HeadShift: Head Pointing with Dynamic Control-Display Gain

HAOPENG WANG, Lancaster University, United Kingdom
LUDWIG SIDENMARK, University of Toronto, Canada
FLORIAN WEIDNER, Lancaster University, United Kingdom
JOSHUA NEWN, Lancaster University, United Kingdom
HANS GELLERSEN, Lancaster University, United Kingdom and Aarhus University, Denmark

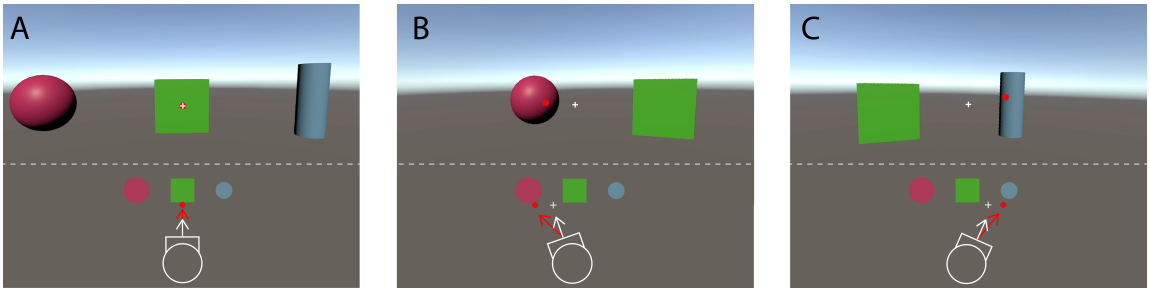


Fig. 1. *HeadShift* improves ergonomics and precision of head pointing in HMDs by employing dynamic gain. The technique reduces the amount of head movement required for cursor (red dot) positioning and makes targets selectable within a comfortable gaze range around the head vector (white cross, included for illustration and not shown to the user).

Head pointing is widely used for hands-free input in head-mounted displays (HMDs). The primary role of head movement in an HMD is to control the viewport based on absolute mapping of head rotation to the 3D environment. Head pointing is conventionally supported by the same 1:1 mapping of input with a cursor fixed in the centre of the view, but this requires exaggerated head movement and limits input granularity. In this work, we propose to adopt dynamic gain to improve ergonomics and precision, and introduce the *HeadShift* technique. The design of HeadShift is grounded in natural eye-head coordination to manage control of the viewport and the cursor at different speeds. We evaluated HeadShift in a Fitts' Law experiment and on three different applications in VR, finding the technique to reduce error rate and effort. The findings are significant as they show that gain can be adopted effectively for head pointing while ensuring that the cursor is maintained within a comfortable eye-in-head viewing range.

CCS Concepts: • **Human-centered computing** → **Pointing**; **Virtual reality**.

Additional Key Words and Phrases: Pointing, Control-Display Gain, Virtual Reality, Head Mounted Display

ACM Reference Format:
Haopeng Wang, Ludwig Sidenmark, Florian Weidner, Joshua Newn, and Hans Gellersen. 2024. HeadShift: Head Pointing with Dynamic Control-Display Gain. 1, 1 (August 2024), 28 pages. https://doi.org/10.1145/3689434

Authors' addresses: Haopeng Wang, Lancaster University, Lancaster, United Kingdom, h.wang73@lancaster.ac.uk; Ludwig Sidenmark, University of Toronto, Toronto, Ontario, Canada, lsidenmark@dgp.toronto.edu; Florian Weidner, Lancaster University, Lancaster, United Kingdom, f.weidner@lancaster.ac.uk; Joshua Newn, Lancaster University, Lancaster, United Kingdom, j.newn@lancaster.ac.uk; Hans Gellersen, Lancaster University, Lancaster, United Kingdom and and Aarhus University, Aarhus, Denmark, h.gellersen@lancaster.ac.uk.

