUSION

We present MouseRing, a ring-shaped IMU device that accurately tracks fingertip movements and enables continuous cursor control. Through data analysis, we identified several physical constraints that govern the sliding process of the index finger. We have achieved high-precision fingertip tracking by combining physical prior with machine-learning methods, with a remarkable mean angular error of 6.61° . We believe that leveraging the inherent knowledge embedded within the physical structure and movement patterns of the human body can enhance the perceptual capabilities of IMU sensors. In a lab evaluation, the dual MouseRing demonstrated input efficiency comparable to a TouchPad. In real-life tasks, both the single and dual MouseRing devices exhibited robust and swift 2D cursor control on surfaces of varying hardness and flatness and in standing and sitting postures. MouseRing holds immense potential for various applications, including AR/VR, large display interaction, IoT, commuting, and sports scenarios.

REFERENCES

- [1] Subatai Ahmad. 1994. A usable real-time 3D hand tracker. In *Proceedings of 1994 28th Asilomar Conference on Signals, Systems and Computers*, Vol. 2. IEEE, 1257–1261.
- [2] Teo Babic, Harald Reiterer, and Michael Haller. 2018. Pocket6: A 6dof controller based on a simple smartphone application. In *Proceedings of the 2018 ACM Symposium on Spatial User Interaction*. 2–10.

