

Aligner, Nodder, and Winker: Creating Complete, Hands-Free Interaction Techniques that Unify Precise, UI-Independent Selection and Dragging

BAOSHENG JAMES HOU, Lancaster University, United Kingdom

PAVEL MANAKHOV, Lancaster University, United Kingdom

JINGHUI HU, Lancaster University, United Kingdom

HAOPENG WANG, Lancaster University, United Kingdom

HANS GELLERSEN, Lancaster University, United Kingdom and Aarhus University, Denmark

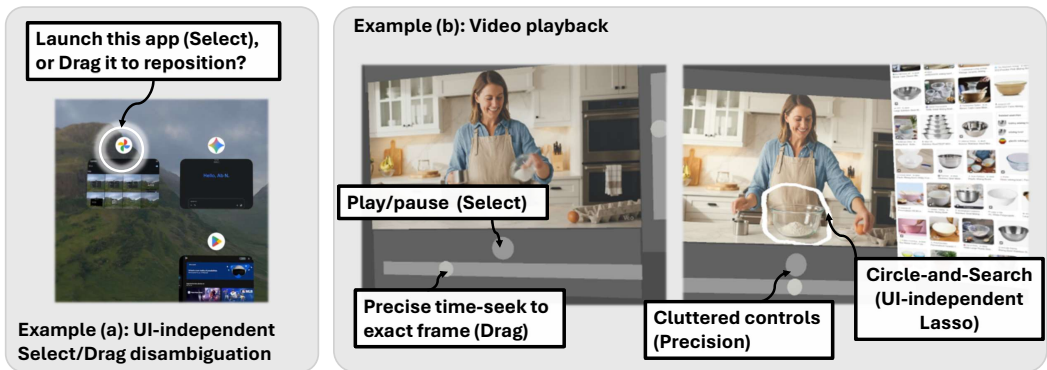


Fig. 1. Core challenges for hands-free techniques: (a) Disambiguating *Select* vs. *Drag* where UI context is insufficient, e.g., an app icon can be selected to launch or dragged to reposition. (b) Supporting UI-independent, free-form tasks, including precise dragging (time-seeking) and lasso selection (circle-and-search). Cluttered controls further highlight the need for precision.

Hands-free interaction is essential when users' hands are occupied with primary tasks. A key challenge is unifying precise, UI-independent selection and continuous dragging without explicit mode switching. This paper introduces two novel techniques: ALIGNER (spatial eye-head alignment) and NODDER (gestural decomposition of a nod); and evaluates them against an established WINKER technique, a gestural clutch using single-eye closure. A user study with controlled Fitts' law and dragging tasks, alongside a practical application, evaluated the techniques. Results show that ALIGNER offers superior selection precision but creates a visuomotor conflict during dragging; NODDER optimizes movement efficiency for large-amplitude targets despite higher physical effort; and WINKER provides the fastest performance but is susceptible to

Authors' Contact Information: Baosheng James Hou, Computing and Communications, Lancaster University, Lancaster, United Kingdom, b.hou2@lancaster.ac.uk; Pavel Manakhov, Lancaster University, Lancaster, United Kingdom, p.manakhov@lancaster.ac.uk; Jinghui Hu, Lancaster University, Lancaster, United Kingdom, j.hu23@lancaster.ac.uk; Haopeng Wang, Computing and Communications, Lancaster University, Lancaster, United Kingdom, h.wang73@lancaster.ac.uk; Hans Gellersen, Lancaster University, Lancaster, United Kingdom and Aarhus University, Aarhus, Denmark, hwg@cs.au.dk.



This work is licensed under a Creative Commons Attribution International 4.0 License.

© 2026 Copyright held by the owner/author(s).

ACM 2573-0142/2026/5-ARTETRA016

<https://doi.org/10.1145/3806030>

accidental activation. These findings inform the design of future hands-free systems by highlighting the context-dependent nature of this interaction challenge.

CCS Concepts: • **Human-centered computing** → **Pointing; Gestural input; Virtual reality; Mixed / augmented reality.**

Additional Key Words and Phrases: Pointing, Gestural Input, Virtual Reality, Augmented Reality, Interaction Techniques

ACM Reference Format:

Baosheng James Hou, Pavel Manakhov, Jinghui Hu, Haopeng Wang, and Hans Gellersen. 2026. Aligner, Nodder, and Winker: Creating Complete, Hands-Free Interaction Techniques that Unify Precise, UI-Independent Selection and Dragging. *Proc. ACM Hum.-Comput. Interact.* 10, 3, Article ETRA016 (May 2026), 21 pages. <https://doi.org/10.1145/3806030>

1 Introduction

When users' hands are unavailable, how can we enable both precise *Select* and continuous *Drag* operations? Here, *Drag* refers to any continuous control, such as repositioning a panel, moving a slider, or free-form drawing. The system must resolve the ambiguity - a problem UI context alone cannot solve: a user must be able to either launch an app (Select) or reposition that same app (Drag). This capability must be precise and UI-independent, a user must be able to make a discrete "click" (like drawing a dot) or perform a continuous "lasso" to circle-and-search a non-UI region anywhere on the screen. This problem is common in everyday scenarios, see Figure 1. For example, a user following a cooking tutorial on a tablet or AR glasses with messy hands may need to pause the video, reposition the window, seek to a specific frame, and lasso around an ingredient on the screen to search for alternatives.

While solutions exist for individual components (e.g., bimodal pointers for cursor control [41], dwell for selection [19], head gestures to issue commands [56]), they are not integrated and often rely on discrete UI elements. Combining them to support both selection and dragging typically requires an explicit mode switch, adding cognitive load and friction. The core challenge, therefore, is to design interaction techniques that cohesively integrate Point, Select, and Drag using only eye and head data, where the interaction inherently distinguishes between selection and dragging.

This paper addresses the problem space by introducing two alternative interaction techniques — ALIGNER and NODDER — based on different interaction metaphors. ALIGNER is a novel technique that employs the natural (mis)alignment of the eyes and head [40, 42], enabling command multiplexing based on how a user spatially aligns their head with their eye gaze. NODDER, explores a gestural decomposition metaphor, decomposing a single nod into a "down-up" for Select and a "down-and-hold" for Drag. This is supported by *E+H Pointer*, an optimized bimodal pointer that enables rapid, coarse gaze-based targeting followed by precise head-based refinement, extending prior work [7, 41].

We compared the proposed techniques against WINKER — a technique based on a gestural clutch metaphor, where a single-eye closure is used to 'grab' and 'hold' an object for manipulation. The technique is known from prior literature [11] but has never been empirically studied. To understand the techniques' trade-offs, we conducted a comprehensive evaluation comprising three parts: (1) a Fitts' law task (Amplitudes: 10°, 30°; Widths: 1°, 2.5°, 5°) to measure selection performance, (2) a dragging task (Amplitudes: 10°, 30°) to evaluate drag-and-drop performance, and (3) an application study with a video player to simulate a realistic, sporadic interaction context.

Our results reveal nuanced trade-offs. ALIGNER was highly precise for selection, but its drag mechanic conflicted with the natural tendency to look ahead to the target. NODDER was physically more demanding, but its *E+H Pointer* excelled at large-amplitude movements. WINKER, though

fatiguing in repetitive tasks, showed strengths in the application study, where actions were sporadic and less demanding.

The contributions of this work are threefold:

- We define the challenge of and expand the design space for enabling precise, UI-independent selection and dragging within a unified, hands-free framework without explicit mode switching.
- We introduce two techniques — ALIGNER and NODDER — each representing a different metaphorical solution to this challenge.
- We present a comprehensive empirical evaluation of these interaction techniques, revealing the fundamental performance and ergonomic trade-offs of each approach.

2 Related Work

Our work introduces hands-free techniques that build upon and extend established interaction paradigms. We situate our designs—ALIGNER, WINKER, NODDER and *E+H Pointer*—with the physiological foundations of eye-head coordination and existing techniques that inform advanced hands-free systems.

2.1 The Landscape of Hands-Free Interaction

The hands-free design space is broad. The most common method is dwell [3, 19, 23, 29, 32], which is versatile for simple selection [13, 20, 22, 29, 47, 53], toggling states [13], and marking the start and end of continuous actions like swipe typing [22] or releasing a dragged object [20]. However, it introduces the “Midas Touch” problem [9], forcing unnatural viewing behaviours and offering limited cues to distinguish overlapping actions like selection and dragging.

Explicit gestures provide more direct control. Head gestures offer a rich vocabulary, such as a nod or roll for selection [1, 4, 5, 31, 35, 44, 52, 56], turns for scrolling [4, 31, 32, 46–48], Z-axis movement for zooming [48, 56, 59], or maintaining a roll to switch modes [38]. Eyelid gestures provide another low-effort channel. Previous work has used holding one eye closed as a “clutch” to grip and drag objects [11, 36], or used the degree of eye closure to adjust depth [19]. However, if the drag is controlled by the single open eye’s gaze, users can experience fatigue and degraded pointing accuracy [36].

Many of these techniques, while powerful, are dependent on predefined UI elements [8, 12, 37, 54, 55]. Recent innovations like EyeShadows [33] utilizes saccadic confirmation for rapid selection of predefined targets. We extend these efforts to support continuous manipulation and UI-independent interaction. This enables free-form tasks where target locations are not known a priori, addressing an opportunity for fully UI-independent techniques, like the three techniques we evaluate in this paper.