Manuscript submitted to ACM

1 INTRODUCTION

Head pointing is the most readily available modality for spatial interaction in head-mounted displays (HMDs). Using the head for pointing affords hands-free input and is more precise than gaze due to users' fine-grained control over their head movement [14]. As the head always points at the centre of an HMD, the state-of-the-art approach is to support head pointing with a cursor fixed in the display centre [26]. Head movement is mapped 1:1 to cursor movement over the 3D scene, and the cursor is rigidly coupled with the display. This has several disadvantages. Selections are only possible in the display centre and users need to move the entire view for selection. The cursor speed is tied to the head rotation speed, which limits efficiency for positioning over larger distances and precision for selecting small targets [9]. Users need to fully turn their heads toward a pointing target, which is not natural as we usually orient our heads only as far as necessary to bring a target into a comfortable eye-in-head viewing range [33]. The movement required to fully align the head with a target increases effort and can be experienced as exaggerated and uncomfortable, particularly with targets high up or low down in a 3D scene [45]. All this compromises the efficiency and ergonomics of the interaction.

This work introduces *HeadShift*, a novel head pointing technique with dynamic control-display (CD) gain. Pointing with dynamic gain is a default behaviour with relative input devices decoupled from the display, such as a mouse or trackpad in graphical user interfaces [6]. CD gain is manipulated based on the speed of the input device to amplify cursor movement in the fast transition toward a target while ensuring fine control over the cursor at the end of the ballistic movement. However, our motivation for using dynamic gain extends beyond the possible optimisation of input speed and accuracy, to improve the ergonomics of head pointing. Our idea is based on the natural coordination of the eye and head in the visual acquisition of targets: The eyes move faster and reach targets first, while the head follows more slowly and not all the way to the target [33]. Accordingly, we consider dynamic gain such that the cursor is controlled by the head but moves faster than the head toward a target, with the head only needing to follow to within a comfortable range for precise positioning of the cursor on the target.

Designing a head pointing technique with dynamic gain is not straightforward. As head movement is tracked as a vector in motor space, it lends itself more intuitively to raycasting with an absolute mapping to the display space. When the mapping is modified by gain, this should conform with the natural relationship between eye and head to preserve the impression of direct pointing. Head movement supports eye saccades and most gaze fixations occur at offsets of 10-15° from the head vector [33] but head pointing becomes uncomfortable and less precise at larger gaze angles [14]. In an HMD we have the additional challenge that head movement already controls the viewport. With a different mapping for pointing than for viewport control, the cursor can move at an offset from the display centre, leading to the problem that offset can accumulate and move the cursor beyond a comfortable range. In light of these challenges, prior work has considered gain only for refinement modes, triggered manually or by target proximity [9, 19].

We designed HeadShift to work without any target assistance based on a non-linear transfer function. We developed the technique iteratively through testing on a Fitts' Law task in an HMD and analysing primary and secondary pointing submovements. Primary submovements are aimed to land the cursor at the centre of the target, but noise in the neuromotor system may cause under- or overshooting and trigger a secondary submovement for correction [23]. Considering this, we designed the transfer function with high- and low-gain states for coarse and fine positioning and modes for moderate versus fast pointer acceleration. We further integrated cursor offset control into the transfer function, to contain the pointer within a comfortable eye-in-head range. Figure 1 illustrates the technique: compared with conventional head pointing, targets can be acquired with less head movement as the cursor moves faster toward the target than the viewport.

We conducted two studies to evaluate performance and usability of HeadShift, compared to conventional head pointing in two different HMDs. In a Fitts' Law experiment, we found no significant difference in movement time but HeadShift had a lower error rate. A usability study on pointing in three different virtual reality (VR) applications showed HeadShift to improve ergonomics in particular for tasks that require vertical pointing, and efficiency for tasks that require high precision. Collectively, the applications demonstrate the generality and effectiveness of the technique, including for pointing over wider ranges vertically and horizontally, and at targets that are dynamically revealed. We have open-sourced the technique[1], including the implementation with a Unity demo scene, experiment data, and scripts for generating figures and statistics.

Although developed and evaluated in VR headsets, HeadShift is adaptable to other types of displays, and the design principle of using gain while ensuring that the cursor remains in comfortable eye-in-head view is generalisable.