- [3] Marc Baloup, Thomas Pietrzak, and G ry Casiez. 2019. Raycursor: A 3d pointing facilitation technique based on raycasting. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–12.
- [4] Sandra Bardot, Bradley Rey, Lucas Audette, Kevin Fan, Da-Yuan Huang, Jun Li, Wei Li, and Pourang Irani. 2022. One Ring to Rule Them All: An Empirical Understanding of Day-to-Day Smartring Usage Through In-Situ Diary Study. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 3 (2022), 1–20.
- [5] Sebastian Boring, Marko Jurmu, and Andreas Butz. 2009. Scroll, tilt or move it: using mobile phones to continuously control pointers on large public displays. In *Proceedings of the 21st Annual Conference of the Australian Computer-Human Interaction Special Interest Group: Design: Open 24/7*. 161–168.
- [6] Alexander Buryanov and Viktor Kotiuk. 2010. Proportions of hand segments. *Int. J. Morphol* (2010), 755–758.
- [7] G ry Casiez and Nicolas Roussel. 2011. No more bricolage! Methods and tools to characterize, replicate and compare pointing transfer functions. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*. 603–614.
- [8] G ry Casiez, Nicolas Roussel, and Daniel Vogel. 2012. 1  filter: a simple speed-based low-pass filter for noisy input in interactive systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2527–2530.
- [9] Ke-Yu Chen, Shwetak N Patel, and Sean Keller. 2016. Finexus: Tracking precise motions of multiple fingertips using magnetic sensing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 1504–1514.
- [10] Mungyeong Choe, Yeongcheol Choi, Jaehyun Park, and Hyun K. Kim. 2019. Comparison of Gaze Cursor Input Methods for Virtual Reality Devices. *International Journal of Human-Computer Interaction* 35, 7 (2019), 620–629. <https://doi.org/10.1080/10447318.2018.1484054> arXiv:https://doi.org/10.1080/10447318.2018.1484054
- [11] Barrett Ens, Ahmad Byagowi, Teng Han, Juan David Hincapi -Ramos, and Pourang Irani. 2016. Combining ring input with hand tracking for precise, natural interaction with spatial analytic interfaces. In *Proceedings of the 2016 Symposium on Spatial User Interaction*. 99–102.
- [12] Ali Erol, George Bebis, Mircea Nicolescu, Richard D Boyle, and Xander Twombly. 2007. Vision-based hand pose estimation: A review. *Computer Vision and Image Understanding* 108, 1-2 (2007), 52–73.
- [13] Augusto Esteves, Yonghwan Shin, and Ian Oakley. 2020. Comparing selection mechanisms for gaze input techniques in head-mounted displays. *International Journal of Human-Computer Studies* 139 (2020), 102414. <https://doi.org/10.1016/j.ijhcs.2020.102414>
- [14] Mark Euston, Paul Coote, Robert Mahony, Jonghyuk Kim, and Tarek Hamel. 2008. A complementary filter for attitude estimation of a fixed-wing UAV. In *2008 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 340–345.
- [15] Gabriel Evans, Jack Miller, Mariangely Iglesias Pena, Anastacia MacAllister, and Eliot Winer. 2017. Evaluating the Microsoft HoloLens through an augmented reality assembly application. In *Degraded environments: sensing, processing, and display 2017*, Vol. 10197. SPIE, 282–297.
- [16] Masaaki Fukumoto and Yasuhito Suenaga. 1994. “FingerRing” a full-time wearable interface. In *Conference companion on Human factors in computing systems*. 81–82.
- [17] Henning Graf and Klaus Jung. 2012. The smartphone as a 3D input device. In *2012 IEEE Second International Conference on Consumer Electronics-Berlin (ICCE-Berlin)*. IEEE, 254–257.
- [18] Yizheng Gu, Chun Yu, Zhipeng Li, Weiqi Li, Shuchang Xu, Xiaoying Wei, and Yuanchun Shi. 2019. Accurate and low-latency sensing of touch contact on any surface with finger-worn imu sensor. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. 1059–1070.
- [19] Yizheng Gu, Chun Yu, Zhipeng Li, Zhaohe Li, Xiaoying Wei, and Yuanchun Shi. 2020. Qwertyring: Text entry on physical surfaces using a ring. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 4 (2020), 1–29.
- [20] Aakar Gupta, Cheng Ji, Hui-Shyong Yeo, Aaron Quigley, and Daniel Vogel. 2019. Rotoswype: Word-gesture typing using a ring. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–12.
- [21] Juan David Hincapi -Ramos, Kasim Ozacar, Pourang P Irani, and Yoshifumi Kitamura. 2015. GyroWand: IMU-based raycasting for augmented reality head-mounted displays. In *Proceedings of the 3rd ACM Symposium on Spatial User Interaction*. 89–98.
- [22] Roberto Hoyle, Robert Templeman, Steven Armes, Denise Anthony, David Crandall, and Apu Kapadia. 2014. Privacy behaviors of lifeloggers using wearable cameras. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 571–582.
- [23] Robert JK Jacob. 1991. The use of eye movements in human-computer interaction techniques: what you look at is what you get. *ACM Transactions on Information Systems (TOIS)* 9, 2 (1991), 152–169.
- [24] Lei Jing, Zixue Cheng, Yinghui Zhou, Junbo Wang, and Tongjun Huang. 2013. Magic ring: A self-contained gesture input device on finger. In *Proceedings of the 12th International Conference on Mobile and Ubiquitous Multimedia*. 1–4.