2.2 Eye-Head Coordination During Target Acquisition

Effective hands-free design is informed by natural eye-head coordination. When a user shifts their gaze, the eyes move first (a rapid saccade), followed by a slower, supportive head rotation. This head movement is not for pointing; it re-centers the eyes to maintain a comfortable posture and prepare for the next gaze shift. As the head moves, the eyes perform a compensatory counter-rotation (the vestibulo-ocular reflex, or VOR) to keep the gaze locked and vision stable [39]. Critically, eye and head vectors are almost always misaligned; they are within 5° of each other only 7.5% of the time [40]. This predictable misalignment has profound design implications. The head movement following a saccade is a pre-programmed motor action, not under immediate voluntary control [6, 7, 41]. Systems using head input must therefore differentiate this natural, gaze-driven movement

from an intentional gesture. This physiological principle provides the groundwork for *E+H Pointer* and ALIGNER.

2.3 Selection and Manipulation through Alignment

Using the alignment of multiple inputs as a confirmation mechanism is a powerful interaction primitive [24, 25, 50]. In XR, this is particularly effective when applied to the eye-head relationship, leveraging their inherent misalignment as the basis for an explicit, intentional action.

Building on this, several techniques use the intentional alignment of gaze and head as an unambiguous command. Eye+Head displays a "convergence area" on a gaze-targeted UI element, confirming selection when the user moves their head pointer into this area [40]. Radi-Eye applies this to radial menus, where a user looks to pre-select a wedge and performs a head-crossing motion to confirm or manipulate a slider [42].

ALIGNER generalizes this alignment metaphor. While the aforementioned techniques are highly intuitive, they are fundamentally UI-bound, depending on a priori knowledge of target locations. ALIGNER decouples the interaction from UI semantics, extending the principle of intentional alignment to support both selection and dragging anywhere on the screen, not just on predefined UI elements.

2.4 Combining Speed of Gaze and Precision of Head

Refinement is crucial for pointing, allowing users to transition from coarse, rapid movements to fine-grained adjustments. Gaze can provide fast acquisition, refined by a mouse, touch, free-hand gestures, or aided by target expansion or zoom lenses [2, 16, 34, 45, 49, 57]. In hands-free systems, this is commonly achieved by leveraging the different dynamic capabilities of the eyes and head. A primary paradigm is the bimodal combination of fast, coarse eye gaze with slow, precise head movements for refinement [7, 10, 15, 16, 41, 43, 58].

The effectiveness of these systems hinges on the mode-switching mechanism. The heuristic approach, exemplified by BimodalGaze [41], required a minimum head velocity of $15^\circ/\text{s}$ to distinguish intentional refinement from unintentional movement. While stable, this high threshold made small, subtle refinements difficult, as users had to "exaggerate" their head motion to trigger it. The system also relied on a 150ms post-saccadic cooldown [39] and a $< 20^\circ$ trajectory difference. It reverted to gaze pointing if eye velocity exceeded $160^\circ/\text{s}$ or the eye-cursor distance was $> 10^\circ$.

The machine learning (ML)-based GazeSwitch [7] allows for earlier detection of intentional gestures. It accurately classified small head movements and entered head mode 120ms earlier than BimodalGaze, thus requiring less overall head movement for a more seamless user experience. However, it suffers from less predictability and is prone to "Late-Trigger errors" [14]. Users also observed its accuracy was highly dependent on eye tracking quality.

Our new technique *E+H Pointer* addresses these trade-offs. Instead of an ML-based model like GazeSwitch [7], we retain the predictable, heuristic-based approach of BimodalGaze [41], but use Bayesian-optimized thresholds. This provides a robust, stable state-transition that feels responsive to subtle movements without requiring exaggeration or sacrificing predictability.

NODDER provides the gestural command mechanism for this pointer. While common head gesture systems suffer from positional constraints (requiring a neutral start) or classification latency, NODDER is more responsive. It decomposes the familiar nod into its initiation phases, providing immediate feedback by freezing the cursor at the gesture's start. A quick "down-up" motion acts as a "click," while a "down-and-hold" initiates dragging. This multiplexes two fundamental actions onto a single gesture, enabling an instant, UI-independent drag from any pixel.

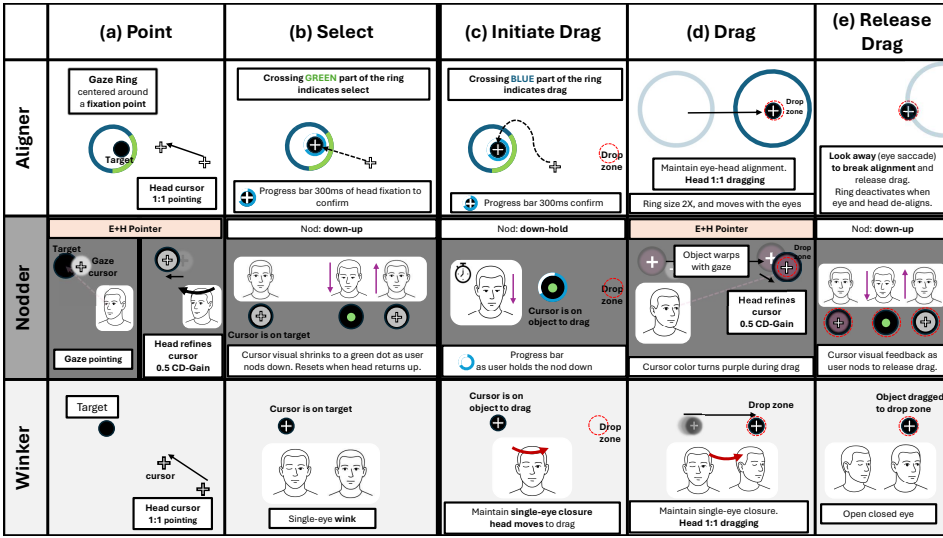


Fig. 2. When hands are occupied, a secondary hands-free modality is needed. It should distinguish a user's intent to select vs. drag, enable precise manipulation in both UI and free-form contexts (such as lasso and precise slider release). These three techniques are complete solutions to this challenge. Top: **ALIGNER** - selection and dragging achieved by aligning eye and head orientation. Middle: **NODDER** - selection and dragging through nodding, and *E+H Pointer* - cursor control via an eye-head bimodal pointer. Bottom: **WINKER** - selection and dragging through single-eye closure combined with head movement.

3 Novel Hands-free Interaction Techniques: **ALIGNER** and **NODDER**

We explored techniques that rely solely on the user's eye and head movements to support fundamental operations, such as selection and dragging without requiring knowledge of UIs or explicit mode switch: **ALIGNER** extends the metaphors in subsection 2.3 via a split-ring design that inherently disambiguates intent without a priori target knowledge. **NODDER** introduces novel multiplexing of selection and dragging through temporal gesture phases, while *E+H Pointer* refines the features and thresholds of BimodalGaze [41] to achieve robust and effortless pointing. We detail the two techniques below.

3.1 Aligner

ALIGNER translates the spatial relationship between the user's line of sight and head orientation into commands. It uses an always-visible head cursor (a cross) and an initially-hidden gaze ring. The interaction sequence begins when a user fixates on a region of interest and then moves their head cursor toward that fixation point, causing a 3° radius gaze ring independent of the targeted UI control to appear at the gaze position (see Figure 2, column (a)). To prevent visual clutter, the ring only triggers when the head moves toward the eyes, not in the opposite direction.

To distinguish between selection and dragging intents, the ring's circumference is divided into two colored sections: a green section (120°) and a blue section (240°). The green section is dynamically positioned to face the approaching head cursor. Crossing the green boundary signals a *selection intent* (Figure 2b), while crossing the blue boundary (which necessitates a curved path) signals a *drag intent* (Figure 2c). The user fixate their gaze on the target and simply steer the head cursor to enter the ring through the desired color-coded sector. The actual command (either selection or drag initiation) is confirmed after the head cursor stabilizes inside the ring for 80ms,

followed by a 300ms dwell time. A circular progress bar, displayed around the head cursor, provides visual feedback during this confirmation process.

When in a drag state, the ring expands to a 6° radius and follows the user's gaze (Figure 2d). To maintain the drag, the user must actively keep their head cursor aligned inside this expanded gaze ring. The drag is released when this eye-head alignment is broken, i.e., when the user's gaze shifts more than 6° away from the head cursor (Figure 2e). After release, the gaze ring disappears.

Depending on the targeted UI control, selection and drag commands translate into various actions. For example, a selection could place a single dot on a canvas, while a drag could draw a continuous line. Other examples include instantly warping a slider to a specified position versus time-seeking, or activating a window versus moving it in space, among others. While the function of the commands is determined by the UI control, the control itself does not define whether the user executes a selection or a drag.

3.2 Nodder

NODDER employs a complex bimodal system, we refer to as *E+H Pointer*, for pointing and a single, multi-phase head gesture (nod) for distinguishing between selection and drag-and-drop operations. *E+H Pointer* enables dynamic switching between fast, gaze-based control for coarse targeting and precise, head-based control for fine-grained refinement, for both selection and dragging. The switching happens by the system differentiating between gaze-supported head movements and voluntary head movements. Initially, a user attempts to select a target using only their gaze. If the cursor is slightly off-target due to eye-tracking inaccuracies, the user can then make a small, voluntary head movement to correct its position. *E+H Pointer* automatically detects this voluntary head rotation and activates the fine-tuning mode (see Figure 2, column (a)).

