Extending Gaze and Pinch with Thumb-to-Finger Microgestures for Cross-Context Interaction

Kagan (Yan-Ling) Taylor
BSc (Hons)

A dissertation submitted for the degree of
Master of Science in Computer Science

Supervised by *Dr. Hans Gellersen*

School of Computing and Communications
Lancaster University

January, 2026

Declaration

I declare that the work presented in this dissertation is, to the best of my knowledge and belief, original and my own work. The material has not been submitted, either in whole or in part, for a degree at this, or any other university. Estimated word count is: **16375**

Name: **Kagan (Yan-Ling) Taylor**
Date: **January, 2026**

Extending Gaze and Pinch with Thumb-to-Finger Microgestures for Cross-Context Interaction

Kagan (Yan-Ling) Taylor, BSc (Hons).
School of Computing and Communications, Lancaster University

A dissertation submitted for the degree of *Master of Science* in Computer Science.
January, 2026

Abstract

Gaze+Pinch is a simple and effective model for XR interaction, using gaze for pointing and pinch for selection. Yet, for many tasks, its expressivity is lacking. To address this problem, we extend the pinch gesture with additional thumb to finger microgestures to leverage the expressivity afforded by the hand. Rather than designing a gesture set that tackles an individual problem, this project's approach is to design a system-level gesture set that maps gestures to application-specific functionality, depending on the application in focus. Utilising a system-level gesture set aims to provide interactions that are learnable and discoverable across diverse contexts, while application-specific mappings aim to provide task-specific utility, while leaving the original Gaze+Pinch interaction untouched. To support users across contexts changes, we employed two established desktop metaphors that describe how gestures will behave across contexts (the buttons and scroll wheel of a mouse, and function-key shortcuts on a keyboard). In a 16 participant user study across two practical tasks (travel planning and spreadsheeting), we compared Gaze+Pinch against Gaze +Microgestures (a condition which restricted participants to the extended gesture set) and Combined (where participants were able to freely choose between Gaze+Pinch and Gaze+ Microgesture interactions). Results from the study revealed the viability of our approach, with the Combined offering the fastest, most preferred, and most satisfying performance. Participants engaged with the extended gesture set in the combined condition, but preference fell when restricted to the additional gestures in Gaze+Microgestures. This, in conjunction with qualitative feedback, underscores the importance of user agency and customisability, especially for gestural input, where differences in hand dimensions, finger dexterity, and personality can result in significant impacts on user experience. These findings demonstrate that optional microgestures can enhance expressivity without compromising the usability of the original interaction, pointing towards a design strategy that enables more expressive and flexible gaze-based input systems.

Acknowledgements

This thesis was made possible through the support of my parents, sister, partner, my supervisor Prof. Hans Gellersen, Dr. Florian Weidner, my examiners Dr. Daniel Fitton and Dr. Augusto Esteves, and all of the GEMINI team.

Thank you all for your encouragement and guidance.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Gaze+Pinch has emerged as a promising controller-free input technique for virtual and augmented reality (Pfeuffer et al., 2017; Pfeuffer et al., 2024). The technique integrates eye gaze to indicate intent and a pinching gesture to confirm/manipulate. By leveraging the natural synergy of looking and pinching, Gaze+Pinch provides a minimal input structure that allows users to point and select in a manner similar to the way a single-button mouse has been used on the desktop (Pang, 2002). The technique is beginning to enter the consumer space, featured prominently in devices such as the Apple Vision Pro (Apple, 2025) and Microsoft HoloLens 2 (Microsoft, 2025). Its appeal lies in its subtlety, low effort, and consistency across system contexts.

The problem is that Gaze+Pinch alone lacks the expressivity that many tasks demand. To compensate, current systems rely on long-press gestures and context menus to distinguish selections from operations such as copy-and-paste, opening a link in a new tab/window, and moving an element, as opposed to scrolling a page (Apple, 2025). This design keeps the interaction space simple and learnable, but frequent menu summoning and navigation adds latency and overhead that limits performance and flow (GadgetHacks, 2025; Jinwook Kim et al., 2025). This matters because efficient and fluid interaction is essential for the adoption of XR as a medium for everyday use; users not only need techniques that are simple to learn, but also ones that support and scale with expertise (Nielsen, 1994; Card et al., 1980). Prior work has demonstrated the feasibility of extending the pinch gesture to a broader gesture set for specific application scenarios, such as AR map navigation (Danyluk et al., 2024), UI mobility (Pei et al., 2024), and cockpit multi-tasking (Wambecke et al., 2021). While these approaches improve expressivity within the scope of their application, in a system-wide context where users often interact with upwards of 40 unique applications (Tila, 2021; Tila, 2022; Tila, 2025), unique gesture sets per application fragment learnt knowledge across contexts, hindering learnability and expertise development (Shneiderman,

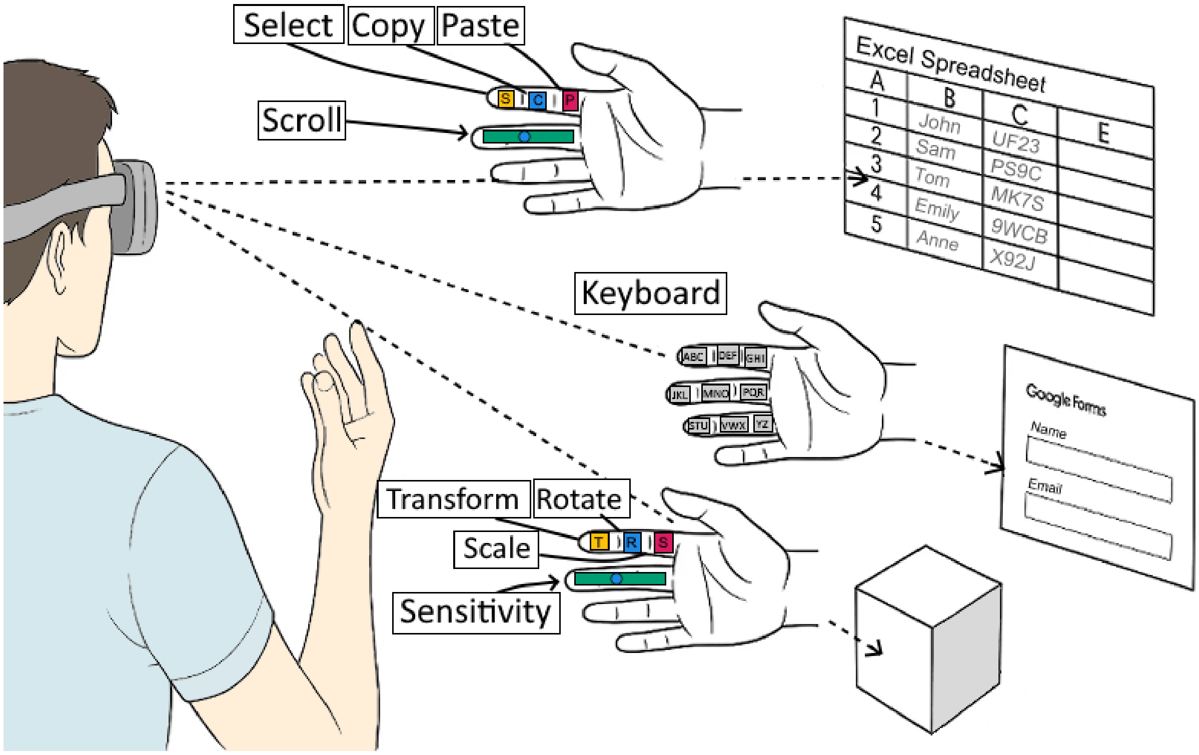


Figure 1.1: The extension of Gaze+Pinch to Gaze+Microgestures enables users to combine gaze for target selection with subtle hand gestures for input. This allows the hands to function as a flexible tool for common XR tasks. This figure illustrates how the same microgestures can be used to both copy and paste information in a spreadsheet, enter simple text into a form, and manipulate a virtual object in 3D.

1997). Other approaches extend the pinch to support dedicated functionality, such as multi-select (Shi et al., 2024; Jinwook Kim et al., 2025), and object translation modes (Zhang et al., 2025), however this sacrifices task-specific functionality. In summary, Gaze+Pinch offers simple, but low-expressivity hands-free input, and while prior works have explored gesture sets tailored to individual applications, and specialised gestures for specific functionality, it remains unclear how Gaze+Pinch can be extended in a way that preserves its simplicity and learnability, while supporting task-specific interactions.

1.2 Objectives

This dissertation aims to extend the expressivity of Gaze+Pinch through the introduction of a compact set of microgestures. This subdivides into 3 objectives:

1. Design and implement a representative extension of Gaze+Pinch that preserves pinch-

to-select as the default input.

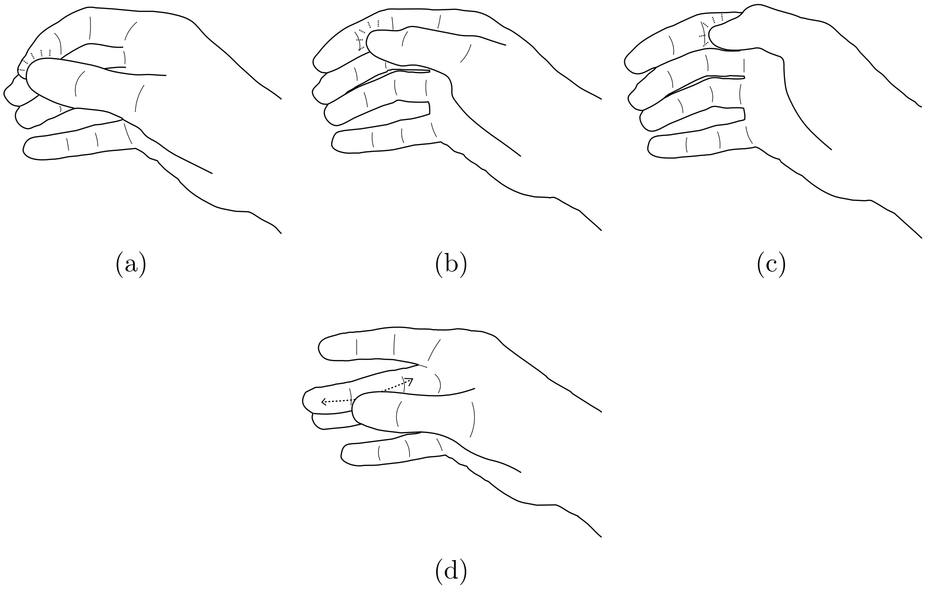
2. Define strategies to map application-specific functionality to the gesture set, and evaluate them against Gaze+Pinch across representative tasks, analysing performance, error rates, physical effort, workload, and preference.

3. Implement a gesture detection system using commodity hardware to support the study of Gaze+Microgestures.

1.3 Methodology

This project explores the extension of Gaze+Pinch with additional thumb-to-finger microgestures that dynamically map to context-specific functionality. Figure 1.1 illustrates the concept. This approach defines a gesture set at the system-level to support familiarity and discoverability across contexts, and maps each gesture to functionality specific to the application at the gaze point, supporting tailored and task-relevant interactions. The index fingertip is reserved for standard Gaze+Pinch interaction to support novice users, while additional, optional microgestures accommodate the progressive adoption of expert interactions. To explore this concept, a detection strategy was developed to enable the detection of thumb to finger microgestures. Then, a representative microgesture set was designed to serve as an interface to interact with applications, consisting of the following four microgestures: pinch of thumb to tip of the index finger (retained as the default for selection), thumb taps to the middle and base segments of the index finger, and a sliding gesture of the thumb along the middle finger for continuous input (fig. 1.2). To support users across contexts, two representative metaphors were employed that leveraged familiar desktop interactions: the first describes the gestures as left-, middle-, and right-click plus scroll (mirroring the functionality of a three-button mouse), and the second describes the additional gestures as function-key shortcuts. Each metaphor is tested against the baseline (Gaze+Pinch) through two practical tasks (travel planning and spreadsheeting), as well as in a combined condition (Combined) where users can freely choose between Gaze+Pinch and the extended gesture set. Finally, the findings from this project are used to present a guide to designing for Gaze+Microgestures.

The novelty of this work lies in balancing the simplicity of Gaze+Pinch with the added expressivity of a small but powerful set of microgestures. The design rationale preserves pinch-to-select as the universal default while opening new possibilities for expert users. Results from the user study demonstrate the feasibility of the approach, with participants performing fastest and rating the technique as the easiest and the most satisfying to use when they were able to choose whether they completed an interaction using Gaze+Pinch or Gaze+Microgestures. Preference mirrored this pattern: when given all options, participants overwhelmingly preferred Combined. Participants attributed their preference to the flexibility



(a) (b) (c)

(d)

Figure 1.2: Basic set of employed gestures. Three pinches (a-c) plus one sliding motion (d).

and agency afforded by Combined, which allowed participants to selectively engage with polarising gestures, and, in one case, support the novice to expert transition. These factors overshadowed the greater learnability and memory demands imposed by the extended gesture set, suggesting that, through careful gesture-set design and employment of effective metaphors, Gaze+Pinch can be effectively extended to offer expressive, application-specific functionality for expert users, while preserving the simplicity of the original technique.

This work contributes a design approach that can be applied at a system-level to enhance functionality across a broad range applications ranging from web browsing and spreadsheeting, to gaze-locomotion, 3D manipulation and drawing –without undermining the simplicity of the interaction. The results from this project underscore the importance of flexible design and user agency, suggesting a future where users can customise their own gesture sets and functionality mappings.

This work demonstrates an extension of Gaze+Pinch with additional microgestures that dynamically map to application-specific functionality defined by the gaze point. The project reveals the positive impact this approach has on performance, satisfaction and user experience, and lays the foundation for the design of more complex and expressive gesture sets and application mappings. More generally, it points towards XR systems that combine simplicity and scalability, enabling both casual use and expert-level productivity without reliance on handheld controllers.

1.4 Contributions

This work makes the following contributions to the design and study of controller-free XR interaction:

- The extension of Gaze+Pinch with additional thumb to finger microgestures to increase expressivity while preserving pinch as the default selection.
- The study of two desktop-inspired mapping strategies (mouse buttons and function keys) that demonstrate alternative ways of integrating Gaze+Microgestures into XR interactions.
- Empirical evidence from two tasks and a combined condition that shows that combining Gaze+Pinch and with additional gestures yields the fastest and most satisfying performance, without compromising the simplicity of Gaze+Pinch.
- A guide to designing for Gaze+Microgestures.

1.5 Dissertation Structure

The remainder of this dissertation is as follows: Chapter 2 provides an overview of the related literature. Chapter 3 explores the conceptual design space of Gaze+Microgestures, exploring the gesture set and mapping strategies designed, and proposing several applications where Gaze+Microgestures show potential to improve the user experience. Chapter 4 explains how microgesture detection was implemented, and how the system was designed to enable interactions between the microgestures and applications. Chapter 5 describes the method and measures used to study two designed mapping strategies against Gaze+Pinch and a combined condition, where users were free to use either technique. Chapter 6 reports the empirical findings from the study. Chapter 7 discusses the findings from the study, interpreting the results and discussing their implications for the future of Gaze+Microgesture interaction. Chapter 8 then uses the findings from this study to offer a guide to designing for Gaze+ Microgestures. Chapter 9 concludes the dissertation, giving an overview of what was learnt from the project, what challenges were faced, and the broader impact this work has on interaction in XR.

Chapter 2

Related Works

This project builds primarily on two lines of prior research: the extension of gaze, specifically Gaze+Pinch with gestures; and the design of microgestures and hand-proximate interfaces.

2.1 Extension of Gaze with Gestures

Gaze+Pinch has become a strong baseline for input in XR (Pfeuffer et al., 2017). The technique is grounded in the natural coordination of eye and hand, and leverages the relative strengths of gaze for fast and implicit pointing at targets with manual input for deliberate selection and manipulation (Johansson et al., 2001; Pfeuffer et al., 2014; Pfeuffer et al., 2024). Studies have found the combination of gaze and pinch to outperform other selection methods for XR (Wagner et al., 2023; Mutasim et al., 2021) and demonstrated benefits of indirect input by Gaze+Pinch for avoiding occlusion, reducing effort and 6DOF input from a comfortable position (Lystbæk et al., 2024; Mikkelsen et al., 2025).

Gaze+Pinch employs a single thumb-to-finger gesture for interaction, following a ”time-multiplexed” input scheme – one input method is used to control different functions at different points in time (for example, by selecting different UI elements). Distributing functionality spatially across multiple inputs can improve performance (Fitzmaurice and Buxton, 1997), benefiting from reduced pointing and shallower menu navigation, as direct access to functionality bypasses the menu navigation required for common tasks such as copy and paste.

Numerous works have explored extensions of Gaze+Pinch to increase the technique's expressivity. This includes modification of the pinch gesture itself to support half-pinch (Jinwook Kim et al., 2025; Patel et al., 2025) and force-pinch input (Zhang et al., 2025), and mid-air gestures with the pinching hand (L. Lu et al., 2021). In this project, modifications to the core use of Gaze+Pinch are avoided, and instead the focus is on the distribution of functionality spatially across gestures, which offers a more expressive and versatile interaction space.

Other works have combined Gaze+Pinch with in-air gestures (Chatterjee et al., 2015) and used Gaze+Pinch followed by microgestures executed in serial (Wambecke et al., 2021; Chaffangeon Caillet et al., 2023). Then, more recently, thumb to finger microgestures have been integrated with gaze input for UI mode switching (Pei et al., 2024) and continuous input (Jina Kim et al., 2025). The common pattern is to keep gaze for spatial intent and to layer short-hand actions for mode changes or secondary commands. This work follows the idea but organises the additional input systematically across finger segments. Rather than proposing solutions for particular tasks, the focus of this work is on mapping strategies for a given set of microgestures to task demands across applications, and tailored to applications.

2.2 Microgestures and Hand-Proximate User Interfaces

Thumb-to-finger microgestures are fast, subtle, and always available on-body gestures (Soliman et al., 2018; E. Chan et al., 2016; Wolf et al., 2011). They are less physically demanding and more subtle than mid-air gestures (Li et al., 2025; Chaffangeon Caillet et al., 2023). This subtlety lends thumb-to-finger microgestures to public and on-the-go input (L. Chan et al., 2013; Danyluk et al., 2024; Boldu et al., 2018), as social acceptability studies find people prefer smaller, body-proximate gestures to larger, more conspicuous gestures (Ahlström et al., 2014; Rico and Brewster, 2010; You et al., 2019). However, a key advantage of thumb-to-finger microgestures is that they can be performed eyes-free (Shariff A M Faleel et al., 2024; Darling et al., 2024); the segments and knuckles of the fingers act as proprioceptive landmarks (Darling et al., 2024), while the tactile feedback of the fingers complement proprioceptive cues with haptic confirmation (Gustafson et al., 2013). A number of works have therefore explored thumb-to-finger-segment microgestures for eyes-free text entry (Jiang et al., 2019; Huang et al., 2016; Whitmire et al., 2017), map panning and zooming (Danyluk et al., 2024), and application selection and operation (Tsai et al., 2016; Perella-Holfeld et al., 2023). As microgestures have no inherent pointing capabilities, current microgesture ecosystems rely on swipe-through menu navigation to transition between applications (*Meta AI glasses — Ray-Ban Meta, Oakley Meta &amp; Display glasses* 2025). In this project, the eyes-free affordance of thumb-to-finger gestures are leveraged to combine with gaze for spatial input and seamless context switching. Unlike previous work, context switching with gaze allows us to explore Gaze+Microgesture across a number of tasks and contexts, rather than focussing only on one domain.

Thumb-to-finger microgestures have been used for continuous input, as well as discrete (Jina Kim et al., 2025; Kin et al., 2024; Perella-Holfeld et al., 2023; X. Chen et al., 2023; Danyluk et al., 2024). Some works overlap continuous microgestures over discrete (Kin et al., 2024; Jina Kim et al., 2025; Danyluk et al., 2024), while others use a dedicated space to display visual feedback (Perella-Holfeld et al., 2023). Although users can execute thumb-to-finger microgestures eyes-free, users must learn and remember what those gestures do (Shariff A M Faleel et al., 2024; Darling et al., 2024), and many works therefore employ gesture feedback on the hand or in the world (Perella-Holfeld et al., 2023; Whitmire et al., 2017; Gustafson et al., 2013). This project uses the dedicated space for gesture input to enable visual feedback for novice users, who tend to focus attention at the on-hand UI and shift towards eyes-free with practice (Shariff A M Faleel et al., 2024).

Prior works have extensively explored using thumb-to-finger microgestures for changing the mode of direct touch input (Tsai et al., 2017; Faisandaz et al., 2023; Lim et al., 2018; Faisandaz et al., 2022), improving performance by combining command selection with direct manipulation (Guimbretière et al., 2005). In this project, the concept is extended to the indirect space, using microgestures for command selection to change the mode of gaze input and improve performance. This project also explores using gaze for command selection, changing the commands microgestures execute depending on the application in the user's gaze (see Section 3.2.

A wide range of work is focussed on the design and structure of microgestures. Chan et al. conducted one of the first elicitation studies for thumb-to-finger gestures (E. Chan et al., 2016), and FingerInput provides a key taxonomy that defines thumb-to-finger gestures by segment locations (Soliman et al., 2018). Work on hand-proximate user interfaces (HPUI) provides relevant guidelines for using finger segments for UI placement, and several works report comfort ratings for thumb-to-finger segment interactions (S. A. Faleel et al., 2021; Huang et al., 2016; Jiang et al., 2019; Dewitz et al., 2019; Kuo et al., 2009). This project builds on these findings and maps segment-specific gestures to system-wide functions. The focus is on mapping strategies rather than optimisation of gesture sets, for which a representative set of three discrete thumb-to-finger taps and a thumb slider has been implemented.

A number of works have explored microgestures while grasping objects (Wolf et al., 2011; Sharma et al., 2024; Caillet et al., 2025), however these works have focused on users who are engaged in a primary task and are encumbered. In this work, microgestures are explored as the primary interaction, with users unencumbered and free to issue a wider range of microgestures.

A wide range of sensing approaches have been proposed. These include magnetic tracking (Huang et al., 2016), wrist-worn or ring cameras (Prätorius et al., 2014; L. Chan et al., 2015), conductive gloves (Whitmire et al., 2017), capacitive wearables (Y. Lu et al., 2024), electromagnetic rings (T. Chen et al., 2023), IR bracelets (McIntosh et al., 2017), and hover or pressure sensing (Dobinson et al., 2022). Meta also presented machine learning classifiers for small thumb-to-finger gesture sets using ocular signals (Kin et al., 2024). This work stays agnostic to the sensing method and instead focus on mappings and technique design that work across detectors and are compatible with commodity XR hardware.

Chapter 3

Gaze+Microgesture Interaction

Traditionally, AR/VR input has relied on handheld controllers, introducing the need for an extra device, which hinders availability and accessibility. In contrast, mobile AR/VR applications benefit from controller-free interaction solutions. Gaze+Pinch is a prominent controller-free approach that is both easy to learn and efficient. Its strength lies in its simplicity: gaze specifies the target, while a pinch confirms the input. This basic input model can be extended, for instance, by holding the pinch longer to open a context menu or by moving the hand during the pinch to translate or rotate objects (Apple, 2025). In this way, users can perform a wide range of essential interactions with a minimal gesture vocabulary. The overall interaction design is analogous to the single-button mouse (Pang, 2002): moving the mouse determines which user interface element receives input, while pressing the button confirms the action. Gaze+Pinch adopts this principle of simplicity to provide an intuitive, controller-free input method for AR/VR.

Similar to how the mouse evolved from a single-button device into a multi-button input device with extended functionality, introducing additional microgestures to Gaze+Pinch promises to retain the simplicity of gaze-based selection while offering a more versatile range of interactions. The idea is to combine gaze input with a richer set of hand-based actions: additional pinches on different parts of the hand can function like extra mouse buttons or act as shortcuts comparable to function keys on a keyboard. Beyond discrete commands, thumb-to-finger microgestures also enable continuous input, for example, by sliding the thumb tip along the inside of the middle finger or rolling the thumb along the top of the index finger, which can emulate the functionality of a mouse scroll wheel. However, instead of designing a gesture set for a particular application, the approach used in this project is to design a generic gesture set that is mapped to application-specific functionality depending on the application at the gaze point.

This approach gives rise to two key considerations. First, the set of microgestures itself: an extended gesture set must carefully balance the simplicity, expressivity, and the functional range of the hand. Second, functionality must be mapped to gestures appropriately, either mapping to the currently active window, application, or widget, or instead mapping to whichever element the user's gaze is currently hovering over. Relevant metaphors are then essential to explain this behaviour transparently to users. Together, these design choices determine what input is available and how Gaze+Microgestures map input to user interface elements, specifying how the technique operates in practice and how it can be applied effectively within AR/VR ecosystems.

Once these design decisions are established, a Gaze+Microgestures system can resolve the tracked gaze point, detect, interpret, and execute the corresponding microgesture event, and ultimately determine the underlying user interface element that should receive input and execute the input action. The following sections discuss these two key design considerations in turn, followed by a section that covers strategies to improve the learnability and memorability of an extended microgesture set.

3.1 An Extended Microgesture Set

While prior research offers a rich body of work on gesture vocabularies (see Section 2.2), there is an inherent tension between simplicity and functional range. In principle, microgestures can replicate the full functionality of mouse and keyboard input (Danyluk et al., 2024; Whitmire et al., 2017). However, this increases the complexity of the input system as users have to learn and retain a much larger set of microgestures. Therefore, the number of microgestures included in the gesture set must be carefully limited.

The gesture set designed in this project (Gaze+Microgestures) extends Gaze+Pinch by introducing two discrete thumb-to-finger pinches to the palmar side of the index middle and base phalanges, and a sliding gesture of the thumb along the length of the middle finger for continuous input Figure 1.2. Pinches to the finger segments were used as they serve as proprioceptive landmarks that can be identified without the use of gaze (Darling et al., 2024; Shariff A M Faleel et al., 2024), and inputs were distributed across the fingers to broaden the interaction space. Input was restricted to the first two fingers to keep interactions in the most comfortable areas of the hand (S. A. Faleel et al., 2021; Huang et al., 2016; Jiang et al., 2019; Dewitz et al., 2019; Kuo et al., 2009). Gestures are positioned spatially near to each other to reduce effort when switching between them, but do not overlap to allow for clear visual feedback to be displayed (see Section 3.3.3).

Although this design is grounded in microgesture elicitations (Perella-Holfeld et al., 2023), comfort regions (Dewitz et al., 2019), and social acceptability studies (Rico and Brewster, 2010). Future work that involves end-users in the design process may yet elicit more optimal or expressive gesture sets. For example, this design does not employ finger-to-thumb microgestures, microgestures to the radial, dorsal, or ulnar side of the hand, continuous gestures about the finger, or symbolic microgestures (such as flicks, or drawn shapes) (Soliman et al., 2018; E. Chan et al., 2016). The designed gesture set acts as an informed and representative set, balancing expressivity with simplicity to serve as a vehicle to explore

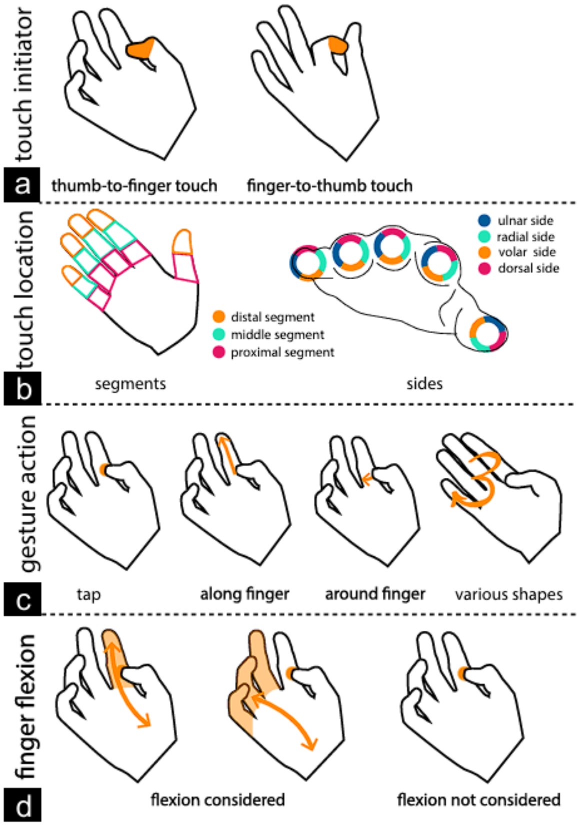


Figure 3.1: Taxonomy of thumb-to-finger microgestures by (Soliman et al., 2018). The extended gesture set explored employs a limited subset of the total range of thumb-to-finger microgestures.

the space.

3.2 Mapping Microgestures to Functionality

Gaze+Pinch issues a generic command to targets at the gaze point. This creates a consistent interaction that reacts reliably across contexts, and is easily learnable. Gaze+Microgestures preserves this behaviour at the index fingertip, but can map the additional gestures to application-specific functionality, allowing the gestures to be dynamically tailored to the application in focus, rather than defined statically at the system level. Similar to desktop systems, the application in focus can be defined as the last selected application; however, in multi-window scenarios, the user's gaze can be leveraged instead to map to the application currently under the user's gaze. Figure 3.2 illustrates an example. As the user looks across applications, the dynamically defined gestures change functionality to be application-specific

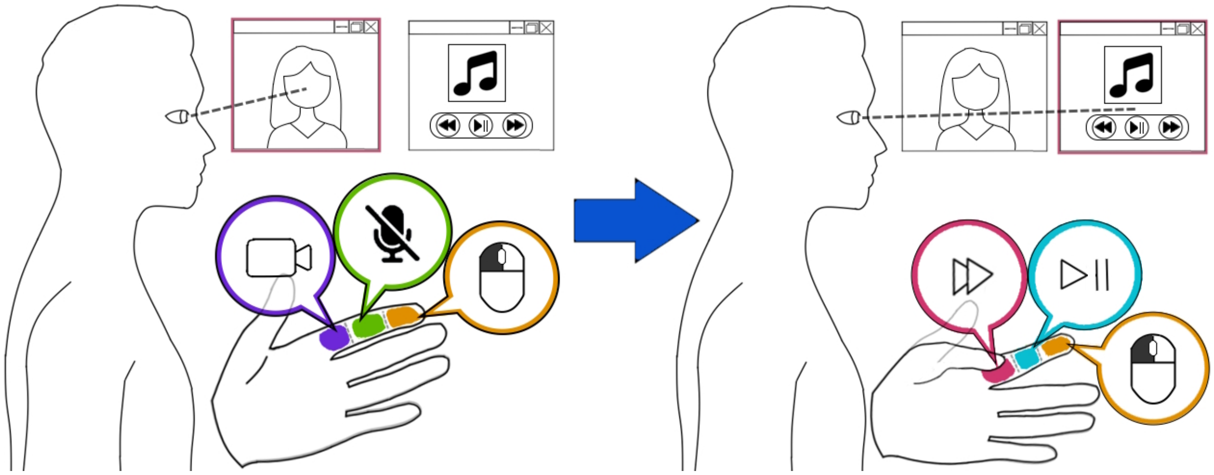
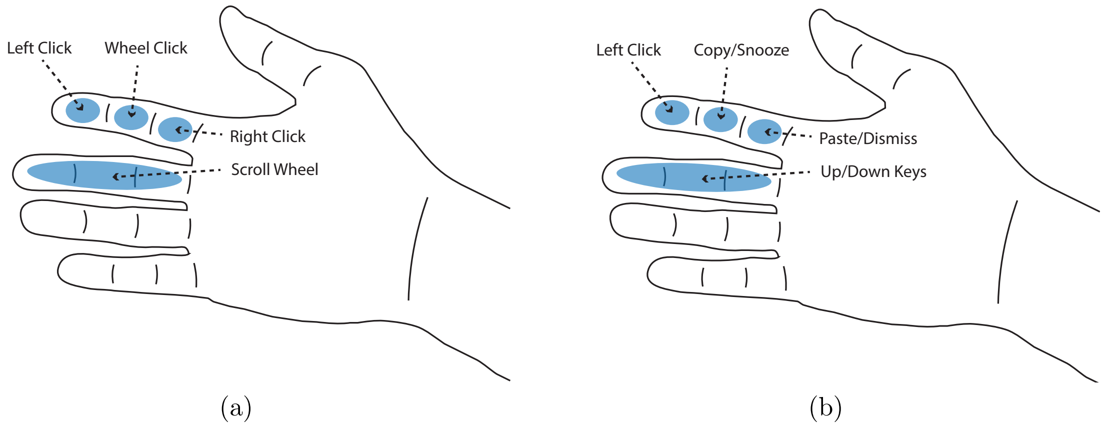


Figure 3.2: Gaze+Microgestures preserves the functionality of Gaze+Pinch at the fingertip. Its functionality is statically defined, and does not change depending on gaze location. The other two pinch locations change dynamically, depending on where the user looks. In this example, the user controls their camera and microphone when looking at their video call, and when looking at their music player they can instead play/pause and skip forward — without having to fixate on the buttons.

while the fingertip is statically defined, and maintains the same functionality across contexts.

When application-wide functions such as skip-forward or toggle-mute are mapped to microgestures, they are able to be executed without pointing to their respective UI elements, effectively enlarging the target to the whole window, reducing the pointing needed to execute them. This can be particularly impactful for frequently used functionality that is hidden behind a context menu, such as copy and paste, or when navigation controls are placed distant to the content, requiring repeated back-and-forth gaze pointing—for example, when flipping through photos in a photo gallery. This has further implications when windows partially obscure each other. Instead of needing to bring a window into the foreground, pointing to and selecting the UI element, then sending it back to the background, if the desired functionality is mapped to a gesture on their hand, they can simply look at an exposed part of the window and issue the command directly. To put it another way, the interaction in fig. 3.2 would work exactly the same if the two windows partially obscured each other. This allows for interactions with Gaze+Microgestures that were previously not possible on desktop or with Gaze+Pinch.

Rather than functionality being application-wide, functionality can also be directed to the gaze-point, similar to Gaze+Pinch, or mouse input. This can be used to perform alternative types of selection, such as selections with modifier keys, such as CTRL-click, or shift-click, or alternatively, the buttons of the mouse, enabling full mouse emulation in XR.



(a) (b)

Figure 3.3: Two implementations of Gaze+Microgestures: (a) three-button mouse metaphor, (b) function key metaphor.

3.3 Learnability and Memorability Strategies

With Gaze+Microgestures, the gesture set remains unchanged across contexts, supporting the the gesture set's learnability. However, a remaining challenge for Gaze+Microgestures is that users need to learn and recall a set of gestures for each application, which can discourage them from using the additional input options altogether (Bailly et al., 2023). To mitigate this, gesture mappings should follow strategies that support learnability and transition smoothly across contexts. In particular, microgesture mappings should align with established mobile and XR standards (Shariff A. M. Faleel et al., 2020), and frequently used functions should be assigned to gestures that are comfortable for most users (Shariff A. M. Faleel et al., 2020).

Metaphors can be utilised to combat learnability challenges, leveraging established patterns users are already familiar with, such as left-, middle, and right-click, or select, copy, and paste. These can map spatially onto the gesture set to imitate established and recognisable patterns. This project implements and tested two variants of Gaze+ Microgestures that build on familiar desktop metaphors: one modelled after function keys and another modelled after a three-button mouse (see Figure 3.3).

3.3.1 Mouse Buttons on The Hand

In the three-button mouse variation, the three phalanges of the index finger are mapped to the three mouse buttons, while the palmar side of the middle finger serves as an input surface for scrolling via sliding (see Figure 3.3a). This design mirrors the functionality of computer mice and touchpads that typically offer a left, right, and middle button click. Gaze controls the cursor, and the three pinches between thumb and index finger (thumb-to-tip, thumb-to-medial, and thumb-to-proximal) correspond to left, middle, and right mouse buttons. Each of these microgestures execute functionality directly at the gaze point: for example, a left-click on a button executes its function (as in Gaze+Pinch), a right-click opens a context menu at the cursor location, and a middle-click closes the browser tab currently under the gaze cursor.

3.3.2 Function-Keys on The Hand

In the function key variation, the same set of microgestures are employed, but apply a different input metaphor for the proximal and middle phalanges (see Figure 3.3b). This draws on established desktop conventions, where function keys provide quick access to global commands independent of cursor location. Here, actions are not tied to the precise gaze cursor location but instead operate at the level of windows, documents, or applications that the gaze hovers over. For example, in a drawing application, users can switch between pen and eraser without diverting their gaze to the toolbar, or briefly glance at a music player and perform a microgesture to skip a song, then immediately return to their work with context-specific gestures. In this setup, the gesture mappings are dynamically remapped depending on the application the user glances at, reducing the need to bring background applications to the foreground to execute commands.

3.3.3 Feedback

When users forget gesture mappings, feedback is necessary to help them recover and execute their intended function. Discoverability and reminders of functionality become increasingly important as the number of applications a user interacts with grows. Gaze+Microgestures employs visual feedback displayed on the hand. An icon is placed on each finger segment on the index finger, and a slider is rendered along the middle finger. Each icon depicts the functionality that would be executed when a pinch is performed at the corresponding segment. To support this interaction, the last-looked-at target remains selected until the gaze shifts to a new application. This allows users to briefly divert their gaze towards their hand to check the iconography and then execute the gesture without losing their intended target. In practice, this mechanism makes the gesture set easier to use and reduces potential errors caused by forgotten mappings.

Unlike audio feedback (Gustafson et al., 2013), or displaying a representation of the gestures in the world (Whitmire et al., 2017), showing feedback on the hand offers a non-intrusive mechanism that supports both novice and expert users, and facilitates the novice to expert transition, while being expressive enough to display functionality across a broad range of applications. Visual feedback on the hand is accessible at a glance, supports memorability, and reduces the cognitive load associated with the recall of all gesture-function mappings.

It is expected that over a large number of applications users will not recall all microgesture mappings, but instead remember a subset of frequently used ones, similar to the selective uptake of keyboard shortcuts in desktop environments (e.g., space-bar for pause, Control+F for search). Likewise, some microgestures may prove more useful than others. A key strength of Gaze+Microgestures is that the original Gaze-and-Pinch functionality remains available, allowing users to opt out of additional complexity if desired, while enabling more confident users to benefit from richer interaction possibilities.

3.4 Applications of Gaze+Microgestures

Gaze+Microgestures can be applied across a wide spectrum of XR scenarios, spanning productivity tasks, object manipulation, and navigation (also see Figure 1.1). This section covers several applications where Gaze+Microgestures shows potential to improve the user experience.

3.4.1 3D Manipulation

While Gaze+Pinch combines degrees of freedom to form natural interactions, when precision is needed, Gaze+Microgestures can separate them for fine manipulation.

Gaze+Pinch can be used for 3D manipulation by pinching while looking at an object, and moving the hand to manipulate (Pfeuffer et al., 2017). When used uni-manually, objects translate relative to the hand's movement, and rotate about the hand (Pfeuffer et al., 2017). Translation and rotation coupling can make precise manipulation difficult to achieve, especially at long distances (Jeong et al., 2025). Gaze+Microgestures can be used to separate rotation from translation by introducing distinct gestures for each, enabling in-place rotation. A gesture for uniform scaling can be introduced to allow for uni-manual scaling, and the slider can be used for translation towards and away from the user, for which clutching is normally required.

Alternatively, Gaze+Microgestures could be used to eliminate the hand movement, allowing for more socially acceptable interactions for public situations. In this case, index finger gestures can be mapped to the X, Y and Z axes, and the slider can control acceleration along the selected axis.

Index Tip	**Index Middle**	**Index Base**	**Slider**
Translate	Rotate	Scale	Depth translation
X axis	Y axis	Z axis	Move along selected axis

Table 3.1: Gaze+Microgesture mappings for 3D manipulation.

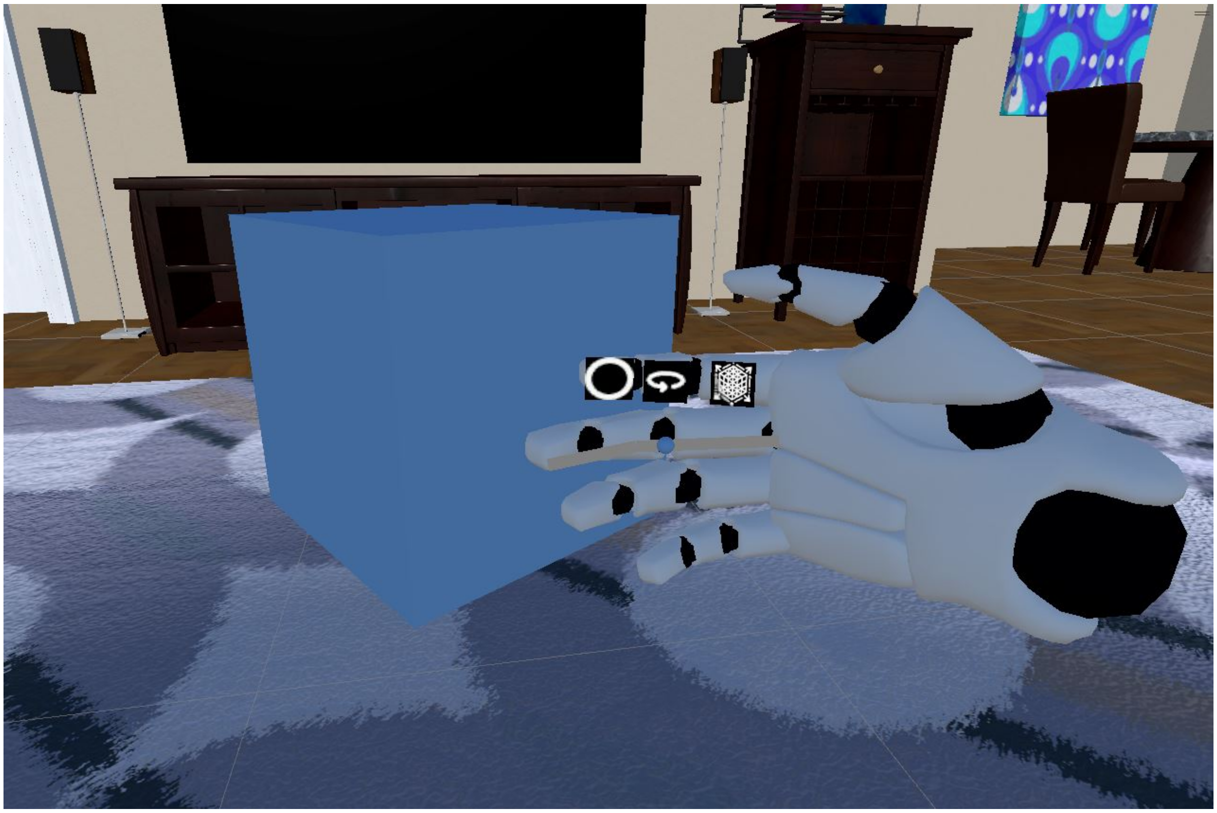


Figure 3.4: Example of 3D manipulation with Gaze+Microgestures. Translation, rotation and uniform scaling have been disseminated across the index finger. The user can slide along the middle finger to move the looked-at object towards, or away from themselves.

3.4.2 Drawing

Gaze+Microgestures can be used to keep attention on critical tasks, or to combine command selection with manipulation; instead of performing mode switches in the user interface, Gaze +Microgestures can execute commands directly using microgestures.

3D, or canvas drawing with Gaze+Pinch can be streamlined by introducing gestures that change the mode of the primary input, such as toggle eraser, or switch primary and secondary colours, and the slider can control the brush size. This allows users to maintain their gaze on the drawing, without needing to avert their attention to a toolbar.

Alternatively, microgestures can combine command selection with direct input by allowing users to draw directly with the other microgestures (Guimbretière et al., 2005); The index finger tip draws using the primary colour, while the index middle segment draws with the secondary colour, and the index base segment erases.

Index Tip	Index Middle	Index Base	Slider
Draw	Switch primary/secondary colour	Toggle eraser	Brush size
Draw with primary colour	Draw with secondary colour	Erase	Brush size
Draw	Re-Do	Un-Do	Brush size

Table 3.2: Gaze+Microgesture mappings for drawing applications.

3.4.3 Disambiguation

Gaze+Microgestures can be used to disambiguate dense or imprecise selections. For example, when an ambiguous gaze selection is made, items in the selection can be numbered or highlighted to correspond with a microgesture, similar to (Delamare et al., 2022). Executing the corresponding microgesture would select the corresponding item. This could be used within a system for elements such as minimise, maximise, and close, which are found in the top corner of browser windows.

Index Tip	Index Middle	Index Base	Slider
Select Option 1	Select Option 2	Select Option 3	NA

Table 3.3: Gaze+Microgesture mappings for disambiguation.

3.4.4 Gaze Locomotion

Gaze+Microgestures can enhance immersive experiences by offering direct access to functionality without breaking the immersion with menus and user interfaces.

Gaze locomotion can act as an accessible and efficient method of locomotion to explore 3D environments. Gaze-locomotion consists of translating in the direction of the eye-gaze, and rotating the viewport in the direction of the gaze (Mine, 1995; H. S. Lee et al., 2024). This means that during locomotion, users would need to fixate in the direction they wish to travel. With Gaze+Microgestures, additional inputs allow for the separation of rotation and translation. The index fingertip can be used to translate and rotate together, while the index middle segment can be used to rotate only, and the index base can be used to translate in the direction indicated by the gaze when the gesture was first pressed. This allows users to rotate in place, or to translate in a direction, then look around as they travel in that direction, observing scenery they pass by.

Index Tip	Index Middle	Index Base	Slider
Translate and Rotate	Rotate in place	Translate in one direction	Translation Speed

Table 3.4: Gaze+Microgesture mappings for gaze locomotion.

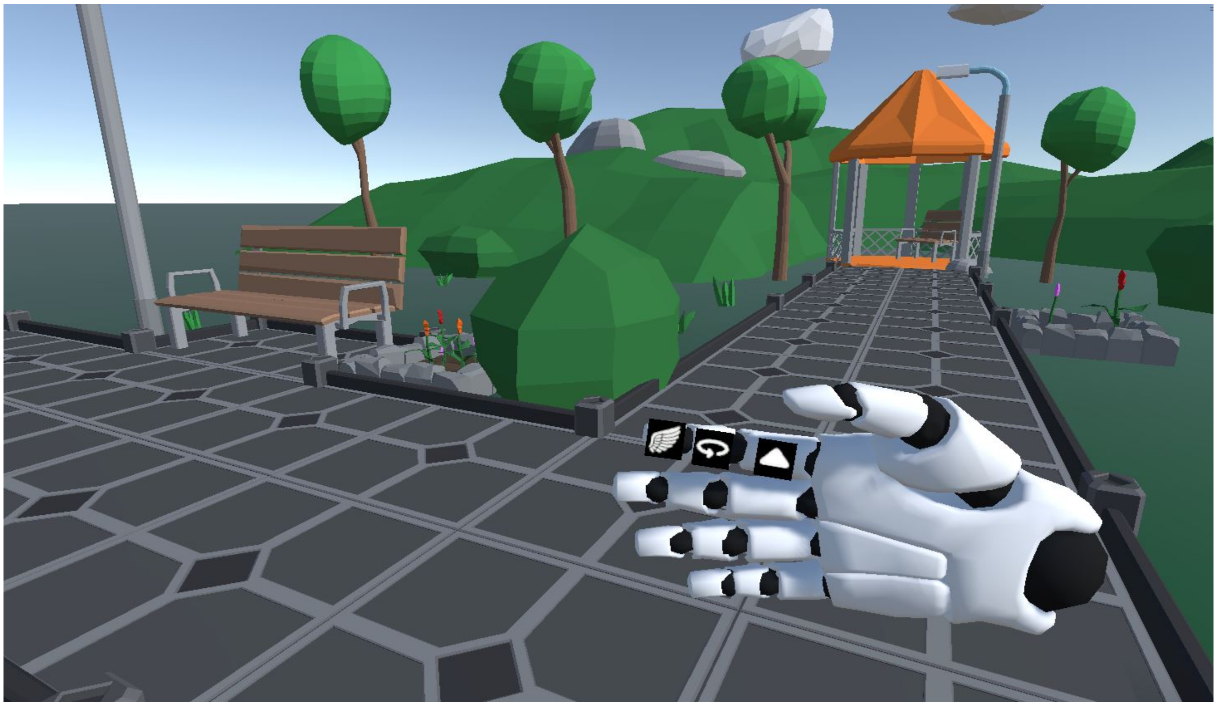


Figure 3.5: Example of gaze locomotion with Gaze+Microgestures. Holding a pinch to the index fingertip translates and rotates the viewport in the gaze direction. Holding the middle segment of the index finger only rotates the viewport for in-place rotation. Holding the base segment of the finger moves the viewport in the direction the user looked at when the pinch began, allowing free browsing of the environment during locomotion.

3.4.5 Gaze Cursor Control

Gaze+Microgestures can be used to enhance accessibility by enabling direct access to alternative gaze cursor modes.

Previous works have explored using gestures to switch the mode of the gaze cursor to avoid the 'Midas Touch' problem (Istance et al., 2008). In a similar way, gaze cursor modes can be mapped to the gesture set. The index fingertip can remain for selection, while the index middle can toggle the gaze cursor on and off, and the index base can place the cursor on the screen, or pick it up again. The slider can adjust the size of the cursor, allowing either for multi-selection, or for a more generous cursor size for selections. Alternatively, gestures can be used to enter or exit gaze refinement mode for pinpointing (Kytö et al., 2018).

3.4.6 Text Entry

Extending the gesture set allows Gaze+Microgestures to be used for controller-less text entry. Mid-air keyboards in AR/VR often underperform, suffering from low input speed, high error

Index Tip	Index Middle	Index Base	Slider
Select	Toggle cursor	Pick up/Put down cursor	Adjust cursor size
Select	Toggle pinpointing	NA	NA

Table 3.5: Gaze+Microgesture mappings for gaze gaze cursor control.

rates, and high physical demand (L. H. Lee et al., 2020; J. H. Kim et al., 2014; J. Hu et al., 2025) A compact T9-style keyboard controlled through microgestures (L. H. Lee et al., 2019) could offer a lightweight alternative for short text tasks such as filling in forms, entering names, or typing search queries (see Figure 1.1). Gaze can select text input fields while thumb-to-finger gestures perform the typing operation. Rather than replacing full virtual keyboards, this approach could complement them by reducing physical effort and improving efficiency for situations where only brief text input is needed (such as short messages, contact info, or passwords).

3.4.7 Other Application Shortcuts

Gaze+Microgestures offers shortcuts to functionality present within the UI, improving efficiency for power users, by reducing pointing, menu traversal, and enabling interaction with partially obscured windows (see Section 3.2). Table 3.6 shows a list of other potential mappings for applications where Gaze+Microgestures stands to logically provide benefit compared to Gaze+Pinch alone. For these applications, the index fingertip is always mapped to Gaze+Pinch.

Application	Index Middle	Index Base	Slider
Media Player	Play/Pause	Skip forward	Volume
Photo Viewer	Last Photo	Next Photo	Zoom
Social Media	Like	Comment	Scroll page
Video Call	Toggle mute	Toggle Camera	Call volume
Presentation	Laser pointer	Previous slide	Zoom in
Email	Archive	Delete	Scroll email contents
Map	Pan to gaze point	Zoom to gaze point	Global zoom

Table 3.6: Gaze+Microgesture mappings for various applications. In these examples, the index fingertip is always mapped to Gaze+Pinch.

Chapter 4

Microgesture Detection and System Design

To extend Gaze+Pinch, two additional thumb-to-finger pinches to the middle and proximal phalanges of the index finger and a sliding gesture of the thumb along the palmar side of the middle finger were introduced. Discrete gestures were detected using Unity colliders on the thumb and index finger, and the sliding gesture was detected by calculating the distance from the thumb to the middle finger. Long-pinches were detected by holding a pinch for longer than 0.5 seconds, following its description by Pfeuffer et al. where a 'fast-pinch' is a pinch less than 0.5 seconds Pfeuffer et al., 2017.

The following chapter details the detection of these gestures, and the system-level design employed to map these gestures to functionality.

4.1 Microgesture Detection

To track the finger positions, the Manus Quantum Metaglove was used (*Motion Capture Gloves — Quantum Metagloves by MANUS* 2025) in Unity v2022.3, with a Vive Tracker 3.0 for positional tracking (Vive, 2025b) and the Vive Pro Eye VR headset (Vive, 2025a). A glove-based solution was chosen to avoid occlusion problems encountered by ocular tracking, and to ensure consistent interactions regardless of hand and finger visibility (Soliman et al., 2018). This also permits more natural and expressive hand positioning by removing line-of-sight tracking constraints. Quantum pinch correction was disabled for 1:1 tracking of the fingers. This allowed us to track pinches to segments of the finger other than the fingertips, but resulted in a large discrepancy between the model and the real hand position. To compensate for this discrepancy, each user performed a custom calibration process.

4.1.1 Gesture Calibration

Sphere colliders with a radius of 0.005 metres were placed on the index finger's distal, middle and proximal segments. A sphere collider with radius of 0.0075 metres was attached to the thumb-tip, and was offset by 0.005 metres towards the base of the thumb to align the sphere with the round of the thumb. A discrete gesture was triggered when the thumb tip collided with a collider on the index finger using Unity's OnTriggerEnter and OnTriggerExit methods. To compensate for the discrepancy between the tracked model and the real hand position, the position of the index finger's sphere colliders were adjusted through the following procedure: In VR, users were asked to hold a pinching gesture to the top, middle, or base segment of the index finger. Once users verbally confirmed they were holding the gesture, the corresponding sphere collider was repositioned to the thumb-tip. This way, when the user performed the gesture in real life, the model aligned the thumb with the appropriate collider in space. The virtual hand was not rendered during this process, as pilot testing showed that some users tended to align the virtual hand rather than their real hand, resulting in skewed gestures in real-life. Additionally, users were shown three numbered boxes in front of them that each corresponded to one segment of the index finger. When the system recognised a gesture, the associated box lit up as feedback to the user. Following initial calibration, users were instructed to make each box light up individually. This task was provided as pretesting found that some users would perform gestures differently when task-driven, as opposed to when asked only to perform a gesture, resulting in poor calibration during the study.

To calibrate the continuous microgesture along the middle finger, users were asked to slide their thumb up and down the palmar side of their middle finger to find their comfortable range of motion. When they were ready, they were asked to pause at the bottom, middle and top of their comfortable range, and placed transforms at each of the thumb-tip's positions, anchoring the transforms to the middle finger's metacarpophalangeal (MCP) joint, middle phalanx, and fingertip respectively.

4.1.2 Sliding Detection

The finger's sliding path was approximated using two line segments: one from the base to the middle transform, and another from the middle to the tip. Each frame, the distance between the thumb tip and the nearest line segment was calculated in three orthogonal dimensions: palmar (away from the palm), radial (towards the index finger), and proximal (along the length of the segment). A sliding gesture was registered when the thumb tip's distances in all three dimensions were below the respective thresholds for either segment.

The threshold for proximal distance was 0, meaning an orthogonal line could be made between the thumb and line segment. The radial threshold was set to $\leq$ 0.001 metres. Palmar thresholds differed between segments: for the middle to base segment the threshold was set to $\leq$ 0.001 metres, and for the middle to tip segment the threshold was set to $\leq$ 0.01 metres. Different palmar thresholds were used as pretesting observed a greater disparity between the model and real hand towards the fingertips. Threshold values were derived experimentally through iterative in-lab testing. Values that were set too low resulted in intermittent detection, while values that were set too high caused the slider to feel ’sticky’, as gestures continued to register after the thumb had begun to move away. The chosen thresholds balanced these effects across users to reliably detect sliding while tolerating hand-tracking error. Further testing showed that modelling the path with two line segments offered the best trade-off. A single segment from the base to the tip of the finger was too permissive, and would fail to detect when a user had released their thumb when the middle finger was bent. Three segments (one per phalanx) aligned too closely with the tracked model, and resulted in unreliable detection. Two segments were found to provide a balance between reliable gesture detection and robustness against tracking inaccuracies.

When a sliding gesture was detected, the amount of movement was computed each frame and reported to the scene. The closest point on the line segment to the thumb tip was identified, and its relative position along the total slider length was expressed as a percentage. This value was compared with the corresponding percentage from the previous frame to calculate the displacement. The percentage position was then scaled by the physical length of the slider to obtain the absolute distance moved. Finally, the sliding displacement was amplified by a factor of 3, while displacement from dragging with Gaze+Pinch was amplified by a factor of 1.5. Both dragging and scrolling were implemented with an inertia that reduced the movement by 90% each second when not scrolling or dragging.

4.1.3 Visual Feedback

0.02x0.02 centimetre quads were rendered on the palmar surface of the index distal, middle and proximal phalanges which displayed an icon representing the current functionality mapped to that finger segment. Icons were designed with high-contrast colours and minimal detail to remain legible in peripheral vision without obscuring the finger itself. This feedback changed colours when pressed, acting as feedback for if a user forgets the mappings on their hand. Similarly, a line of width 0.005 metres was rendered along the palmar side of the middle finger. This line would light blue when a slide gesture was registered, and was grey otherwise. A sphere of diameter 0.0075 metres moved to the closest point on the slider to the thumb-tip while sliding. This feedback can be seen in Figure 3.4.

Eye gaze was tracked using the SR Anipal SDK and the Vive Pro Eye (Vive, 2025a), then filtered using a €1 filter (Casiez et al., 2012). A gaze cursor was displayed to reduce tracking difficulty and focus task demands on the microgestures.

4.2 System Design

To resolve gestural input, the following components are needed:

- Gesture Detection,
- Gaze Tracking,
- In-World Gaze Targets,
- Scene Behaviour.

The scene behaviour defines the interaction between gesture input, gaze input, and in-world gaze targets.

In this project, in-world gaze targets implement an interface that exposes functionality for each microgesture. Gaze tracking is implemented using ray-casts: when the user's gaze intersects a valid gaze target, an event containing the target is published. The scene behaviour subscribes to these events, storing a reference to the currently looked at target and loading its associated icons onto the hand when triggered. The scene behaviour subscribes also to microgesture events; detection occurs when the thumb tip's OnTriggerEnter or OnTriggerExit functions register contact with a finger segment, or when the slider detects a sliding gesture. Each event contains the position of the gesture, its state (start, hold, release), and, if relevant, how far has been slid or dragged. When a microgesture begins, the scene behaviour locks the currently looked at target until the gesture is released. Once the gesture is released, interaction may shift to other gestures or gaze targets. The scene behaviour can also use default, system-level gesture functionality as either a higher, or lower priority to gaze-target functionality. Additionally, the scene behaviour employs a 'sticky gaze', keeping the last looked at target selected so that users can look down to their hands and see the functionality assigned to it.

This architecture is agnostic to the specific gaze targets and their functions. Targets may represent window-sized applications, or individual UI elements within them. For example, in an email client, the email list and message body can be registered separate gaze targets, mapping 'archive' and 'delete' to microgestures when the email list is in view, but mapping 'reply' and 'forward' when looking at the message contents.

Chapter 5

User Study

Extending the gesture set of Gaze+Pinch enriches the expressivity and functionality of interactions, however mapping these gestures to functionality must be done in ways that remain learnable and effective.

In this project, two strategies grounded in established desktop metaphors are explored. The mouse mapping aligns the index finger's segments and the middle finger slider with the left-, middle-, and right-click functions and the scroll wheel, supporting full mouse emulation in XR. The function key mapping instead ties the gestures to application-level shortcuts, enabling users to trigger context-specific commands without needing to summon context menus, or divert their gaze to interface elements. Together, these mappings represent two alternative strategies: one that uses gestures to deliver system-level input to the user's gaze point, and another that uses gestures as application-level shortcuts. The user study reported in this chapter evaluates the user experience, task load, and performance of these designs to better understand their usability, learnability, and potential to support controller-free interaction in AR/VR.

Participants engaged in two practical tasks — travel planning and spreadsheet data manipulation — selected as representative productivity tasks that users could realistically encounter in XR environments. These scenarios were chosen because they reflect common office and everyday computing activities: travel booking combines navigation, selection, and confirmation across multiple interface elements, while spreadsheets emphasise precise manipulation, data entry, and editing. The tasks were chosen to emphasise speed and efficiency and gain meaningful performance metrics, while avoiding artificial task setups. The results of the two Gaze+Microgestures variants are compared against Gaze+Pinch and against a combined condition where both Gaze+Pinch and Gaze+Microgestures can be jointly used. To study Gaze+Microgesture interactions, the Gaze+Microgestures condition has Gaze+Pinch interactions disabled, while the Combined condition enabled both forms of interaction.

The research questions addressed in this study are as follows:

1. Does Gaze+Microgestures improve performance and user experience compared to Gaze +Pinch?

2. Does Gaze+Microgestures increase users' self-reported task load compared to Gaze+ Pinch?

3. How do users adopt and integrate Gaze+Microgestures when both Gaze+Pinch and Gaze+Microgestures are available?

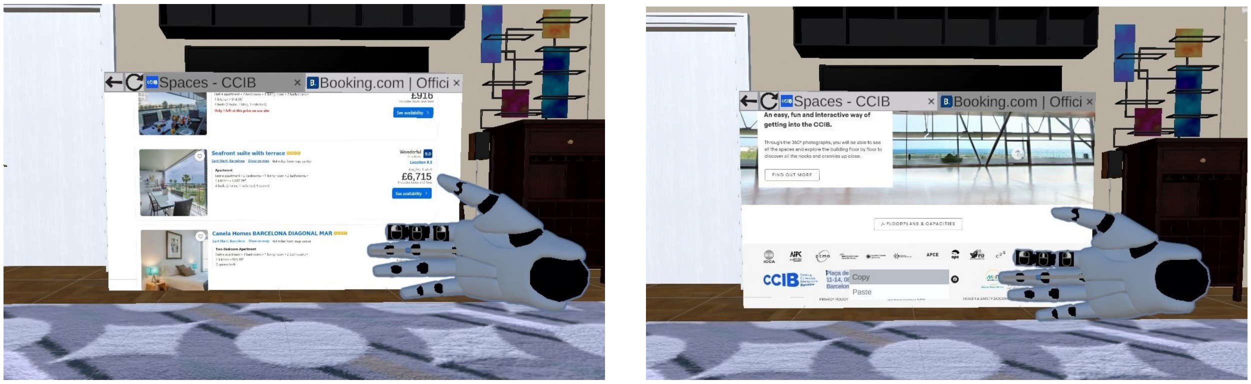
The independent variable was the interaction technique, with three levels: Gaze+ Microgestures, Gaze+Pinch, and Combined. These were evaluated across two tasks: travel planning and spreadsheeting.

5.1 Study Procedure

To begin, participants signed an informed consent form and completed a demographics questionnaire, including their prior experience with XR. They then put on the Manus Quantum gloves (*Motion Capture Gloves — Quantum Metagloves by MANUS* 2025) and performed the manufacturer's native calibration. After this, the study structure was explained: participants had to complete two tasks (travel planning and spreadsheet manipulation), with task order counterbalanced across participants. Each task was performed with each condition (Gaze+Pinch, Gaze+Microgestures, Combined). For each condition, they performed one training run and two experimental trials, resulting in six valid trials per participant. The order of Gaze+Pinch and Gaze+Microgestures was randomised, while the Combined condition was always last.

At the start of each task, participants were introduced to the assigned task and instructed on how to complete it with the given technique. They then put on the HTC Vive Pro Eye headset (Vive, 2025a), completed eye-tracking calibration, and underwent a dedicated microgesture-detection calibration (Section 4.1.1). Following calibration, participants performed one practice run of the task, then completed two trials with the assigned technique. Afterwards, they removed the headset and completed a post-condition questionnaire, followed by a short opportunity to comment on techniques (see. Section 5.4). This procedure was repeated for the second technique, with recalibration before the practice runs. Finally, participants completed the Combined condition, in which both interaction styles were available. They were encouraged to complete the tasks using whichever input style they preferred. At the end, they filled out questionnaires and provided feedback.

After finishing all three conditions for the first task, participants repeated the same process with the second task. The full study lasted approximately two hours, and participants received £10 compensation for their time. The study duration was determined through pilot testing, where explanation, calibration, training, and the issuing of questionnaires for each of the 3 conditions and two tasks were found to consume a considerable amount of time.


Spaces - CCIB
Booking.com | Offici
£916
Seafront suite with terrace
£6,715
Camela Homes BARCELONA DIAGONAL MAR
Spaces - CCIB
Booking.com | Offici
An easy, fun and interactive way of getting into the CCIB.
CCIB
Copy
Paste

(a) (b)

Figure 5.1: Screenshots from the Travel Planning Task: (a) User scrolls down a list of hotels. (b) User opens a context menu to copy text.

5.2 Travel Planning Task

In the travel planning task, the extended gesture set was mapped to mouse button functionality. The task was designed as a compound task to elicit interactions spanning the full range of mouse buttons, ensuring that participants engaged not only with frequently used actions (left- and right-click) but also with less common operations typically assigned to the middle mouse button. The following section describes the task, and how these same interactions are performed with the three techniques (Gaze+Pinch, Gaze+Microgestures, Combined).

In this task, participants interacted with a virtual browser window to book a hotel near a given venue in VR (see 5.1a) The scenario was designed to be concise and plausible while eliciting interactions that engage the full range of the extended gesture set. Participants were told to imagine that they want to book a hotel within specified budget and distance constraints. To complete the task, they located candidate hotels within the budget, compared their distances to the venue, and reserved the closest option. The budget and distance constraints were included only to provide narrative justification for the task steps; no real decision-making was required from participants. At each step of the task, the next element to interact with was highlighted with a blue border to indicate the subsequent action.

The exact task sequence is as follows:

1. The task starts with the venue's page open.

2. Participants scrolled to the bottom of a venue's webpage.

3. They copied the address.

4. They switched to a hotel listing,

5. scrolled a fixed distance down, and
6. opened three highlighted hotels in new tabs as they encountered them.
7. After all three hotels were opened, they navigated to the first hotel's tab,
8. opened its "map view," and
9. pasted the copied address to reveal the distance to the venue.
10. They repeated this process for the remaining hotel tabs, pasting the address into each map view.
11. Finally, after pasting the address into the third hotel,
12. participants closed the map view
13. and reserved the hotel by selecting the reserve button.

The travel planning task included copying and pasting to elicit "right-click" interactions for opening context menus (see 5.1b), opening links in new tabs to elicit "middle-click" interactions, and scrolling to compare the extended gesture set with the baseline scrolling technique. Table 5.1 summarises how each action was performed across the three techniques. In all cases, the default pinch – thumb to finger — served as the basic "left-click."

The total microgesture count for one trial is: 18 thumb-to-tip pinches, 3 thumb-to-medial pinches, 4 thumb-top-proximal pinches, and 0.5 metres of scrolling. Using Gaze+Pinch, the total gesture count comes to: 20 index tip pinches, 7 long-presses, and 1 metre of dragging. Note that scrolling and dragging feature momentum and inertia, which means users will not move their arm an equal distance to the scroll or drag amount. Visual guides indicate the next object users must interact with.

The browser window was positioned 2.5 m in front of the user, spanning a horizontal field of view of 50° and a vertical field of view of 30° (visual angle), including the top taskbar. Participants received auditory notifications when performing a correct or incorrect gesture. This feedback helped them distinguish between cases where a gesture was not recognised and cases where the wrong gesture for the task was executed. The context menu which had a size of 15.4 x 6.71° (visual angle). Elements on the webpage and context menu options reacted to both gaze and gesture input by changing colours when looked at, or selected.

5.3 Spreadsheeting Task

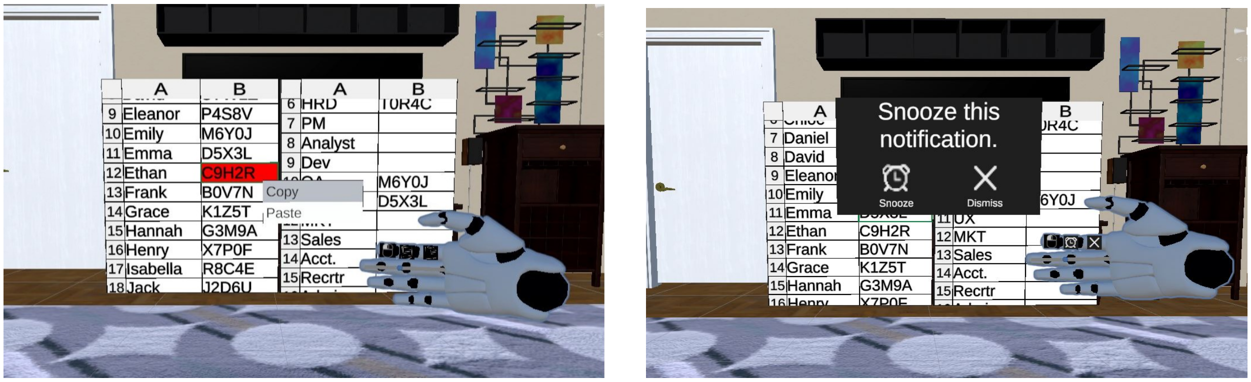
In this task, participants copied and pasted values across two spreadsheets in VR while periodically responding to notifications that had to be either snoozed or dismissed. The scenario was designed to emphasise speed while remaining simple and plausible in an XR

	Copy/Paste	Open in New Tab	Scroll
Gaze+Pinch	Long-pinch while looking at highlighted text to open the context menu.	Long-pinch while looking at a link to open the context menu.	Pinch and drag the webpage.
Gaze+Microgestures (Mouse)	Tap the index finger's proximal phalanx to open the context menu.	Tap the index finger's medial phalanx while looking at a link.	Slide the thumb along the middle finger's palmar side.
Combined	Long-pinch on tip OR tap on proximal phalange to open context menu.	Long-pinch to open context menu OR tap the middle phalanx.	Pinch and drag OR slide along palmar side of middle finger.

Table 5.1: How to perform key actions in the Travel Planning task using Gaze+Microgestures, Gaze+Pinch, and Combined.

setting. Participants copied a highlighted cell from a spreadsheet on the left and pasted it into a highlighted cell on the right; targets were marked in red, and if out of view, a red bar at the top or bottom indicated the scroll direction. Copy and paste first required cell selection, mirroring keyboard shortcuts as application-wide commands that could operate on single cells or larger selections, and could be issued while looking anywhere in the workspace. Randomly throughout the task, a total of 4 notifications appeared in the centre of the workspace that participants were required to either snooze or dismiss. An instructional text on the notification told them which action to choose (see 5.2a). Notifications appeared only after copy or paste actions to prevent accidental dismissal, and were shown at a large size (1.5 metres wide, 0.85 metres tall) with options immediately visible to ensure they were addressed promptly.

The function key metaphor was adopted to map additional microgestures to application-wide shortcuts: just as function keys change meaning depending on the application in focus, shortcut functionality varied depending on the window currently being looked at. To perform copy and paste with Gaze+Microgestures, participants selected a cell by pinching their



(a) (b)

Figure 5.2: Screenshots from the Spreadsheet Task: (a) User opens a context menu to copy a cell from a spreadsheet. The icons on their index middle and proximal phalanges are mapped to copy and paste respectively. (b) The user receives a notification instructing them to snooze it. The icons on their index finger now correspond to 'Snooze' and 'Dismiss'.

index fingertip and then issued the corresponding copy or paste microgesture. With Gaze +Pinch, participants long-pinched the index finger's tip and copy/pasted via the context menu. Scrolling worked analogously to task 1, either via pinching-and-dragging or by sliding along the middle finger. In all cases, the default pinch — thumb to index fingertip — served as the basic "click". Table 5.2 summarises the interaction patterns for Gaze+Pinch, Gaze+ Microgestures, and Combined in this task.

For this task, left-handed participants had the middle and proximal pinch functions swapped. This ensured that snooze and dismiss remained on the left and right sides, matching the notification icons, and that copy and paste also stayed consistently left and right.

For each trial, users completed 10 copies and 10 pastes, experienced 4 notifications, and scrolled the length of both spreadsheets. Using the baseline condition, that results in 24 pinches, 20 long-presses, and 2 metres of dragging. Using Gaze+Microgestures technique, users performed a minimum of 20 index tip pinches, 10 pinches to the middle segment of the index finger, 10 to the base, and 1 metre of scrolling along the middle finger.

Each window spanned 25° horizontally and was placed side by side, 2.5 m in front of the user. Individual spreadsheet cells measured 12° in width and 3° in height. The notification window had a size of 32.2° by 18.6° visual angle. Participants received auditory feedback: a negative sound when an incorrect gesture was detected and a positive sound when the correct gesture was detected. This feedback helped them distinguish between cases where a gesture was not detected and cases where the wrong gesture for the task was executed. Interactable elements including spreadsheet cells, context menu options, and notification buttons reacted to both gaze and gesture input by changing colour when looked at or interacted with.

5.4 Measures

During each trial, completion time was measured for each trial, along with total hand movement, total eye movement, and the number of incorrect input actions. After each technique, participants filled out the Raw NASA TLX (Byers and Hill, 1989). In addition, they had to answer a 5-point satisfaction question, ranging from 0 (Very dissatisfied) to 5 (Very satisfied). Finally, they were asked a Single Ease Question (SEQ) which ranged from 0 (Very difficult) to 7 (Very easy).

Following this, participants had the chance to comment on the techniques. The following open-ended questions were asked:

- How did you find that condition? Did anything stand out as especially good, or especially bad?
- Could you see yourself using this?

After users experienced their second condition, they also were asked to give their preference between the two they had tried, then explain their reasoning. After the final Combined condition, users were asked to give their preference over the three conditions they had now tried, and to explain their reasoning. They were then asked the following:
”If you could design your own gestures, what would you choose to include, or not include, and where would you put it?”
To answer this question, they were provided a list of the functionality they used in the task, and buckets representing the different parts of the finger they used. Two additional buckets for ’Other Segments’ and ’Unassigned’ were provided. Users were told they could place gestures wherever they liked, and could include, or not include, whatever functionality they saw fit.

5.5 Apparatus

The study was run in Unity v2022.3 on a 64-bit Windows 10 PC, fitted with an Intel Core i7-12700KF CPU, a NVIDIA GeForce RTX 3070Ti and 32GB of RAM. An HTC VIVE Pro Eye VR headset was used for the study, with 110° diagonal FOV, 2880×1600 pixels resolution, and 90 Hz refresh rate and the Manus Quantum Metagloves (*Motion Capture Gloves — Quantum Metagloves by MANUS* 2025) for hand tracking, with quantum pinch correction disabled to allow pinches to different fingers and finger segments.

5.6 Participants

16 participants (9 self-identified as male, 6 as female, and 1 as a third/non-binary gender) aged 21-60 (M = 29.81, SD = 12.72) were recruited from university campus grounds via word

of mouth and posters. 9 participants reported normal vision, while 7 reported corrected-to-normal vision, with 2 reporting as left hand dominant and 14, as right hand dominant. 2 participants reported no prior VR experience, 4 as having tried it once or twice before, 6 as having used it 3-10 times in the past, 3 as having had tried it more than 10 times in the past, and 1 participant reported as currently using VR frequently. 1 participant reported having used the Apple Vision Pro (and therefore the Gaze and Pinch technique) between 3 to 10 times in the past, while the remaining 15 users reported no experience. 8 participants reported no experience with eye tracking, 2 reported having used it once or twice in the past, 5 as having tried it 3-10 times in the past, and 1 participant reported having tried it more than 10 times in the past. 8 participants reported as never having used 'pinch-to-select' in VR (as is present in headsets like the Meta-Quest, Vive Focus, and Apple Vision Pro), 5 reported as having tried it once or twice in the past, 1 reported as having tried it 3-10 times in the past, 1 as having tried it more than 10 times in the past, and 1 as currently using it frequently. 1 participant reported as no experience with a trackpad, 1 reported having used it 3-10 times in the past, while the remaining participants reported that they currently use a trackpad frequently. 1 participant reported having used a mouse between 3-10 times in the past, while the remaining 15 reported that they currently use a mouse frequently. Lancaster university's ethics committee approved this study.

	Copy/Paste	Notifications (Snooze/Dismiss)	Scroll
Gaze+Pinch	Long-pinch the index tip to open context menu.	Pinch on index fingertip while looking at the respective button.	Pinch and drag the spreadsheet.
Gaze+Microgestures (function keys)	Look and pinch the index fingertip to select the cell, then tap the index finger's proximal/medial phalanx to copy/paste.	Tap the index finger's proximal/medial phalanx to snooze/dismiss.	Slide the thumb along the middle finger's palmar side.
Combined	Long-pinch to open context menu OR tap the index tip to select cell + tap the proximal/medial phalange to copy/paste there.	Select icon with pinch on index fingertip OR tap on proximal/medial phalange.	Pinch with index fingertip to drag OR slide along palmar side of the middle finger.

Table 5.2: How to perform key actions in the Spreadsheet task using Gaze+Microgestures, Gaze+Pinch, and Combined.

Chapter 6

Results

Unless stated otherwise, quantitative data were analysed using one-way repeated-measures ANOVA ($\alpha = .05$). When sphericity was violated (Mauchly's test), Greenhouse-Geisser corrections were applied. Normality was assessed via Shapiro-Wilk tests, Q-Q plots, and histograms; if violated, data were analysed with the Aligned Rank Transform (ART) (Wobbrock et al., 2011). Post-hoc tests were performed only following significant effects, with Bonferroni correction. Pairwise t-tests were used when parametric assumptions held, and ART-C otherwise. Effect sizes are reported as η_g^2 for parametric tests and η_p^2 for ART. Subjective metrics were analysed using Wilcoxon signed-rank tests and reported effect sizes using rank-biserial correlations (r).

6.1 Travel Planning Task

6.1.1 Objective Results

Task Completion Time: A significant effect of Technique was found ($F(2, 30) = 5.82$, $p < .05$, $\eta_g^2 = .12$), indicating differences in task completion time between techniques. Post-hoc tests showed that Combined (M=81.2 seconds, 95% CI [67.1, 95.3]) outperformed Gaze +Microgestures (M=105 seconds, 95% CI [89.4, 110], $p = .020$), while no difference between Gaze+Microgestures and Gaze+Pinch (M=96.9 seconds, 95% CI [82.6, 111]) was observed, nor between Gaze+Pinch and Combined ($p > .05$).

Total Hand Movement: A significant effect of Technique emerged ($F(2, 30) = 25.8$, $p < .05$, $\eta_g^2 = .633$), showing that hand movement differed across techniques. Post-hoc pairwise comparisons (Bonferroni-corrected) revealed that Gaze+Microgestures (M=0.707 metres, 95% CI [0.137, 1.28]) reduced hand movement compared to Gaze+Pinch (M=2.92

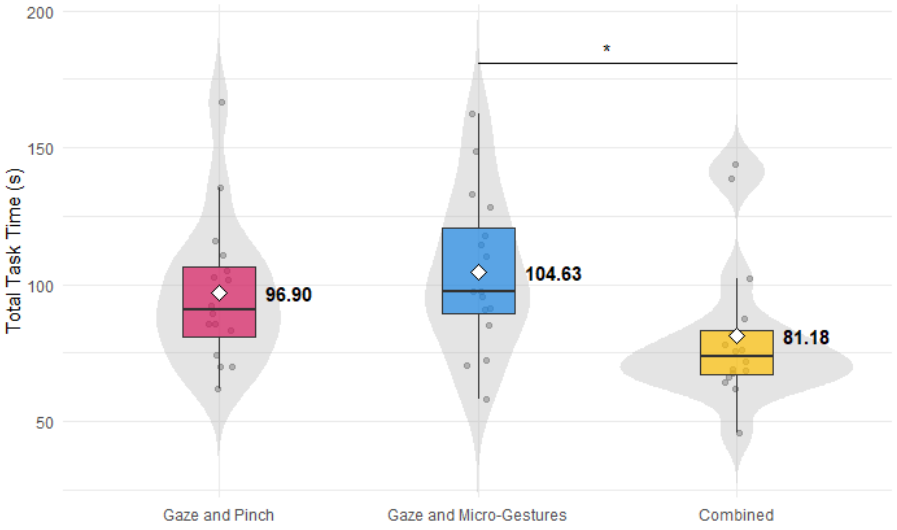


Figure 6.1: Average Task Time for the Travel Planning task

metres, 95% CI [2.60, 3.24], $p = 4.36e-7$) and Combined (M=2.81 metres, 95% CI [1.68, 3.94], $p = 2.36e-5$).

Total Eye Movement: There was no significant effect of Technique on eye movement ($F(1.44, 21.6) = 0.409$, $p = .604$, $\eta_g^2 = .006$) (Summary: M=14.8, 95% CI [13.8, 15.8]) (Gaze +Pinch: M=14.4, 95% CI [12.5, 16.4]) (Gaze+Microgestures: M=14.9, 95% CI [13.2, 16.7]) (Combined: M=14.7, 95% CI [12.6, 16.8]).

Total Errors: Any pinch detected that was not the next pinch required to complete the task, or any correct microgesture directed to the wrong gaze target, was counted as an error. There was no significant effect of Technique on the number of errors ($F(2, 30) = 2.29$, $p = .119$, $\eta_p^2 = .132$) (Summary: M=7.73, 95% CI [6.45, 9.01]) (Gaze+Pinch: M=8.44, 95% CI [5.81, 11.1]) (Gaze+Microgestures: M=8.06, 95% CI [5.85, 10.3]) (Combined: M=6.69, 95% CI [4.47, 8.91]).

Total Number of Fixations at Hand: No significant effect of Technique appeared for the number of fixations on the hand ($F(2, 30) = 1.50$, $p = .240$, $\eta_p^2 = .091$). Participants fixated on their hand on average 0.813 times with Gaze+Pinch (95% CI [0.154, 1.47]), 2.28 times with Gaze+Microgestures (95% CI [0.902, 3.66]), and 0.938 times with Combined (95% CI [0.250, 1.62]).

Behaviour with the Combined condition: Users chose the microgesture to copy instead of long-pressing 56.25% of the time, and to paste 45% of the time. They used the middle click to open a new tab 62.5% of the time. In 42.19% of instances, users scrolled down webpages using the slider, rather than dragging.

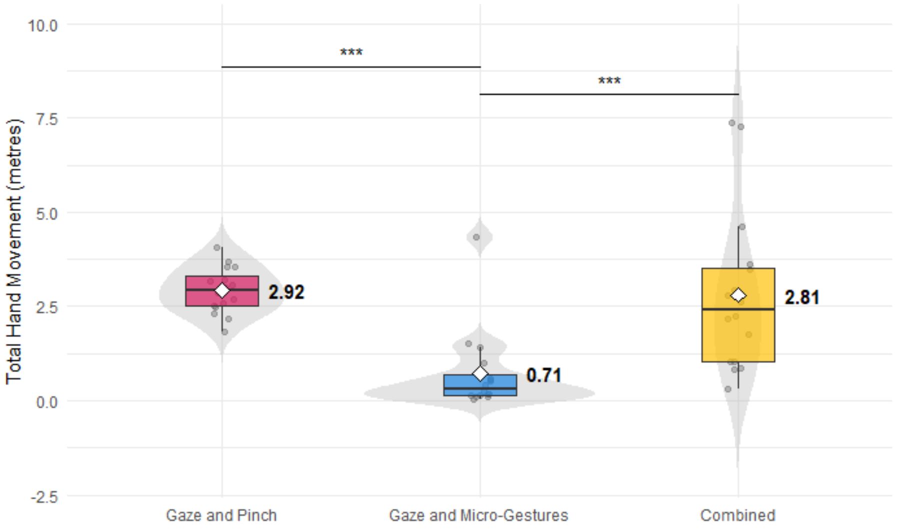


Figure 6.2: Average hand movement during the Travel Planning task.

Table 6.1: ANOVA results for effect of Technique on NASA-TLX dimensions in the Travel Planning Task.

Dimension	F **(df)**	p	**Effect size**	**Mean**	**95% CI**
Mental Demand	$F(1.34, 20.1) = 0.045$	.896	$\eta_g^2 = .001$	5.79	[5.37, 6.20]
Physical Demand	$F(2, 30) = 1.07$	.356	$\eta_g^2 = .020$	5.20	[3.36, 7.05]
Temporal Demand	$F(1.46, 21.9) = 1.42$	.259	$\eta_g^2 = .006$	4.49	[3.40, 5.57]
Performance	$F(1.21, 18.1) = 3.39$	.076	$\eta_g^2 = .095$	3.69	[0.95, 6.43]
Effort	$F(2, 30) = 1.11$	.342	$\eta_p^2 = .069$	6.70	[4.38, 9.03]
Frustration	$F(2, 30) = 1.89$	.169	$\eta_p^2 = .112$	3.99	[2.02, 5.95]

6.1.2 Subjective Results

NASA TLX: Table 6.1 illustrates the results of the Raw NASA-TLX. No significant effect of Technique on overall NASA-TLX score (or subscales) was observed for the travel planning task ($F(1.30, 19.4) = 1.07$, $p = .334$, $\eta_p^2 = .022$).

Single Ease Question: A significant effect of Technique was found ($F(2, 30) = 3.78$, $p < .05$, $\eta_p^2 = .201$), indicating differences in ease of use across techniques. Post-hoc pairwise comparisons showed that Combined (M=5.75, 95% CI [5.30, 6.70]) outperformed Gaze+ Microgestures (M=5.06, 95% CI [4.54, 5.84], $p = .0304$), while no difference was observed between Gaze+Microgestures and Gaze+Pinch, nor between Gaze+Pinch and Combined ($p > .05$).

Satisfaction Question: A significant effect of Technique was found ($F(2, 30) = 4.38$,

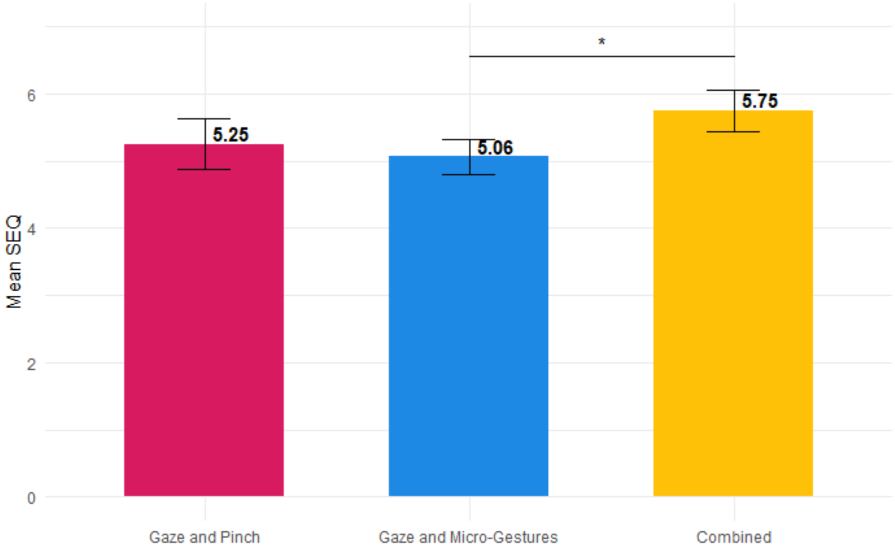


Figure 6.3: Average SEQ results for the Travel Planning task

$p < .05$, $\eta_p^2 = .226$), indicating differences in satisfaction across techniques. Post-hoc pairwise comparisons (Bonferroni-corrected) showed that Combined (M=4.5, 95% CI [4.16, 4.84]) outperformed Gaze+Microgestures (M=4.00, 95% CI [3.66, 4.34], $p = .0191$), while no difference was observed between Gaze+Microgestures and Gaze+Pinch, nor between Gaze+ Pinch and Combined ($p > .05$).

Technique Preference: When asked to choose between Gaze+Pinch and Gaze+ Microgestures, 75% of users preferred Gaze+Pinch. When asked to choose between Gaze+ Pinch, Gaze+Microgestures, and Combined, 75% of users preferred Combined, with 18.75% preferring Gaze+Pinch, and 6.25% preferring Gaze+Microgestures.

6.1.3 Qualitative Results

Custom Gesture Sets: For this task, 100% of users chose to include the left-click at the index fingertip. The middle and right-click buttons each were included in 62.5% of designs. The long-press was kept in 75% of designs, 68.75% keeping it at the index fingertip. All participants retained the pinch-to-drag functionality at the index fingertip, and 68.75% of users chose to include the slider in their design.

Willingness to Adopt Techniques: When asked ”Could you see yourself using this technique to complete a task like this?”, 68.75% (11) of participants indicated that they could see themselves using Gaze+Pinch, 68.75% (11) for Gaze+Microgestures, and 75% (12) indicated they could see themselves using Combined. Several participants provided a conditional response, indicating a specific barrier that prevented them from adopting the technique. For Gaze+Pinch, Gaze+Microgestures, and Combined, 25% (4) of participants

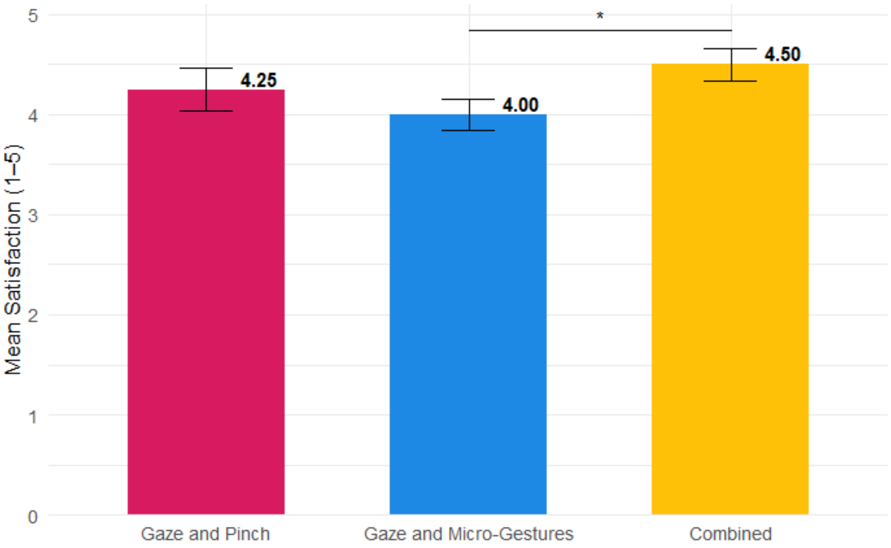


Figure 6.4: Average Satisfaction question results for the Travel Planning Task

gave conditional responses. Responses indicating conditional adoption were systematically coded into five categories, which represented barriers to adoption: Social, Task Complexity, Practice, Environment, and Keyboard Presence. For Gaze+Pinch, 18.75% (3) of participants mentioned social barriers, 12.5% (2) of participants indicated that the complexity of the task was a barrier, 6.25% (1) of participants stated that their adoption depended on the environment they were in, and 6.25% (1) stated that they would only use the technique if they had access to a keyboard. For Gaze+Microgestures, 6.25% (1) of participants indicated social barriers, 12.5% (2) of participants indicated that the complexity of the task was a barrier, and 6.25% (1) of participants stated they would only use the technique if they were well practised. For Combined, 12.5% (2) of participants indicated social barriers, 6.25% (1) indicated that the complexity of the task was a barrier, and 6.25% (1) stated they would only use the technique if they were well practised.

6.2 Spreadsheeting Task

6.2.1 Objective Results

Task Completion Time: A significant effect of Technique was found ($F(2, 30) = 7.82$, $p < .05$, $\eta_g^2 = .096$), indicating differences in task time across techniques. Post-hoc pairwise comparisons (Bonferroni-corrected) showed that the Combined condition (M=98.8, 95% CI [81.4, 116]) outperformed Gaze+Microgestures (M=127, 95% CI [107, 146], $p = .001$), while

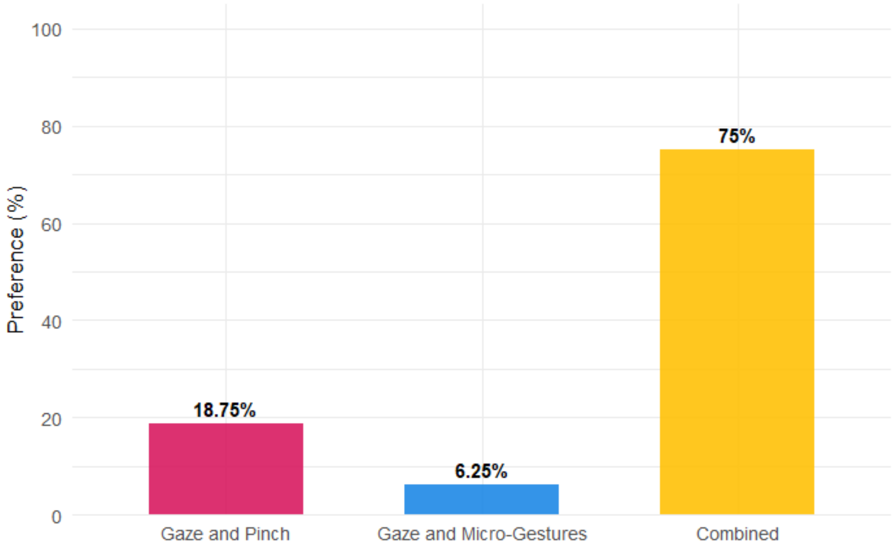


Figure 6.5: Distribution of user preferences after having experienced all three conditions in the Travel Planning task.

no difference was observed between Gaze+Microgestures and Gaze+Pinch (M=117, 95% CI [95.1, 139]), nor Gaze+Pinch and Combined ($p > .05$).

Total Hand Movement: There was a significant effect of Technique ($F(2, 30) = 42.0$, $p < .05$, $\eta_p^2 = .737$), indicating differences in hand movement across techniques. Post-hoc pairwise comparisons (Bonferroni-corrected) showed that Gaze+Microgestures (M=0.475 metres, 95% CI [0.210, 0.740]) outperformed Combined (M=2.58, 95% CI [1.46, 3.69], $p = 2.32e - 5$), which in turn outperformed Gaze+Pinch (M=4.07, 95% CI [3.21, 4.93], $p = 0.00241$).

Total Eye Movement: No significant effect of Technique on eye movement was observed for the spreadsheeting task ($F(1.4, 21.0) = 2.85$, $p = .095$, $\eta_g^2 = .053$) (Summary: M=14.7, 95% CI [13.5, 15.9]) (Gaze+Pinch: M=13.6, 95% CI [11.6, 15.6]) (Gaze+Microgestures: M=16.1, 95% CI [13.4, 18.9]) (Combined: M=14.6, 95% CI [12.6, 16.5]).

Total Errors: Any pinch detected that was not the next pinch required to complete the task, or any correct microgesture directed to the wrong gaze target, was counted as an error. Technique had a significant effect on the number of total errors ($F(2, 30) = 4.91$, $p < .05$, $\eta_p^2 = .246$), indicating differences in errors across techniques. Post-hoc pairwise comparisons (Bonferroni-corrected) showed that Gaze+Microgestures (M=21.4, 95% CI [13.5, 29.3]) resulted in more errors than Gaze+Pinch (M=10.3, 95% CI [6.80, 13.9], $p = .0245$), and Combined (M=11.3, 95% CI [6.47, 16.2], $p = .0456$).

Total Number of Fixations at Hand: There was no significant effect of Technique on the number of fixations at the hand ($F(2, 30) = 0.147$, $p = .864$, $\eta_p^2 = .00969$). When using

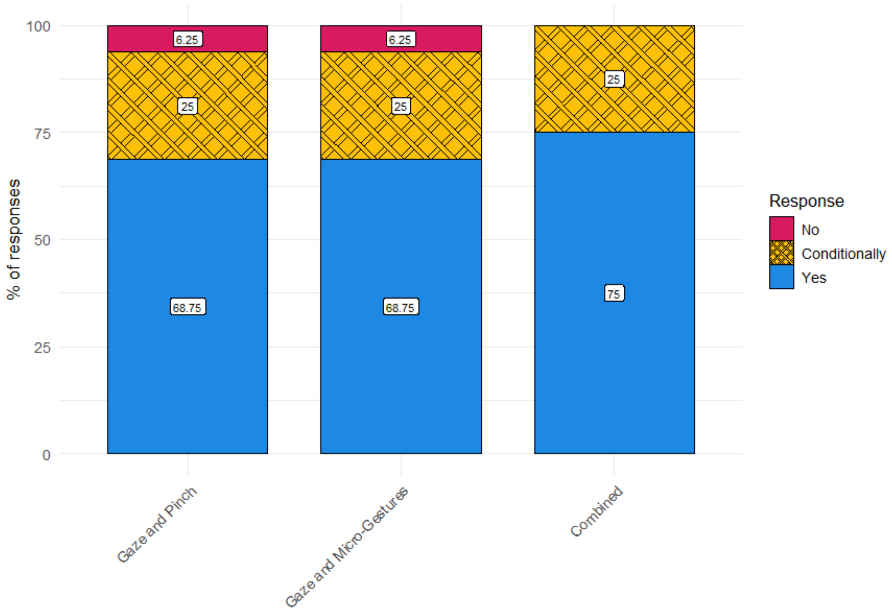


Figure 6.6: User responses when asked if they could see themselves using each technique to perform this travel planning task in real life. Responses that indicated their answer was conditional contribute to the hashed portion of each bar.

Gaze+Pinch: (M=1.59, 95% CI [0.644, 2.54]), Gaze+Microgestures: (M=3.75, 95% CI [0.170, 7.33]), and Combined: (M=2.56, 95% CI [0.814, 4.31]). When using the Gaze+Microgestures technique, 19.8% of fixations on the hand were when a notification had appeared. With the Combined technique, 9.75% of hand fixations occurred while a notification was on screen, and with Gaze+Pinch 0% of hand fixations occurred while a notification was present.

Behaviour with The Combined Technique: 55% of cells copied were copied using shortcuts, and 45% by using a long-press. 56% of cells pasted were pasted using shortcuts, and 44% by using a long-press. 41% of notifications were snoozed using the snooze shortcut, and 59% were snoozed by selecting the snooze button. 36% of notifications were dismissed using the dismiss shortcut, and 64% were dismissed by selecting the dismiss button. Users looked at their hand before dismissing a notification 21.9% of the time while using Gaze+ Microgestures, 6.25% of the time while using Combined, and 0% of the time while using Gaze +Pinch.

6.2.2 Subjective Results

NASA TLX: No significant effect of Technique on overall NASA-TLX score was observed for the spreadsheeting task ($F(2, 30) = 2.91$, $p = .07$, $\eta_g^2 = .024$, see Table 6.2).

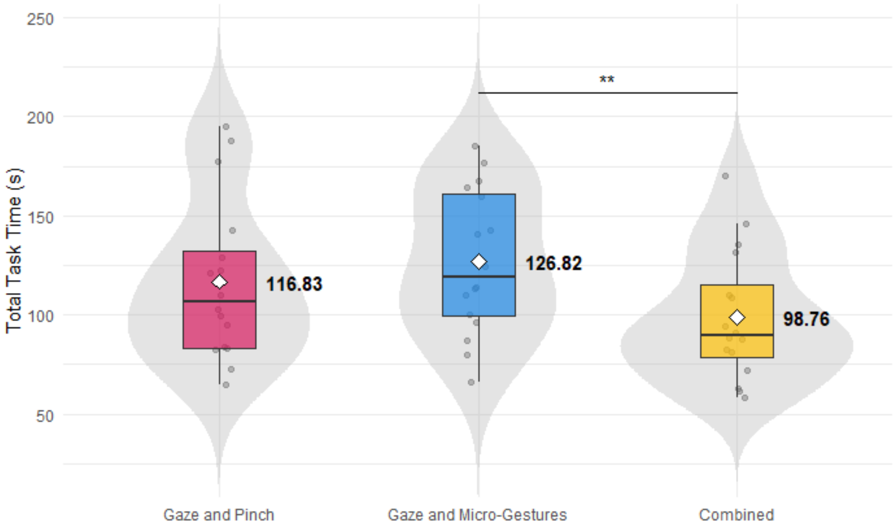


Figure 6.7: Average task completion time for the Spreadsheeting task.

Table 6.2: ANOVA results for effect of Technique on NASA-TLX dimensions in the Spreadsheet Task.

Dimension	F **(df)**	p	**Effect size**	**Mean**	**95% CI**
Mental Demand	$F(2,30) = 3.50$	.189	$\eta_p^2 = .040$	4.65	[0.97, 8.33]
Physical Demand	$F(2,30) = 1.00$	.380	$\eta_p^2 = .063$	4.91	[2.61, 7.21]
Temporal Demand	$F(2,30) = 0.562$	.578	$\eta_p^2 = .036$	3.80	[2.79, 4.82]
Performance	$F(2,30) = 1.27$	.295	$\eta_p^2 = .078$	3.68	[2.52, 4.83]
Effort	$F(1.28,19.2) = 3.04$	.089	$\eta_g^2 = .052$	6.04	[2.67, 9.41]
Frustration	$F(2,30) = 0.544$	.586	$\eta_p^2 = .035$	3.15	[0.97, 5.33]

Single Ease Question: For the spreadsheet task, a significant effect of Technique was found ($F(2,30) = 4.49$, $p < .05$, $\eta_g^2 = .071$). Post-hoc pairwise comparisons (Bonferroni-corrected) showed no statistical significance between the techniques: Gaze+ Pinch: (M=5.5, 95% CI [4.85, 6.15]), Gaze+Microgestures (M=5.19, 95% CI [4.54, 5.84]), Combined (M=6.00, 95% CI [5.30, 6.70]).

Technique Preference: When asked to choose between Gaze+Pinch and Gaze+ Microgestures, 62.5% of users preferred Gaze+Pinch. When asked to choose between Gaze +Pinch, Gaze+Microgestures, and Combined, 75% of users preferred Combined, with 12.5% preferring Gaze+Pinch, and 12.5% preferring Gaze+Microgestures.

Satisfaction Question: For the spreadsheet task, a significant effect of Technique was found ($F(2,30) = 4.05$, $p < .05$, $\eta_p^2 = .213$), indicating differences in satisfaction across

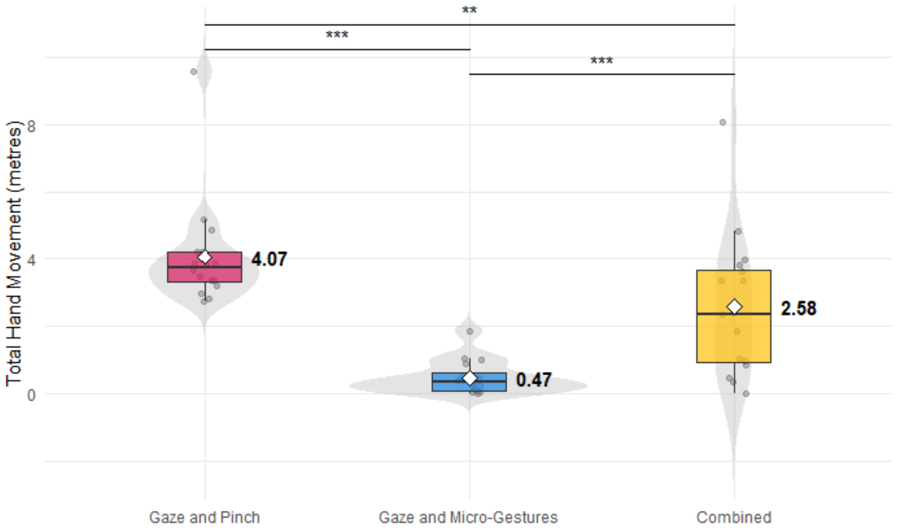


Figure 6.8: Average hand movement for the Spreadsheeting task.

techniques. Post-hoc pairwise comparisons (Bonferroni-corrected) showed that Combined (M=4.63, 95% CI [4.30, 4.95]) outperformed Gaze+Pinch (M=3.94, 95% CI [3.44, 4.43], $p = .0259$), while no difference was observed between Gaze+Pinch and Gaze+Microgestures, nor Gaze+Microgestures and Combined ($p > .05$).

Custom Gesture Feature Set: For this task, 93.75% of users chose to include the left-click in their design, with 85.5 choosing to keep it at the index fingertip. The copy shortcut was included 81.25% of the time, while the paste shortcut was included in 100% of the designs. Snooze and Dismiss shortcuts were included in 43.75% of designs. Long-Press was included in 75% of designs, Pinch-to-Drag was included in 81.25% of designs, and the slider was included in 68.75% of designs.

Willingness to Adopt Techniques: When asked ”Could you see yourself using this technique to complete a task like this?”, 50% (8) of participants indicated that they could see themselves using Gaze+Pinch, 37.5% (6) for Gaze+Microgestures, and 87.5% (14) indicated they could see themselves using Combined. Several participants provided a conditional response, indicating a specific barrier that prevented them from adopting the technique. 37.5% (6) of participants indicated a conditional adoption of Gaze+Pinch, 43.75% (7) for Gaze+Microgestures, and 6.25% (1) for Combined. Responses indicating conditional adoption were systematically coded into six categories, which represented barriers to adoption: Social, Task Complexity, Task Criticality, Practice, Environment, and Keyboard. For Gaze+Pinch, 18.75% (3) of participants mentioned social barriers, 12.5% (2) of participants indicated that the complexity of the task was a barrier, 6.25% (1) stated that their use of the technique depended on how critical the task was, 6.25% (1) stated they

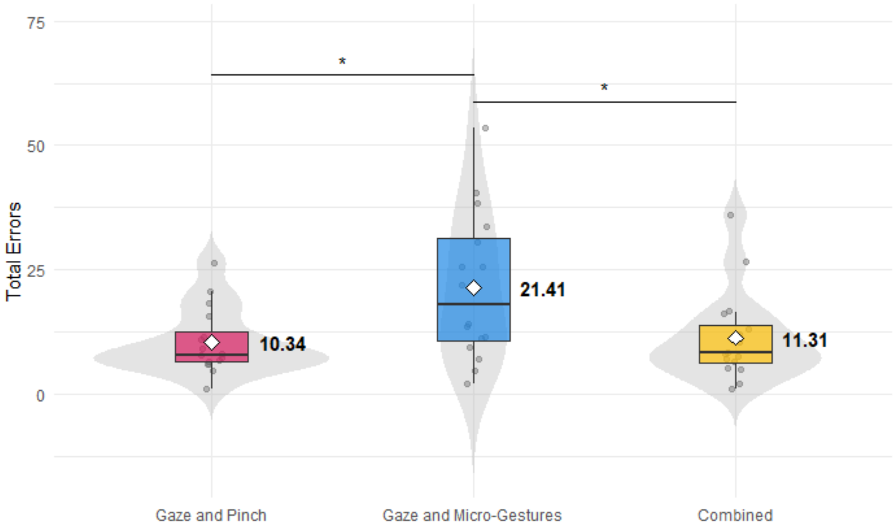


Figure 6.9: Average number of errors detected during the Spreadsheeting task.

would only use the technique if they were well practised, and 12.5% (2) stated that their adoption depended on the environment they were in. For Gaze+Microgestures, 12.5% (2) of participants indicated social barriers, 12.5% (2) indicated that the complexity of the task was a barrier, 6.25% (1) of participants stated that they would only use the technique if they were well practised, 12.5% (2) stated their adoption depended on the environment they were in, and 6.25% (1) indicated they would only use the technique if they had a keyboard available. For Combined, 6.25% (1) of participants indicated that they would only use the technique if they were well practised.

6.3 Qualitative Themes

Participants were asked how they found each condition for each task in a semi-structured interview. Their responses revealed a number of recurring themes that highlight the strengths and challenges of the techniques. What follows is a summary of the themes that spanned across more than one participant's responses, with how often they were raised across each condition.

Practice was identified as an important factor by 87.5% (14) of participants. 43.75% (7) in the Gaze+Pinch condition, 68.75% (11) in the Gaze+Microgestures condition, and 56.25% (9) in the Combined condition. 37.5% (6) of participants mentioned that practice improved their performance with Gaze+Pinch, and 12.5% (2) of participants mentioned that practice would improve their performance with Gaze+Pinch. With Gaze+Microgestures, 62.5% (10) of participants said they improved with performance, and 25% (4) of participants

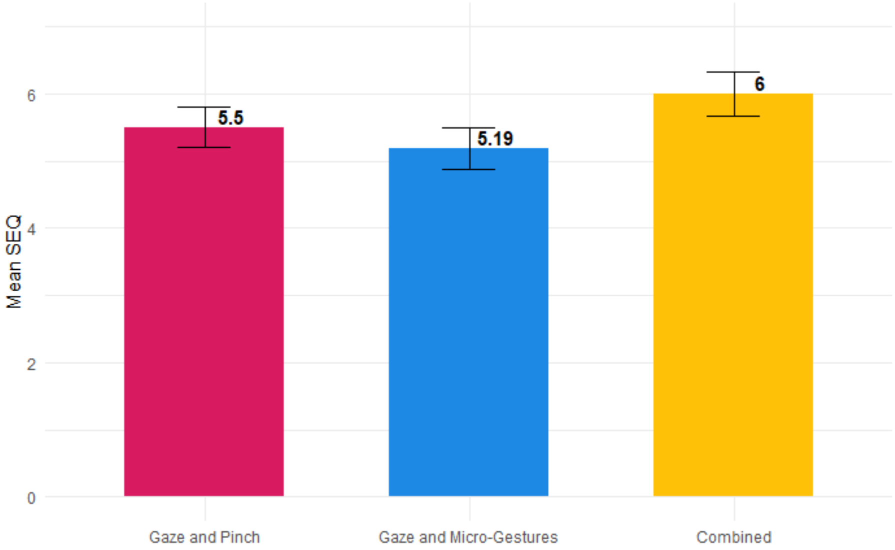


Figure 6.10: Average SEQ responses for the Spreadsheeting task.

believed they would improve with practice. For Combined, 56.25% (9) of participants said they improved with performance, and 25% (4) of participants believed they would improve with practice.

Gesture tracking was a negative theme in responses from 87.5% (14) of participants. This theme was mentioned negatively by 50% (8) of participants for Gaze+Pinch, 68.75% (11) for Gaze+Microgestures, and 68.75% (11) for Combined.

81.25% (13) of participants mentioned that they liked the option to use different gestures in the Combined condition.

75% (12) of participants stated that in the combined condition they combined interactions from both Gaze+Pinch and Gaze+Microgestures.

68.75% (11) of participants commented positively on the speed of a technique (25% (4) for Gaze+Pinch, 50% (8) for Gaze+Microgestures, and 18.75% (3) for Combined).

56.25% (9) of participants noted that the effectiveness of a technique would be better for some tasks than others (43.75% (7) for Gaze+Pinch, 31.25% (5) for Gaze+Microgestures, 25% (4) for Combined).

56.25% (9) of participants noted that they would be uncomfortable using the technique in public settings (43.75% (7) for Gaze+Pinch, 25% (4) for Gaze+Microgestures, 6.25% (1) for Combined).

56.25% (9) of participants commented that techniques were physically demanding (43.75% (7) for Gaze+Pinch, 31.25% (5) for Gaze+Microgestures, 12.5% (2) for Combined), while 62.5% (10) of participants commented that techniques were not physically demanding (18.75% (3) for Gaze+Pinch, 43.75% (7) for Gaze+Microgestures, 37.5% (6) for Combined).

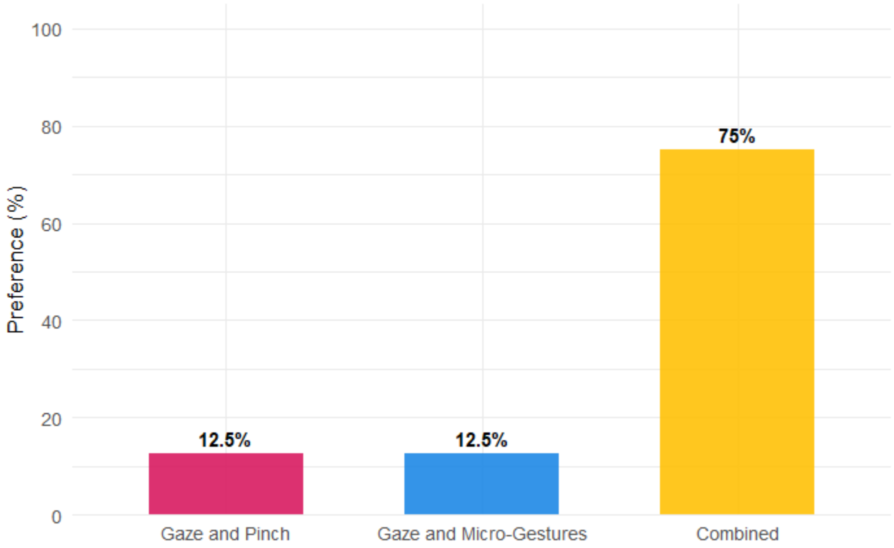


Figure 6.11: Distribution of user preferences after having experienced all three conditions in the Spreadsheeting task.

56.25% (9) of participants commented that they liked the slider along the middle finger, while 37.5% (6) of participants commented that they didn't.

50% (8) of participants commented on the naturalness of techniques (37.5% (6) for Gaze +Pinch, 18.75% (3) for Gaze+Microgestures, and 31.25 (5) for Combined), and 25% (4) commented that techniques were not natural (6.25% (1) for Gaze+Pinch, 12.5% (2) for Gaze +Microgestures, and 6.25% (1) for Combined).

43.75% (7) of participants stated that they mixed up fingers when using Gaze+ Microgestures.

43.75% (7) of participants stated that the gestures were hard to remember (12.5% (2) for Gaze+Pinch, 25% (4) for Gaze+Microgestures, 12.5% (2) for Combined), while 25% (4) of participants noted that gestures were easy to remember (12.5% (2) for Gaze+Pinch, 18.75% (3) for Gaze+Microgestures, and 0% for Combined).

43.75% (7) of participants stated that the task felt easier using a technique (12.5% (2) for Gaze+Pinch, 18.75% (3) for Gaze+Microgestures, 31.25% (5) for Combined). 18.75% (3) of participants mentioned that the task felt harder with Gaze+Pinch.

37.5% (6) of participants mentioned that a technique felt similar to existing devices (6.25% (1) for Gaze+Pinch, 31.25% (5) for Gaze+Microgestures, and 12.5% (2) for Combined).

25% (4) of participants mentioned accidentally triggering long-presses with Gaze+Pinch.

18.75% (3) of participants mentioned that they would rather use a phone, or laptop than any of the techniques.

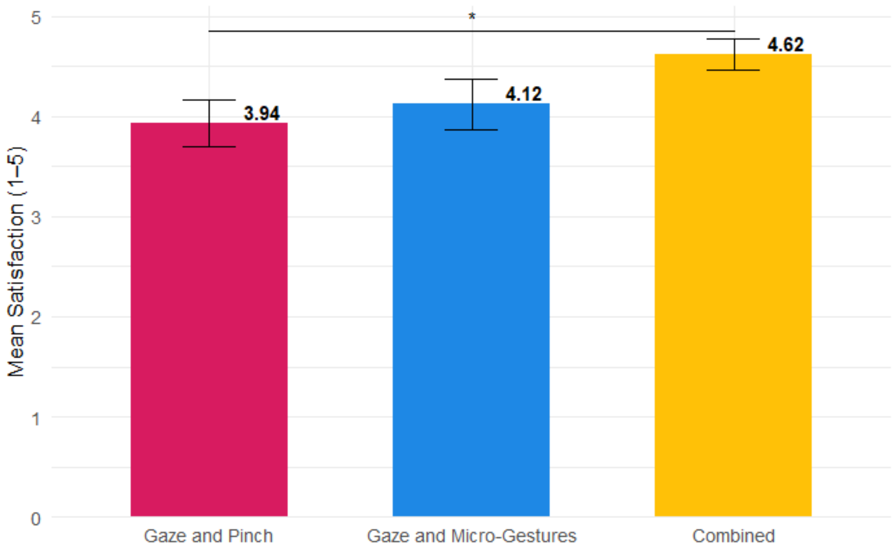


Figure 6.12: Average Satisfaction question responses for the Spreadsheeting task.

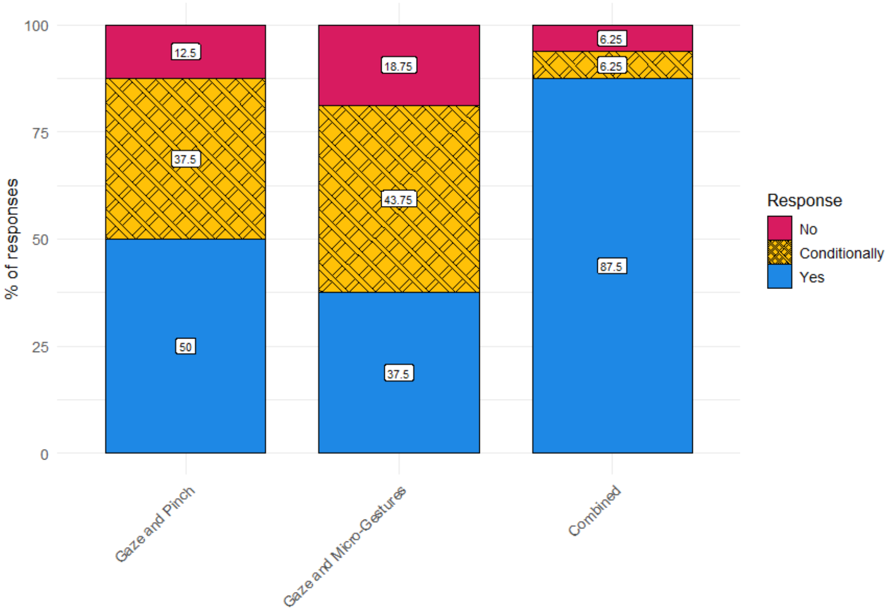


Figure 6.13: User responses when asked if they could see themselves using each technique to perform this spreadsheeting task in real life. Responses that indicated their answer was conditional contribute to the hashed portion of each bar.

Chapter 7

Discussion

This project set out to address the limited expressivity of Gaze+Pinch by extension to Gaze +Microgestures, testing two desktop-inspired variants of it (mouse and function keys) against Gaze+Pinch and a Combined condition, where both Gaze+Pinch and Gaze+Microgesture were enabled. Across both experimental tasks (travel planning and spreadsheeting), Combined was the fastest and most preferred (75% overall preference), with users actively mixing gestures from Gaze+Pinch and Gaze+Microgestures. Combined outperformed Gaze +Microgestures and Gaze+Pinch in completion time, ease, and satisfaction, while Gaze +Microgestures reduced overall hand movement relative to both baselines, but was more sensitive to tracking and memory demands, which contributed to higher error rates in the spreadsheeting task. Practice effects were stronger for Gaze+Microgestures and Combined than for Gaze+Pinch, indicating steeper learning curves and further potential for performance improvement. Measures of eye rotation, workload, and attention to the hand showed no reliable differences across techniques. Taken together, these findings support a design stance that retains Gaze+Pinch as the default input, while enabling optional microgestures to support user agency and contextual, task-specific functionality.

Across both tasks, participants expressed a strong preference (68.75%) towards Gaze +Pinch over Gaze+Microgestures. This can be attributed to several factors. Although no significant difference in cognitive load was found, Gaze+Microgestures imposes higher memory demands: More users reported that the Gaze+Microgestures were hard to remember when using Gaze+Microgestures than Gaze+Pinch (Gaze+Microgestures 25% vs. Gaze+ Pinch 12.5%), and 43.75% of participants reported some finger confusion when using Gaze +Microgestures (Manser-Smith et al., 2018); by contrast, more participants described Gaze +Pinch as natural (Gaze+Pinch 37.5% vs. Gaze+Microgestures 18.75%). Additionally, tracking quality had a greater effect on Gaze+Microgestures than Gaze+Pinch (Gaze+ Microgestures 60.8% vs. 50% Gaze+Pinch). This in part explains the high number of errors in the spreadsheeting task for Gaze+Microgestures, as observation of participants found that the rapid and repetitive execution of gestures caused users to issue subtler gestures than in the travel planning task, leading to more false activations. These false activations registered as erroneous input, and would play a negative audio cue to the user.

Despite this, when afforded choice in the Combined condition, participants overwhelmingly preferred Combined to Gaze+Pinch or Gaze+Microgestures (75%), and engaged with the full range of gestures in the extended microgesture set (copy/paste via shortcuts in 55%/56% of cases; new-tab via middle-click in 62.5% of cases; slider-based scrolling in 42.19% of cases). Qualitative feedback reinforces this pattern: in the Combined condition, 75% of participants reported switching between Gaze+Pinch and Gaze+Microgestures during the task and 81.25% valued the freedom to choose which technique to use. Participants explicitly described mixing modalities (e.g., P17: " I can mix and match the different parts that work best for me."; P23: "It's the nice in-between I wanted.") and highlighted agency (P11: "You get to choose rather than being given a technique.").

These results mark the importance of user agency. In conventional mouse and keyboard interaction, often, users can achieve the same outcome a number of different ways, for example, by right-clicking, selecting a UI element, issuing a keyboard shortcut with the CTRL key, or issuing a shortcut with the ALT key. Results show that users appreciated how the Combined condition allowed them to perform the task the way they wanted to. In some cases, this encouraged users to explore the options available to them (P20: "allowing you to choose what's best for you allows you to get to grips with what's comfortable"), while when users were forced into an interaction, the negative feedback when making mistakes impacted confidence and satisfaction (P12: "...it's hard because there's a lot of errors and buzzes.").

An important theme across conditions was the role of practice, mentioned by 87.5% of participants. More participants reported improvement during the task when using Gaze+ Microgestures and Combined than when using Gaze+Pinch (Gaze+Pinch 37.5% vs. Gaze +Microgestures 62.5% vs. Combined 56.25%). This pattern suggests that while additional microgestures introduce a steeper learning curve, they also offer greater scope for learning and mastery over time. Participant comments reinforced this trajectory, with several noting that gestures that were confusing at first became easier once practised (P14: "Initially it was difficult, but eventually [I] got the hang of it"). On the other hand, Gaze+Pinch, was easier to adopt initially, and appeared to plateau more quickly: fewer participants reported improvement during the task, and fewer anticipated further improvement (Gaze+Pinch 12.5% vs. Gaze+Microgestures 25% vs. Combined 25%). These findings indicate that Gaze+ Microgestures and Combined, despite their greater initial complexity, may support higher long-term performance than captured in this study, provided users are given sufficient time to develop fluency through repeated exposure.

The scrolling gesture of Gaze+Microgestures also influenced the preference results. Use of the scroller resulted in significantly reduced hand movement, and explains why comments on physical demand were less prevalent for Gaze+Microgestures and Combined than for Gaze +Pinch (P13: "The up and down is demanding on your arm."). This reduction in hand movement also explains why more participants commented on social discomfort for Gaze+

Pinch than for Gaze+Microgestures and Combined (P21: ”I’m more likely to use this on buses or public spaces because you’re not waving your hand around like a madman.”). While the scrolling gesture appeared to reduce effort and was more socially acceptable for many participants, the gesture was polarising overall, with 56.25% of participants offering positive comments towards the gesture, and 37.25% offering negative comments. Comments show users felt strongly for or against the gesture (P24: ”Scrolling is the biggest part, it increases the navigability”, P8: ”The scrolling part is the most troublesome part”).

To support contrasting user behaviours and user agency, a Gaze+Microgesture system could expose user-configurable gesture sets as the OS or application level. Results show that XR systems should retain Gaze+Pinch as the default input for selection (100% of users chose to keep the left-click at the index fingertip for the travel planning task, and 85.5%, for the spreadsheet task), however allowing a degree of personalisation supports agency and custom fit. This could be implemented similarly to current desktop software for mice (e.g., Logitech G Hub (Logitech, 2025)), where users can tailor button assignments and functions to their workflow and applications.

43.75% of participants noted that the effectiveness of Gaze+Pinch would vary depending on the task, which was more than for Gaze+Microgestures (31.25%) or Combined (25%). This was supported in other responses: more participants expressed hesitancy to use Gaze +Pinch in public settings (Gaze+Pinch 43.75% vs. Gaze+Microgestures 25% vs. Combined 6.25%), and more participants commented on the physical demand of Gaze+Pinch than Gaze+Microgestures or Combined (Gaze+Pinch 43.75% vs. Gaze+Microgestures 31.25% vs. Combined 12.5%). Taken together, these findings suggest that users regarded Gaze+ Microgestures as more versatile and adaptable across different contexts, while Gaze+Pinch was limited by concerns around comfort, social acceptability, and task fit. This points to the potential of microgestures in combination with Gaze+Pinch not only to broaden the range of tasks that can be supported effectively, but also to reduce barriers to adoption by offering a less conspicuous and physically demanding form of input.

For example, results indicate that Gaze+Microgestures offers particular promise to enhance navigation or travel applications (e.g., GoogleEarthVR (Google, 2025)), allowing users to move through space, zoom in or out, or drop pins, without disrupting the immersion or flow with menu summoning or user interface. This points to a lightweight alternative to controller-based navigation, supporting tasks such as exploring virtual maps, inspecting large datasets, or touring virtual environments. The flexibility of this approach resonates with the strong performance of the combined condition in the study, where participants appreciated being able to adapt their strategy to the task and select whichever input felt most natural.

Results also suggest Gaze+Microgestures’ suitability for interaction on the go. In mobile AR or public settings, users often need to perform quick actions such as checking schedules, replying to short messages, or pulling up navigation cues without wanting to take out a controller or use a full mid-air keyboard. Gaze+Microgestures could provide discreet and efficient input in such contexts, reducing the physical and social overhead of larger movements. This direction aligns with participant feedback, highlighting the comfort of fallback to Gaze+Pinch when tasks were simple, while still offering more expressive input options when needed (P12: ”If tracking was perfect I would still combine...”). As XR devices and smart glasses increasingly extend beyond stationary use, the ability to interact seamlessly while walking, commuting, or waiting could be a critical factor for adoption.

Together, these applications show how Gaze+Microgestures can extend the simplicity of Gaze+Pinch into a richer input language beyond the tested gesture set, supporting both everyday productivity and immersive interaction in XR.

Overall, these findings demonstrate that Gaze+Microgestures is a viable concept, capable of enhancing controller-free interaction without undermining the simplicity of Gaze+Pinch. Participants were able to adopt the gestures quickly, and their performance suggests that the concept can further scale with advances in sensing and recognition. Rather than replacing existing techniques, Gaze+Microgestures offers a flexible layer of additional functionality that can be tailored to different contexts. At the same time, results surfaced a number of open challenges, spanning detection reliability, comfort, discoverability, and memorability, that must be addressed to fully unlock the potential of Gaze+Microgestures in XR.

7.1 Challenges when Designing For Gaze + Microgestures

Designing with and for Gaze+Microgestures presents a distinct set of challenges that span technical reliability, ergonomic comfort, and cognitive support for learning.

With respect to gaze tracking, participants occasionally encountered noise and accuracy issues, leading to early or late triggers that disrupted flow (P19: ”It was just slightly off with the gaze calibration ... Didn’t want to look over towards the corners”). Future work should investigate more robust gaze filtering and event-triggering mechanisms, for example, by correcting the measured eye gaze from the eye tracker with gaze predicted by machine learning (Z. Hu et al., 2025). For microgestures, detection reliability also proved challenging, discouraging consistent use in some cases (P23: ”I would much prefer G+MGs over G+P if it was smoother tracked”). Advances in sensing hardware and recognition algorithms could make gesture detection more reliable and less dependent on specialised gloves. Promising solutions could be EMG-wristbands, which stand to detect discrete and continuous thumb to finger microgestures across fingers, without facing occlusion issues and without the need for conspicuous and cumbersome hand-tracking gloves (Kaifosh and Reardon, 2025). Comfort and flexibility varied across gestures, with certain positions described as fatiguing or awkward (P13: ”The base of the finger I didn’t use at all, only as it’s a bit awkward and discomforting...”), despite prior work suggesting the selected set as suitable (Perella-Holfeld et al., 2023). Other gestures were perceived as prone to accidental activation, raising the Midas touch problem (P12: ”If you make an error in real life it’s more catastrophic. [It was]

Intuitive, but when I scroll I mix up fingers”). Addressing these issues may involve refining gesture sets, exploring other mappings (e.g., the top of the index finger (Kin et al., 2024)), and designing adaptive thresholds that minimise false positives. Systems could implement custom gesture sets that allow users to place gestures in positions comfortable for them, ensuring comfort across the whole user group. Fatigue was a recurring concern (P10: ”I can feel a strain in my wrist and arm so it would be healthy to vary it[the gestures] throughout the task”), since microgestures recruit small hand muscles, making some positions unsustainable over time without sustained use. Long-term studies with repeat sessions could help uncover how comfort, endurance, and gesture strategies evolve with prolonged use and practice.

Together, these challenges motivate future work on technical improvements, ergonomic refinements, and adaptive mapping strategies to fully unlock the potential of Gaze+ Microgestures in XR interaction.

Findings also surfaced conceptual challenges inherent to the approach: Discoverability, learnability, memorability, and mode confusion remain central hurdles. Icons alone may not be descriptive enough to help users distinguish or remember gestures, with 25% of users mentioning that gestures were hard to remember with Gaze+Microgestures. Trial-and-error exploration, though useful for discovery, may not be appropriate when operations are critical. In such cases, richer feedback strategies could support users more effectively, for example, by presenting gesture cues on the target or near the gaze point, triggered automatically, after a dwell, or with a dedicated gesture. Another challenge is user apathy: because pinch remained a reliable fallback, some participants questioned whether learning additional gestures was worth the effort (P6: ”Wasn’t thinking about the finger at all so [I] went back to using one finger”, P16: ”I went for the simplest option.”). This mirrors prior observations of underused input extensions such as Apple’s Touch Bar, where the benefits were not compelling enough to overcome the learning cost (Cranz, 2023) and generally the legacy bias (Morris et al., 2014). Taken together, these challenges suggest that while microgestures can meaningfully enrich controller-free XR interaction, their adoption depends on careful support for learning and a clear demonstration of value in everyday use.

While this study revealed both technical and conceptual challenges for Gaze+Microgestures, the benefits observed — reduced effort, improved satisfaction, and quick adoption — point to promising opportunities for applying Gaze+Microgestures beyond controlled study tasks.

7.2 Limitations

While this study provides insights into the potential of Gaze+Microgestures, several limitations should be considered when interpreting the results.

First, the study was conducted with a glove for microgesture detection. Although this setup ensured satisfactory recognition for experimental purposes, it also introduced friction in gesture performance, resulting in significantly high errors, particularly in the Gaze+Microgesture condition in the spreadsheeting task, where microgestures where issued repeatedly. Future sensing solutions, for example, EMG-based (Kaifosh and Reardon, 2025) or camera-based systems (Kin et al., 2024), are capable of detecting discrete and continuous thumb to finger interactions, and would further support interactions such as the ones used in this work.

Second, the study focused on two desktop-inspired mappings (mouse and function keys) and two task domains (travel planning and spreadsheets). These choices allowed for structured evaluation with meaningful performance measures, but necessarily constrain the generalisability of the results. Exploring mappings outside of selections, such as those proposed for 3D design, gaze locomotion, or mobile scenarios, may reveal different strengths and weaknesses of Gaze+Microgestures.

Thirdly, long-term memorability or fatigue was not investigated. It remains unclear how gesture use evolves with extended practice or performs in everyday contexts where users may combine Gaze+Microgestures with other modalities.

Fourth, the evaluation took place in a controlled lab environment. While this enabled systematic comparison across techniques, it limits ecological validity: factors such as mobility and social acceptance. Future work should explore Gaze+Microgestures in the wild to assess its robustness and adoption in everyday XR use.

Finally, due to the nature of the task setup, Combined was always presented last. This introduces potential order and carry-over effects, but the sequencing was deliberate: participants needed to experience Gaze+Pinch and Gaze+Microgestures first to make meaningful use of the combined option, whose value depends on familiarity with the individual techniques. This resulted in the reliance on qualitative feedback to further justify results.

Chapter 8

Guide to Designing for Gaze+ Microgestures

This chapter uses the project's empirical findings to produce a design guide for extending controller-less XR input with Gaze+Microgestures. This chapter translates the observes trade-offs into guiding principles on when to deploy Gaze+Microgestures, how to define the gesture set and mapping strategies, and how to support users with effective feedback.

8.1 Understanding the System's Context

The first step in designing with Gaze+Microgestures is to consider the broader system context. Results show that while participants initially found Gaze+Microgestures harder to remember and more prone to finger confusion than Gaze+Pinch, many reported that performance improved with practice and that they expected further improvement over time. This suggests that Gaze+Microgestures are most effective in scenarios that support repeated use, such as productivity tools, collaborative workspaces, or personal XR devices, where users have the opportunity to practise and memorise gestures. At an interaction-level, Gaze +Microgestures can benefit interactions that require: a sense of immersion, by reducing the UI navigation required by moving controls to the hand; frequent menu summoning, or deep menu navigation by providing direct shortcuts; repetitive gaze pointing, when context and controls are located spatially apart from each other, by bringing the controls to the hand, allowing the gaze to remain on the content. In these contexts, microgestures can best deliver on their strengths of efficiency, subtlety, and reduced physical effort.

On the other hand, in public, demonstrative settings such as museum displays, participants are unlikely to receive extensive training or repeated exposure. In these scenarios, mid-air gestures may be more suitable because they are more intuitive and easier to learn in a short interaction window (Havlucu et al., 2017). However, mid-air gestures are more conspicuous, and can therefore face larger social barriers in environments such as offices or public transport, where larger, attention-drawing gestures could hinder the system's adoption.

In addition to the application domain, the target user group must also be taken into account. A visitor to a museum, a professional XR developer, and a VR gamer each bring different expectations, levels of technological experience, and tolerance for errors. Microgestures can be particularly beneficial for experienced users who are willing to invest time in learning shortcuts and value reduced effort in the long term. For novice or transient users, however, designers may wish to prioritise gestures that emphasise simplicity and discoverability over efficiency. Recognising these differences ensures that microgestures are designed according to the capabilities of your users.

8.2 Defining a Gesture-Set

Once there is an understanding of the application domain and the users-base, we can design our microgestures for Gaze+Microgesture interaction.

The primary design constraint is the precision and reliability of gesture detection. Given that constraint, the gesture set can be designed with the following considerations:

Firstly, one or multiple fingers can be used for thumb-to-finger input. Using a single finger reduces finger confusion (Manser-Smith et al., 2018) and occlusion (for when optical tracking is used), but limits the input space. The overall number of gestures should be balanced; too many gestures creates learnability and memorability challenges, while too few hampers expressivity and functionality.

Secondly, non-overlapping gestures should be used if on-hand visual feedback is employed. Overlapping gestures should find alternative onboarding and teaching mechanisms for gesture discovery and recall.

Thirdly, thumb-to-finger gestures should be performed at locations and in ways that are supported by proprioception, and can consistently be performed eyes-free. The segments of the finger act as proprioceptive landmarks for microgestures, and works have shown that the segments can be further divided while still supporting the eyes-free affordance (Tsai et al., 2017; Whitmire et al., 2017). Other gestures are possible, including sliding gestures along or about the finger, symbolic shapes drawn over the fingers, or finger-to-thumb pinches (Soliman et al., 2018).

Finally, reachability should be factored, as not all parts of the finger are equally comfortable for the thumb to reach (S. A. Faleel et al., 2021; Kuo et al., 2009). Critical and high-risk commands can be placed in harder to reach locations, such as the base of fingers, or the backs of fingernails, to reduce accidental activations (Wambecke et al., 2021).

8.3 Defining a Mapping Strategy

Once a gesture set has been defined, we can define how functionality is mapped to it. The simplest approach is to assign default, system-wide functionality to each gesture in the set. This approach supports learnability by providing consistent behaviour across contexts. Task-specific functionality can be introduced through dynamically assigned microgestures. These stand to improve performance and user satisfaction, but require a mechanism to switch the functionality of gestures between tasks. These mechanisms can follow desktop behaviour, binding gestures to the application in focus. This can be defined as the last application selected, or the last application looked at. The former is more reliable and gives more control to users, while the latter enables faster interactions by leveraging the rapid pointing of the eyes.

The individual mappings for each task must also be clear to users. Wherever possible, mappings should use layouts consistent with mobile and other MR UI standards (Perella-Holfeld et al., 2023). Be aware that different user-groups will have different familiarity to different UI standards, and mappings should cater to the target audience of the system. Additionally, the comfort of gestures should be considered; the most frequently used thumb-to-finger interactions should be in comfortable positions on the hand. Functionality that is non-reversible or critical should be mapped to less accessible gestures to avoid critical slips and mistakes. Whenever possible, established metaphors should be employed to explain mapping strategies to users, especially when multiple contexts are included.

8.4 Providing Effective Feedback

With a gesture set and mapping strategy defined, users must discover, learn and remember functionality mappings. Depending on the number of contexts and mappings, this can be a significant challenge. To combat this challenge, feedback can be employed.

Assuming users have learnt the gesture set, discovery of functionality can be done through trial and error. This may be appropriate for low-stakes applications, however for critical functions, this is not feasible. Instead, feedback can be displayed through icons on the hand, although this requires distinct, non-overlapping feedback. Feedback can be displayed in the world near gaze targets, or near the gaze point, manifesting as a representation of the functionality on the hand. This approach, along with audio feedback, can be disruptive for more immersive experiences, however. Alternatively, feedback can be triggered automatically, or after a gaze-dwell, or with a dedicated gesture.

Good feedback will support the novice to expert transition by signifying the functionality available to them. Additionally, it can empower users to experiment by reducing the mental cost associated with forgetting a gesture or mapping.

Applied together, these principles enable Gaze+Microgestures to apply across a broad array of contexts, ranging from text entry, to 3D manipulation and gaze locomotion.

With future improvements to gesture detection and further optimisation of gesture sets, This chapter can serve as practical guidance for translating Gaze+Microgestures from lab prototypes into deployable, real-world systems.

Chapter 9

Conclusions

9.1 Review of Objectives

This dissertation set out to extend the expressivity of Gaze+Pinch through a compact set of microgestures (Gaze+Microgestures) that dynamically map to applications in focus. The work addressed the three stated objectives as follows.

9.1.1 Objective 1: Design and implement a compact Gaze+Microgesture vocabulary

A small, learnable set of thumb-to-finger microgestures was specified and implemented to augment Gaze+Pinch while retaining pinch-to-select as the default. The vocabulary was designed with comfort, proprioception, and capability for visual feedback, in mind. Gaze+ Pinch was preserved at the index fingertip, increasing the expressivity through the additional gestures, but without undermining the simplicity of the original technique, as indicated by the results of the user study, where the condition that supported both Gaze+Pinch and Gaze +Microgesture interactions was overwhelmingly preferred.

9.1.2 Objective 2: Define mapping strategies and evaluate them against Gaze+Pinch

Two mapping strategies were introduced to map Gaze+Microgestures to application functionality and were evaluated against Gaze+Pinch across two representative tasks. The evaluation demonstrated the feasibility of Gaze+Microgestures, with participants achieving the greatest speed, preference, and satisfaction, when allowed to choose between Gaze+Pinch and Gaze+Microgesture interactions. Results also underscored the importance of user agency and flexibility; opinions on individual gestures and mappings were highly polarising, leading to reduced satisfaction when restricted to either Gaze+Pinch or Gaze+Microgestures. These results substantiate the design stance that Gaze+Pinch should remain the default, with Gaze +Microgestures exposed as complementary shortcuts that increase user agency and task-specific efficiency, and points towards a future where users can customise their own gesture sets and application mappings to suit their individual ergonomic and interaction preferences.

9.1.3 Objective 3: Implement a commodity-hardware gesture detection system

Using the Manus Quantum Metaglove (*Motion Capture Gloves — Quantum Metagloves by MANUS* 2025) and the Vive Pro Eye headset (Vive, 2025a), a gesture recognition system was implemented using unity colliders, and thumb-to-finger distance. This system used a custom calibration sequence to mitigate for disparities between the tracked hand model, and the real life hand. The system proved sufficiently robust to run the study and capture performance, workload, and preference measures. The implementation provided the necessary foundation to evaluate Gaze+Microgestures and the combined condition in practice. However, gesture tracking suitable for real-world use still remains challenging. Any number of errors can discourage users, especially when gesture detection fails during the training and learning process, and gesture tracking that is both robust, discrete, and precise enough to capture thumb-to-finger-segment gestures, is still an open problem.

9.1.4 Summary

Collectively, these outcomes show that augmenting Gaze+Pinch with a compact set of microgestures can increase expressivity without undermining its core simplicity. Application-specific microgestures improved satisfaction, ease of use, and performance. The findings motivate flexible, user-configurable gesture sets in XR that allow users to mix techniques, adapt to context, and personalise their workflows.

9.2 Conclusion

This dissertation has explored the potential of extending Gaze+Pinch to Gaze+Microgestures by introducing a compact set of thumb-to-finger microgestures (Gaze+Microgestures)that dynamically map to application-specific functionality, depending on the target at the gaze point. By using additional inputs as compliment to Gaze+Pinch, this work has sought to address the limited expressivity of Gaze+Pinch without undermining its simplicity across contexts.

Across two tasks, results revealed that, when given the choice, participants chose to engage with the additional microgestures, and found it most satisfying, easiest to use, and were fastest when able to choose between Gaze+Pinch and Gaze+Microgesture interactions.

Similarly, opinions on individual gestures were highly polarising. These findings underscore the importance of user agency and flexibility, especially for gaze and microgesture interaction, where imperceptible differences in ergonomics and personality have significant impacts on user experience.

The significance of this work lies in demonstrating how microgestures can enrich gaze-based input without undermining simplicity. Instead of a one-size-fits-all mapping, users benefit from the flexibility to mix techniques, adapt to context, and choose how they engage with a gesture set. This points to a future where XR systems expose configurable microgestures, akin to how desktop software lets users tailor mouse buttons or keyboard shortcuts. And with the development of advanced, unencumbered microgesture detection techniques (such as EMG wristbands (Kaifosh and Reardon, 2025)), the significance of discrete but expressive and powerful inputs becomes increasingly relevant, especially with participants citing the social perception of interactions as a barrier to adoption for Gaze+ Pinch.

In short, the key takeaways from this project are: (1) Extending Gaze+Pinch with additional thumb to finger microgestures offers measurable benefits for performance, satisfaction, ease of use, and social acceptance, but only when able to choose how they engage with the system. (2) Individual differences in ergonomics and interaction preference play a significant role in gaze and microgesture interaction. (3) Supporting these differences through flexibility and customisation is essential for scaling controller-free input in everyday XR use.

The combination of gaze and microgestures opens a rich space of opportunities for other XR applications. Examples like text-entry and locomotion illustrate that Gaze+ Microgestures is not limited to this narrow set of shortcuts but offers a general design principle for extending controller-free input in XR.

References

Ahlström, David et al. (2014). "Are You Comfortable Doing That? Acceptance Studies of around-Device Gestures in and for Public Settings". In: *Proceedings of the 16th International Conference on Human-Computer Interaction with Mobile Devices & Services* (Toronto, ON, Canada). MobileHCI '14. New York, NY, USA: Association for Computing Machinery, pp. 193–202. ISBN: 978-1-4503-3004-6. DOI: 10.1145/2628363.2628381. URL: https://doi.org/10.1145/2628363.2628381.

Apple (2025). *Use Gestures with Apple Vision Pro*. URL: https://support.apple.com/en-us/117741 (visited on 08/02/2025).

Bailly, Gilles et al. (2023). "Computational Model of the Transition from Novice to Expert Interaction Techniques". In: *ACM Trans. Comput.-Hum. Interact.* 30.5, 66:1–66:33. ISSN: 1073-0516. DOI: 10.1145/3505557. (Visited on 09/12/2025).

Boldu, Roger et al. (2018). "Thumb-In-Motion: Evaluating Thumb-to-Ring Microgestures for Athletic Activity". In: *Proceedings of the 2018 ACM Symposium on Spatial User Interaction* (Berlin, Germany). SUI '18. New York, NY, USA: Association for Computing Machinery, pp. 150–157. ISBN: 978-1-4503-5708-1. DOI: 10.1145/3267782.3267796. URL: https://doi.org/10.1145/3267782.3267796.

Byers, James C and Susan G Hill (1989). "Traditional and raw task load index (TLX) correlations: Are paired comparisons necessary". In: pp. 481, 485.

Caillet, Adrien Chaffangeon et al. (2025). "Microgesture + Grasp: A Journey from Human Capabilities to Interaction with Microgestures". In: *International Journal of Human-Computer Studies* 195, p. 103398. ISSN: 1071-5819. DOI: 10.1016/j.ijhcs.2024.103398. URL: https://www.sciencedirect.com/science/article/pii/S1071581924001812.

Card, Stuart K. et al. (1980). "The keystroke-level model for user performance time with interactive systems". In: *Commun. ACM* 23.7, pp. 396–410. ISSN: 0001-0782. DOI: 10.1145/358886.358895. URL: https://doi.org/10.1145/358886.358895.

Casiez, Géry et al. (2012). "1 € Filter: A Simple Speed-Based Low-Pass Filter for Noisy Input in Interactive Systems". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '12. New York, NY, USA: Association for Computing Machinery, pp. 2527–2530. ISBN: 978-1-4503-1015-4. DOI: 10.1145/2207676.2208639. (Visited on 09/11/2025).

Chaffangeon Caillet, Adrien et al. (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)*, pp. 800–809. DOI: 10.1109/ISMAR59233.2023.00095.

Chan, Edwin et al. (2016). "User Elicitation on Single-hand Microgestures". In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI '16. San Jose, California, USA: Association for Computing Machinery, pp. 3403–3414. ISBN: 9781450333627. DOI: 10.1145/2858036.2858589. URL: https://doi.org/10.1145/2858036.2858589.

Chan, Liwei et al. (2013). "FingerPad: Private and Subtle Interaction Using Fingertips". In: *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (St. Andrews, Scotland, United Kingdom). UIST '13. New York, NY, USA: Association for Computing Machinery, pp. 255–260. ISBN: 978-1-4503-2268-3. DOI: 10.1145/2501988.2502016. URL: https://doi.org/10.1145/2501988.2502016.

Chan, Liwei et al. (2015). "CyclopsRing: Enabling Whole-Hand and Context-Aware Interactions Through a Fisheye Ring". In: *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*. UIST '15. Charlotte, NC, USA: Association for Computing Machinery, pp. 549–556. ISBN: 9781450337793. DOI: 10.1145/2807442.2807450. URL: https://doi.org/10.1145/2807442.2807450.

Chatterjee, Ishan et al. (2015). "Gaze+Gesture: Expressive, Precise and Targeted Free-Space Interactions". In: *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*. ICMI '15. New York, NY, USA: Association for Computing Machinery, pp. 131–138. ISBN: 978-1-4503-3912-4. DOI: 10.1145/2818346.2820752. URL: https://dl.acm.org/doi/10.1145/2818346.2820752 (visited on 09/07/2025).

Chen, Taizhou et al. (2023). "EFRing: Enabling Thumb-to-Index-Finger Microgesture Interaction through Electric Field Sensing Using Single Smart Ring". In: *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6.4. DOI: 10.1145/3569478. URL: https://doi.org/10.1145/3569478.

Chen, Xin et al. (2023). "Compass+Ring: A Multimodal Menu to Improve Interaction Performance and Comfortability in One-handed Scenarios". In: *2023 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 473–482. DOI: 10.1109/ISMAR59233.2023.00062.

Cranz, Alex (2023). *Goodbye, Touch Bar, You Held Incredible Promise.* https://www.theverge.com/2023/10/31/23938841/apple-macbook-pro-touch-bar-discontinued-proof-of-concept. (Visited on 09/12/2025).

Danyluk, Kurtis et al. (2024). "Understanding Gesture and Microgesture Inputs for Augmented Reality Maps". In: *Proceedings of the 2024 ACM Designing Interactive Systems Conference*. DIS '24. Copenhagen, Denmark: Association for Computing Machinery, pp. 409–423. ISBN: 9798400705830. DOI: 10.1145/3643834.3661630. URL: https://doi.org/10.1145/3643834.3661630.

Darling, Warren G. et al. (2024). "Proprioceptive Acuity for Landmarks on the Hand and Digits". In: *Experimental Brain Research* 242.2, pp. 491–503. ISSN: 0014-4819, 1432-1106. DOI: 10.1007/s00221-023-06761-z. (Visited on 08/15/2025).

Delamare, William et al. (2022). "MultiFingerBubble: A 3D Bubble Cursor Variation for Dense Environments". In: *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems*. CHI EA '22. New Orleans, LA, USA: Association for Computing Machinery. ISBN: 9781450391566. DOI: 10.1145/3491101.3519692. URL: https://doi.org/10.1145/3491101.3519692.

Dewitz, Bastian et al. (2019). *Functional Workspace for One-Handed Tap and Swipe Microgestures*. Bonn: Gesellschaft für Informatik e.V. DOI: 10.18420/muc2019-ws-440.

Dobinson, Rhett et al. (2022). "MicroPress: Detecting Pressure and Hover Distance in Thumb-to-Finger Interactions". In: *Proceedings of the 2022 ACM Symposium on Spatial User Interaction*. SUI '22. Online, CA, USA: Association for Computing Machinery. ISBN: 9781450399487. DOI: 10.1145/3565970.3567698. URL: https://doi.org/10.1145/3565970.3567698.

Faisandaz, Gauthier Robert Jean et al. (2022). "Keep in Touch: Combining Touch Interaction with Thumb-to-Finger µGestures for People with Visual Impairment". In: *Proceedings of the 2022 International Conference on Multimodal Interaction* (Bengaluru, India). ICMI '22. New York, NY, USA: Association for Computing Machinery, pp. 105–116. ISBN: 978-1-4503-9390-4. DOI: 10.1145/3536221.3556589. URL: https://doi.org/10.1145/3536221.3556589.

— (2023). "µGeT: Multimodal Eyes-Free Text Selection Technique Combining Touch Interaction and Microgestures". In: *Proceedings of the 25th International Conference on Multimodal Interaction* (Paris, France). ICMI '23. New York, NY, USA: Association for Computing Machinery, pp. 594–603. ISBN: 979-8-4007-0055-2. DOI: 10.1145/3577190.3614131. URL: https://doi.org/10.1145/3577190.3614131.

Faleel, Shariff A M et al. (2024). "Validating Eyes-free Affordance of On-Finger Hand Proximate User Interfaces in In-situ Scenarios". In: *2024 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 1283–1292. DOI: 10.1109/ISMAR62088.2024.00145.

Faleel, Shariff A. M. et al. (2020). "User gesture elicitation of common smartphone tasks for hand proximate user interfaces". In: *Proceedings of the 11th Augmented Human International Conference*. AH '20. Winnipeg, Manitoba, Canada: Association for Computing Machinery. ISBN: 9781450377287. DOI: 10.1145/3396339.3396363. URL: https://doi.org/10.1145/3396339.3396363.

Faleel, Shariff AM et al. (2021). "HPUI: Hand Proximate User Interfaces for One-Handed Interactions on Head Mounted Displays". In: *IEEE Transactions on Visualization and Computer Graphics* 27.11, pp. 4215–4225. DOI: 10.1109/TVCG.2021.3106493.

Fitzmaurice, George W. and William Buxton (1997). "An Empirical Evaluation of Graspable User Interfaces: Towards Specialized, Space-Multiplexed Input". In: *Proceedings of the*

ACM SIGCHI Conference on Human Factors in Computing Systems. CHI '97. New York, NY, USA: Association for Computing Machinery, pp. 43–50. ISBN: 978-0-89791-802-2. DOI: `10.1145/258549.258578`. URL: `https://dl.acm.org/doi/10.1145/258549.258578` (visited on 09/09/2025).

GadgetHacks (2025). *Why Your Eyes Aren't Enough: Apple's Next Vision Pro Breakthrough*. URL: `https://mixed.reality.news/news/why-your-eyes-arent-enough-apples-next-vision-pro-breakthrough/` (visited on 01/27/2026).

Google (2025). *Geo VR*. URL: `https://www.google.com/intl/en%5C_uk/earth/education/tools/geo-vr/` (visited on 09/12/2025).

Guimbretière, François et al. (2005). "Benefits of Merging Command Selection and Direct Manipulation". In: *ACM Trans. Comput.-Hum. Interact.* 12.3, pp. 460–476. ISSN: 1073-0516. DOI: `10.1145/1096737.1096742`. URL: `https://dl.acm.org/doi/10.1145/1096737.1096742` (visited on 09/09/2025).

Gustafson, Sean G. et al. (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*. CHI '13. New York, NY, USA: Association for Computing Machinery, pp. 889–898. ISBN: 978-1-4503-1899-0. DOI: `10.1145/2470654.2466114`. URL: `https://dl.acm.org/doi/10.1145/2470654.2466114` (visited on 09/09/2025).

Havlucu, Hayati et al. (2017). "It Made More Sense: Comparison of User-Elicited On-skin Touch and Freehand Gesture Sets". In: *Distributed, Ambient and Pervasive Interactions*. Ed. by Norbert Streitz and Panos Markopoulos. Cham: Springer International Publishing, pp. 159–171. ISBN: 978-3-319-58697-7.

Hu, Jinghui et al. (2025). "Seeing and Touching the Air: Unraveling Eye-Hand Coordination in Mid-Air Gesture Typing for Mixed Reality". In: *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. CHI '25. Association for Computing Machinery. ISBN: 9798400713941. DOI: `10.1145/3706598.3713743`. URL: `https://doi.org/10.1145/3706598.3713743`.

Hu, Zhiming et al. (2025). "HOIGaze: Gaze Estimation During Hand-Object Interactions in Extended Reality Exploiting Eye-Hand-Head Coordination". In: *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*. SIGGRAPH Conference Papers '25. Association for Computing Machinery. ISBN: 9798400715402. DOI: `10.1145/3721238.3730692`. URL: `https://doi.org/10.1145/3721238.3730692`.

Huang, Da-Yuan et al. (2016). "DigitSpace: Designing Thumb-to-Fingers Touch Interfaces for One-Handed and Eyes-Free Interactions". In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA). CHI '16. New York, NY, USA: Association for Computing Machinery, pp. 1526–1537. ISBN: 978-1-4503-3362-7. DOI: `10.1145/2858036.2858483`. URL: `https://doi.org/10.1145/2858036.2858483`.

Istance, Howell et al. (2008). “Snap clutch, a moded approach to solving the Midas touch problem”. In: *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications*. ETRA ’08. Savannah, Georgia: Association for Computing Machinery, pp. 221–228. ISBN: 9781595939821. DOI: 10.1145/1344471.1344523. URL: https://doi.org/10.1145/1344471.1344523.

Jeong, Jaejoon et al. (2025). “GazeHand2: A Gaze-Driven Virtual Hand Interface with Improved Gaze Depth Control for Distant Object Interaction”. en. In: *Electronics* 14.13, p. 2530. ISSN: 2079-9292. DOI: 10.3390/electronics14132530. URL: https://www.mdpi.com/2079-9292/14/13/2530 (visited on 09/28/2025).

Jiang, Haiyan et al. (2019). “HiFinger: One-Handed Text Entry Technique for Virtual Environments Based on Touches between Fingers”. In: *Sensors* 19.14. ISSN: 1424-8220. DOI: 10.3390/s19143063. URL: https://www.mdpi.com/1424-8220/19/14/3063.

Johansson, Roland S. et al. (2001). “Eye–Hand Coordination in Object Manipulation”. In: *Journal of Neuroscience* 21.17, pp. 6917–6932. ISSN: 0270-6474. DOI: 10.1523/JNEUROSCI.21-17-06917.2001. eprint: https://www.jneurosci.org/content/21/17/6917.full.pdf. URL: https://www.jneurosci.org/content/21/17/6917.

Kaifosh, Patrick and Thomas R. Reardon (2025). “A Generic Non-Invasive Neuromotor Interface for Human-Computer Interaction”. In: *Nature*, pp. 1–10. ISSN: 1476-4687. DOI: 10.1038/s41586-025-09255-w. URL: https://www.nature.com/articles/s41586-025-09255-w (visited on 09/07/2025).

Kim, Jeong Ho et al. (2014). “Differences in typing forces, muscle activity, comfort, and typing performance among virtual, notebook, and desktop keyboards”. In: *Applied Ergonomics* 45.6, pp. 1406–1413. ISSN: 0003-6870. DOI: https://doi.org/10.1016/j.apergo.2014.04.001. URL: https://www.sciencedirect.com/science/article/pii/S000368701400043X.

Kim, Jina et al. (2025). “T2IRay: Design of Thumb-to-Index based Indirect Pointing for Continuous and Robust AR/VR Input”. In: *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. CHI ’25. Association for Computing Machinery. ISBN: 9798400713941. DOI: 10.1145/3706598.3713442. URL: https://doi.org/10.1145/3706598.3713442.

Kim, Jinwook et al. (2025). “PinchCatcher: Enabling Multi-selection for Gaze+Pinch”. In: *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. CHI ’25. New York, NY, USA: Association for Computing Machinery, pp. 1–16. ISBN: 979-8-4007-1394-1. DOI: 10.1145/3706598.3713530. URL: https://dl.acm.org/doi/10.1145/3706598.3713530 (visited on 09/07/2025).

Kin, Kenrick et al. (2024). “STMG: A Machine Learning Microgesture Recognition System for Supporting Thumb-Based VR/AR Input”. In: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA). CHI ’24. New York, NY, USA: Association for Computing Machinery. ISBN: 979-8-4007-0330-0. DOI: 10.1145/3613904.3642702. URL: https://doi.org/10.1145/3613904.3642702.

Kuo, Li-Chieh et al. (2009). “Functional Workspace for Precision Manipulation between Thumb and Fingers in Normal Hands”. In: *Journal of Electromyography and Kinesiology* 19.5, pp. 829–839. ISSN: 10506411. DOI: 10.1016/j.jelekin.2008.07.008. URL: https://linkinghub.elsevier.com/retrieve/pii/S1050641108001193 (visited on 08/15/2025).

Kytö, Mikko et al. (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*. CHI ’18. Montreal QC, Canada: Association for Computing Machinery, pp. 1–14. ISBN: 9781450356206. DOI: 10.1145/3173574.3173655. URL: https://doi.org/10.1145/3173574.3173655.

Lee, Hock Siang et al. (2024). “Snap, Pursuit and Gain: Virtual Reality Viewport Control by Gaze”. In: *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. CHI ’24. Honolulu, HI, USA: Association for Computing Machinery. ISBN: 9798400703300. DOI: 10.1145/3613904.3642838. URL: https://doi.org/10.1145/3613904.3642838.

Lee, Lik Hang et al. (2019). “Quadmetric Optimized Thumb-to-Finger Interaction for Force Assisted One-Handed Text Entry on Mobile Headsets”. In: *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3.3. DOI: 10.1145/3351252. URL: https://doi.org/10.1145/3351252.

Lee, Lik Hang et al. (2020). “From seen to unseen: Designing keyboard-less interfaces for text entry on the constrained screen real estate of Augmented Reality headsets”. In: *Pervasive and Mobile Computing* 64, p. 101148. ISSN: 1574-1192. DOI: https://doi.org/10.1016/j.pmcj.2020.101148. URL: https://www.sciencedirect.com/science/article/pii/S1574119220300365.

Li, Xiang et al. (2025). *Evaluating the Usability of Microgestures for Text Editing Tasks in Virtual Reality*. Version Number: 2. DOI: 10.48550/ARXIV.2504.04198. URL: https://arxiv.org/abs/2504.04198 (visited on 09/27/2025).

Lim, Hyunchul et al. (2018). “Touch+Finger: Extending Touch-based User Interface Capabilities with ”Idle” Finger Gestures in the Air”. In: *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany). UIST ’18. New York, NY, USA: Association for Computing Machinery, pp. 335–346. ISBN: 978-1-4503-5948-1. DOI: 10.1145/3242587.3242651. URL: https://doi.org/10.1145/3242587.3242651.

Logitech (2025). *G HUB Gaming Software— Logitech G*. URL: https://www.logitechg.com/en-gb/software/ghub (visited on 09/12/2025).

Lu, Lu et al. (2021). “Gaze-Pinch Menu: Performing Multiple Interactions Concurrently in Mixed Reality”. In: *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pp. 536–537. DOI: 10.1109/VRW52623.2021.00150.

Lu, Yu et al. (2024). “HCMG: Human-Capacitance based Micro Gesture for VR/AR”. In: *Companion of the 2024 on ACM International Joint Conference on Pervasive* *and Ubiquitous Computing*. UbiComp '24. Melbourne VIC, Australia: Association for Computing Machinery, pp. 766–770. ISBN: 9798400710582. DOI: 10.1145/3675094.3678386. URL: https://doi.org/10.1145/3675094.3678386.

Lystbæk, Mathias N. et al. (2024). "Hands-on, Hands-off: Gaze-Assisted Bimanual 3D Interaction". In: *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. UIST '24. Pittsburgh, PA, USA: Association for Computing Machinery. ISBN: 9798400706288. DOI: 10.1145/3654777.3676331. URL: https://doi.org/10.1145/3654777.3676331.

Manser-Smith, Kelda et al. (2018). "Tactile Confusions of the Fingers and Toes". In: *Journal of Experimental Psychology: Human Perception and Performance* 44.11, pp. 1727–1738. ISSN: 1939-1277. DOI: 10.1037/xhp0000566.

McIntosh, Jess et al. (2017). "SensIR: Detecting Hand Gestures with a Wearable Bracelet using Infrared Transmission and Reflection". In: *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. UIST '17. Québec City, QC, Canada: Association for Computing Machinery, pp. 593–597. ISBN: 9781450349819. DOI: 10.1145/3126594.3126604. URL: https://doi.org/10.1145/3126594.3126604.

Meta AI glasses — Ray-Ban Meta, Oakley Meta &amp; Display glasses (2025). en. URL: https://www.meta.com/de/en/ai-glasses/ (visited on 09/27/2025).

Microsoft (2025). *Gaze and Commit - Mixed Reality*. URL: https://learn.microsoft.com/en-us/windows/mixed-reality/design/gaze-and-commit (visited on 08/27/2025).

Mikkelsen, Thorbjørn et al. (2025). "Mapping 6DOF Hand Movement to Concurrent 3DOF Controls of Translation and Rotation for Manipulation in Extended Reality". In.

Mine, Mark R. (1995). *Virtual Environment Interaction Techniques*. Tech. rep. USA.

Morris, Meredith Ringel et al. (2014). "Reducing legacy bias in gesture elicitation studies". In: *Interactions* 21.3, pp. 40–45. ISSN: 1072-5520. DOI: 10.1145/2591689. URL: https://doi.org/10.1145/2591689.

Motion Capture Gloves — Quantum Metagloves by MANUS (2025). URL: https://www.manus-meta.com/products/quantum-metagloves (visited on 09/12/2025).

Mutasim, Aunnoy K et al. (2021). "Pinch, Click, or Dwell: Comparing Different Selection Techniques for Eye-Gaze-Based Pointing in Virtual Reality". In: *ACM Symposium on Eye Tracking Research and Applications*. ETRA '21 Short Papers. Virtual Event, Germany: Association for Computing Machinery. ISBN: 9781450383455. DOI: 10.1145/3448018.3457998. URL: https://doi.org/10.1145/3448018.3457998.

Nielsen, Jakob (1994). "Enhancing the Explanatory Power of Usability Heuristics". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '94. New York, NY, USA: Association for Computing Machinery, pp. 152–158. ISBN: 978-0-89791-650-9. DOI: 10.1145/191666.191729. (Visited on 01/27/2026).

Pang, Alex Soojung-Kim (2002). "The making of the mouse". In: *American Heritage of Invention and Technology* 17.3, pp. 48–54.

Patel, Khushman et al. (2025). “H2E: Hand, Head, Eye a Multimodal Cascade of Natural Inputs”. In: *2025 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pp. 84–89. DOI: 10.1109/VRW66409.2025.00026.

Pei, Siyou et al. (2024). “UI Mobility Control in XR: Switching UI Positionings between Static, Dynamic, and Self Entities”. In: *Proceedings of the CHI Conference on Human Factors in Computing Systems*. CHI ’24: CHI Conference on Human Factors in Computing Systems. Honolulu HI USA: ACM, pp. 1–12. ISBN: 979-8-4007-0330-0. DOI: 10.1145/3613904.3642220. URL: https://dl.acm.org/doi/10.1145/3613904.3642220 (visited on 08/02/2025).

Perella-Holfeld, Francisco et al. (2023). “Evaluating design guidelines for hand proximate user interfaces”. In: *Proceedings of the 2023 ACM Designing Interactive Systems Conference*. DIS ’23. Pittsburgh, PA, USA: Association for Computing Machinery, pp. 1159–1173. ISBN: 9781450398930. DOI: 10.1145/3563657.3596117. URL: https://doi.org/10.1145/3563657.3596117.

Pfeuffer, Ken et al. (2014). “Gaze-Touch: Combining Gaze with Multi-Touch for Interaction on the Same Surface”. In: *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*. UIST ’14. New York, NY, USA: Association for Computing Machinery, pp. 509–518. ISBN: 978-1-4503-3069-5. DOI: 10.1145/2642918.2647397. URL: https://dl.acm.org/doi/10.1145/2642918.2647397 (visited on 09/07/2025).

Pfeuffer, Ken et al. (2017). “Gaze + pinch interaction in virtual reality”. In: *Proceedings of the 5th Symposium on Spatial User Interaction*. SUI ’17. Brighton, United Kingdom: Association for Computing Machinery, pp. 99–108. ISBN: 9781450354868. DOI: 10.1145/3131277.3132180. URL: https://doi.org/10.1145/3131277.3132180.

Pfeuffer, Ken et al. (2024). “Design Principles and Challenges for Gaze + Pinch Interaction in XR”. In: *IEEE Computer Graphics and Applications* 44.3, pp. 74–81. DOI: 10.1109/MCG.2024.3382961.

Prätorius, Manuel et al. (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. New York, NY, USA: Association for Computing Machinery, pp. 9–18. ISBN: 978-1-4503-3253-8. DOI: 10.1145/2671015.2671029. URL: https://doi.org/10.1145/2671015.2671029.

Rico, Julie and Stephen Brewster (2010). “Usable Gestures for Mobile Interfaces: Evaluating Social Acceptability”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Atlanta, Georgia, USA). CHI ’10. New York, NY, USA: Association for Computing Machinery, pp. 887–896. ISBN: 978-1-60558-929-9. DOI: 10.1145/1753326.1753458. URL: https://doi.org/10.1145/1753326.1753458.

Sharma, Adwait et al. (2024). “GraspUI: Seamlessly Integrating Object-Centric Gestures within the Seven Phases of Grasping”. In: *Proceedings of the 2024 ACM Designing Interactive Systems Conference* (Copenhagen, Denmark). DIS ’24. New York, NY, USA:

Association for Computing Machinery, pp. 1275–1289. ISBN: 979-8-4007-0583-0. DOI: 10.1145/3643834.3661551. URL: https://doi.org/10.1145/3643834.3661551.

Shi, Rongkai et al. (2024). "Experimental Analysis of Freehand Multi-object Selection Techniques in Virtual Reality Head-Mounted Displays". In: *Proc. ACM Hum.-Comput. Interact.* 8 (ISS). DOI: 10.1145/3698129. URL: https://doi.org/10.1145/3698129.

Shneiderman, Ben (1997). *Designing the User Interface: Strategies for Effective Human-Computer Interaction.* 3rd. USA: Addison-Wesley Longman Publishing Co., Inc. ISBN: 978-0-201-69497-0.

Soliman, Mohamed et al. (2018). "FingerInput: Capturing Expressive Single-Hand Thumb-to-Finger Microgestures". In: *Proceedings of the 2018 ACM International Conference on Interactive Surfaces and Spaces.* ISS '18. Tokyo, Japan: Association for Computing Machinery, pp. 177–187. ISBN: 9781450356947. DOI: 10.1145/3279778.3279799. URL: https://doi.org/10.1145/3279778.3279799.

Tila, Christy (2021). *U.S. Monthly App Open Volume 2021.* https://www.statista.com/statistics/1288955/us-apps-opened-per-month/. (Visited on 01/27/2026).

— (2022). *China: Number of Apps on Phones by User Age Group.* https://www.statista.com/statistics/279761/number-of-apps-on-the-phones-of-mobile-internet-users-in-china-age-group/. (Visited on 01/27/2026).

— (2025). *Average Amount of Smartphone Apps Germany 2025.* https://www.statista.com/statistics/1545617/average-amount-smartphone-applications-germany/. (Visited on 01/27/2026).

Tsai, Hsin-Ruey et al. (2016). "ThumbRing: private interactions using one-handed thumb motion input on finger segments". In: *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct.* MobileHCI '16. Florence, Italy: Association for Computing Machinery, pp. 791–798. ISBN: 9781450344135. DOI: 10.1145/2957265.2961859. URL: https://doi.org/10.1145/2957265.2961859.

Tsai, Hsin-Ruey et al. (2017). "SegTouch: Enhancing Touch Input While Providing Touch Gestures on Screens Using Thumb-To-Index-Finger Gestures". In: *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems.* CHI EA '17. Denver, Colorado, USA: Association for Computing Machinery, pp. 2164–2171. ISBN: 9781450346566. DOI: 10.1145/3027063.3053109. URL: https://doi.org/10.1145/3027063.3053109.

Vive (2025a). *VIVE Pro Eye Overview — VIVE Southeast Asia.* https://www.vive.com/sea/product/vive-pro-eye/overview/. (Visited on 09/12/2025).

— (2025b). *VIVE Tracker (3.0) — VIVE.* EN. URL: https://www.vive.com/accessory/tracker3/ (visited on 09/28/2025).

Wagner, Uta et al. (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.* CHI '23. Hamburg, Germany: Association for Computing Machinery.

isbn: 9781450394215. doi: 10.1145/3544548.3581423. url: https://doi.org/10.1145/3544548.3581423.

Wambecke, Jeremy et al. (2021). “M[Eye]Cro: Eye-gaze+Microgestures for Multitasking and Interruptions”. In: *Proc. ACM Hum.-Comput. Interact.* 5 (EICS). doi: 10.1145/3461732. url: https://doi.org/10.1145/3461732.

Whitmire, Eric et al. (2017). “DigiTouch: Reconfigurable Thumb-to-Finger Input and Text Entry on Head-mounted Displays”. In: *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1.3. doi: 10.1145/3130978. url: https://doi.org/10.1145/3130978.

Wobbrock, Jacob O. et al. (2011). “The aligned rank transform for nonparametric factorial analyses using only anova procedures”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’11. Vancouver, BC, Canada: Association for Computing Machinery, pp. 143–146. isbn: 9781450302289. doi: 10.1145/1978942.1978963. url: https://doi.org/10.1145/1978942.1978963.

Wolf, Katrin et al. (2011). “A Taxonomy of Microinteractions: Defining Microgestures Based on Ergonomic and Scenario-Dependent Requirements”. In: *Human-Computer Interaction – INTERACT 2011*. Ed. by Pedro Campos et al. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 559–575. isbn: 978-3-642-23774-4.

You, Chuang-Wen et al. (2019). “Understanding Social Perceptions towards Interacting with On-Skin Interfaces in Public”. In: *Proceedings of the 2019 ACM International Symposium on Wearable Computers* (London, United Kingdom). ISWC ’19. New York, NY, USA: Association for Computing Machinery, pp. 244–253. isbn: 978-1-4503-6870-4. doi: 10.1145/3341163.3347751. url: https://doi.org/10.1145/3341163.3347751.

Zhang, Chenyang et al. (2025). *ForcePinch: Force-Responsive Spatial Interaction for Tracking Speed Control in XR*. doi: 10.1145/3746059.3747694. arXiv: 2507.18510 [cs]. url: http://arxiv.org/abs/2507.18510 (visited on 09/07/2025). Pre-published.