- [25] Ricardo Jota, Miguel Nacenta, Joaquim Jorge, Sheelagh Carpendale, and Saul Greenberg. 2009. A comparison of ray pointing techniques for very large displays. Technical Report. University of Calgary.
- [26] Mohamed Kari and Christian Holz. 2023. HandyCast: Phone-based Bimanual Input for Virtual Reality in Mobile and Space-Constrained Settings via Pose-and-Touch Transfer. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–15.
- [27] Keiko Katsuragawa, Krzysztof Pietroszek, James R Wallace, and Edward Lank. 2016. Watchpoint: Freehand pointing with a smartwatch in a ubiquitous display environment. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*. 128–135.
- [28] Daniel Kharlamov, Brandon Woodard, Liudmila Tahai, and Krzysztof Pietroszek. 2016. TickTockRay: smartwatch-based 3D pointing for smartphone-based virtual reality. In *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology*. 365–366.
- [29] Wolf Kienzle and Ken Hinckley. 2014. LightRing: always-available 2D input on any surface. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. 157–160.
- [30] Junhyeok Kim, William Delamare, and Pourang Irani. 2018. Thumbtext: Text entry for wearable devices using a miniature ring. In *Graphics Interface*.
- [31] Carsten Kirstein and Heinrich Muller. 1998. Interaction with a projection screen using a camera-tracked laser pointer. In *Proceedings 1998 MultiMedia Modeling. MMM’98 (Cat. No. 98EX200)*. IEEE, 191–192.
- [32] Yuki Kubo. 2022. Ring-Type Indirect Pointing Device for Large Displays Using Three-Axis Pressure Sensor. In *Proceedings of the 2022 ACM Symposium on Spatial User Interaction (Online, CA, USA) (SUI ’22)*. Association for Computing Machinery, New York, NY, USA, Article 33, 2 pages. <https://doi.org/10.1145/3565970.3568185>
- [33] James John Kuch. 1994. Vision-based hand modeling and gesture recognition for human computer interaction. (1994).
- [34] JMF Landsmeer. 1963. The coordination of finger-joint motions. *JBJS* 45, 8 (1963), 1654–1662.
- [35] 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*. 321–333.
- [36] Jintae Lee and Tosiya L Kunii. 1993. Constraint-based hand animation. In *Models and techniques in computer animation*. Springer, 110–127.
- [37] Jintae Lee and Tosiya L Kunii. 1995. Model-based analysis of hand posture. *IEEE Computer Graphics and applications* 15, 5 (1995), 77–86.
- [38] Chen Liang, Chi Hsia, Chun Yu, Yukang Yan, Yuntao Wang, and Yuanchun Shi. 2023. DRG-KeyBoard: Enabling Subtle Gesture Typing on the Fingertip with Dual IMU Rings. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 4 (2023), 1–30.
- [39] Chen Liang, Chun Yu, Yue Qin, Yuntao Wang, and Yuanchun Shi. 2021. DualRing: Enabling subtle and expressive hand interaction with dual IMU rings. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 5, 3 (2021), 1–27.
- [40] Richard A Lipka. 2003. Are 2D: 4D finger-length ratios related to sexual orientation? Yes for men, no for women. *Journal of personality and social psychology* 85, 1 (2003), 179.
- [41] Guan hong Liu, Yizheng Gu, Yiwen Yin, Chun Yu, Yuntao Wang, Haipeng Mi, and Yuanchun Shi. 2020. Keep the phone in your pocket: Enabling smartphone operation with an imu ring for visually impaired people. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 2 (2020), 1–23.
- [42] Michael McGuffin, Nicolas Burtynk, Gordon Kurtenbach, and King Street. 2002. FaST Sliders: Integrating marking menus and the adjustment of continuous values. In *Graphics Interface*. 35–42.
- [43] Mathieu Nancel, Olivier Chapuis, Emmanuel Pietriga, Xing-Dong Yang, Pourang P Irani, and Michel Beaudouin-Lafon. 2013. High-precision pointing on large wall displays using small handheld devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 831–840.
- [44] Kai Nickel and Rainer Stiefelhagen. 2003. Pointing gesture recognition based on 3d-tracking of face, hands and head orientation. In *Proceedings of the 5th international conference on Multimodal interfaces*. 140–146.
- [45] Takehiro Niikura, Yoshihiro Watanabe, and Masatoshi Ishikawa. 2014. Anywhere surface touch: utilizing any surface as an input area. In *Proceedings of the 5th Augmented Human International Conference*. 1–8.
- [46] The Metropolitan Museum of Art. 2023. . <https://www.metmuseum.org>
- [47] Ju Young Oh, Jun Lee, Joong Ho Lee, and Ji Hyung Park. 2017. Anywhere-touch: Finger tracking method on arbitrary surface using nailed-mounted imu for mobile hmd. In *HCI International 2017-Posters’ Extended Abstracts: 19th International Conference, HCI International 2017, Vancouver, BC, Canada, July 9–14, 2017, Proceedings, Part I 19*. Springer, 185–191.
- [48] Farshid Salemi Parizi, Eric Whitmire, and Shwetak Patel. 2019. Auraring: Precise electromagnetic finger tracking. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 3, 4 (2019), 1–28.
- [49] Vladimir I Pavlovic, Rajeev Sharma, and Thomas S. Huang. 1997. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on pattern analysis and machine intelligence* 19, 7 (1997), 677–695.