To perform a selection, which is analogous to a standard mouse click, the user executes a quick, continuous nod. This action consists of a rapid downward phase immediately followed by an upward phase, returning the head to a neutral position (Figure 2b). The system registers the "select" event as soon as the upward phase is detected, however, the position where the command is executed is based on where the nod gesture began.

The drag-and-drop operation maps to the concept of holding down a mouse button. To begin a drag, the user performs the downward phase of a nod and then briefly pauses for 800ms at the lowest point of the gesture (Figure 2c). This "down-and-hold" action signals the intent to grab an object, metaphorically holding the mouse button down. Similar to selection, the position of the grabbing point is determined based on where the nod gesture began.

The system acknowledges the start of the drag operation with immediate visual feedback, a progress bar is filled during the "holding" period, then the cursor's colour changes to red, indicating dragging is enabled. The selected object is then virtually attached to the cursor, allowing it to be controlled via *E+H Pointer* with both fast gaze-based targeting and precise head-based refinement. To release the object, the user performs another nod (a quick down-up gesture). This final nod acts as a confirmation of the drop position, metaphorically "affirming the drop" and completing the drag-and-drop sequence.

3.2.1 *E+H Pointer: Coarse Rapid Eye Pointing and Precise Head Refinement.* *E+H Pointer* is a bimodal system that unifies fast gaze-based control for coarse targeting with precise head-based refinement for fine-grained accuracy. We improved upon the BimodalGaze mechanism [41] by enhancing features and Bayesian-optimized thresholds (Figure 3) to ensure the pointing experience is both robust and effortless.

Distinguishing nod gestures from functional refinement is challenging, as a downward refinement can be confused with a nod. To resolve this ambiguity, we employ a person-specific calibration

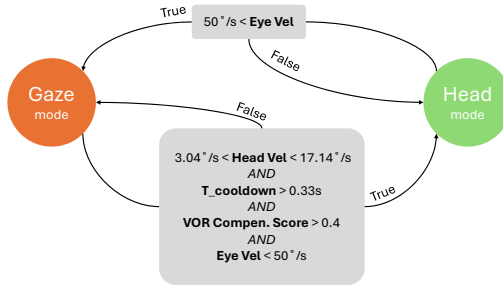


Fig. 3. *E+H Pointer* algorithm summary. To trigger head mode, head velocity (**Head Vel**) must be within a range to filter noise and reject ballistic shifts, the **VOR Compensation score** (cosine similarity of eye/head angular velocities) must exceed a threshold for eye-head correlation, and enough time (**T_cooldown**) period must pass to distinguish intentional from natural gaze shifts. The eye velocity (**Eye Vel**) threshold identifies the saccade that returns control to gaze.

procedure that models individual velocity profiles. This allows the system to accurately discriminate between a user's intent to refine the cursor position and their intent to issue a command.

All technical implementation details can be found in *Appendix*. Full implementation parameters and the Unity source code are available via Zenodo¹.

4 Empirical Study

To empirically evaluate the performance of our eye and head input techniques, we conducted a controlled within-subject experiment assessing their efficiency and accuracy for selecting and dragging under varying conditions, followed by an application-based task to understand how the techniques perform in a simulated real-world context. HTC Vive Pro Eye was used to record the eye and head tracking data in applications built with Unity 2021.3.24f1.

4.1 Techniques

We compare three techniques: ALIGNER, NODDER (section 3), and WINKER. WINKER provides an empirical evaluation of the single-eye closure clutch proposed conceptually by Kim et al. [11]. A selection is executed when a single eye closure is detected and the user subsequently reopens the eye (Figure 2, bottom row). To avoid false activation, our implementation uses a head velocity threshold (15°/s) during the single eye closure to initiate dragging. Furthermore, to avoid the pointing degradation and instability reported by Ramirez Gomez et al. [36] when tracking a single open eye, we assigned cursor control entirely to head pointing during the closure. Eye tracking is thus used solely to detect eye closure via the headset SDK's eye-openness metric.

4.2 Participants

18 participants (9 women), age ($M=27.6$, $SD=6.06$) took part in the study. 12 had good uncorrected vision, 6 had corrected vision with glasses or contact lenses. 5 had never used a VR or eye tracking device before. 12 had occasional VR and eye tracking experience. 1 was a weekly VR user with occasional eye tracking experience.

¹<https://doi.org/10.5281/zenodo.19201250>

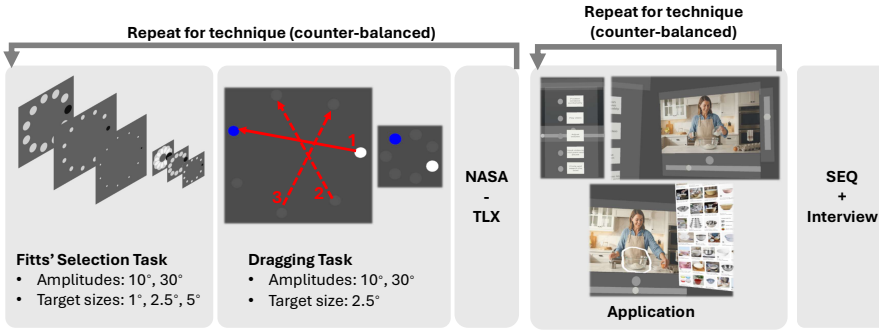


Fig. 4. Study procedure and task screenshots. In the dragging task, first three trials and their dragging directions are shown as red arrows. In application, top left picture shows two app windows overlaying. Participants need to drag to separate them and position them comfortably. They then play the video, adjust volume, and drag to seek time (top right). They then circle-and-search the bowl in the video (bottom).

4.3 Study Tasks and Measures

4.3.1 Evaluating selecting via 2D Fitts' task. For the selection task, we employed a classic 2D Fitts' Law paradigm [28]. We tested two amplitudes (10° and 30°) and three target widths (1°, 2.5°, and 5°) to cover a range from pixel-level precision to standard UI element sizes. A trial was marked as erroneous if any area outside the designated target was selected, timed out (10s), or if a drag operation was initiated.

The primary measures for the selection task included: **Selection Time (ST)**: The total trial duration from target presentation to successful selection. **Error Rate (ER)**: The percentage of trials where a selection was made outside the target, timed out, or a drag was initiated. **Throughput (TP)**: calculated using the ISO 9241-9 standard's Shannon formulation [26, 27] ($TP = ID_e / ST$), it combines speed and accuracy, using the effective index of difficulty: $ID_e = \log_2(A_e / W_e + 1)$, where A_e and W_e represent the effective amplitude and width, respectively. **Head Movement (HM)**: The accumulated angular distance of head rotation during a trial, quantifying physical effort.

4.3.2 Evaluating dragging-and-dropping via modified Fitts' sequence. Our dragging task was inspired by the Fitts' law task, using a modified sequence. Participants picked up an object, dropped it at the subsequent target, and then picked up the object for the ensuing trial from the third subsequent position in the Fitts' pattern (Figure 4). We tested two amplitudes (10° and 30°) and a fixed target width of 2.5°. A trial was marked as erroneous if the object was released beyond 1° off the destination's center, if the drag started outside the object area, or if any selection action occurred instead of a drag. Visual feedback was provided: the object to be dragged turned red when hovered over, and a red ring appeared around the destination when the object was within 1° of it.

The primary metrics for the dragging task included: **Task completion time**: Total trial duration, decomposed into **Drag Initiation Time** (from trial start to drag start) and **Drag Duration** (from drag start to release). **Release offset**: The distance in degrees from the position of drag release to the center of the drop zone. **Error rate** and **head movement** were measured as in the selection task. Perceived workload was measured using raw NASA-TLX.

4.3.3 Application. Following the controlled experiment, participants engaged with an explorative application to evaluate the techniques in a more representative, context-driven scenario. The study followed a scripted task list, which was identical for each technique and took approximately 5 minutes to complete. This involved a video player interface where users performed window

positioning (dragging), playback control (selection), volume adjustment (vertical dragging), seeking (horizontal dragging), and lasso-selection (Figure 4). While Sections 5 and 6 provide rigorous quantitative benchmarks for speed and precision, this phase focused on qualitative usability and ecological validity to avoid redundant metrics, as the primary objective was to verify functional coverage and user experience across a diverse set of realistic interaction requirements. After completing the tasks with each technique, participants completed a Single Ease Question (SEQ). Finally, qualitative feedback was collected.

4.4 Procedure

Participants were seated comfortably and provided informed consent before being briefed about the study. Each interaction technique was introduced and practiced individually. Techniques were evaluated one at a time, with participants given an initial practice round to familiarize themselves with the tasks. Before the trials began (and every time after taking the headset off), a 5-point eye-tracking calibration was performed.

Each participant completed two repetitions for each experimental condition (i.e., technique $\times$ amplitude $\times$ target size for selection, and technique $\times$ amplitude for dragging). Each condition contained 10 targets, giving 360 selecting and 120 dragging data points per participant. The order of techniques was counterbalanced across participants, while the amplitude and target size conditions were randomized within each technique block.

Participants first completed the Fitts' law selection task, followed by the dragging task. At the end of the dragging task, they completed the NASA-TLX workload survey. Next, participants engaged in an application scenario, again using each technique in the same counterbalanced order. Afterward, they completed the SEQ to report their experience with the application and provided qualitative feedback on the techniques more broadly.

Ethics and privacy: The study, which collected gaze and head movement data, was approved by our institutional review board and all participants provided informed consent. NODDER and other gestural techniques may require tuning to prevent false positives from natural movements, which could impact their user privacy and acceptance.

The full procedure is illustrated in Figure 4. The study took 90 minutes on average.

5 Results

Temporal outliers (duration $> Q3 + 3 \times IQR$) were removed per technique. Spatial outliers were retained to capture important error information (Figure 7). Trials shorter than 0.5s (accidental double-clicks) were excluded. The first Fitts' and dragging trial of each sequence were also removed. Two participants could not perform wink and their WINKER data were invalid. Overall, 3% of trials were removed. Analyses, other than error rate, included only successful trials. Each dependent variable was modeled using generalized linear mixed models (GLMMs) as a function of technique, target amplitude, and target size (for selection only), including all interactions, with participant-specific random intercepts and slopes for technique to account for fixed and random variation.

This random-slope structure allows the effect of technique to vary idiosyncratically across participants. For the majority of our dependent variables, likelihood ratio tests confirmed that a model including random slopes for technique per participant provided a significantly better fit than a simpler random-intercept-only model (all $ps < .05$). We assessed model assumptions using the DHARMA package. All checks indicated well-behaved residuals and no violations of model assumptions. All reported statistics are based on estimated marginal means derived from the fitted GLMMs, representing population-level predictions after accounting for random effects of participants. We found no significant effect of technique on perceived workload across any of the six NASA-TLX dimensions (all $ps > .05$).

5.1 Fitts' law selection task

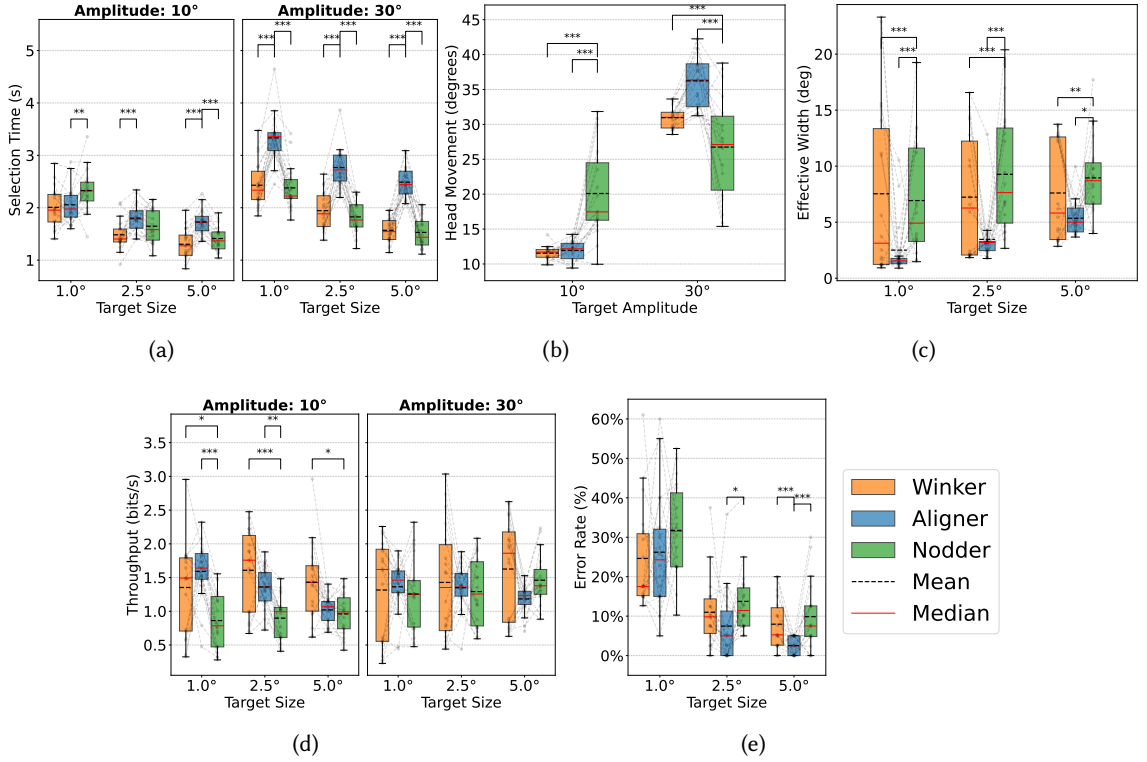


Fig. 5. Fitts law selection task empirical metrics. Statistical significance is shown as * for $p < .05$, ** for $p < .01$, and *** for $p < .001$. p-values shown are based on estimated marginal means contrasts averaged over each condition. Full size-specific contrasts are reported in the text. (c) Although the boxplots summarize participant-level data and show overlapping distributions for WINKER and NODDER, the mixed-effects analysis accounts for each participant's individual variability and within-subject pairing, revealing a statistically significant difference between the techniques.

Selection Time. ALIGNER (Median=1.98s) was consistently slower than WINKER (Median=1.59s) and NODDER (Median=1.63s), particularly at high amplitudes where it was 20-40% slower than WINKER ($p < .0001$). WINKER and NODDER performed comparably, except for 1° targets at 30° where WINKER was significantly faster ($p < .05$; Figure 5a).

Head Movement. A strong crossover effect emerged ($\chi^2(2) = 1602.65, p < .001$): NODDER produced the most head movement at low amplitude ($p < .0001$) but the least at high amplitude ($p < .02$), where it was more efficient than both WINKER and ALIGNER. (See Figure 5b).

Effective Width. ALIGNER (Median= 3.15°) and WINKER (Median= 3.85°) were consistently more precise (smaller W_e) than NODDER (Median= 7.12°) ($p < .001$). This precision gap was largest at the 1° target size, where NODDER's W_e was nearly double that of WINKER and ALIGNER ($p \leq .002$). (See Figure 5c).

Error Rate. While all techniques were comparable for small and medium targets, ALIGNER was uniquely accurate for the largest (5°) targets. At this size, ALIGNER's error rate (e.g., 0.8% at low amplitude) was significantly lower than both WINKER ($p < .02$) and NODDER ($p < .01$). (See Figure 5e).

Throughput. Performance was highly context-dependent. NODDER had the lowest throughput at low amplitude ($p < .05$) but became competitive at high amplitude. Conversely, ALIGNER achieved the highest throughput for the smallest targets at low amplitude ($p < .001$). WINKER offered the most stable performance across conditions. (See Figure 5d).

5.2 Dragging task

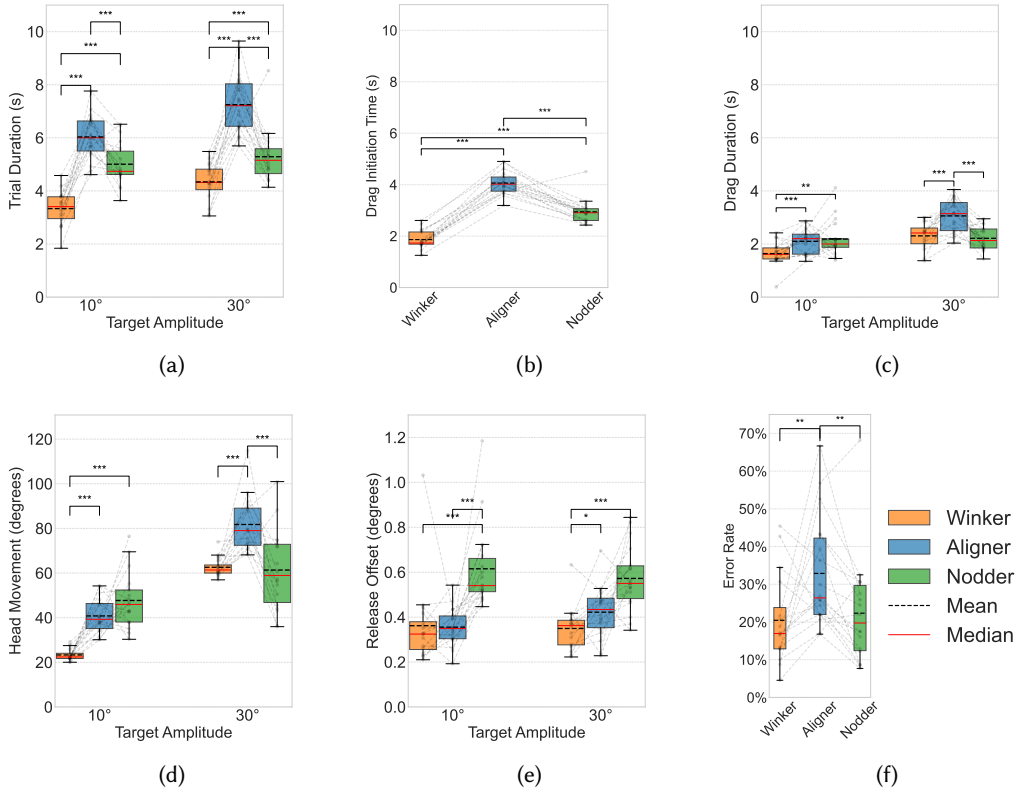


Fig. 6. Dragging task empirical metrics. Statistical significance is shown as * for $p < .05$, ** for $p < .01$, and *** for $p < .001$.