2 RELATED WORK

The background to our work comprises head pointing research over decades, insights on head movement in relation to gaze, and prior work on pointing models and transfer functions.

2.1 Head Pointing

Head movement has a long history of study as a pointing modality [17, 25, 29]. Head pointing affords hands-free input at a distance, and is viewed as a natural modality for pointing without a pointing device [1]. Rotation of the head provides a wide input range of over 50° to left or right relative to the body with precise control over movements as small as 0.3° [14]. Head pointing has been used as a proxy for gaze, as we naturally turn towards objects of interest for interaction [27, 32]. In comparison with eye gaze pointing, studies have consistently found head pointing to perform better for precise input as it affords more stable control [2, 19, 28]. Head pointing has been investigated for desktop interfaces [20, 29], mobile devices [40], large displays [27] and smart environments [32] but is most readily available with HMDs where head-tracking is already built-in for viewport control [26]. In this work, we are motivated by the specific challenges of head pointing within a head-referenced display. While we have developed HeadShift in VR, the technique is not limited to HMDs and is transferable to other display environments.

In an HMD, the head always points at the centre of the display. The default for head pointing is thus to align the entire display centrally over the target for selection, with a static cursor in the display centre for guidance [26]. The technique is straightforward as it is based on the same absolute mapping of head rotation to both viewport control and pointing, but the simultaneous use of head movement for view control and input has also been noted as conflicting [26]. The coupling of display and pointer is problematic for performance and usability as users would not normally turn their heads all the way to align a target centrally in their field of view [33, 34]. Head pointing is generally experienced as uncomfortable and fatiguing as it requires more than natural head rotation to align with a target [2]. With a 1:1 mapping of the pointer, there is also no scope for adjusting cursor sensitivity, which limits both speed and accuracy [9]. The rigid coupling of pointer and display has also been found to induce more overshooting when pointing targets are dynamically revealed in an HMD [37]. In spite of evident problems, there has only been limited work on alternative mappings of head movement for pointing in HMDs.

Recent related work proposed a dual-gain head pointing technique with a higher gain mode for the ballistic cursor movement toward a target and a lower gain mode for the final positioning on the target [9]. For horizontal pointing,

[1] https://doi.org/10.5281/zenodo.11368509

fixed gains of 1.8 for coarse and 0.5 for fine positioning were found to improve efficiency without compromising usability. However, the dual-gain technique relies on a priori knowledge of target positions to switch modes when the cursor comes into proximity of the target. This limits the technique's applicability to interfaces specifically designed to support it. Earlier work had also proposed sticky targets to slow a cursor down for head-based selection by users with limited motor control [39]. Other work has proposed explicit switching between default head pointing (gain of 1) and a refinement mode (gain of 0.5), using a manual trigger [19]. There is also work using other modalities as primary input and head movement for refinement in mappings relative to the primary modality [15, 19, 35, 36]. HeadShift, in contrast, modifies cursor gain solely based on properties of the head movement, using a transfer function without any target assistance or explicitly switched modes.

2.2 Head Movement in Relation to Gaze

Head movement has long been used as a proxy for gaze in HMDs, but the relationship with gaze has not been considered further in prior head pointing literature. Past research has described head input techniques as "gaze-directed" [25, 43] and many works refer to the head vector as "head-gaze" [3, 9, 26]. As eye trackers become more widely integrated with headsets, one might expect gaze to replace its proxy, but there are fundamental differences in the control we have over either movement. Eye movement is entirely driven by visual processes that limit the precision of input irrespective of eye-tracker performance, for example, making it impossible to pinpoint or nudge a cursor [44]. Head movement naturally supports eye movement to acquire targets but we can also move our heads independently of visual processes, including for small and precise movements. Recent work suggests distinguishing between "head-gaze" and "head-gestures" as different types of head movement, each of different utility for interaction [14]. Head pointing naturally leverages gaze-driven head movement for cursor positioning toward a target, while gestural head movement enables us to align a cursor more precisely for selection than we would be able to do with gaze.