- [50] James M Rehag and Takeo Kanade. 1994. Digiteyes: Vision-based hand tracking for human-computer interaction. In *Proceedings of 1994 IEEE workshop on motion of non-rigid and articulated objects*. IEEE, 16–22.
- [51] James R Rehag and Takeo Kanade. 1994. *Visual Tracking of Self-Occluding Articulated Objects*. Technical Report. CARNEGIE-MELLON UNIV PITTSBURGH PA SCHOOL OF COMPUTER SCIENCE.
- [52] Xiyuan Shen, Yukang Yan, Chun Yu, and Yuanchun Shi. 2022. ClenchClick: Hands-Free Target Selection Method Leveraging Teeth-Clench for Augmented Reality. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 3 (2022), 1–26.
- [53] Yilei Shi, Haimo Zhang, Kaixing Zhao, Jiashuo Cao, Mengmeng Sun, and Suranga Nanayakkara. 2020. Ready, steady, touch! sensing physical contact with a finger-mounted IMU. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 2 (2020), 1–25.
- [54] Adrian Spurr, Umar Iqbal, Pavlo Molchanov, Otmar Hilliges, and Jan Kautz. 2020. Weakly supervised 3d hand pose estimation via biomechanical constraints. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVII* 16. Springer, 211–228.
- [55] Sophie Stellmach and Raimund Dachselt. 2012. Look & touch: gaze-supported target acquisition. In *Proceedings of the SIGCHI conference on human factors in computing systems*. 2981–2990.
- [56] Sophie Stellmach and Raimund Dachselt. 2012. *Look and Touch: Gaze-Supported Target Acquisition*. Association for Computing Machinery, New York, NY, USA, 2981–2990. <https://doi.org/10.1145/2207676.2208709>
- [57] Daniel Vogel and Ravin Balakrishnan. 2005. Distant freehand pointing and clicking on very large, high resolution displays. In *Proceedings of the 18th annual ACM symposium on User interface software and technology*. 33–42.
- [58] Jennifer M. Vojtech, Surbhi Hablani, Gabriel J. Cler, and Cara E. Stepp. 2020. Integrated Head-Tilt and Electromyographic Cursor Control. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 28, 6 (2020), 1442–1451. <https://doi.org/10.1109/TNSRE.2020.2987144>
- [59] Xing-Dong Yang, Tovi Grossman, Daniel Wigdor, and George Fitzmaurice. 2012. Magic finger: always-available input through finger instrumentation. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*. 147–156.
- [60] Tae Suk Yoo, Sung Kyung Hong, Hyok Min Yoon, and Sungsu Park. 2011. Gain-scheduled complementary filter design for a MEMS based attitude and heading reference system. *Sensors* 11, 4 (2011), 3816–3830.
- [61] Shumin Zhai, Carlos Morimoto, and Steven Ihde. 1999. Manual and Gaze Input Cascaded (MAGIC) Pointing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Pittsburgh, Pennsylvania, USA) (CHI '99). Association for Computing Machinery, New York, NY, USA, 246–253. <https://doi.org/10.1145/302979.303053>
- [62] Yuliang Zhao, Xianshou Ren, Chao Lian, Kunyu Han, Liming Xin, and Wen J Li. 2021. Mouse on a Ring: A Mouse Action Scheme Based on IMU and Multi-Level Decision Algorithm. *IEEE Sensors Journal* 21, 18 (2021), 20512–20520.

A DATA ANALYSIS PROCESS OF KEY JOINT MOVEMENT LAWS

We induce several motion laws of key joint movement in Section 5. Here, we provide a detailed account of the processes through which various conclusions were verified or falsified via data analysis. Falsifying the conclusions is relatively easy. The linearity of Conclusion 1 can be easily falsified through analysis of the data plots. We have falsified Conclusions 3 and 5 by demonstrating that the length error exceeds 20%. Due to measurement errors, we cannot rigorously establish the strict validity of the constraint relationship through hypothesis testing. Instead, we consider the conclusion valid if more than 95% of the user data points satisfy our assumption of minimal error.

A.0.1 Conclusion 1. $\theta_{DIP} \neq \frac{2}{3}\theta_{PIP}$ (θ_{DIP} is defined as the angle between the phalanxs connected by the DIP joints. θ_{PIP} is the angle between the phalanxs connected by the PIP joint.)

While $\theta_{DIP} = \frac{2}{3}\theta_{PIP}$ is widely used in VR hand reconstruction, we found that it does not hold during finger sliding. We visualized the relationship between θ_{DIP} and θ_{PIP} for five representative participants out of the total twelve participants in Fig.4(b). The data from these participants were categorized into three distinct classes, which provide representative coverage of the entire participant group. For most participants (P3, P6, P7, P9), θ_{DIP} and θ_{PIP} show a linear relationship, but the force exerted by the surface on the distal phalanx makes θ_{DIP} smaller than its relaxed state. For P3 and P11, the ratio is still around $\frac{2}{3}$. However, the ratios for P6 and P9 are reduced. In addition, for a few participants (e.g., P5, green points in the scatter plot), θ_{DIP} increases and then decreases with θ_{PIP} , and the linear relationship does not hold. We explained that some participants applied greater force to the surface, causing the supporting effect of the ligament near the DIP to disappear (Fig.4(c)), resulting in an unnatural posture of the finger.

A.0.2 Conclusion 2. The three phalanxs of the index finger are in the same plane.

Despite the additional forces exerted on the bones and ligaments during lateral sliding, we found that the three phalanxes of the index finger remained in the same plane in three different finger sliding modes (RW, RTM, RP), shown in Fig.4(d). We represented the corresponding vectors of the three bones as L_1 : the vector from DIP to the nail tip, L_2 : the vector from PIP to DIP, and L_3 : the vector from MP to PIP. We calculated the angle between L_3 and the plane formed by L_1 and L_2 for each sliding mode. Even in the presence of measurement errors, if the angle is sufficiently small, we can consider coplanarity to hold true. The average angle size is 1.9° for RW, 2.1° for RTM, and 2.1° for RP. The proportion of data points with angles $\leq 5^\circ$ reached 99.7%(RW), 98.1%(RTM), and 98.4%(RP). The above analysis indicates that the coplanarity holds.

A.0.3 Conclusion 3. Each person's skeletal length is fixed, but the ratio of the phalanxs' lengths varies significantly between individuals.

We suspected that skin deformation when the index finger is bent would cause a change in the distance between adjacent OptiTrack markers placed on the skin's surface. We measured the standard deviation of the distance change between adjacent joints of the same participant. The average STD was 1.4%. 95.6% of the data

points achieved an error of less than 3%. The result indicates that using reflective markers on the skin to measure finger bone length ensures stable data.

Although medical literature[6] provides an average length ratio of 2.52:1.42:1 for the three phalanxes of the index finger, we found that the length ratios of the three bones of the participants varied greatly. Taking the length ratio between the intermediate phalanx and distal phalanx of twelve participants as an example, we obtained a similar average length ratio (1.42:1). However, the standard deviation of the ratio reached 0.22, indicating a difference of over 15% between each participant's ratio and the average ratio. Applying bone length ratios as prior knowledge to a physical model may further amplify joint velocity prediction errors in the propagation of the kinetic chain.

A.0.4 Conclusion 4. The displacement of the fingertip projection on the physical surface is approximately equal to the displacement of the contact point.

The nail tip is well located at the end of the kinetic chain, which, together with DIP, consists of the two distal phalanx endpoints. On the other hand, the contact point is located on the soft part of the fingertip, making it difficult to connect with the kinetic chain. We studied the average length and angle errors between the fingertip projection and the contact point. The average length error was 0.98mm (4.69% of the total length) for each stroke. 96.1% of the data points had an error of less than 2mm. It is reasonable to the error in the subsequent physical modeling and to use the displacement of the fingertip projection to represent the movement of the contact point on the physical surface (Fig.4(f)).

A.0.5 Conclusion 5. Among all three sliding modes (RW, RTM, RP), the displacement of MP cannot be ignored.

One of the motivations behind proposing three different finger motion modes was the belief that constraints from the palm and other fingers could reduce the degrees of freedom in the sliding finger process. Placing the palm, thumb, or middle finger on a surface can strongly restrict the movement of MP of the index finger. However, in the RP mode, the average displacement of MP in one stroke still reached 2.9mm (18% of fingertip displacement). The average displacement was 16.2mm (40%) for the RW mode and 12.2mm (28%) for the RTM mode. More substantial constraints can effectively reduce the displacement of MP but cannot eliminate its impact on the kinetic chain (Fig.4(g)-(i)).

A.0.6 Conclusion 6. There is a strong correlation between the projected velocities of the fingertip, DIP, and PIP on the physical surface:

$$\arccos\left(\frac{\mathbf{v}_{\text{NailTip}} \cdot \mathbf{v}_{\text{DIP}}}{|\mathbf{v}_{\text{NailTip}}| |\mathbf{v}_{\text{DIP}}|}\right) \leq 13^\circ, \arccos\left(\frac{\mathbf{v}_{\text{DIP}} \cdot \mathbf{v}_{\text{PIP}}}{|\mathbf{v}_{\text{DIP}}| |\mathbf{v}_{\text{PIP}}|}\right) \leq 15^\circ,$$

$$\arccos\left(\frac{\mathbf{v}_{\text{NailTip}} \cdot \mathbf{v}_{\text{PIP}}}{|\mathbf{v}_{\text{NailTip}}| |\mathbf{v}_{\text{PIP}}|}\right) \leq 30^\circ$$

Table 3: Mean angles and 95% confidence interval of the angles' distribution between the projected velocities of the nail tip, DIP, PIP, and MP.

| Mean Angle / 95% Interval | v_{nailtip} | v_{DIP} | v_{PIP} | v_{MP} |
|---------------------------|----------------------|------------------|------------------|-----------------|
| v_{nailtip} | 0°/0° | - | - | - |
| v_{DIP} | 6.2°/12.6° | 0°/0° | - | - |
| v_{PIP} | 10.6°/30.1° | 7.8°/15.8° | 0/0 | - |
| v_{MP} | 13.2°/40.7° | 12.9°/34.8° | 10.4°/32.1° | 0°/0° |

When sliding the index finger leftwards or rightwards, the finger exhibits an approximately fan-shaped trajectory (Fig.4(j)). We hypothesize that the velocity between the key points should have a strong correlation. We calculated the angles between the projected velocity vectors of the fingertip, DIP, PIP, and MP under three different finger sliding modes. We then fit the velocity angles to a normal distribution and calculated the range of the 95% confidence interval (Table 3). The mean angle between the projected velocity of the MP and other key points exceeded 10 degrees, which indicates that the velocity relationship between the MP and other points is relatively weak. The fingertip and DIP, DIP, and PIP are the endpoints of two finger bones, respectively, so they have a strong velocity correlation. We summarize the range of the 95% confidence interval as the velocity constraints between the projected velocities.

B A DETAILED INTRODUCTION OF THE WIRELESS MOUSERING PROTOTYPE

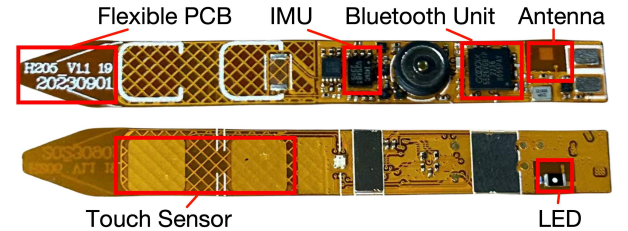


Figure 12: The Flexible PCB in the wireless version of MouseRing.

The wireless version of MouseRing is designed and manufactured on a flexible, elongated PCB, as is shown in Fig.12. The elongated PCB is then curved into a circular shape and secured within a metal ring for user wearability. The PCB board incorporates several sensors and communication components. We used the MPU9250 chip, identical to the one utilized in the data collection section, to carry out real-time collection of IMU motion data from the index finger. The Bluetooth module and antenna on the PCB can communicate with a remote computer to transmit the IMU data at 200 Hz. We have also integrated a touch-capacitive sensor and an LED light. They are used for device debugging and status feedback only and are unrelated to the design of the MouseRing.



Figure 13: Interactive pages of exhibits.

C DETAILED SETUP OF LARGE-SCREEN USER EXPERIMENT

Figure 13 shows the interactive pages of exhibits in the real-world user study. All text and image content is sourced from the official

website of the Metropolitan Museum of Art[46]. Circular semi-translucent components are interactable buttons. The sizes and positions were designed in advance to cover different indexes of difficulty.