Task Completion Time. A clear performance ranking emerged ($\chi^2(2) = 231.95$, $p < .001$). WINKER was fastest (Median=3.72s), followed by NODDER (Median=4.75s), and ALIGNER (Median=6.33s). The gap between NODDER and WINKER shrank significantly at high amplitude (from 1.4s to 0.78s) ($p < .001$). (See Figure 6a).

Drag Initiation Time. All techniques were significantly different ($p < .001$). WINKER had the fastest initiation time (Median=1.76s), followed by NODDER (Median=2.63s), and ALIGNER (Median=3.67s). (See Figure 6b).

Drag Duration. WINKER was consistently fast. NODDER's performance improved with distance: at low amplitude, it was significantly slower (Median=1.87s) than WINKER (Median=1.55s) ($p = .001$), but at high amplitude, its duration (Median=1.91s) became statistically comparable to WINKER's (Median=2.16s). ALIGNER was consistently the slowest ($p < .001$). (See Figure 6c).

Head Movement. We found a strong crossover effect ($\chi^2(2) = 177.78$, $p < .001$). At low amplitude, NODDER produced the most head movement (Median=43.41°), significantly more than

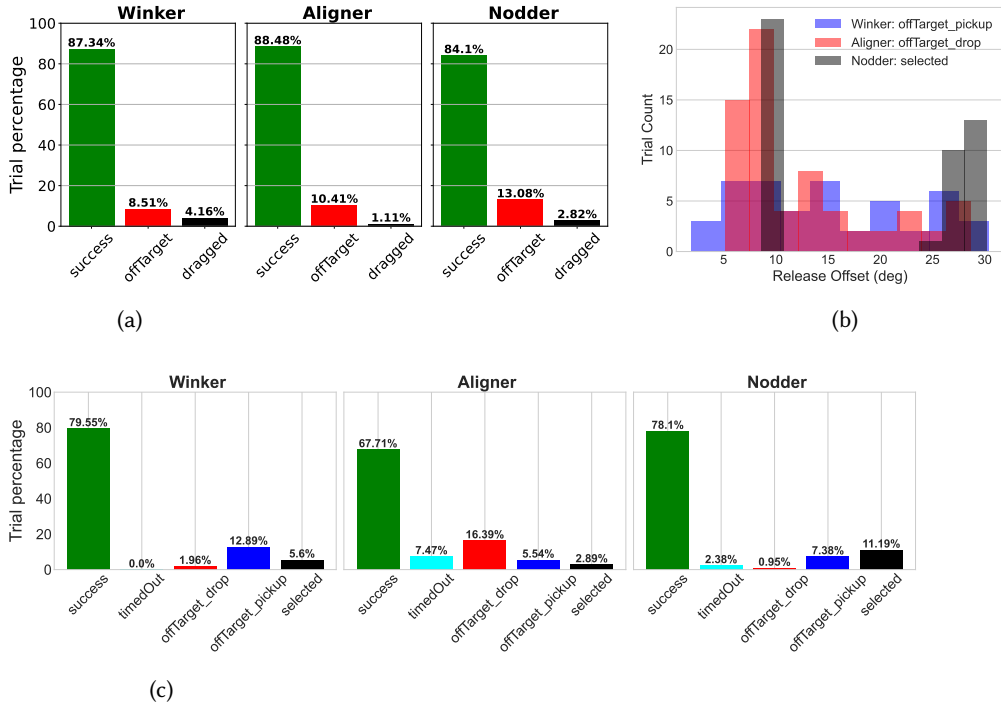


Fig. 7. Sources of errors for each technique in the (a) Fitts' selection task, (c) dragging task, and (b) the distribution of the distance-to-drop-zone at which the errors occurred in the dragging task.

ALIGNER and WINKER ($ps < .001$). This inverted at high amplitude, where NODDER produced the least (Median=57.27°), becoming significantly more efficient than ALIGNER ($p < .001$) and comparable to WINKER. (See Figure 6d).

Release Offset. NODDER was consistently the least precise technique (Median=0.54° at low amplitude). At low amplitude, WINKER (Median=0.31°) and ALIGNER (Median=0.33°) were equally and highly precise. At high amplitude, ALIGNER's precision degraded (Median=0.38°), making it significantly less precise than WINKER ($p < .05$), which remained the most precise. (See Figure 6e).

Error Rate. ALIGNER had a significantly higher error rate (Median=0.27%) than both NODDER (Median=0.17%) ($p < .01$) and WINKER (Median=0.18%) ($p < .01$). The error rates for NODDER and WINKER were not significantly different. (See Figure 6f).

6 Application

We fitted a cumulative link mixed-effects model to analyze the SEQ scores, accounting for the ordinal nature of the data. The fixed effect of **technique** was significant ($\chi^2(2) = 14.44$, $p < .001$). Post-hoc analysis showed that WINKER was rated as significantly easier to use than both NODDER ($p = .028$) and ALIGNER ($p < .001$). This subjective preference was mirrored in the overall ranks, where WINKER was the favorite for 10/18 participants. Qualitative feedback revealed highly context-dependent experiences, with perceptions differing markedly between the repetitive Fitts' task and the practical application, as well as between selection and dragging.

WINKER: Fatiguing in Repetition, Intuitive in Practice. WINKER's user experience was a tale of two contexts. In the repetitive Fitts' task, participants complained of physical fatigue ("eye

strain” P1, P4) and inaccuracy from headset shift (“it throws it off when winking,” P10). However, in the application’s sporadic, larger-target context, WINKER became the favored technique. P7 praised its intuitiveness, calling it “the easiest for video, like the mouse button, most intuitive... I had more control when to drop it”.

ALIGNER: Accurate for Selection, Unnatural for Dragging. ALIGNER’s reception was sharply divided. While praised for selection (“accurate” and “really controlled,” P12), its dragging mechanic was criticized. Requiring users to maintain eye-head alignment and avoid looking at the destination was “counter-intuitive” (P9), forcing them to “fight natural inclinations” (P3) and making the technique feel “mentally taxing” (P10) in the application.

NODDER: Controllable Pointer, Complex Gestures. NODDER’s experience was multifaceted, separating the “pointer” from the “gesture”. A highlight was the *E+H Pointer* (P7: “much much quicker with NODDER, I can zoom across screen... head refinement is intuitive”). However, this was often overshadowed by the “complex interaction gestures.” While the “down-up” nod for selection was easy, the “down-and-hold” for dragging was a consistent point of physical strain (P9: “requires neck muscle a lot, when holding down”), compounded by the progress bar moving out of sight (P14).

7 Discussion

Our evaluation of three hands-free techniques reveals a complex interplay between task context, action type, and individual user characteristics. While we used a VR headset for prototyping, the underlying principles of eye-head coordination and gestural kinematics are rooted in human physiology rather than display technology. Their practical success across different hardware will evolve as tracking technology matures and factors like headset shifts or eye-openness detection are further refined.

7.1 Bound by Alignment: The Trade-Off Between Precision and Speed

Aligner’s high selection precision results from its confirmation mechanism, which enforces head stability via a 300ms dwell to filter motor noise. In contrast, techniques requiring active gestures are more susceptible to error: winking can cause headset shifts, and NODDER’s gesture introduces movement variability. NODDER’s precision is further limited by its back-tracing algorithm, which registers selections at the last stable fixation before the nod.

During dragging, Aligner’s required eye-head alignment creates a fundamental mismatch with natural visuomotor strategies. Users instinctively look ahead to the destination [17, 18], which breaks alignment and triggers unintended release. They moved more carefully - an experience one participant called “meditative for focus” (P17). We acknowledge this as a context-dependent trade-off: ALIGNER provides superior precision for selection, while WINKER and NODDER better accommodate natural look-ahead behavior by decoupling gaze from control, freeing the eyes for movement planning and faster dragging.

7.2 Mismatched Metaphors: Error Sources and Coordination Mechanics

An effective metaphor for selection may not work for dragging. For all techniques, selection errors were dominated by off-target failures on demanding 1° targets, whereas distinguishing intent (accidental drags) was a minimal challenge (1–4% of errors). The multi-phase dragging task exposed unique, design-rooted sources of error for each technique: for WINKER, accidental activation was the primary cause (13%) of drag errors. These occurred almost uniformly, hinting the wink detection was tuned to be too sensitive for tasks requiring sustained focus. For ALIGNER, the look-ahead conflict caused 16% of failures (off-target drops), peaking 5–10° away from the drop zone. This premature release explains qualitative feedback calling the mechanic “counter-intuitive.”

Finally, NODDER's intent ambiguity (11%) indicates the "down-and-hold" gesture, while allowing for an immediate system response and multiplexing two commands onto a single gesture, caused physical strain and felt overwhelming for some due to the cognitive load of the full sequence, overshadowing its technical precision.

NODDER's performance benefited from its optimized *E+H Pointer* thresholds. Unlike systems such as [7], users reported no instability issues while praised it as intuitive and easy to control. This efficiency raised NODDER's throughput (Figure 5d) and minimized head movement at large amplitudes (Figure 5b, Figure 6d). At smaller amplitudes, head movement was greater, as the nod gesture is inherently more "expensive" than fixation.

NODDER's performance was the most idiosyncratic. Our GLMMs confirmed that random slopes per participant significantly improved model fit. Some users refined consistently while others (especially for larger targets) almost never did, suggesting a personal trade-off between speed and precision. We speculate a negative interplay between refinement and the nod: some participants, worried the cursor might shift, hesitated during the nod. Those who nodded more decisively (faster, shorter motions) were less likely to notice the small visual delay before back-tracing engaged, while slower motions led to perceived delay and hesitation.

7.3 Benchmarking and Functional Distinction

Recent selection-only techniques [30, 33] report higher throughput (3.5–4.0 bps) than our general-purpose techniques (1.3–1.8 bps). The difference in throughput corresponds to differing interaction scopes: selection-only systems optimize for known UI targets, whereas our metaphors enable a unified interaction loop (point, select and/or drag, maintain, and release) at any location without prior target knowledge or explicit mode switches. Notably, our dragging completion times (Winker: 3.72s, Nodder: 4.75s) are comparable to established benchmarks like Gaze+Hold (4.2s) [36], suggesting they are robust solutions for multi-modal tasks.

7.4 Limitations and Future Directions

Our study used isolated targets; ALIGNER's precision suggests strength for cluttered environments, but further validation is required to confirm that head-cursor confirmation remains robust in dense UI contexts. Similarly, for large targets lacking salient features, further investigation is warranted to understand if gaze wandering occurs. While NODDER's algorithm utilizes physiological coordination for robustness, its resilience to incidental movements in the wild must be formally tested.

Finally, it remains an open question regarding long-term ergonomics; specifically, whether the steep learning curves of ALIGNER and NODDER would flatten with practice, or if WINKER's simplicity would be overshadowed by long-term muscle fatigue. This inspires a vision for future interfaces that automatically transition to secondary modalities by detecting when hands have become occupied [21], intelligently drawing upon unique strengths based on the interaction context.

8 Conclusion

This paper addresses the core challenge of integrating hands-free, precise, and UI-independent selection and dragging without explicit mode switches — a problem that component-level solutions for pointing or selection alone do not solve. We implemented and empirically compared three techniques as solutions to this integration problem: a spatial multiplexer (ALIGNER), a gestural decomposer (NODDER), and a gestural clutch (WINKER). Our findings reveal this is a complex design challenge fraught with deep, context-dependent trade-offs between precision, ergonomics, and sensorimotor congruence. This work highlights that the future of usable hands-free systems lies not in optimizing isolated components, but in navigating the fundamental trade-offs inherent to these integrated interaction techniques.

Acknowledgments

This work was supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (Grant No. 101021229 GEMINI: Gaze and Eye Movement in Interaction). This project benefited from the High End Computing facility at Lancaster University. We gratefully acknowledge the GEMINI team at Lancaster and Aarhus Universities for the collaborative brainstorming sessions that shaped the conceptualization of these techniques.

References

- [1] Stephen Brewster, Joanna Lumsden, Marek Bell, Malcolm Hall, and Stuart Tasker. 2003. Multimodal 'eyes-free' interaction techniques for wearable devices. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 473–480.
- [2] Ishan Chatterjee, Robert Xiao, and Chris Harrison. 2015. Gaze+ gesture: Expressive, precise and targeted free-space interactions. In *Proceedings of the 2015 ACM on international conference on multimodal interaction*. 131–138.
- [3] Tatiana V Evreinova, Grigori Evreinov, and Roope Raisamo. 2011. Integrating discrete events and continuous head movements for video-based interaction techniques. *Behaviour & Information Technology* 30, 6 (2011), 739–746.
- [4] Wenxin Feng, Jiangnan Zou, Andrew Kurauchi, Carlos H Morimoto, and Margrit Betke. 2021. Hgaze typing: Head-gesture assisted gaze typing. In *ACM Symposium on eye tracking research and applications*. 1–11.
- [5] D Gorodnichy, S Malik, and G Roth. 2002. Nouse 'Use your nose as a joystick or a mouse'-a new technology for handsfree games and interfaces. In *Proceedings of the International Conference on Vision Interface (VI2002)*.
- [6] Baosheng James Hou, Joshua Newn, Ludwig Sidenmark, Anam Ahmad Khan, Per Bækgaard, and Hans Gellersen. 2023. Classifying head movements to separate head-gaze and head gestures as distinct modes of input. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–14.
- [7] Baosheng James Hou, Joshua Newn, Ludwig Sidenmark, Anam Ahmad Khan, and Hans Gellersen. 2024. GazeSwitch: Automatic Eye-Head Mode Switching for Optimised Hands-Free Pointing. *Proceedings of the ACM on Human-Computer Interaction* 8, ETRA (2024), 1–20.
- [8] Toshiya Isomoto, Shota Yamanaka, and Buntarou Shizuki. 2020. Gaze-based Command Activation Technique Robust Against Unintentional Activation using Dwell-then-Gesture. In *Graphics Interface 2020*.
- [9] Robert JK Jacob. 1990. What you look at is what you get: eye movement-based interaction techniques. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 11–18.
- [10] Shahram Jalaliniya, Diako Mardanbegi, and Thomas Pederson. 2015. MAGIC pointing for eyewear computers. In *Proceedings of the 2015 ACM International Symposium on Wearable Computers*. 155–158.
- [11] Jinhyuk Kim, Jaekwang Cha, Hojun Lee, and Shiho Kim. 2017. Hand-free natural user interface for VR HMD with IR based facial gesture tracking sensor. In *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology*. 1–2.
- [12] Taejun Kim, Auejin Ham, Sunggeun Ahn, and Geehyuk Lee. 2022. Lattice menu: A low-error gaze-based marking menu utilizing target-assisted gaze gestures on a lattice of visual anchors. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–12.
- [13] Rick Kjeldsen. 2006. Improvements in vision-based pointer control. In *Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility*. 189–196.
- [14] Manu Kumar, Jeff Klingner, Rohan Puranik, Terry Winograd, and Andreas Paepcke. 2008. Improving the accuracy of gaze input for interaction. In *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications* (Savannah, Georgia) (ETRA '08). ACM, New York, NY, USA, 65–68. doi:10.1145/1344471.1344488
- [15] Andrew Kurauchi, Wenxin Feng, Carlos Morimoto, and Margrit Betke. 2015. HMAGIC: head movement and gaze input cascaded pointing. In *Proceedings of the 8th ACM international conference on Pervasive technologies related to assistive environments*. 1–4.
- [16] Mikko Kytö, Barrett Ens, Thammathip Piumsomboon, Gun A Lee, and Mark Billinghurst. 2018. Pinpointing: Precise head-and-eye-based target selection for augmented reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–14.
- [17] Michael Land, Neil Mennie, and Jennifer Rusted. 1999. The roles of vision and eye movements in the control of activities of daily living. *Perception* 28, 11 (1999), 1311–1328.
- [18] Michael F Land and Mary Hayhoe. 2001. In what ways do eye movements contribute to everyday activities? *Vision research* 41, 25–26 (2001), 3559–3565.
- [19] Yi Lin, Xingwei Chen, Chuyang Zhang, and Jingjing Chen. 2025. A virtual object manipulation method based on eye-gaze selection and head-eye coordination movement. *International Journal of Human-Computer Studies* 197 (2025), 103458.

- [20] Chang Liu, Alexander Plopski, and Jason Orlosky. 2020. OrthoGaze: Gaze-based three-dimensional object manipulation using orthogonal planes. *Computers & Graphics* 89 (2020), 1–10.
- [21] Xingyu Bruce Liu, Jiahao Nick Li, David Kim, Xiang‘Anthony’ Chen, and Ruofei Du. 2024. Human i/o: Towards a unified approach to detecting situational impairments. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–18.
- [22] Xueshi Lu, Difeng Yu, Hai-Ning Liang, and Jorge Goncalves. 2021. itext: Hands-free text entry on an imaginary keyboard for augmented reality systems. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. 815–825.
- [23] Xueshi Lu, Difeng Yu, Hai-Ning Liang, Wenge Xu, Yuzheng Chen, Xiang Li, and Khalad Hasan. 2020. Exploration of hands-free text entry techniques for virtual reality. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 344–349.
- [24] Mathias N Lystbæk, Ken Pfeuffer, Jens Emil Sloth Grønbaek, and Hans Gellersen. 2022. Exploring gaze for assisting freehand selection-based text entry in ar. *Proceedings of the ACM on Human-Computer Interaction* 6, ETRA (2022), 1–16.
- [25] Mathias N Lystbæk, Peter Rosenberg, Ken Pfeuffer, Jens Emil Grønbaek, and Hans Gellersen. 2022. Gaze-hand alignment: Combining eye gaze and mid-air pointing for interacting with menus in augmented reality. *Proceedings of the ACM on Human-Computer Interaction* 6, ETRA (2022), 1–18.
- [26] Ian Scott MacKenzie. 2013. A note on the validity of the Shannon formulation for Fitts’ index of difficulty. *Open Journal of Applied Sciences* 3, 6 (2013), 360–368.
- [27] I Scott MacKenzie. 2018. Fitts’ law. *The wiley handbook of human computer interaction* 1 (2018), 347–370.
- [28] I Scott MacKenzie and William Buxton. 1992. Extending Fitts’ law to two-dimensional tasks. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 219–226.
- [29] Pavel Manakhov, Ludwig Sidenmark, Ken Pfeuffer, and Hans Gellersen. 2024. Gaze on the go: Effect of spatial reference frame on visual target acquisition during physical locomotion in extended reality. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–16.
- [30] Aunnoy Mutasim, Anil Ufuk Batmaz, Moaaz Hudhud Mughrabi, and Wolfgang Stuerzlinger. 2022. Performance analysis of saccades for primary and confirmatory target selection. In *Proceedings of the 28th ACM Symposium on Virtual Reality Software and Technology*. 1–12.
- [31] Adam Nowosielski. 2016. Minimal interaction touchless text input with head movements and stereo vision. In *International Conference on Computer Vision and Graphics*. Springer, 233–243.
- [32] Tomi Nukarinen, Jari Kangas, Oleg Špakov, Poika Isokoski, Deepak Akkil, Jussi Rantala, and Roope Raisamo. 2016. Evaluation of HeadTurn: An interaction technique using the gaze and head turns. In *Proceedings of the 9th Nordic Conference on Human-Computer Interaction*. 1–8.
- [33] Jason Orlosky, Chang Liu, Kenya Sakamoto, Ludwig Sidenmark, and Adam Mansour. 2024. Eyeshadows: Peripheral virtual copies for rapid gaze selection and interaction. In *2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*. IEEE, 681–689.
- [34] Khushman Patel, Vrushank Phadnis, Eric J Gonzalez, Hans Gellersen, Ken Pfeuffer, and Mar Gonzalez-Franco. 2025. H 2 E: Hand, Head, Eye a Multimodal Cascade of Natural Inputs. In *2025 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE, 84–89.
- [35] Thammathip Piumsomboon, Gun Lee, Robert W Lindeman, and Mark Billinghurst. 2017. Exploring natural eye-gaze-based interaction for immersive virtual reality. In *2017 IEEE symposium on 3d user interfaces (3DUI)*. IEEE, 36–39.
- [36] Argenis Ramirez Ramirez Gomez, Christopher Clarke, Ludwig Sidenmark, and Hans Gellersen. 2021. Gaze+ hold: eyes-only direct manipulation with continuous gaze modulated by closure of one eye. In *ACM symposium on eye tracking research and applications*. 1–12.
- [37] David Rudi, Ioannis Giannopoulos, Peter Kiefer, Christian Peier, and Martin Raubal. 2016. Interacting with maps on optical head-mounted displays. In *Proceedings of the 2016 Symposium on Spatial User Interaction*. 3–12.
- [38] Rongkai Shi, Nan Zhu, Hai-Ning Liang, and Shengdong Zhao. 2021. Exploring head-based mode-switching in virtual reality. In *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 118–127.
- [39] Ludwig Sidenmark and Hans Gellersen. 2019. Eye, head and torso coordination during gaze shifts in virtual reality. *ACM Transactions on Computer-Human Interaction (TOCHI)* 27, 1 (2019), 1–40.
- [40] Ludwig Sidenmark and Hans Gellersen. 2019. Eye&head: Synergetic eye and head movement for gaze pointing and selection. In *Proceedings of the 32nd annual ACM symposium on user interface software and technology*. 1161–1174.
- [41] Ludwig Sidenmark, Diako Mardanbegi, Argenis Ramirez Gomez, Christopher Clarke, and Hans Gellersen. 2020. Bimodalgaze: Seamlessly refined pointing with gaze and filtered gestural head movement. In *ACM Symposium on Eye Tracking Research and Applications*. 1–9.

- [42] Ludwig Sidenmark, Dominic Potts, Bill Bapisch, and Hans Gellersen. 2021. Radi-eye: Hands-free radial interfaces for 3d interaction using gaze-activated head-crossing. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–11.
- [43] Oleg Špakov, Poika Isokoski, and Päivi Majaranta. 2014. Look and lean: accurate head-assisted eye pointing. In *Proceedings of the symposium on eye tracking research and applications*. 35–42.
- [44] Oleg Špakov and Päivi Majaranta. 2012. Enhanced gaze interaction using simple head gestures. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*. 705–710.
- [45] Sophie Stellmach and Raimund Dachsel. 2012. Look & touch: gaze-supported target acquisition. In *Proceedings of the SIGCHI conference on human factors in computing systems*. 2981–2990.
- [46] Yang Tian, Youpeng Zhang, Yukang Yan, Shengdong Zhao, Xiaojuan Ma, and Yuanchun Shi. 2025. AmplitudeArrow: On-the-Go AR Menu Selection Using Consecutive Simple Head Gestures and Amplitude Visualization. *IEEE Transactions on Visualization and Computer Graphics* (2025).
- [47] Herman Tolle, Ismiarta Aknuranda, Mahardeka Tri Ananta, Komang Candra Brata, and Hanifah Muslimah Az-Zahra. 2016. Design of keyboard input control for mobile application using Head Movement Control (HEMOCS). In *2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*. IEEE, 518–523.
- [48] Robert Van Liere and Wim de Leeuw. 1999. Exploration of large image collections using virtual reality devices. In *Proceedings of the 1999 workshop on new paradigms in information visualization and manipulation in conjunction with the eighth ACM international conference on Information and knowledge management*. 83–86.
- [49] Uta Wagner, Matthias Albrecht, Haopeng Wang, Ken Pfeuffer, and Hans Gellersen. 2024. Gaze, Wall, and Racket: Combining Gaze and Hand-controlled Plane for 3D Selection in Virtual Reality. In *ISS'24 ACM Interactive Surfaces and Spaces*.
- [50] Uta Wagner, Mathias N Lystbæk, Pavel Manakhov, Jens Emil Sloth Grønbaek, Ken Pfeuffer, and Hans Gellersen. 2023. A fitts' law study of gaze-hand alignment for selection in 3d user interfaces. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 1–15.
- [51] Haopeng Wang, Ludwig Sidenmark, Florian Weidner, Joshua Newn, and Hans Gellersen. 2025. HeadShift: Head Pointing with Dynamic Control-Display Gain. *ACM Transactions on Computer-Human Interaction* 32, 1 (2025), 1–28.
- [52] Zhibo Wang, Yijie Li, Bonan Jin, Qian Wang, Yunhe Feng, Yanjun Li, and Huejie Shao. 2019. AirMouse: Turning a Pair of Glasses into a Mouse in the Air. *IEEE Internet of Things Journal* 6, 5 (2019), 7473–7483.
- [53] Matthew R Williams and Robert F Kirsch. 2015. Evaluation of head orientation and neck muscle EMG signals as three-dimensional command sources. *Journal of neuroengineering and rehabilitation* 12, 1 (2015), 25.
- [54] Wenge Xu, Hai-Ning Liang, Yuxuan Zhao, Tianyu Zhang, Difeng Yu, and Diego Monteiro. 2019. Ringtext: Dwell-free and hands-free text entry for mobile head-mounted displays using head motions. *IEEE transactions on visualization and computer graphics* 25, 5 (2019), 1991–2001.
- [55] Yukang Yan, Yingtian Shi, Chun Yu, and Yuanchun Shi. 2020. Headcross: Exploring head-based crossing selection on head-mounted displays. *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies* 4, 1 (2020), 1–22.
- [56] Yukang Yan, Chun Yu, Xin Yi, and Yuanchun Shi. 2018. Headgesture: Hands-free input approach leveraging head movements for hmd devices. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 4 (2018), 1–23.
- [57] Shumin Zhai, Carlos Morimoto, and Steven Ihde. 1999. Manual and gaze input cascaded (MAGIC) pointing. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 246–253.
- [58] Ben Zhang, Yu-Hsiang Chen, Claire Tuna, Achal Dave, Yang Li, Edward Lee, and Björn Hartmann. 2014. HOBS: head orientation-based selection in physical spaces. In *Proceedings of the 2nd ACM symposium on Spatial user interaction*. 17–25.
- [59] Kaining Zhang, Catarina Moreira, Pedro Belchior, Gun Lee, Mark Billingham, and Joaquim Jorge. 2025. HeadZoom: Hands-Free Zooming and Panning for 2D Image Navigation Using Head Motion. *arXiv preprint arXiv:2508.01765* (2025).

A Signal Processing

All eye and head tracking data are processed in local headset coordinates. Angular velocities are computed via the central difference method and smoothed using a three-frame moving average filter. The eye-head coordination metrics, specifically the VOR Compensation score, are calculated using a five-frame sliding window to filter high-frequency jitter and ensure stable state transitions. While the default cursor mapping for all techniques is 1:1, the system applies a 0.5 Control-Display (CD) gain to the head cursor specifically during the E+H Pointer's refinement mode [16]. This

reduction in sensitivity allows users to acquire small targets with significantly higher precision than raw head-pointing.

B E+H Pointer: Threshold Optimization and Pilot Study

E+H Pointer builds upon BimodalGaze [41]. Its core principle is to detect transitions from rapid eye movements (saccades) used for coarse targeting, to slower head movements used for precise refinement. The switch from gaze- to head-pointing is triggered by the Vestibulo-Ocular Reflex (VOR), in which the eyes remain fixed on a target while the head refines the cursor's position. The challenge lies in distinguishing VOR generated during intentional head-based refinement from the VOR that naturally occurs at the end of each gaze shift.

BimodalGaze relied on hard thresholds to achieve this distinction. Since the head typically lags the eye by approximately 150ms during a natural gaze shift [39], VOR activity occurring after this **cooldown period** is more likely to reflect refinement, particularly when the trajectory difference between eye and head is less than 20° . These thresholds were derived from gaze shift data analysis [39, 41]. Additionally, a **minimum head velocity** of 15° was required to filter out unintentional movement. A switch back to gaze pointing was detected when a saccade exceeded a **minimum eye velocity** of 160° , or when the eye-cursor distance grew larger than 10° .

E+H Pointer introduces several refinements to these feature calculations, See Figure 3 for algorithm summary. Instead of projection-based trajectory measures, it directly computes the correlation between eye and head movements by calculating the cosine similarity of their angular velocities, smoothed over a 5-frame window. We refer to this as the **VOR Compensation score**, where values close to 1 indicate strong correlation of eye and head movement typical of VOR activity.

We further introduce both a lower and an **upper bound on head velocity** to trigger head mode. The lower bound filters out unintentional drift, while the upper bound distinguishes deliberate refinement from the faster, ballistic head movements that often accompany large gaze shifts [51].

The gaze-cursor distance threshold is removed to better support large-scale manipulation tasks. For example, during a lasso selection, users often keep their gaze centered on the target region for spatial context while moving their head to trace the boundary. In this design, the transition back to gaze pointing depends solely on the occurrence of a saccade.

The head-movement cooldown period since the last eye saccade is retained as an important feature.

Finally, all threshold values are optimized through experiments that elicit both natural gaze shifts and intentional head refinements, ensuring the system is tuned for task relevance.

B.1 Data Collection and Threshold Optimization

To determine the optimal thresholds for our algorithm, we first collected ground-truth data in a pilot study with 12 participants. They each completed 100 trials of a manual eye-and-head pointing task [16], selecting 1-degree targets. In this task, participants aimed initially with their gaze, then pressed and held a controller thumbpad to switch to head-pointing for refinement, and released it to confirm the selection. All frames with thumbpad held down are marked as head-pointing, other frames are marked as gaze-pointing. We used these manually segmented interaction data to optimize the algorithm's parameters via Bayesian optimization. The process, configured with 20 random starts and 30 optimization steps, maximized frame-wise balanced accuracy (the average of recall obtained on each class) and yielded the following thresholds: Eye saccade velocity threshold of $50^\circ/\text{s}$ with a post-saccadic cooldown of 0.33s. The VOR Compensation metric threshold of 0.4, and a head refinement velocity constrained between $3.04^\circ/\text{s}$ and $17.14^\circ/\text{s}$.

Using leave-one-participant-out cross-validation, our algorithm achieved a frame-wise balanced accuracy of 0.81 in predicting the user's intended mode (gaze or head pointing), indicating it

generalizes well to new users. To validate its practical performance, we then conducted a pilot test with five new users who were not in the original training set. This test confirmed the algorithm's usability, as participants were able to successfully acquire and select small (1°) targets at a large distance (30° movement amplitude) in under 1.5 seconds per target.

A technical challenge is distinguishing intentional command gestures from functional movements like *E+H Pointer's* cursor refinement, for example, a downward refinement may be confused with a downward nod. NODDER resolves this by analyzing the kinematics of the head motion. An intentional nod is differentiated from a slower, downward refinement movement by its velocity.

C Nodder: Statistical Calibration and Kinematic Analysis

We conducted a pilot study to understand how people perform a nod gesture. The results indicated that head rotation velocity profiles may be person-specific (see Figure 8). To ensure our system is robust and responsive to these individual differences, each user of NODDER should complete a calibration process before use. We detail the implementation of this procedure below.

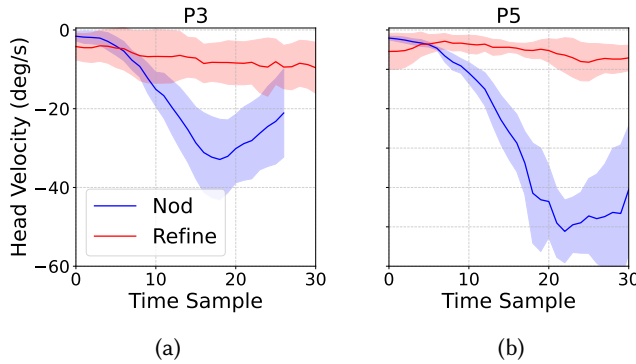


Fig. 8. Two participants with distinct nodding patterns. P3's nods were shorter and more subtle than P5's longer, more aggressive nods. This highlights the challenge in establishing a single threshold for nod detection across all users. Means with 95% confidence interval are shown.

C.1 Data Collection

The calibration process begins with a nine-trial calibration phase to model a user's idiosyncratic downward head movements for both deliberate cursor refinement and natural nodding. The calibration task follows a grid-based trial structure. In each trial, a target (1° dot) appears at one of nine positions. To elicit and capture clear examples of downward refinement movements, thereby generating the data needed to reliably differentiate them from nod gestures, a random upward offset, uniformly distributed between 2.5° and 5° , is artificially applied to the gaze cursor. The user first acquires the target by refining the cursor onto it, which triggers visual feedback. The calibration procedure implies that, following acquisition, the user performs a single, natural head nod while pressing and holding a controller thumbpad. This action segments the nod gesture in the recorded data. Releasing the thumbpad concludes the trial, and the next target appears.

C.2 Data Preprocessing

We then refine the segmentation of nodding from users' trackpad actions. We first identify the downward phase of the nod in each trial, as any continuous period where two conditions were met concurrently: the user's input on the trackpad was active (i.e., clicked), and, the vertical head

velocity exceeded a downward threshold of 5 deg/s. To ensure robustness, a temporal filter is applied to discard spurious events with a duration of less than five samples (approx. 55ms). Trials containing anything other than exactly one valid downward nod event are deemed ambiguous and excluded from subsequent analysis.

To precisely locate the gesture's onset, the start time of the identified downward nod is retrospectively adjusted. The algorithm traced backward from the initial detected start point (for a maximum of 10 samples, approx. 110ms) to find the end point of the previous head fixation. The event's onset is then extended to here, more accurately anchoring the beginning of the nodding movement.

Then, the segmented time-series data for refining movements and nodding movements are temporally aligned. Both datasets are then truncated to a consistent length of 20 samples (approx. 220m) to enable a direct point-by-point comparison.

C.3 Statistical Analysis to Identify Periods of Significant Difference

To quantitatively differentiate between nodding and refining head movements, we perform a statistical analysis on their angular velocity profiles. Our method is designed to identify the most salient temporal window where the two movements differ and to derive velocity thresholds for classification. Our statistical approach involves a multi-stage process to ensure robust identification of significant differences.

The initial stage of the analysis involves conducting independent-samples t-tests at each time sample to identify moments of significant divergence in mean angular velocity between the nodding and refining groups. To mitigate the inflated Type I error rate associated with performing multiple comparisons, a Benjamini-Hochberg False Discovery Rate (FDR) correction is subsequently applied to the resulting p-values.

To corroborate these point-wise findings with a global measure of significance, a non-parametric permutation test ($n=2000$) is then employed. This procedure confirms that the maximum t-statistic observed in the empirical data is significantly greater than the maximum t-statistics generated from the permuted data. This result provides robust evidence for a statistically significant difference between the two entire time-series profiles.

Finally, the set of time points deemed significant by the FDR procedure is analyzed to locate the longest continuous sequence of divergence. This segment was designated as the most characteristic temporal window for discriminating between the two gestures.

C.4 Identifying Classification Thresholds

The characteristic timing of the most significant period (T) of divergence between the two gestures is identified. For the real-time algorithm a grace period of half of this window length is added.

Within this significant time window, a velocity threshold is established (V1) to distinguish the movements, derived from the velocity profiles at the single most statistically significant time point. We identify the gap between the lower bound of the 95% confidence interval for refining movements and the upper bound of the 95% confidence interval for nodding movements. A more liberal threshold is then set as the less negative (nod downward velocity is negative) between a point 70% across this gap and the nod's upper confidence bound.

The velocity threshold for identifying a nod during a drag-and-release interaction (V2) is set to be more conservative to discourage false release, set as the more negative between a point 70% across the confidence interval gap and the nod's lower (more negative velocity) confidence bound.

Finally, to establish a general baseline, we first calculate the upper bound of the 95% confidence interval for the nodding velocity across the entire analysed time window. The global threshold is then set as the absolute minimum value of this upper bound. This represents a robust, timing-independent minimum peak velocity that a movement must achieve to be considered a nod.

Therefore a nod down is detected, if the global velocity threshold is met, or, if currently not in drag mode - the velocity threshold V_1 is met within T , and if currently in drag mode - the velocity threshold V_2 is met within T .

By parsing a single nod gesture into its constituent phase, a rapid down-up versus a down-and-hold, NODDER can multiplex two fundamental operations. This design aims to avoid the cognitive overhead of learning and executing two entirely different gestures, with the hope of creating a unified and efficient interaction paradigm.