Recent studies of eye and head movement in HMDs provide more nuanced insight into head movement in relation to gaze. Head movement routinely supports eye saccades to acquire gaze targets but the head moves more slowly than the eyes and only to within 10-20 degrees from the target [33]. This efficient behaviour limits effort and energy expenditure while ensuring that the eyes reach a comfortable eye-in-head position before the next gaze shift [38]. It explains why conventional head pointing gestures are often experienced as exaggerated [2, 45] and inspires us to use dynamic gain for the cursor to move in advance of the head vector. Head pointing relies on gaze guidance of a cursor to align with a target, for which it is important that the cursor remains within a range from the head vector that is comfortably covered by eye-in-head rotation. A study of head-assisted gaze pointing observed that users tended to use an eye-in-head range of up to 20 degrees, but less in the upward direction [33]. This is also reflected in office ergonomics guidance for displays to be arranged with the upper edge at eye height and viewing ranges of 15-20 degrees from the central line of sight [10]. Recent work on head pointing at different gaze angles found a drop in comfort and precision when head-gaze offsets exceeded 30 degrees [14]. Our design of HeadShift reflects the relation of head vector and gaze, both in the use of gain and in the integration of bounding mechanisms that keep the cursor in a comfortable interaction range.

2.3 Pointing Models and Pointer Acceleration

We draw on long-established pointing models to develop and evaluate our technique. Fitts' Law estimates the time required to perform an aimed movement according to target properties (distance and width) and has long been shown to also hold for head pointing [16]. Meyers' optimised initial impulse model describes the aimed movement itself, as composed of a primary ballistic submovement programmed in the neuromotor system to land the pointer at the centre

of the target and a secondary submovement for correction in case the primary submovement over- or undershoots [23]. The model is reflected in the prior design of transfer functions that manipulate CD gain for pointer acceleration, increasing gain during the ballistic phase of the movement but reducing it for fine positioning and any corrective movement [6]. This is achieved by coupling gain to the velocity of the input device in motor space. At the start of an input movement, the pointer is in a low-gain state which may require more movement in motor space than of the cursor on the display, but gain increases rapidly with input speed before dropping back to low-gain at the end of the ballistic movement. The effect of such a dynamic gain transfer function is a dynamic reduction of target distance in the ballistic phase, and a dynamic increase of target width as the pointer approaches the targetin the corrective phase [6].

Head pointing gain and transfer functions have been studied for 2D desktop and handheld displays [17, 20, 31, 40]. In such settings, displays occupy a narrow field of view (FOV) in which content can be comfortably viewed with eye-in-head rotation, without the need for head movement. As the display is fixed in the world, head movement can be adopted for cursor control on the entire screen, including with a limited degree of gain [17, 40]. Head pointing has also been studied for larger displays in the world, where head rotation is necessary for orienting toward targets. A problem in such settings is the increase in target size at oblique viewing angles, addressed in prior work with a transfer function for coarse input by head-gaze combined with manual input on a touchpad [27]. Head pointing in an HMD presents different challenges as the display itself is head-referenced, as a viewport to a surrounding environment. Pointing in this context is generally based on an absolute mapping of a vector tracked in motor space to a ray cast in the display environment. Prior work has proposed transfer functions that modify ray-cast input in subtle ways, with reduced gain for precise pointing carefully compensated during ballistic movement to maintain a perception of absolute input [12, 18]. Our work is similar in how it facilitates more precise input but more liberal in using gain to accelerate the cursor. We are not aiming to make cursor offsets imperceptible but have designed HeadShift for head vector and cursor to move in accordance with our natural experience of offsets between head movement and gaze direction.

3 DESIGN OF HEADSHIFT

The principal idea for the design of HeadShift is to have a head pointer that moves faster toward the target than the head vector itself, to reduce required head movement and enable target selection at a natural gaze offset. We aimed for the technique to afford precise input without any target assistance. We therefore based our technique on the design of a transfer function to modify cursor gain solely based on properties of the head movement.

Methodologically, we used a Fitts' Law task to test performance for different task conditions throughout the development process. As the input is a vector, target widths were controlled as visual angles, including as small as 1° for precise input, while targets were presented at a fixed depth in the 3D scene. The task is described in detail in the next section, as we also used it for summative evaluation for comparison with 1:1 head pointing. For testing and development, we parsed head movement to segment it into primary and secondary submovements in accordance with Meyer's model [23], using smoothed head speed and acceleration magnitudes (cf. Appendix A for a description of the method). This allowed us to optimise the HeadShift transfer function by analysing the effect of design choices on ballistic versus corrective phases of input.

The two key parts of HeadShift developed in the process are (1) the design of the core of the transfer function for pointer acceleration and (2) the integration of cursor offset control into the transfer function to ensure that the cursor remains within an ergonomic eye-in-head movement range. The technique is dependent on the integration of both parts as pointer acceleration alone would result in a pointer prone to cursor drift beyond comfortable viewing range.

3.1 Head pointer acceleration

Pointer acceleration in head pointing is commonly based on sigmoid functions that provide a smooth transfer between low- and high-gain states [17, 27, 40]. We adopted the following function:

$$G(x) = \frac{G_{Max} - G_{Min}}{1 + e^{k(P_{inf} - x)}} + G_{Min} \tag{1}$$

where x is the property of head movement used as input for the transfer function. In pointer acceleration literature [11, 30], transfer functions are generally described as velocity-based with x as the speed of the input device. G_{Min} and G_{Max} are minimum and maximum values that define the CD gain range. k determines the slope of the sigmoid function at $x = P_{inf}$ where P_{inf} is the inflection point of the sigmoid function.

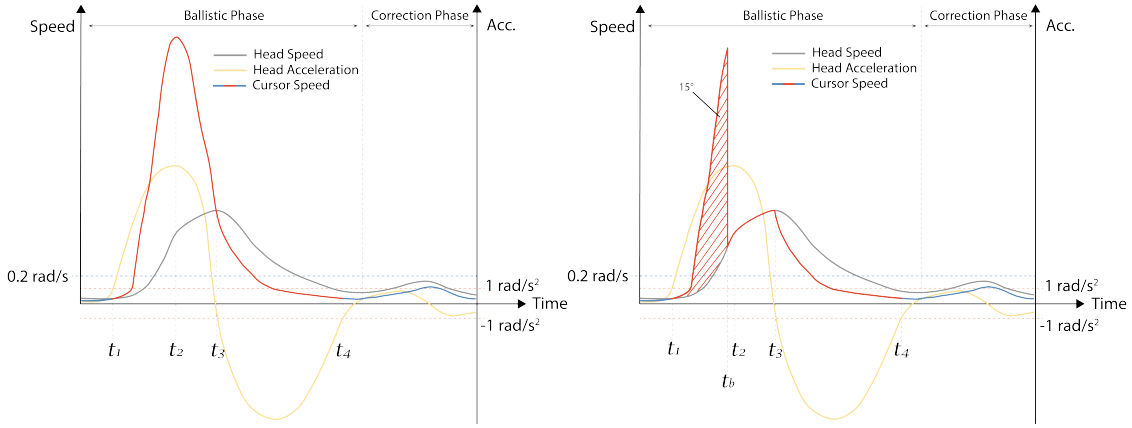
In experimentation, we found it challenging to optimise for faster ballistic movement and high precision solely based on first-order control, i.e. with speed as input. When the cursor was made too sensitive to changes in speed, then it tended to overshoot at the onset of corrective submovements. Lower sensitivity, on the other hand, resulted in the cursor gaining momentum too slowly for efficient coarse positioning. We therefore added second-order control (i.e. acceleration input) into our transfer function

$$G_{core} = \begin{cases} G_{fast}(acc), & \text{if } speed \geq 0.2\ rad/s, \\ & or\ |acc| \geq 1\ rad/s^2 \\ G_{moderate}(speed), & \text{otherwise} \end{cases} \tag{2}$$

where $G_{moderate}$ and G_{fast} define two modes of the transfer function that are automatically switched depending on speed and acceleration thresholds. In moderate mode, the function takes speed as input, obtained as the magnitude of HMD angular velocity, while the fast mode is based on acceleration, calculated as the derivative of speed with respect to time between two frames. In addition to switching order of control, each mode is optimised with different gain parameters, to allow for a higher gain range in fast ballistic movement while capping gain for shorter and corrective movements. The high-gain state for the combined function is defined by the fast mode parameters as the function switches to fast mode before the theoretical high-gain state of the moderate mode can be reached. On the contrary, low-gain states can be reached in both modes, with fast mode parameters determining cursor behaviour during the slowing down of the head in the ballistic phase, and moderate mode parameters controlling cursor behaviour for refinement (c.f. Table 1 for parameters and Section 4 for evaluation of different parameter sets).

Figure 2a illustrates the behaviour of the HeadShift pointer acceleration function. At the start of a ballistic movement, the gain is moderate, but HeadShift switches into fast mode as soon as the acceleration threshold is reached. The speed threshold ensures that HeadShift remains in fast mode when acceleration drops below a threshold at maximum speed. At the end of the ballistic phase, HeadShift switches back to first-order control of the cursor to facilitate cursor refinement in the corrective phase. The mode switching thresholds have been determined in pilot testing to reach fast mode quickly in ballistic movements while ensuring that HeadShift remains in moderate mode during corrective submovements. In either mode, gain is based on unfiltered input for maximum sensitivity, while mode switching is based on filtered data to ensure robustness[2]. Note that not all pointing involves a corrective phase, as the primary submovement may already land the cursor on the target [23]. The HeadShift transfer function supports that by the low gain state of the fast mode, for the cursor speed to drop well below head speed in the latter part of the ballistic phase.

[2]We use 1€ Filter [5], with default parameter values $f_{c_{min}} = 1$ and $\beta = 0$ and sampling frequency 90Hz for all 1€ Filters reported in this work.



(a) Pointer acceleration with HeadShift.

(b) Cursor offset bounding.

Fig. 2. HeadShift pointer behaviour. (a) Initially, cursor gain is first-order controlled by head speed (moderate mode; blue speed curve). Second-order control (fast mode; red speed curve) is activated once a minimum acceleration threshold is reached (t_1). After the head has reached maximum acceleration (t_2), cursor gain reaches a peak and thenrapidly decreases. By the time the head reaches maximum speed (t_3), gain has dropped back to approximately 1 by design. HeadShift remains in fast mode as the head decelerates, for cursor speed to drop faster than head speed, and only switches back to moderate mode when both speed and acceleration fall below threshold at the end of the ballistic phase (t_4). In the corrective phase, gain remains low as corrective movements do not reach fast mode activation thresholds. (b) Cursor offset increases when the cursor moves faster than the head and corresponds to the area between the two curves. In HeadShift, we cap gain at 1 when the offset reaches 15° (t_b). The maximum offset may be reached depending on the amplitude of the movement, and the position of the cursor at the start.

3.2 Cursor Offset Control

As a result of gain, the cursor can move at an offset from the head vector. In contrast to other work on raycast enhancement with dynamic gain [12, 18], we treat offset not as a problem but as desirable for the acquisition of targets with less head movement and at natural gaze angles. However, cursor offsets need to be controlled to ensure that targets can be acquired within comfortable eye-in-head range. Also, with the cursor offset from the head vector at the end of the pointing movement, the next pointing action will start from an offset, and offsets can accumulate over time [11, 12].

As there is no natural clutching mechanism with a head pointer, we constrain cursor movement to within an ergonomic range. We do this by extending our transfer function to cap the gain at 1 when the cursor reaches a maximum offset from the head vector. From the literature, we considered offsets up to 20° [33] and, through pilot testing, established 15° as a range for the cursor to remain naturally associated with head control, and comfortably usable. Figure 2b illustrates the bounding effect: a cursor can accelerate faster than the head until the maximum offset is reached, at which point its speed becomes capped at head speed. In an HMD, this restricts the cursor position to a circular bounding area in the centre of the display.

In pilot testing, we observed an upward bias in cursor offset accumulation. Analysis of head movements revealed asymmetries in vertical movement. