

T2IRay: Design of Thumb-to-Index based Indirect Pointing for Continuous and Robust AR/VR Input

Jina Kim

KAIST

Daejeon, Republic of Korea

jina1190@kaist.ac.kr

Yang Zhang

University of California, Los Angeles

Los Angeles, California, USA

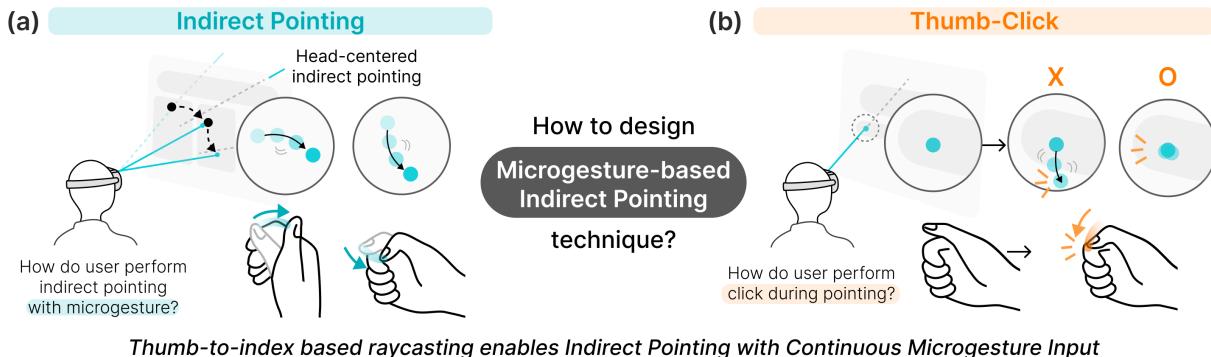
yangzhang@ucla.edu

Sang Ho Yoon

KAIST

Daejeon, Republic of Korea

sangho@kaist.ac.kr



Thumb-to-index based raycasting enables Indirect Pointing with Continuous Microgesture Input

Figure 1: T2IRay is an indirect, free-hand fine-motor pointing framework based on thumb-to-index (T2I) motion. Our study empirically provides system design guidance for effective interaction by focusing on two key aspects: (a) ensuring precise control and (b) maintaining stable ray.

Abstract

Free-hand interactions have been widely deployed for AR/VR interfaces to promote a natural and seamless interaction experience. Among various types of hand interactions, microgestures are still limited to support discrete inputs and lack of continuous interaction theme. To this end, we propose a new pointing technique, T2IRay, which enables continuous indirect pointing by microgestures for continuous spatial input. We employ a own local coordinate system based on the thumb-to-index finger relationship to map computed raycasting direction for indirect pointing in a virtual environment. Furthermore, we examine various mapping methodologies and collected thumb-click behaviors to formulate thumb-to-index microgesture design guidelines to foster continuous, yet reliable input. We evaluated the design parameter for mapping indirect pointing with acceptable speed, depth, and range. We collected and analyzed the characteristics of click behaviors for future detection implementation. Our research demonstrates the potential and practicality of free-hand micro-finger input methods for advancing future interaction paradigms.

CCS Concepts

- Human-centered computing → Pointing; Virtual reality; Gestural input.

Keywords

AR/VR, Thumb-to-Index interaction, Interaction Technique, Micro-gestures

ACM Reference Format:

Jina Kim, Yang Zhang, and Sang Ho Yoon. 2025. T2IRay: Design of Thumb-to-Index based Indirect Pointing for Continuous and Robust AR/VR Input. In *CHI Conference on Human Factors in Computing Systems (CHI '25), April 26-May 1, 2025, Yokohama, Japan*. ACM, New York, NY, USA, 16 pages. <https://doi.org/10.1145/3706598.3713442>

1 Introduction

Free-hand interaction is a fundamental and crucial modality in immersive AR/VR that enables seamless and natural interactions. Among diverse ways of utilizing the hand, thumb-to-finger gestures have recently been highlighted with one-handed interaction [9]. It is performed by manipulating the thumb against the other fingers in a subtle manner to minimize physical fatigue and enhance social acceptability [27, 41]. To this end, the main body of microgesture works has focused on recognizing gestures and enabling discrete inputs [25, 29, 34]. Also, previous studies have shown versatile design spaces for microgestures including grasping [43] or across distinctive hand postures [27]. With the advancement in aforementioned sensing techniques and flexibility in design space, recent works have started to focus on utilizing continuous microgestures. However, they still remain in offering 1-DOF level control [10] or stroke drawing [33] that cannot support fully continuous input such as pointing.

In AR/VR, pointing-based input with free-hand interactions is formed by using the angle of the ray stemming from the palm or wrist, and selection is often done by pinch gesture. However,

the conventional hand interaction with direct input requires users to hold up their entire hand within the field-of-the-view (FoV) of the camera causing undesirable side effects mentioned earlier. To mitigate this, recent work suggested a mid-air indirect input method with a finger pointing to achieve finer motor control and more deliberate aiming [6]. However, this approach still requires a mid-air interaction plane in a fixed configuration which reduces the flexibility of performing input tasks. Recent HMDs have started to be equipped with wide FoV cameras facing multi-direction to easily recognize microgestures in various postures [2]. This opens a new opportunity to enable indirect input-based pointing in AR/VR.

Based on these backgrounds, our goal is to extend the previously limited discrete thumb-to-index (T2I) gesture into a readily available indirect input for pointing in AR/VR. Rather than constrained to a fixed interaction plane, we focus on bringing indirect input with any hand position or orientation. To this end, we propose a **T2IRay**, an indirect pointing technique with a thumb-to-index gesture, designed to utilize fine thumb manipulation over the index finger. For T2IRay, we allow users to perform indirect input within the FoV of head-oriented FoV. As combining microgestures with other modalities reduces fatigue and improves interaction speed [8, 49], we incorporate natural head movements to ensure always-available and fluid continuous input.

Given the concept of T2IRay, there still remains a lack of understanding on how to design indirect input using thumb-to-index microgesture. To establish precise and robust T2IRay, the key challenge is to properly map the thumb tracking to pointing and ensure accurate detection of a thumb-click gesture. Here, the thumb-click refers to the gesture where the thumb taps the side of the index finger. Since inferring discrete taps during continuous interaction is ambiguous [56] and finger movement associated with thumb-click alters pointing accuracy [15], it is important to systematically evaluate user behavior. For this reason, we investigate 1) mapping methods to align continuous tracking to pointing in AR/VR and 2) finger kinematics during thumb-click.

In this paper, we propose a T2I pointing interaction technique and explore the design parameters by addressing the following questions: 1) How should we map continuous thumb tracking to pointing in AR/VR? and 2) What are the common user behaviors associated with thumb-clicks? To answer this, we conducted a Fitts' law study to evaluate the system's performance in pointing tasks with various mapping methods. Then, we examine finger kinematics during the thumb-click. Through our study results, we formulate a mapping strategy to enable robust pointing while discovering parameters to avoid pointing jitter and false positives for robust thumb-clicks.

Our contributions are as follows:

- A novel indirect input technique transforming thumb-to-index microgesture into a continuous and robust pointing for AR/VR;
- An in-depth analysis by Fitts' law study to compare different mapping methods to align continuous thumb tracking to pointing input;
- An investigation of user behaviors on thumb-clicks to formulate design guidelines to reduce pointing jitters and false positives.

2 Related Work

2.1 Microgesture-based Interactions

In emerging AR/VR environments, there is a increasing focus on free-hand interactions. Free-hand interaction has long been utilized as a versatile input modality, often replacing controllers and other devices. A common approach is to use the hand as an interaction space [17, 25, 39] or hand posture as a control cue [37, 59]. Recently, the significance of microgestures has been highlighted for their flexible use of finger movements [9, 52]. Among various microgestures, previous works explored T2I gesture recognition [10, 30, 33, 34, 46] with a ring-form factor, touchpad devices [4, 5, 18, 24, 55] and external sensors [20, 36, 47, 51, 57]. Furthermore, other works have demonstrated the potential of utilizing T2I interactions as discrete interactions [29, 38, 45].

Still, the T2I interaction research has mainly focused on providing robust discrete gestures (e.g., tap, swipe, stroke). While discrete gestures are essential for interaction, expanding their functionality to include continuous input would enhance their applicability and practicability [11]. Therefore, we aim to bring continuous input capability for T2I interaction where we also explore various mapping approaches for pointing in AR/VR.

2.2 Pointing Techniques in AR/VR

Pointing with controllers and free-hand gestures is fundamental for continuous interaction in AR/VR. Direct and indirect methods for pointing have been proposed within the HCI community. For instance, Gunslinger[35] proposed indirect hand-cursor by hand posture with arms-down posture. ARPAd [6] showcased an indirect mid-air continuous pointing in AR by mapping hand movements from a virtual fixed plane to HMD. This work demonstrated that indirect input reduced fatigue compared to direct hand interaction, especially during prolonged or repetitive tasks. Similarly, TriPad [15] developed touch input on any surface while supporting both direct and indirect inputs. Beyond vision-based approaches, Handycast [28] resolved the physical constraints and limitations of traditional raycasting by introducing a bimanual raycasting through mobile phones to access VR in space-constrained. Furthermore, In-Depth Mouse [58] proposed a depth-adaptive pointing system to enhance mouse operation in desktop VR. To improve the performance further, previous research suggested integrating microgestures with other modalities [31, 48].

As mentioned, previous research highlighted the capability of precise and flexible controllability with indirect input approaches. Despite progress in previous work, however, there remains a need to explore indirect pointing with finer-grained movements. To address this, we propose a method that enables fine-grained indirect pointing. By combining head movement with T2I motion, we extend the interaction space to access the full 3D space. Given the findings of previous studies on the relationship between raycasting and 3D pointing factors—such as pointing technique and target depth [26, 44], it is crucial to investigate the 3D pointing factors for key consideration for system design. Thus, our research aims to examine the diverse 3D pointing factors involved in pointing such as depth.

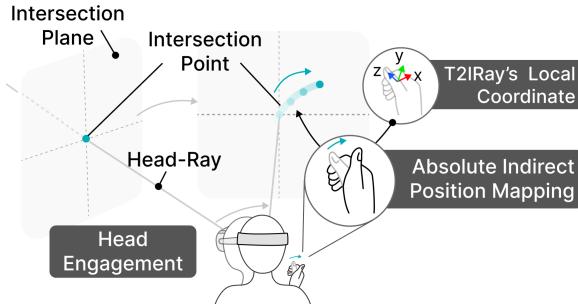


Figure 2: An overview of the T2IRay input mechanism. The ray originated from HMD is centered on the intersection plane. This plane moves as the user's head moves. The user performs pointing by adjusting the intersection point based on the thumb position in T2IRay's local coordinate system. The intersection point only be moved by finger so that head movement doesn't cause any point movement.

2.3 Mapping and User Behavior for Input

For gesture input, some previous work has been widely explored for various interaction purposes. These include examining user behavior such as proprioception-based control on the hand [13, 21] or implementation to map finger manipulation. For example, previous research has successfully implemented the input technique of typing on the index finger using the thumb [53, 54]. In addition, other work has demonstrated thumb-driven interaction to perform thumb-based text input [19]. More recently, cursor-like input has been proposed using various sensing techniques such as mmWave radar [32] or radio frequency [14] by tracking finger positions. In addition, previous work has evaluated proprioceptive gestures [22], which involves recognizing gestures made directly on the skin to provide a more intuitive interaction experience. It has also studied user behavior for one-handed mid-air typing [56], grasping microgesture [42], aiming to understand the feasibility and efficiency of such input.

These studies highlight the importance of understanding user behavior to improve the effectiveness of gesture-based interactions. Despite extensive research on improving hand-based interaction, the behavior associated with microgesture, particularly in the context of continuous input such as pointing, has not been thoroughly investigated. To address this gap, our study aims to understand user performance and behavior during pointing, including thumb-click actions for a possible selection approach with a focus on microgestures. In this way, we aim to contribute to the development of more effective and intuitive microgesture-based interaction techniques.

3 T2IRay Design

3.1 Design Goals

We define T2IRay as an indirect pointing technique as shown in Figure 2. We utilize the thumb's position relative to the T2IRay's local coordinate formed by the thumb and index to manipulate a virtual ray for pointing. This local coordinate system is made

by virtual plane by thumb and index finger for intuitive thumb position tracking. T2IRay allows users to perform pointing input using thumb manipulation over the side of the index finger, such as fine motor raycasting. We design our system with the following 3 design goals:

- **DG1: Fine-grained & Subtle Control:** Thumb movement should keep their comfort range of motion for small inputs and quick transitions to specific command gestures. Previous work showed that the thumb and index area is one of the most comfortable regions [12]. And index are also most preferred and frequent interaction [39]. Therefore, we form our interactions using the thumb and the index finger.
- **DG2: Familiar Input Metaphor:** The input system should resemble existing systems to promote rapid and effortless adoption by users. Since users perceive their hands as a multi-touch sensor [23], we transferred the metaphor of the familiar body touchpad concept [11] onto the side of the index finger. Moreover, we focus on supporting the proposed interaction in a natural hand pose, which enhances microgesture performance [50].
- **DG3: Eyes-free with high controllability:** Users should be able to interact with AR/VR interfaces without needing to maintain direct visual focus on their fingers. In this way, we promote users to concentrate on their interactions rather than their hand movements in eye-free manner [16, 40]. Also, we aim to support always readily available input to users in AR/VR.

To achieve these goals, we ensure that continuous microgesture pointing to support the *comfort reachability* for DG1, DG2 and *seamless control* for DG3.

3.2 T2I Finger Input to Ray Control

This section explains the design components and rationales to meet the requirements of the design goals. Our system components mainly consist of 3 features as shown in Figure 2 and 3. These include the T2IRay's local coordinate system, absolute indirect position mapping, and head engagement. For more details, please refer to Section 4.

3.2.1 T2IRay's Local Coordinate (DG1). Most hand-tracking systems use a local coordinate system based on key joint systems. For recognizing microgesture, relative finger movement would be a practical approach to maintain consistent gestures. However, directly using the values of the fingers' joints for the T2I interaction would miss the meaningful movement information such as thumb position relative to interaction space. To answer DG1 (Fine-grained & Subtle Control), the range of thumb motion should reside within a reasonable space near the index finger. Therefore, we propose a T2IRay's local coordinate that sets an origin reference point at a semi-fist pose to maximize comfort for fine controllability. By setting interaction volume within a small volume, we also encourage users to carry out subtle interaction during pointing interactions.

3.2.2 Absolute Pointing (DG2). We employed absolute pointing with the T2IRay's local coordinate system with the finger plane. The reason is to reflect the characteristics of existing raycasting principles in AR/VR where controller or free-hand ray casting utilize

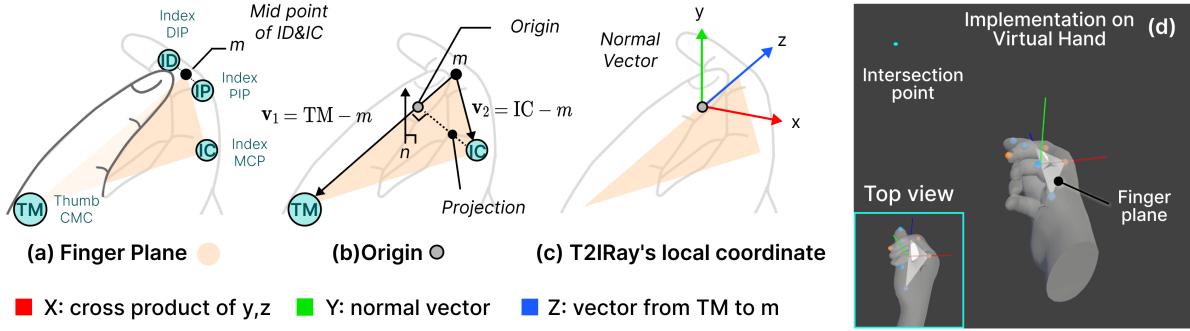


Figure 3: Components of T2IRay's local coordinate system. (a) Finger Plane: A plane formed by the finger joints (TM, IC, mid point by ID and IP) and defines a normal vector n . **(b) Origin:** A reference point in the *Finger Plane* based on projection of v_2 onto vector v_1 . **(c) A formulated T2IRay's local coordinate system,** **(d) The implementation in Unity.**

absolute pointing. Since users perceive their hands as multitouch sensors, we believe that users are likely to perceive the index finger as a touch sensor that performs interaction with proprioception and natural haptic feedback from the index finger [45]. Absolute pointing naturally allows unrestricted movement within the thumb's motion range. Therefore, our system does not distinguish thumb contact on the index finger as a discrete touch signal and allows the thumb to move naturally within the index finger's interaction volume.

3.2.3 Thumb-Click Design Parameters (DG2). As a continuous input, click-down, and click-up events serve as indicators for the trigger state. Our system directly utilizes tapping the thumb on the side of the index finger to carry out the thumb-click gesture. However, a problem arises with ray fluctuations during the finger's landing and release for click which causes instability and pointing jitters. Thus, we explore finger kinematics to identify suitable design parameters for a robust thumb-click, analogous to determining the debouncing mechanism parameters for a computer mouse.

3.2.4 Head-engaged Indirect Input (DG3). As mentioned in DG3, we want to ensure that we support always accessible input to users in AR/VR, we include head engagement in our interaction. To determine a suitable pointing methodology for AR/VR, we utilize the head-ray. The head-ray refers to a virtual ray that originated from the HMD's origin point. The direction of the ray is assigned to an intersection point determined by the absolute position of the finger. Also, we adopted a two-phase 3D interaction technique [8] where we combine head engagement with T2I-based microgestures. We utilize head gaze (commonly through HMDs) to move the FoV while controlling the ray within the FoV using T2IRay. With this, we expand the usable range of indirect input while maintaining the control performance of continuous input.

4 Implementation

4.1 Apparatus & Software

Our system is built with Oculus Quest 3 Interaction SDK¹. For software, we built the system on Unity (2022.3.22f version) and used

the joint keypoints from hand tracking and the position and rotation data from HMD (head-gaze). With streaming a real-time data, our system continuously computes each proposed algorithm: T2I coordinate and Ray direction in the backend upon availability of the hand tracking in the FoV.

4.2 T2IRay's Local Coordinate Configuration

As shown in Figure 3, we use the thumb and index finger joints as core components to construct a coordinate. Specifically, we utilized thumb's tip (TT), interphalangeal joint (TI), carpometacarpal (TM) and index's distal interphalangeal (ID), proximal interphalangeal joint (IP), and metacarpophalangeal joint (IC). Here, we define the normal vector on the finger plane and the origin position before formulating coordinate axis.

To set a *Finger Plane*, we compute the midpoint position (m) between ID and IP joints. Then, we form the plane with 3 key joints including (m), IC, and TM. The normal vector (n) is defined as the perpendicular vector to this plane. To determine the origin, we computed vector v_1 by subtracting the position of m from TM and vector v_2 by subtracting the position of m from IC. Next, the projection of vector v_2 onto vector v_1 is calculated. The dot product between v_1 and v_2 provides a scalar that represents the component of v_2 in the direction of v_1 . This scalar is divided by the dot product of v_1 with itself. The projection of vector v_2 onto vector v_1 is given by:

$$\text{Proj}_{v_1} v_2 = \left(\frac{v_1 \cdot v_2}{v_1 \cdot v_1} \right) v_1$$

We assigned n as the Y-axis and the vector direction from TM to m as the Z-axis. These two axes remain static regardless of finger movement. The X-axis is calculated as the cross-product of the y-axis and z-axis vectors. Finally, the resulting vectors are normalized to unit length to ensure consistency.

4.3 Determining Ray direction through Finger Position

4.3.1 Absolute Tracking related to Finger Position. The system tracks absolute movement by the TT and TI joints from T2IRay's local coordinates. For horizontal movements, we sum the x-axis positions of the TT and TI. This approach leverages the fact that

¹<https://developers.meta.com/horizon/reference/interaction/v68>

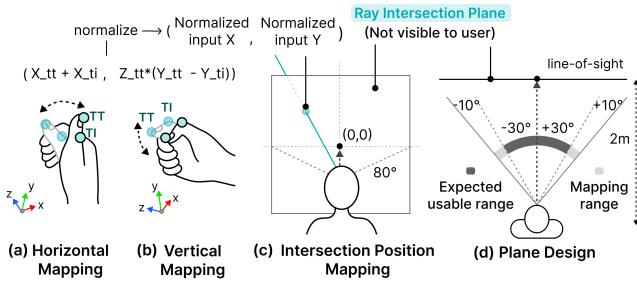


Figure 4: (a) The horizontal ray’s movement is based on where the combined movements of joints TT and TI along the x-axis as seen from the top view. (b) The vertical control is based on the difference in the positions of TT and TI along the y-axis and is further adjusted by TT’s z-axis movement, as seen from the side view. (c) The finger position value (x,y) is normalized and mapped to ray intersection plane range. (d) The intersection plane is designed to be within 80° of the view angle, which is situated at a distance of 2 m from the user in line of sight.

the thumb’s TT and TI joints generally move in the same direction during horizontal motions, allowing for wider and more flexible movements.

$$\text{Horizontal Movement} = x_{TT} + x_{TI}$$

We consider several considerations to deal with vertical movement. First, the range of vertical movement is relatively smaller compared to horizontal movement and involves rotation and translation. To address these challenges, we use the z-axis motion for scaling purposes. Specifically, we calculate the difference in the y-positions between the TT and TI joints:

$$\text{Vertical Movement} = \Delta y \cdot z_{TT}, \Delta y = y_{TT} - y_{TI}$$

The TT joint moves in the intended vertical direction, while the TI joint often moves in the opposite direction, providing stability and refinement to the vertical movement. This difference (Δy) is then multiplied by the z-position of the TT joint to achieve a balance: This calculation compensates for the unique characteristics of thumb motion, where the thumb tends to translate forward and backward.

4.3.2 Mapping Finger Position to Ray Intersection Position. Given the inherent range of movement of the fingers, we can establish boundaries to adjust the scale for plane mapping. We defined an inner boundary based on the actual reachable range along both the x and y axes. We set the input scale ranging from -0.05 m to 0.05 m for x, and from -0.02 m to 0.02 m for y. And each x,y value is scaled with this range and then mapping to the plane size range from 1.67 to 1.67. The unit is meter. In addition, we defined the plane coordinate range at a distance of 2 m from the user. The normalized position is then scaled to fit within the plane size. The origin of the mapping position is the line of sight of the user’s eye. Since the system uses absolute positioning rather than relative mapping, an accelerated gain based on velocity is not feasible. To address this, we implemented transfer functions for plane coordinates, including linear, logistic, quadratic, and interpolation, to facilitate efficient

and dynamic ray control. This approach ensures that users can access the entire plane area at varying rates.

4.3.3 Ray casting from Head. Based on the position derived from the above process, we ensure that the ray can cover the entire range within the field of view (FoV). The ray intersection point is mapped to a virtual plane that is positioned 2m away from the user. The visualized ray extends in the direction of the ray’s position on the plane from the head, allowing the user to interact with elements directly along the path of the ray. We also incorporated head rotation, in which the ray’s position is fixed relative to the HMD but the ray moves in world coordinates. Here, we note that direct control of the ray is managed through finger movements, while head rotations are used to extend the interaction range rather than directly control the interaction itself. The system could provide full control of the ray within and out of the FoV seamlessly and intuitively by leveraging natural head movements with the finger-controlled ray.

5 User Study 1: Evaluation T2IRay Indirect Pointing

The objective of this study is to evaluate the performance of the T2IRay pointing within the near-peripheral FoV (60°), aiming to validate whether T2IRay could cover this area. Note that we constrain head movement during the study since we want to evaluate T2IRay’s microgesture-based tracking performance only. Throughout this study, we aim to answer following research questions:

- **RQ1)** How do the different transfer functions affect the pointing performance? Although existing ray-based systems use a 1-to-1 mapping, it is unclear whether this mapping performs well on our system. So, we explore several transfer functions to identify the impact of different gain for microgesture indirect pointing.
- **RQ2)** How does variation in depth affect pointing performance? In this study, the ray intersection point is mapped onto a virtual plane relative to a reference plane at 2 m which leads to less/more movements for near/far depth planes accordingly. In this way, we answer the practical significance of depth variations for pointing performance which still remains uncertain and requires further investigation.
- **RQ3)** How does variation in target amplitude affect pointing performance? While performance is generally expected to be proportional to target amplitude, the specific effects of different transfer functions on this relationship are unclear. By examining various target amplitudes with each transfer function, we aim to figure out robust interaction range of T2I pointing based on performance in the central FoV region.

The purpose of this study is to measure pointing performance within the primary angle of view and to find design parameters for interaction space and pointing mapping approaches.

5.1 Apparatus and Task

We conducted Fitts’ law study based on the standard ISO 9241-9 and Shannon formulation equation to assess the performance of different mapping methods with various target depths and amplitudes similar to previous work [48]. Participants move the ray to a target

Study 1 Indirect Pointing

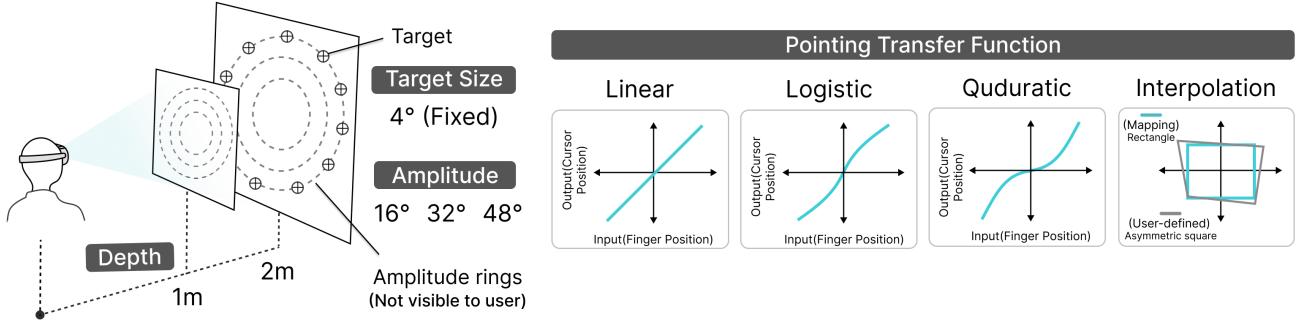


Figure 5: The overview of Fitts' law study. Tasks are designed with two depth planes (1m, 2m) and three target amplitudes (16°, 32°, 48°) with fixed target size (4°). Each transfer function has a different ratio of ray speed.

shown on a flat 2D circular shape as shown in Figure 5. They point to 11 targets that appear from the top side, and the next target appears on the opposite side, and then it continues circularly until the end. The default target is shown in white, which turns green upon successful pointing. Since the ray intersection point is mapped on the 2 m plane, the ray collision on 1 m has a relatively short range compared to 2 m condition, even with the finger movement. We fixed a target size of 4° since we focus on mapping conditions and the effect of different depths. To compensate for the hand tracking error (average fingertip positional error of 1.1cm) [1], we tolerated a relatively large target in this study. First, we set the inner boundary range of plane size to associate with finger movement. The plane size is 1.67 m × 1.67 m with a visual angle of 80°. We adopted *Linear*, *Logistic*, *Quadratic* and *Interpolation*. We map a 2D position (X, Y) on a virtual plane using different transfer functions. Participants used the keyboard pressing to select targets.

Linear. The absolute position value of the finger is linearly amplified to the fixed plane size. The method ensures that the point is proportionally adjusted within the boundaries of the input space.

Logistic. It is the non-linear mapping using the logistic function to the finger position value. After normalizing the finger position value, the logistic function is applied to produce an S-shaped curve that is less sensitive to extreme values. The ray moves slowly at the outer boundaries but more quickly in the center. We used the logistic scaling constant of 1.5, which was empirically chosen not to exceed a linear rate.

$$\text{Output} = \frac{1}{1 + e^{-k \times x}}$$

Quadratic. This function maps ray more aggressively near the outer boundaries. As the ray goes out of the center region, the ray moves significantly faster but much slower around the center region compared to *Logistic*. We used the quadratic scaling constant of 1.3 to ensure the movement does not become too slow.

$$\text{Output} = (x)^k$$

Interpolation. This transfer function involves geometric transformations to maintain spatial relationships in accordance with the user's hand structure. Users were asked to reach their thumb

comfortably toward the corners of a rectangle, and repeated measurements were taken to calculate the average reach. Each corner's coordinates were normalized to the mapped intersection plane and applied homography transformation matrix (H) to reflect their thumb movement tendency shape.

$$\text{Homography} = H \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- Pointing Transfer Function (4): We set *Linear*, *Logistic*, *Quadratic*, and *Interpolation* transfer functions for pointing.
- Target Amplitude (3): We set amplitudes in fixed differences including 16° (8° circle radius), 32° (16° circle radius), and 48° (24° circle radius) to systematically cover the whole range.
- Target Depth (2): We set the target location as 1 m and 2 m in front of the user, which typically represents near and far distances for displaying UIs.

5.2 Participants and Procedure

We recruited 16 participants (10 females, mean age of 29) and all participants were right-handed. We carried out an IRB-approved within-subject study with three independent variables. They first had a practice session to familiarize themselves with the proposed interaction (< 30 minutes). Participants conducted 24 conditions in which they experienced each of the 4 transfer functions × 2 depths × 3 target amplitudes. We counterbalanced the order of the task blocks for 4 different transfer functions across participants. Within each transfer function block, participants went through 6 different conditions of depth and target amplitude. The presentation order is blocked by depth, and amplitude is randomized. We asked participants to rest their hands freely between trials or as needed. We collected 4,224 data points (16 users × 4 transfer functions × 2 depth × 3 target amplitude × 11 trials).

5.3 Measures

- Task Completion Time (TCT): TCT refers to the duration in which a participant completes a task. It is calculated by capturing the moment a target appears and the moment it is successfully selected by the participant.

- Throughput (TP): TP indicates the speed and accuracy of pointing as below:

$$TP = \frac{ID_e}{MT} \quad (1)$$

where ID_e is the task's effective index of difficulty, calculated based on the effective target amplitude (A_e) and effective width (W_e). MT is the mean movement time (same as TCT) recorded over a series of trials. A_e represents the mean of the actual movement target amplitudes across the trials. The formula for ID_e expands as follows:

$$ID_e = \log_2 \left(\frac{A_e}{W_e} + 1 \right) \quad (2)$$

- Effective Width (W_e): W_e is derived from the distribution of the hit point, representing the variability in the precision. Using the effective width helps reduce the variability in TP, which is influenced by changes in TCT and ID_e . W_e is calculated as below:

$$W_e = 4.133 \cdot SD_x \quad (3)$$

where SD_x is the standard deviation of the selection coordinates, and 4.133 is a constant factor.

- Error Rate (ER): ER is calculated as the missing target proportion of trials in a block where the pointing was not successfully made inside the target object.
- Subjective Rating: We used a 7-point Likert-scale questionnaire of subjective ratings on different transfer functions on perceived speed, perceived accuracy, naturalness, fatigue, and overall preference.

5.4 Results

For the analysis, we ignore the first target in all conditions since the distance to the first target is less than the amplitude of the ring. In addition, we apply a filter where a trial time exceeds the mean $+ 3 \times SD$. With this, 6.7% of trials were excluded. We first test the normality and apply the transformation (e.g., Aligned Rank Transform) if the data do not meet the normality. Then, we conducted repeated measures ANOVA with Greenhouse-Geisser corrections if sphericity was violated.

5.4.1 Task Completion Time (TCT). The repeated measures ANOVA revealed that there was no significant main effect of the Transfer function on TCT ($F_{27,73}^{1.85} = 1.62, p = .218, \eta^2 = 0.097$). However, there was a marginally significant effect of Depth on TCT ($F_{15}^1 = 4.03, p = .063, \eta^2 = 0.212$), suggesting that TCT may vary with changes in Depth. There was no significant interaction between the Transfer function and Depth ($F_{21,72}^{1.45} = 0.452, p = .58, \eta^2 = 0.029$), indicating that the influence of the Transfer function on TCT does not depend on Depth. However, Amplitude had a highly significant effect on TCT ($F_{16,49}^{1.1} = 117.35, p < .001, \eta^2 = 0.887$), indicating that changes in Amplitude substantially influenced TCT. However, an interaction effect was not found between Transfer function and Amplitude ($F_{32,98}^{2.2} = 2.47, p = .095, \eta^2 = 0.14$), indicating that the effect of Amplitude on TCT was consistent across transfer functions.

5.4.2 Throughput (TP). A significant main effect of Transfer function was observed on TP ($F_{30,54}^{2.04} = 3.78, p = .03, \eta^2 = 0.201$), indicating that different transfer functions resulted in different levels of TP. There was no significant effect of Depth on TP ($F_{15}^1 = 3.09, p = .099, \eta^2 = 0.171$), and the interaction between the Transfer function and Depth was not significant ($F_{25,03}^{1.67} = 0.17, p = .802, \eta^2 = 0.011$). There was a significant effect of Amplitude on TP ($F_{15,85}^{1.06} = 7.16, p = .001, \eta^2 = 0.323$). Moreover, the interaction between Transfer function and Amplitude was significant ($F_{90}^6 = 2.33, p = .03, \eta^2 = 0.135$), suggesting that the effect of transfer function on TP varies according to the Amplitude.

5.4.3 Effective Width (W_e). A significant main effect of Transfer function on W_e ($F_{31,64}^{2.10} = 5.26, p = .01, \eta^2 = 0.260$), demonstrating that the transfer functions had a significant impact on W_e . There was no significant effect of Depth on W_e ($F_{15}^1 = 0.11, p = .743, \eta^2 = 0.007$). However, a marginally significant interaction between Transfer function and Depth was found ($F_{28,46}^{1.90} = 2.83, p = .078, \eta^2 = 0.159$), suggesting that the impact of transfer function on W_e may vary depending on the Depth. The main effect of Amplitude did not have a significant effect on W_e ($F_{19,76}^{1.32} = 1.00, p = .355, \eta^2 = 0.062$). The interaction between the Transfer function and Amplitude was also not significant ($F_{51,08}^{3.41} = 1.59, p = .198, \eta^2 = 0.095$).

5.4.4 Error Rate (ER). A significant main effect of Transfer Function was observed on ER ($F(3, 20) = 1.08, p = .381$), indicating that different transfer functions did not result in significantly different levels of error rate. There was no significant effect of Depth on ER ($F(1, 18) = 0.37, p = .551$), and the interaction between Depth and Amplitude was not significant ($F(2, 18) = 2.04, p = .159$). However, the main effect of Amplitude on ER approached significance ($F(2, 18) = 3.44, p = .054$), suggesting that differences in Amplitude may have a marginal effect on the error rate.

5.4.5 Fitts' Law Model and ID. Each transfer function is modeled as follows, as shown in Figure 7a.: $MT = 1.555 + 0.602ID$ ($R^2 = 0.978$) for Linear, $MT = 1.163 + 0.608ID$ ($R^2 = 0.964$) for Logistic, $MT = 0.342 + 1.096ID$ ($R^2 = 0.995$) for Quadratic, and $MT = 0.799 + 0.873ID$ ($R^2 = 0.999$) for Interpolation. The overall transfer functions showed a good fit to Fitts' law which reflects a strong dependency of TCT on the difficulty level. Linear ($R^2 = 0.978$) and Interpolation ($R^2 = 0.999$) showed a high degree of fit. The Quadratic also reported a good fit to Fitts' law ($R^2 = 0.995$). However, the model's higher slope of $b = 1.096$ suggests a steeper increase in task completion time with increasing difficulty. This may indicate that Quadratic is particularly sensitive to changes in task difficulty. The Logistic also demonstrated a good fit ($R^2 = 0.964$) and less sensitivity compared to Quadratic. Compared to Linear, Logistics exhibited similar rate of TCT increase upon higher ID.

5.4.6 Subjective Rating. All transfer functions have relatively large standard deviations, indicating significant variability in participants' opinions regarding all aspects (Figure 7 b). Nevertheless, the Logistic has the highest mean score for perceived accuracy, fatigue, and overall preference. And many participants still favored the Logistic over other methods. This suggests that while not conclusively superior, the Logistic provided a generally positive experience for participants.

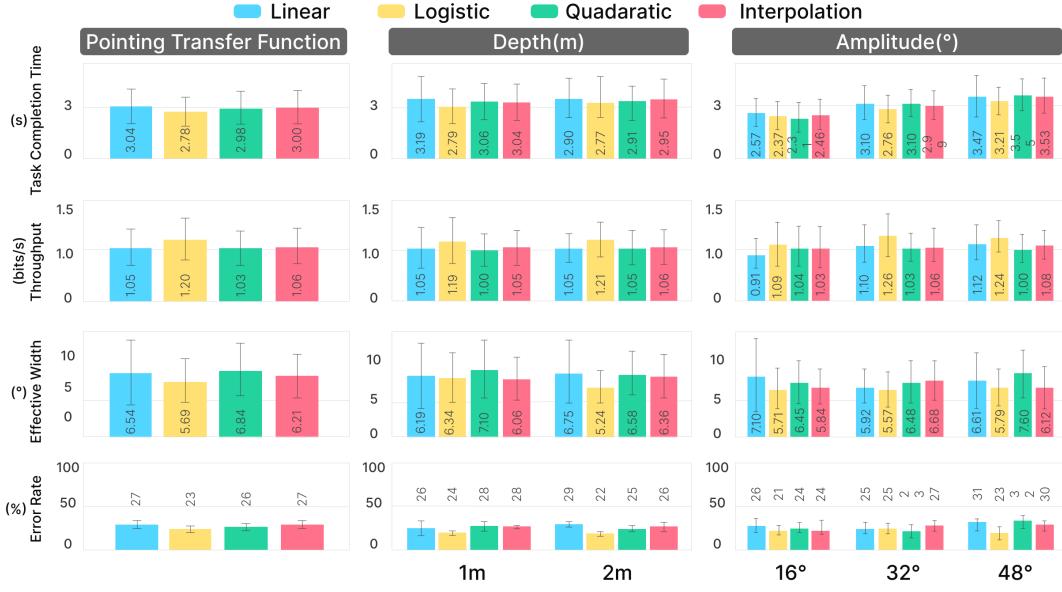


Figure 6: The results of Fitts' law study. The mean values for task completion time, throughput, effective width, and error rate were computed for each transfer function, depth per transfer function, and amplitude per transfer function.

5.4.7 Endpoint distribution of cursor. We examined the endpoint distribution of the selection cursor. Figure 7c shows the intersection points are concentrated towards the center of the target ring. The points are more sparse in the downward target and usually concentrated horizontally but deviate vertically, indicating greater error in the vertical axis. We observed that accuracy decreases with increasing amplitude as the thumb moves further away from the center, particularly along the diagonal directions. In general, the T2IRay appears to present a challenge for accurate targeting in areas requiring considerable finger reach and exertion.

5.5 Discussion

In this section, we discuss our findings from Study 1. Our results indicate that the *Logistic* transfer function offers the best performance. Additionally, we propose an adaptive approach to account for the effects of T2I pointing in system design and usage. To summarize the result, the transfer function significantly affects TP and W_e . Depth has a marginally significant effect on TCT but not on TP or W_e . Amplitude has a highly significant effect on TCT and a significant effect on TP. Interactions between transfer function and Amplitude are significant for TP.

5.5.1 RQ1) How do the different transfer functions affect the pointing performance? As we proposed an approach of microgestures pointing, we had the ultimate research question that microgestures could effectively cover the area of interest in the view angle. We found that there are differences with TCT, TP, and W_e depending on the applied transfer function for depth and amplitude. Remarkably, the *Logistic* had a significant difference as compared to other methods, suggesting that it would enable relatively fast and accurate pointing. We found that while users could quickly move the ray to the target, they often struggled to make fine adjustments near

the target. This resulted in longer selection times. Also, the ray sometimes moved off the target even while correctly positioned due to errors in built-in hand tracking, contributing to higher error rates. We believe results on point and click behaviors of users from this research could provide valuable insights into overcoming this sensor inaccuracy.

5.5.2 RQ2) How does variation in depth affect pointing performance? The influence of the transfer function at different depth levels was not very significant. The transfer function which is slower in the early stages such as *Logistic* and *Quadratic* showed lower W_e as the distance increased. However, W_e increased with depth at relatively faster functions such as *Linear* and *Interpolation*. Also, the error rates increased with depth. It seems that we map the ray intersection at 2 m so that even though the ray moves at the same angle, the same movement allows for finer adjustments at a closer 1 m plane. This could lead to more accurate pointing. This suggests that depth may influence how precisely users approach the target and maintain accuracy as depth changes. We believe that using a finger to access the UI beyond arm's reach seems to provide a consistent experience. However, we recommend allowing a slightly wider target range for closer targets. During the user study, participants reported feeling more frustrated when performing the task at 1 m. While they noted improved accuracy, the task felt slower, suggesting that adaptive speed adjustment based on depth might be a further consideration.

5.5.3 RQ3) How does variation in amplitude affect pointing performance? The amplitude of the target deals with targets at different distances. We examined how user performance varied with different target amplitudes within the expected 60-degree range. As expected, as the distance increased, so did the time required to complete the task. While TCT and TP had significant impacts. The significant interaction between transfer function \times amplitude in

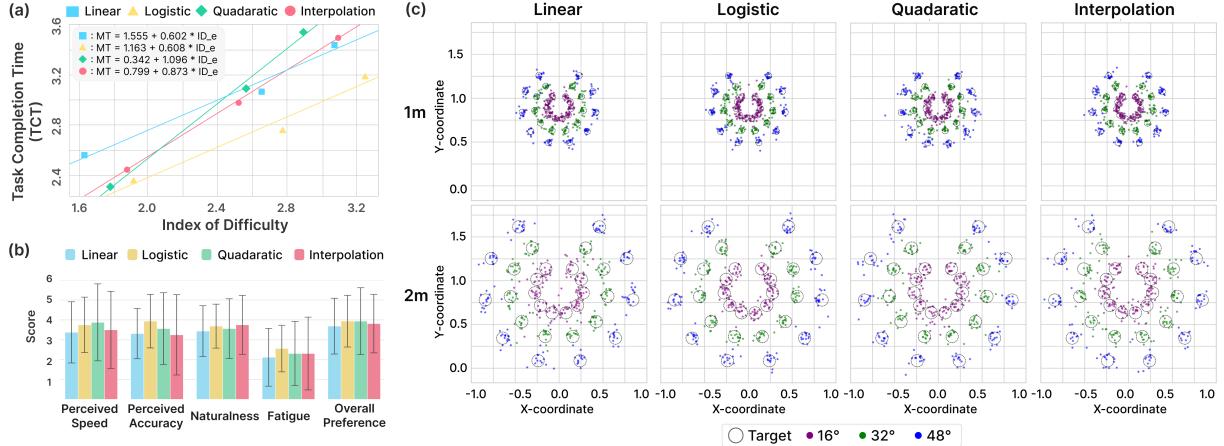


Figure 7: (a) Fitts' law models for various transfer functions in T2IRay. (b) Subjective ratings of different transfer functions. The higher scores indicate a higher matching speed with the finger, higher accuracy, more natural, less fatigue, and a better overall experience. (c) Endpoint distribution of cursor. Each dot indicates the intersection point on the intersection screen upon selection. The mapping intersection plane remains in 2 m, but a 1 m temporary intersection plane was provided for reporting.

TP suggests that the effect of the transfer function and amplitude on TP is related to changes in speed depending on the mapping position. For example, the movement speed of the *Linear* remained constant at 8° and 24°. In contrast, it started slow and sped up as it approached 24° in the *Quadratic*. Similarly, the *Logistic* started slow and accelerated mid-range and then slowed again at 24°. This underscores the need to adjust the appropriate speeds based on the target distance from the center of the mapping to the maximum.

5.5.4 Comparisons with Prior Studies.

Performance. Direct pointing approach like raycasting (Hand-ray) [48] showed as $MT = -0.27 + 0.88ID, R^2 = 0.97$). In comparison, our techniques also demonstrate a good fit (R^2) range in 0.953 and 0.999, indicating that they enable predictable pointing. However, *Quadratic* exhibits a higher slope ($b = 1.096$), suggesting greater sensitivity for more complex tasks. Meanwhile, *Linear* ($b = 0.602$) and *Logistic* ($b = 0.608$) with lower slopes indicate less sensitivity. Raycasting achieves higher TP at the same ID_e , but T2IRay maintains a good fit and comparable sensitivity.

Although the experimental conditions were not identical, we compared the results for context under similar conditions (8° and 15° radius, corresponding to 16° and 32° in our study). These are referred to as small amplitude (SA) and large amplitude (LA), respectively. T2IRay is presented first in each comparison. Regarding TCT, T2IRay showed longer times in SA (2.37 s vs. 2.13 s) but shorter times in LA (2.76 s vs. 2.81 s). TP was lower in SA (1.09 vs. 1.43) but comparable performance in LA (1.26 vs. 1.33). About the W_e , T2IRay showed less accuracy in SA (5.71° vs. 4.23°), but achieved more accuracy in LA (5.57° vs. 6.61°). These results highlight the strength of microgesture pointing with a larger amplitude that favors broader movements. Overall, T2IRay performs comparably well for larger amplitudes, but improvements are needed for smaller amplitudes.

Factors for microgesture pointing. Our findings provide an understanding of 3D pointing with microgesture by examining the impact

of transfer function, target depth, and amplitude. Transfer function related to gain significantly affected performance, aligning with the previous finding [7]. Similarly, the effect of target amplitude is unsurprising, as it is well demonstrated amplitude tendency on TCT in Fitts' law. However, depth works differently for the microgesture pointing. Previously, depth has been identified as a crucial factor in 3D pointing, particularly during the ballistic and homing phases [26]. However, we found no significant effects of depth for microgesture pointing. Instead, our results showed a close relationship to screen-projected techniques [44] where depth has minimal impact if the depth is constant. This may be attributed to the small motion in microgestures, which reduces motor control demands. In contrast, mid-air pointing involves large movements that become harder to control as gain increases under deeper conditions. Therefore, we suspect that microgesture minimizes perceived depth variation, even when depth could alter the perceived gain [26].

5.6 Observation

We have not strictly distinguished the contact state of the thumb on the index finger. Rather, T2IRay enabled the thumb to control the ray in a flexible manner, either by resting on the index finger or lifting above it. While the latter were perceived as more effective for quickly reaching a target, maintaining contact with the index finger surface was preferred for finer adjustments. Some participants reported that selections were more comfortable with lifting, but felt more confident in precision when moving their thumb supported by the index finger. Here, the index finger acts as a tracking surface and offers proprioceptive tactile feedback to track motion paths. We believe this transition is perceived as continuous without explicit mode switching. And allowing interaction in either state provides flexibility and advantages for most users. In addition, participants were observed small adjustments by tilting their fingers slightly, rather than using the traditional sliding or dragging motion. It was also found that users could access all desired interaction regions

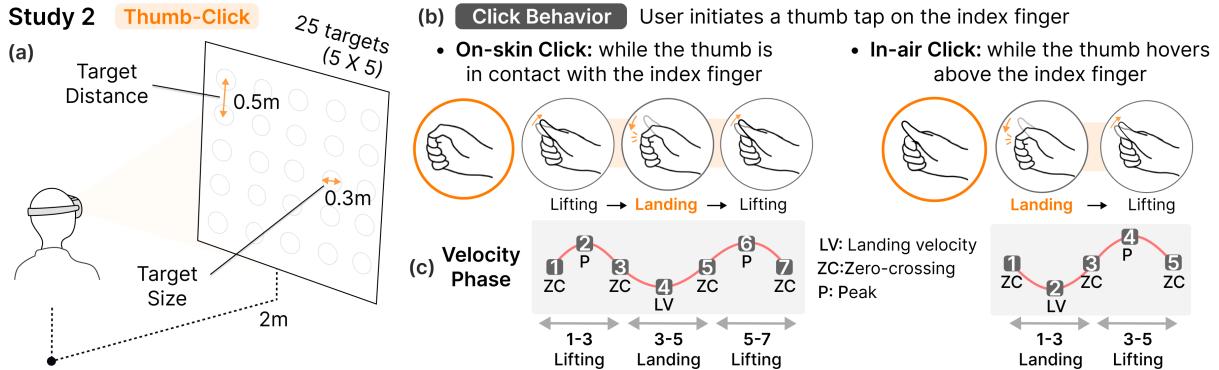


Figure 8: (a) The overview of Study 2. (b) Possible behavior of thumb-click: *On-skin Click* (Left) and *In-air Click* (Right). (c) The conceptual illustrations for velocity and time points of movement. The odd-numbered points indicate zero crossings. Even numbered points indicate extreme points of velocity. In *On-skin*, three phases are identified: first lifting, landing, and second lifting, while there are two phases in *In-air*: landing and lifting. The LV marks the onset of the landing, while P indicates the lifting onset. The interval between ZCs indicates each phase: lifting and landing.

Table 1: The kinematic metrics used for analysis. Each metric serves a distinct purpose: Velocity (speed), SPARC (smoothness), and Duration & Distance (responsiveness).

Metric	Description
Velocity	The speed at the moment of landing onset.
SPARC [3] (Spectral Arc Length)	A smoothness metric that evaluates movement efficiency by analyzing the Fourier magnitude spectrum of the movement speed profile. Larger absolute value indicates less smoothness.
Duration	The total time required to complete a phase of movement.
Distance	The difference in distance between the start and end of the phase.

within the given range without needing to move their hands by just resting on the desk or chair. However, some anatomical limitations were noted, as users had difficulty reaching specific areas due to natural physical constraints. Overall, participants favored horizontal movements over vertical ones. One participant (P9) noted that the interaction felt somewhat like holding a physical controller, but the sensation of moving an ‘empty’ hand felt awkward.

6 User Study 2: Investigating T2IRay Thumb-click Behavior

To validate T2IRay as an indirect pointing method, pointing and selection should be assessed. For pointing, Fitts' Law study was conducted under mapping conditions. In selection, it would be beneficial to consider the characteristics of dynamic gestures inherent in system design, given that a tap is a typical gesture for selection in finger-centric systems. However, existing research has not sufficiently explored how tap characteristics vary across conditions. In this study, we investigate the kinematic features of the thumb during tapping, referred to as “thumb-click”. Although we did not explicitly implement tap-based selection, we explored whether consistent tap could be performed under different conditions and how observed characteristics could inform an effective selection mechanism. Here, we chose the click behavior and target directions as varying conditions. As a fluid transition between two states during

pointing observed in Section 5.6, two possible behaviors could be defined by the state of the click.

- (1) **On-skin Click:** User initiates a thumb tap on the index finger while the thumb is in contact with the index finger.
- (2) **In-air Click:** User initiates a thumb tap on the index finger while the thumb hovers above the index finger.

As continuous pointing involves varying target directions, we include directional variations that induce varied thumb positions. For this, we collected kinematic data of tap gesture during pointing. We aim to provide a comprehensive understanding of thumb click characteristics in terms of speed, smoothness, and responsiveness, which contributes to refining the implementation of selection in T2I systems. Therefore, this study was geared toward two key questions:

- **RQ1)** What are the key kinematic characteristics of thumb-click across different behaviors?
- **RQ2)** How does the kinematic of the thumb-click change with thumb position across different target locations?

6.1 Apparatus and Task

We carried out the Wizard of Oz method, where the system does not detect the selection, but it is designed to make the user feel as though a selection has occurred. We arranged the circle targets in a 5×5 grid within the FoV to guide the thumb’s position toward specific directions. The size of each target is 0.3 m, and the spacing between

adjacent targets is 0.5 m. We collected the position and velocity of the thumb tip joint based on our T2IRay's local coordinate system. Here, we categorized the click behavior into two types (*On-skin* and *In-air*). The difference between these behaviors is whether pointing is performed with the thumb on the finger or above the index finger. In the *On-skin*, the thumb has to be lifted and landed on the index finger and lifted again to complete a thumb-click. On the other hand, in *In-air*, the user can immediately land the thumb and then lift it back to its original position. Note that we collected position data in T2IRay's local coordinate.

6.2 Participants and Procedure

We conducted this study with the same participants as in the earlier study. We collected a total of 3,200 data points (16 users \times 8 sessions \times 25 trials). The order of the targets was randomized, and the target circle was displayed in red. Participants were instructed to point ray as close as possible to the target and press the spacebar upon tapping. Each click was collected for 3 seconds based on keystroke onset. Since we analyzed data by kinematic features, the keystroke did not have to correspond to the actual click moment. The varying target locations were used to reflect variance in direction of the thumb during thumb-click. We divided sessions into two blocks corresponding to different thumb-click behaviors (Figure 8b). We dedicated four sessions for *On-skin*, and the other four for *In-air*. We asked participants to maintain index finger posture to ensure stable coordinates. Participants could reach all targets within the range of motion, but peripheral targets required more effort than others. While slight movement was unavoidable, utilizing the midpoint in coordinate addressed this case.

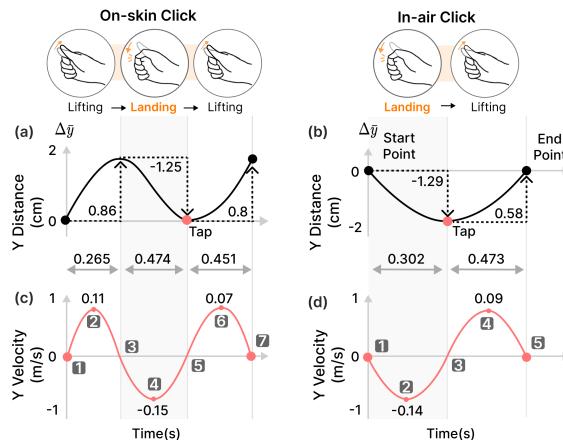


Figure 9: Overall analysis of thumb tip movement during thumb-click. The graphs illustrate the overall behavioral pattern based on the study of both the Y-distance (a and b) and Y-velocity (c and d) (conceptual illustrations for general movement patterns.)

6.3 Results

Our investigation focused on the y-axis features, as vertical movement is particularly relevant in tapping. The metrics are as shown

in Table 1. Each point in the velocity profile (Figure 8c) is identified to determine each phase and onset. The LV point corresponds to landing velocity, entire phase is used for SPARC. And each interval between ZC indicates the submovement of thumb-click.

6.3.1 Overall Kinematic Behaviors. The movements show distinctive patterns as shown in Figure 9. In the *On-skin*, the Y distance graph shows that the finger initially lifts to $\Delta\bar{y} = 0.86$ cm before descending sharply to $\Delta\bar{y} = -1.29$ cm during the landing period, followed by a rise to $\Delta\bar{y} = 0.58$ cm in the second lifting period. The duration of each period is approximately 0.265 s for the first lifting, 0.474 s for the landing, and 0.451 s for the second lifting. The corresponding velocity indicates that the finger reaches a peak upward velocity of 0.11 m/s during the initial lift, drops sharply to $\Delta\bar{y} = 0.15$ m/s during the landing, and then rises again to 0.07 m/s in the second lift. In the *In-air*, it descends to -1.25 cm during landing and rises to $\Delta\bar{y} = 0.8$ cm in the lifting period. The landing duration is 0.302 s and the lifting duration is 0.473 s. The peak downward velocity is -0.14 m/s and the upward velocity is 0.09 m/s.

6.3.2 Velocity. For *On-skin* landing (Figure 10b), the mean velocity was -0.15 m/s (range: -0.23~0.10 m/s, SD=0.03) while -0.14 m/s (range: -0.24~0.08 m/s, SD=0.05) in *In-air* (Figure 10c). In both conditions, velocity decreases as the thumb moves towards the center and upper region, while it increases diagonally towards the lower region. Interestingly, *In-air* exhibits relatively similar speeds horizontally but shows more abrupt changes vertically. In contrast, *On-skin* shows the opposite trend where larger speed variations horizontally and gradual changes vertically. In general, *On-skin* are faster but lower regions are faster in *In-air*.

6.3.3 SPARC. The mean value of *On-skin* including whole click was -2.23 (range: -2.31~-2.12, SD=0.06) (Figure 10d). For *In-air* (Figure 10e), the mean value was -2.45 (range: -2.70~-2.35, SD=0.10), reflecting a less reliable smoothness profile. *In-air* exhibits greater overall variability and vertical patterns show sharper shifts compared to *On-skin*. In contrast, *On-skin* relatively displays smoother and consistent at the center-lower region. This highlights that *On-skin* tends to more stable and controlled movements.

6.3.4 Duration. For the first lifting in *On-skin* (Figure 11a), $\Delta\bar{t}$ was 0.27 s (max=0.33, min=0.21, SD=0.02). For the landing, the $\Delta\bar{t}$ was 0.47 s (max=0.69, the min=0.33, SD=0.11). Finally, for the second lifting period, $\Delta\bar{t}$ was 0.45 s (max=0.69, the min=0.30, SD=0.08). In *In-air* (Figure 11b), for the landing, $\Delta\bar{t}$ was 0.30 s (max=0.35, min=0.25, SD = 0.03). For lifting, $\Delta\bar{t}$ was 0.47 s (max=0.72, min=0.31, SD=0.11). In *On-skin* lifting phase, the duration is generally uniformly distributed. For *On-skin* landing phase, the duration increases in the lower-left region. Then the duration increases towards the upper-left region in the second lifting phase. For *In-air*, the overall duration is relatively consistent with the trend of shorter duration at right in the landing phase. However, the trend of longer duration was observed at the upper-right region in the lifting phase.

6.3.5 Distance. In *On-skin* first lifting (Figure 11c), $\bar{\Delta}d$ was 0.88 cm (max=1.19, min=0.64, SD=0.16). For the landing, the mean $\bar{\Delta}d$ was -1.29 cm (absolute max=-1.86, absolute min=-0.80, SD=0.34). Finally, for the second lifting, the $\bar{\Delta}d$ was 0.57 cm (max=1.19, min=0.23,

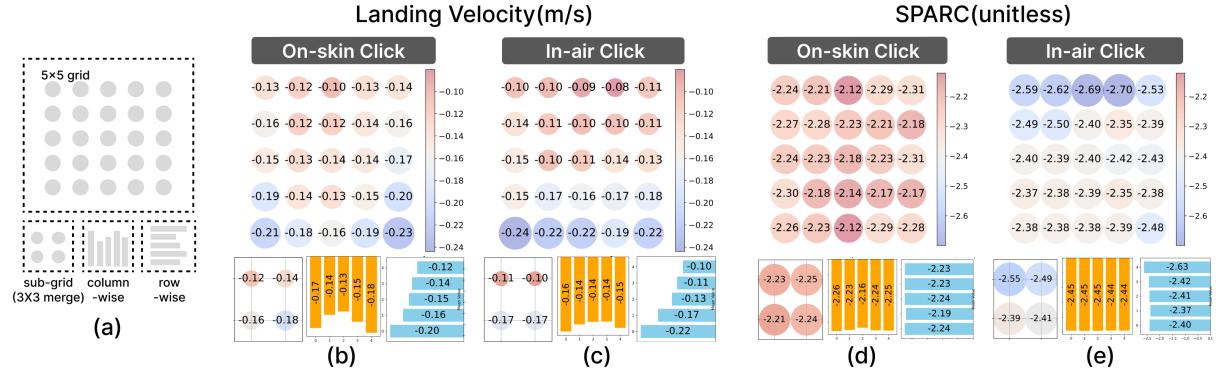


Figure 10: (a) Results are reported in two behaviors by all grids with sub-grid, column-wise, and row-wise. (b, c) The average velocity along the y-axis for landing onset a thumb-click. (d, e) The average SPARC of the whole movement.

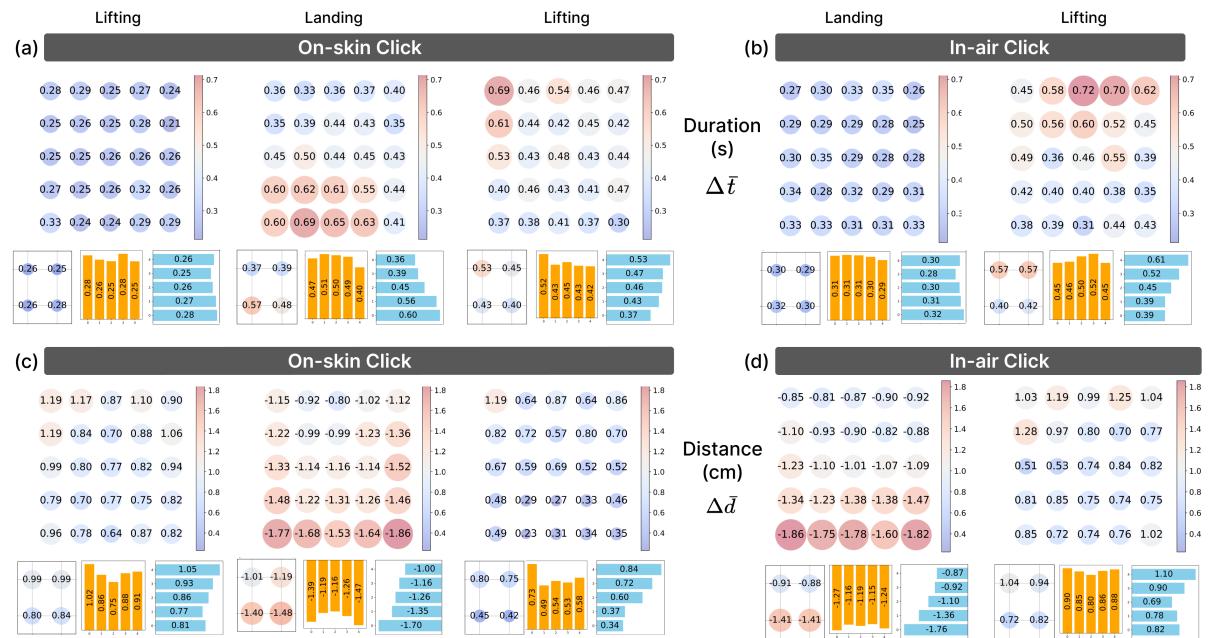


Figure 11: The period is divided into three phases for *On-skin* behavior (Lifting, Landing, and Lifting) and two phases for *In-air* behavior (Landing and Lifting). (a, b) The average duration for different periods of finger movements in a thumb-click across a 5×5 target grid. (c, d) The average distance along the y-axis for different periods (Lifting, Landing, and Lifting) of finger movements in a thumb-click across a 5×5 target grid.

SD=0.23). In *In-air* (Figure 11d) landing, the $\bar{\Delta}d$ was -1.20 cm (absolute max=-1.86, absolute min=-0.81, SD = 0.34). For lifting, the $\bar{\Delta}d$ was 0.86 cm (max=1.28, min=0.51, SD=0.19). In *On-skin* lifting phase, the distance increases at upper-diagonal region. And it increases in the lower diagonal region in the landing phase. The distance increases in the second lifting phase as it moves toward the upper left region. For *In-air*, the trend was shown that large distance at lower region in the landing phase. The overall distance is relatively consistent with the trend of large distance at the upper-left region in the lifting phase.

6.4 Discussion

Our result was examined to derive insights for designing thumb-based selection. The behavior division and grid-based targeting task are designed to account for variations in tapping during pointing. Also, the used posture and tap gesture exhibit common features as adopted in previous thumb-based interactions (e.g., using different side index regions as interfaces [38, 45, 57] or finger-centric interaction [16, 29, 47, 55]). While the results are focused on T2I systems, they also provide implications applicable to the design of thumb-based interaction systems.

6.4.1 RQ1) What are the key kinematic characteristics of thumb-click across different behaviors?

Speed. The patterns were similar (e.g., speed increased at lower or sides). In *In-air*, participants maintained a more consistent speed horizontally. Overall, *On-skin* was performed faster likely due to the influence of initial deliberate tapping intentions in continuous movements.

Smoothness. *On-skin* was more stable although more submovements are included. A notable difference was that while *On-skin* remained consistent overall, *In-air* were less stable in the upper region. This highlights the importance of support provided by the index finger for stable control.

Responsiveness. In *In-air*, the landing duration is much shorter (0.302 s < 0.474 s (*On-skin*)), probably because the action involves only the tapping without the preceding steps. We also noticed that users tend to lift their finger at the target point, followed by a brief pause. Regarding distance, the landing was similar across both behaviors. However, *On-skin* involved less lifting afterward, likely due to the natural resting position of the finger on the surface.

Summary. *On-skin* showed greater stability and consistent speed by utilizing stable support from the index finger. On the other hand, *In-air* was less stable while enabling faster and less constrained movements. We also observed that duration and distance varied even within the same landing state. For example, the *On-skin* landing duration is longer in the lower-left side, while being uniform in *In-air* landing. We suspect these distinct patterns are influenced by the preceding movement. These indicate that the characteristic of thumb-click is influenced by the specific dynamics of each behavior.

6.4.2 RQ2) How does the kinematic of the thumb-click change with thumb position across different target locations?

Speed. The difference between the upper and lower regions was nearly twofold. This may be due to the thumb already being stretched where the motor control is required. Similarly, The faster movements at the lower region indicate greater assertiveness when the thumb is bent.

Smoothness. The smoothest movements were observed in the center region, where the thumb's involvement in bending or stretching was minimal. This indicates that stable tapping is influenced by the thumb's range of motion. In *In-air*, the greatest instability is observed at the upper region, where thumb extension led to a considerable submovement.

Responsiveness. Lifting (after landing) duration showed opposite trends in horizontal. One possible reason is that it takes longer when it is difficult to keep the natural thumb extension depending on finger position during tapping. Regarding distance, it took a larger movement when the thumb landed in the lower region. Distance increased from the center outward and from upper to lower, likely due to the index finger acting as a physical limit. Lifting in the upper and diagonal directions resulted in greater distances, which is relevant to finger bending. Usually, the motion tends to be influenced by diagonal, likely caused by the exertion of finger control.

Summary. Upper region effect slow velocity due to the thumb's stretched position, while bent thumbs aiming at lower region enabled faster motions. The center region had the smoothest movements with minimal motor control. Lifting durations grew longer at upper regions, reflecting challenges in maintaining natural thumb extension. Distance increased outward from the center, especially during effortful diagonal movements.

6.4.3 Implication. We can summarize several reasons for tendencies. First, physical surface constraints limit thumb movement when reaching lower targets due to the index finger and hand structure. In contrast, reaching far from the center required more time and effort without a physical surface. In addition, the motor capability of fingers would be the reason. If the user exerts more effort to reach the target, it reduces the motion range, resulting in more time with greater movements. Additionally, the range of *On-skin* was inherently smaller compared to *In-air* because of the limitations when the fingers remain in contact. However, this constraint resulted in a stable click. Finally, the tactile feedback from the index finger of *On-skin* allowed users to click more consistently. This was due to the sense of touch that the skin provided, making it easier to detect and confirm interactions with controlled movement.

7 Design Guideline

7.1 Precise T2I Indirect Pointing

Acceptable Pointing Transfer Function for Microgesture Control. The *Linear* (1 cm of finger movement corresponding to 33.4 cm on the screen) was slightly fast to move the ray than other approaches. Based on the results, *Logistic* performed better than *Linear*. This indicates that using a lower mapping speed compared to the current ratio with logistic scaling or even less would be proper. Still, we observed that maintaining a faster ratio than *Quadratic* would be necessary. Moreover, the transfer function should avoid spatial distortion, as users were sometimes confused with unexpected movement, which caused degraded performance in *Interpolation*.

Effective View Angle. For view angle up to 80° (mapping range, including the expected usable range of 60°), we observed the trend of performance degradation predicted by the index of difficulty (Figure 7). Therefore, covering this range solely with microgestures is impractical. To maintain consistent accuracy, we suggest not extending the mapping coverage beyond 60°. A narrower mapping angle could enhance resolution and accuracy. In this context, head engagement could provide sufficient coverage within 48° or less, compensating for the narrower range.

Considerations for Depth-adaptive Pointing. As depth influences the TCT marginally, it is important to adjust the transfer function based on the depth at which the user is currently interacting. At less than 2 m, a more rapid response rate might be needed. Because using the same response rate across different depths would cause misalignment with user expectations due to the fixed intersection point at 2 m. So, it is recommended to allow a slightly wider target range for closer distance to provide a more responsive pointing. However, some participants reported that perceived performance in depth was quite significant, suggesting that drastic adjustments might not be advisable.

Unnecessary Additional Finger Adjustments. Participants rarely used the full thumb range, indicating the system could support pointing with skewed finger movement. Reflecting user range might lead to unexpected pointing as shown in *Interpolation* result. The theoretical and actual range of motion might differ, so careful consideration of this concept is essential. Many participants tended to prefer pointing with their thumb above the index finger rather than in contact with it. We recommend avoiding additional complexity, such as requiring contact mode to activate the microgesture interaction. State differentiation for thumb-click could improve usability but may not be necessary for pointing.

7.2 Robust T2I Thumb-Click

Real-Time Movement Correction. Our results suggest the use of predefined values based on finger positions. While the current predefined grid helps to capture general tendencies, more fine-tuned regions will be needed to account for subtle differences in natural finger movements. However, using delta information could enable a correction mechanism with real-time data by implementing hysteresis buffers, such as buffer duration. It could minimize unintentional cursor movements with corrections of each finger position.

Adjustment of Input Sensitivity based on Spatial Information. It can also be used to adjust the sensitivity of the input system based on the user's movement along the target position which induces finger position. For example, in areas where the finger is more erratic due to motor capability (e.g., distant or lower region), decrease sensitivity to minimize detection of small, unintended movements. Conversely, it increases sensitivity in areas that are more controlled and small near the center region. It may be helpful to filter out unintentional movements.

Leveraging Natural Motion Patterns. We observed motion patterns that could be used to adjust detection parameters. For example, faster movements from center to side and down are often seen during landing, while the force applied during the previous landing affects the duration and magnitude of the lift. These insights can guide threshold settings or data collection windows during the learning phase, improving interaction design by reducing errors and increasing detection accuracy. It may be beneficial to distinguish between *On-skin* and *In-air* click behaviors to optimize parameters such as time segment and distance threshold settings.

8 Limitation and Future Work

Direct Comparisons with previous pointing methods. T2IRay introduces the challenge of controlling larger areas with smaller movements compared to prior indirect pointing methods. Also, direct comparisons with prior work may not be feasible due to differences in system design (e.g., areas of use, mapping ratios, and fixed plane), placing them beyond the intended scope. Although this study focused on independently investigating T2IRay and the comparison is included in Section 5.5.4, they are insufficient to fully understand system performance. Further studies will focus on standardized tasks and refined metrics for comprehensive evaluation.

User Study. In this study, we propose exploring a new realm of microgesture raycasting interaction. Our focus lies in investigating the optimal mapping method of thumb movement on the index finger. This may not fully encompass all potential interactions or account for user's preferences. The effectiveness of the transfer function might vary depending on hand dominance, hand sizes, dexterity, and ergonomics. Also, the participants in the study were set in a sitting, leaving the state effects of standing or walking unclear. Therefore, we recognize the need for further research to address broader usability concerns for considering user requirements.

Head Engagement. We excluded head engagement from the study to avoid gaze-related factors. Nevertheless, we believe that combining head and finger inputs could enable faster and wider pointing. It would also be valuable to evaluate head engagement to understand its practical effectiveness and potential advantages. Although we did not directly compare with traditional free-hand raycasting, we observed a reduction in hand movement in that users performed tasks effectively even with their hands in a fixed position. However, additional experiments may help to further understand and confirm the significance of this observation.

Additional System Consideration. Since the current system only provides a basic ray and intersection point, additional visual cues help users adjust fingers for better accuracy. Also, it would be interesting to explore alternative selection methods for microgesture pointing such as dwell time or other gestures, while clicking is the common intuitive gesture. For greater stability, a stabilization method such as a pseudo-static mechanism could be valuable, especially when the ray moves quickly or is unstable near the target. It gradually reduces the ray's sensitivity when movement is detected as minimal or static, improving accuracy and user experience.

9 Conclusion

In this paper, we introduced T2IRay, a thumb-to-index-based indirect pointing technique for AR/VR. We propose a novel way of utilizing the thumb-index relationship to enable continuous input with microgesture. To provide practical design guidelines for T2IRay, we investigated both indirect pointing and thumb-clicks mechanisms. For the indirect pointing, we formulated Fitts' law models for various transfer functions and found that *Logistic* performed superior to other approaches. In terms of thumb-click, we observed distinctive thumb motion patterns across click behaviors and target positions. Based on these findings, we propose design guidelines for both indirect pointing and thumb-click to support robust thumb-to-index interaction. We believe that our work highlights the potential of microgesture as full continuous input to further advance the current state of hand interaction.

Acknowledgments

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2024-RS-2024-00436398) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation) and the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (RS-2023-00210001).

References

- [1] Diar Abdulkarim, Massimiliano Di Luca, Poppy Aves, Mohamed Maaroufi, Sang-Hoon Yeo, R. Chris Miall, Peter Holland, and Joeseph M. Galea. 2024. A methodological framework to assess the accuracy of virtual reality hand-tracking systems: A case study with the Meta Quest 2. *Behavior Research Methods* 56, 2 (01 Feb 2024), 1052–1063. <https://doi.org/10.3758/s13428-022-02051-8>
- [2] Apple Inc. 2023. Apple Vision Pro. <https://www.apple.com/apple-vision-pro/> Accessed: September 1, 2024.
- [3] S Balasubramanian, A Melendez-Calderon, and E Burdet. 2012. A robust and sensitive metric for quantifying movement smoothness. *IEEE Trans. Biomed. Eng.* 59, 8 (Aug. 2012), 2126–2136.
- [4] Sandra Bardot, Surya Rawat, Duy Thai Nguyen, Sawyer Rempel, Huizhe Zheng, Bradley Rey, Jun Li, Kevin Fan, Da Yuan Huang, Wei Li, and Pourang Irani. 2021. ARO: Exploring the Design of Smart-Ring Interactions for Encumbered Hands. In *Proceedings of MobileHCI 2021 - ACM International Conference on Mobile Human-Computer Interaction: Mobile Apart, MobileTogether*. Association for Computing Machinery, Inc. <https://doi.org/10.1145/3447526.3472037>
- [5] Roger Boldu, Alexandra Dancu, Denys J.C. Matthies, Pablo Gallego Cascón, Shanaka Ransir, and Suranga Nanayakkara. 2018. Thumb-in-motion: Evaluating thumb-to-ring microgestures for athletic activity. In *SUI 2018 - Proceedings of the Symposium on Spatial User Interaction*. Association for Computing Machinery, Inc. 150–157. <https://doi.org/10.1145/3267782.3267796>
- [6] Eugenie Brasier, Olivier Chapuis, Nicolas Ferey, Jeanne Vezien, and Caroline Appert. 2020. ARPads: Mid-air Indirect Input for Augmented Reality. In *Proceedings - 2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR) 2020*. Institute of Electrical and Electronics Engineers Inc., 332–343. <https://doi.org/10.1109/ISMAR50242.2020.00060>
- [7] Géry Casiez, Daniel Vogel, Ravin Balakrishnan, and Andy Cockburn. 2008. The Impact of Control-Display Gain on User Performance in Pointing Tasks. *Human-Computer Interaction* 23, 3 (2008), 215–250. <https://doi.org/10.1080/07370020802278163>
- [8] Adrien Chaffageon Caillet, Alix Goguely, and Laurence Nigay. 2023. 3D Selection in Mixed Reality: Designing a Two-Phase Technique To Reduce Fatigue. In *2023 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 800–809. <https://doi.org/10.1109/ISMAR59233.2023.00095>
- [9] Edwin Chan, Teddy Seyed, Wolfgang Stuerzlinger, Xing-Dong Yang, and Frank Maurer. 2016. User Elicitation on Single-hand Microgestures. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) (CHI '16). Association for Computing Machinery, New York, NY, USA, 3403–3414. <https://doi.org/10.1145/2858036.2858589>
- [10] Taizhou Chen, Tianpei Li, Xingyu Yang, and Kening Zhu. 2023. EFRing: Enabling Thumb-to-Index-Finger Microgesture Interaction through Electric Field Sensing Using Single Smart Ring. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 4, Article 161 (jan 2023), 31 pages. <https://doi.org/10.1145/3569478>
- [11] Kurtis Danyluk, Simon Klueber, Aditya Shekhar Nittala, and Wesley Willett. 2024. Understanding Gesture and Microgesture Inputs for Augmented Reality Maps. Association for Computing Machinery (ACM), 409–423. <https://doi.org/10.1145/3643834.3661630>
- [12] Bastian Dewitz, Frank Steinicke, and Christian Geiger. 2019. Functional Workspace for One-Handed Tap and Swipe Microgestures. *Mensch und Computer 2019 - Workshopband*. <https://doi.org/10.18420/muc2019-ws-440>
- [13] Niloofer Dezfuli, Mohammadreza Khalilbeigi, Jochen Huber, Florian Müller, and Max Mühlhäuser. 2012. PalmRC: imaginary palm-based remote control for eyes-free television interaction. In *Proceedings of the 10th European Conference on Interactive TV and Video* (Berlin, Germany) (EuroITV '12). Association for Computing Machinery, New York, NY, USA, 27–34. <https://doi.org/10.1145/232516.2325263>
- [14] Elvis Dsouza, Kevin Chetty, and Shelly Vishwakarma. 2024. RF-Pointer: A Novel Approach to Radio-Frequency Driven Pointer Technology for HCI. In *Proceedings of the IEEE Radar Conference*. Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/RadarConf2458775.2024.10548725>
- [15] Camille Dupré, Caroline Appert, Stéphanie Rey, Houssem Saidi, and Emmanuel Pietriga. 2024. TriPad: Touch Input in AR on Ordinary Surfaces with Hand Tracking Only. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) (CHI '24). Association for Computing Machinery, New York, NY, USA, Article 754, 18 pages. <https://doi.org/10.1145/3613904.3642323>
- [16] Gauthier Robert Jean Faisandaz, Alix Goguely, Christophe Jouffrais, and Laurence Nigay. 2022. Keep in Touch: Combining Touch Interaction with Thumb-to-Finger Gestures for People with Visual Impairment. In *ACM International Conference Proceeding Series*. Association for Computing Machinery, 105–116. <https://doi.org/10.1145/3536221.3556589>
- [17] Shariff A.M. Faleel, Michael Gammon, Kevin Fan, Da Yuan Huang, Wei Li, and Pourang Irani. 2021. HPUi: Hand Proximate User Interfaces for One-Handed Interactions on Head Mounted Displays. *IEEE Transactions on Visualization and Computer Graphics* 27 (11 2021), 4215–4225. Issue 11. <https://doi.org/10.1109/TVCG.2021.3106493>
- [18] Sarthak Ghosh, Hyeong Cheol Kim, Yang Cao, Arne Wessels, Simon T. Perrault, and Shengdong Zhao. 2016. Ringteraction: Coordinated Thumb-index Interaction Using a Ring. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (San Jose, California, USA) (CHI EA '16). Association for Computing Machinery, New York, NY, USA, 2640–2647. <https://doi.org/10.1145/2851581.2892371>
- [19] Hyunjae Gil and Ian Oakley. 2022. ThumbAir: In-Air Typing for Head Mounted Displays. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6 (1 2022). Issue 4. <https://doi.org/10.1145/3569474>
- [20] Jun Gong, Yang Zhang, Xia Zhou, and Xing-Dong Yang. 2017. Pyro: Thumb-Tip Gesture Recognition Using Pyroelectric Infrared Sensing. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (Québec City, QC, Canada) (UIST '17). Association for Computing Machinery, New York, NY, USA, 553–563. <https://doi.org/10.1145/3126594.3126615>
- [21] Sean G. Gustafson, Bernhard Rabe, and Patrick M. Baudisch. 2013. Understanding palm-based imaginary interfaces: the role of visual and tactile cues when browsing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) (CHI '13). Association for Computing Machinery, New York, NY, USA, 889–898. <https://doi.org/10.1145/2470654.2466114>
- [22] Ryo Hajiki, Tamil Selvan Gunasekaran, Chloe Dolma Si Ying Haigh, Yun Suen Pai, Eiji Hayashi, Jaime Lien, Danielle Lottridge, and Mark Billinghurst. 2024. RadarHand: A Wrist-Worn Radar for On-Skin Touch-Based Proprioceptive Gestures. *ACM Transactions on Computer-Human Interaction* 31 (1 2024). Issue 2. <https://doi.org/10.1145/3617365>
- [23] Hayati Havlucu, Mehmet Yarkın Ergin, İdil Bostan, Oğuz Turan Buruk, Tilbe Göksun, and Oğuzhan Özcan. 2017. It Made More Sense: Comparison of User-Elicited On-skin Touch and Freehand Gesture Sets. In *Distributed, Ambient and Pervasive Interactions*, Norbert Streitz and Panos Markopoulos (Eds.). Springer International Publishing, Cham, 159–171.
- [24] Anuradha Herath, Bradley Rey, Sandra Bardot, Sawyer Rempel, Lucas Audette, Huizhe Zheng, Jun Li, Kevin Fan, Da-Yuan Huang, Wei Li, and Pourang Irani. 2022. Expanding Touch Interaction Capabilities for Smart-rings: An Exploration of Continual Slide and Microroll Gestures. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) (CHI EA '22). Association for Computing Machinery, New York, NY, USA, Article 292, 7 pages. <https://doi.org/10.1145/3491101.3519714>
- [25] Da Yuan Huang, Liwei Chan, Shuo Yang, Fan Wang, Rong Hao Liang, De Nian Yang, Yi Ping Hung, and Bing Yu Chen. 2016. Digitspace: Designing Thumb-to-fingers touch interfaces for one-handed and eyes-free interactions. In *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery, 1526–1537. <https://doi.org/10.1145/2858036.2858483>
- [26] Izabelle Janzen, Vasanth K. Rajendran, and Kellogg S. Booth. 2016. Modeling the Impact of Depth on Pointing Performance. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) (CHI '16). Association for Computing Machinery, New York, NY, USA, 188–199. <https://doi.org/10.1145/2858036.2858244>
- [27] Nikhita Joshi, Parastoo Abtahi, Raj Sodhi, Nitzan Bartov, Jackson Rushing, Christopher Collins, Daniel Vogel, and Michael Glueck. 2023. Transferable Microgestures Across Hand Posture and Location Constraints: Leveraging the Middle, Ring, and Pinky Fingers. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) (UIST '23). Association for Computing Machinery, New York, NY, USA, Article 103, 17 pages. <https://doi.org/10.1145/3586183.3606713>
- [28] 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 *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery. <https://doi.org/10.1145/3544548.3550677>
- [29] Kenrick Kin, Chengde Wan, Ken Koh, Andrei Marin, Necati Cihan Camgöz, Yubo Zhang, Yujun Cai, Fedor Kovalev, Moshe Ben-Zacharia, Shannon Hoople, Marcos Nunes-Ueno, Mariel Sanchez-Rodriguez, Ayush Bhargava, Robert Wang, Eric Sausen, and Shugao Ma. 2024. STMG: A Machine Learning Microgesture Recognition System for Supporting Thumb-Based VR/AR Input. In *Proceedings of the CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) (CHI '24). Association for Computing Machinery, New York, NY, USA, Article 753, 15 pages. <https://doi.org/10.1145/3613904.3642702>
- [30] Riku Kitamura, Takumi Yamamoto, and Yuta Sugiura. 2023. TouchLog: Finger Micro Gesture Recognition Using Photo-Reflective Sensors. In *ISWC 2023 - Proceedings of the 2023 International Symposium on Wearable Computers*. Association for Computing Machinery, Inc, 92–97. <https://doi.org/10.1145/3594738.3611371>
- [31] Mikko Kytö, Barrett Ens, Thammathip Piumsomboon, Gun A. Lee, and Mark Billinghurst. 2018. Pinpointing: Precise Head- and Eye-Based Target Selection for Augmented Reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) (CHI '18). Association for Computing Machinery, New York, NY, USA, 1–14. <https://doi.org/10.1145/3173574.3173655>

- [32] Ziheng Li, Zhenyuan Lei, An Yan, Erin Solovey, and Kaveh Pahlavan. 2020. ThuMouse: A Micro-gesture Cursor Input through mmWave Radar-based Interaction. In *2020 IEEE International Conference on Consumer Electronics (ICCE)*. 1–9. <https://doi.org/10.1109/ICCE46568.2020.9040302>
- [33] 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. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 4, Article 170 (jan 2023), 30 pages. <https://doi.org/10.1145/3569463>
- [34] 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 (9 2021), Issue 3. <https://doi.org/10.1145/3478114>
- [35] Mingyu Liu, Mathieu Nancel, and Daniel Vogel. 2015. Gunslinger: Subtle Arms-down Mid-air Interaction. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (Charlotte, NC, USA) (*UIST '15*). Association for Computing Machinery, New York, NY, USA, 63–71. <https://doi.org/10.1145/2807442.2807489>
- [36] Christian Loclair and Sean Gustafson. 2010. PinchWatch: A Wearable Device for One-Handed Microinteractions. <https://api.semanticscholar.org/CorpusID:112927411>
- [37] Siyou Pei, Alexander Chen, Jaewook Lee, and Yang Zhang. 2022. Hand Interfaces: Using Hands to Imitate Objects in AR/VR for Expressive Interactions. In *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery. <https://doi.org/10.1145/3491102.3501898>
- [38] Siyou Pei, David Kim, Alex Olwal, Yang Zhang, and Ruofei Du. 2024. UI Mobility Control in XR: Switching UI Positionings between Static, Dynamic, and Self Entities. In *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery. <https://doi.org/10.1145/3613904.3642220>
- [39] Francisco Perella-Holfeld, Shariff AM Faleel, and Pourang Irani. 2023. Evaluating design guidelines for hand proximate user interfaces. In *Proceedings of the 2023 ACM Designing Interactive Systems Conference* (Pittsburgh, PA, USA) (*DIS '23*). Association for Computing Machinery, New York, NY, USA, 1159–1173. <https://doi.org/10.1145/3563657.3596117>
- [40] Manuel Prätorius, Dimitar Valkov, Ulrich Burgbacher, and Klaus Hinrichs. 2014. DigiTap: an eyes-free VR/AR symbolic input device. In *Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology* (Edinburgh, Scotland) (*VRST '14*). Association for Computing Machinery, New York, NY, USA, 9–18. <https://doi.org/10.1145/2671015.2671029>
- [41] David Rempel, Matt J Camilleri, and David L Lee. 2014. The design of hand gestures for human-computer interaction: Lessons from sign language interpreters. *International journal of human-computer studies* 72, 10–11 (2014), 728–735.
- [42] Adwait Sharma, Michael A. Hedderich, Divyanshu Bhardwaj, Bruno Fruchard, Jess McIntosh, Aditya Shekhar Nittala, Dietrich Klakow, Daniel Ashbrook, and Jürgen Steinle. 2021. SoloFinger: Robust Microgestures while Grasping Everyday Objects. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) (*CHI '21*). Association for Computing Machinery, New York, NY, USA, Article 744, 15 pages. <https://doi.org/10.1145/3411764.3445197>
- [43] Adwait Sharma, Joan Sol Roo, and Jürgen Steinle. 2019. Grasping Microgestures: Eliciting Single-hand Microgestures for Handheld Objects. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) (*CHI '19*). Association for Computing Machinery, New York, NY, USA, 1–13. <https://doi.org/10.1145/3290605.3300632>
- [44] Robert J. Teather and Wolfgang Stuerzlinger. 2013. Pointing at 3d target projections with one-eyed and stereo cursors. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) (*CHI '13*). Association for Computing Machinery, New York, NY, USA, 159–168. <https://doi.org/10.1145/2470654.2470677>
- [45] Hsin Ruey Tsai, Te Yen Wu, Min Chieh Hsiu, Jui Chun Hsiao, Da Yuan Huang, Yi Ping Hung, Mike Y. Chen, and Bing Yu Chen. 2017. SegTouch: Enhancing touch input while providing touch gestures on screens using thumb-to-index-finger gestures. In *Conference on Human Factors in Computing Systems - Proceedings*, Vol. Part F127655. Association for Computing Machinery, 2164–2171. <https://doi.org/10.1145/3027063.3053109>
- [46] Radu Daniel Vatavu. 2023. iFAD Gestures: Understanding Users' Gesture Input Performance with Index-Finger Augmentation Devices. In *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery. <https://doi.org/10.1145/3544548.3580928>
- [47] Anandghan Waghmare, Roger Boldu, Eric Whitmire, and Wolf Kienzle. 2023. OptiRing: Low-Resolution Optical Sensing for Subtle Thumb-to-Index Micro-Interactions. In *Proceedings of the 2023 ACM Symposium on Spatial User Interaction* (Sydney, NSW, Australia) (*SUI '23*). Association for Computing Machinery, New York, NY, USA, Article 8, 13 pages. <https://doi.org/10.1145/3607822.3614538>
- [48] Uta Wagner, Mathias N. Lystbæk, Pavel Manakov, Jens Emil Sloth Grønbæk, Ken Pfeuffer, and Hans Gellersen. 2023. A Fitts' Law Study of Gaze-Hand Alignment for Selection in 3D User Interfaces. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) (*CHI '23*). Association for Computing Machinery, New York, NY, USA, Article 252, 15 pages. <https://doi.org/10.1145/3544548.3581423>
- [49] Jérémie Wambecke, Alix Goguey, Laurence Nigay, Lauren Dargent, Daniel Hautret, Stéphanie Lafon, and Jean-Samuel Louis de Visme. 2021. M[eye]cro: Eye-gaze+Microgestures for Multitasking and Interruptions. *Proc. ACM Hum.-Comput. Interact.* 5, EICS, Article 210 (may 2021), 22 pages. <https://doi.org/10.1145/3461732>
- [50] David Way and Joseph Paradiso. 2014. A Usability User Study Concerning Free-Hand Microgesture and Wrist-Worn Sensors. In *2014 11th International Conference on Wearable and Implantable Body Sensor Networks*. 138–142. <https://doi.org/10.1109/BSN.2014.32>
- [51] Haowen Wei, Ziheng Li, Alexander D. Galvan, Zhuoran Su, Xiao Zhang, Kaveh Pahlavan, and Erin T. Solovey. 2022. IndexPen: Two-finger text input with millimeter-wave radar. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6 (7 2022). Issue 2. <https://doi.org/10.1145/3534601>
- [52] Katrin Wolf, Anja Naumann, Michael Rohs, and Jörg Müller. 2011. A Taxonomy of Microinteractions: Defining Microgestures Based on Ergonomic and Scenario-Dependent Requirements. In *Human-Computer Interaction – INTERACT 2011*, Pedro Campos, Nicholas Graham, Joaquin Jorge, Nuno Nunes, Philippe Palanque, and Marco Winckler (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 559–575.
- [53] Zheer Xu, Weihao Chen, Dongyang Zhao, Jiehui Luo, Te-Yen Wu, Jun Gong, Sicheng Yin, Jialun Zhai, and Xing-Dong Yang. 2020. BiTipText: Bimanual Eyes-Free Text Entry on a Fingertip Keyboard. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) (*CHI '20*). Association for Computing Machinery, New York, NY, USA, 1–13. <https://doi.org/10.1145/3313831.3376306>
- [54] Zheer Xu, Pui Chung Wong, Jun Gong, Te-Yen Wu, Aditya Shekhar Nittala, Xiaojun Bi, Jürgen Steinle, Hongbo Fu, Kening Zhu, and Xing-Dong Yang. 2019. TipText: Eyes-Free Text Entry on a Fingertip Keyboard. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) (*UIST '19*). Association for Computing Machinery, New York, NY, USA, 883–899. <https://doi.org/10.1145/3332165.3347865>
- [55] Yui-Pan Yau, Lik Hang Lee, Zheng Li, Tristan Braud, Yi-Hsuan Ho, and Pan Hui. 2020. How Subtle Can It Get? A Trimodal Study of Ring-sized Interfaces for One-Handed Drone Control. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 2, Article 63 (jun 2020), 29 pages. <https://doi.org/10.1145/3397319>
- [56] Xin Yi, Chen Liang, Haozhan Chen, Jiuxu Song, Chun Yu, Hewu Li, and Yuanchun Shi. 2023. From 2D to 3D: Facilitating Single-Finger Mid-Air Typing on QWERTY Keyboards with Probabilistic Touch Modeling. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 7, 1, Article 38 (mar 2023), 25 pages. <https://doi.org/10.1145/3580829>
- [57] Sang Ho Yoon, Ke Huo, Vinh P. Nguyen, and Karthik Ramani. 2015. TIMMi: Finger-worn textile input device with multimodal sensing in mobile interaction. In *TEI 2015 - Proceedings of the 9th International Conference on Tangible, Embedded, and Embodied Interaction*. Association for Computing Machinery, Inc, 269–272. <https://doi.org/10.1145/2677199.2680560>
- [58] Qian Zhou, George Fitzmaurice, and Fraser Anderson. 2022. In-Depth Mouse: Integrating Desktop Mouse into Virtual Reality. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) (*CHI '22*). Association for Computing Machinery, New York, NY, USA, Article 354, 17 pages. <https://doi.org/10.1145/3491102.3501884>
- [59] Fengyuan Zhu, Ludwig Sidenvik, Mauricio Sousa, and Tovi Grossman. 2023. PinchLens: Applying Spatial Magnification and Adaptive Control-Display Gain for Precise Selection in Virtual Reality. In *Proceedings - 2023 IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2023*. Institute of Electrical and Electronics Engineers Inc., 1221–1230. <https://doi.org/10.1109/ISMAR59233.2023.00139>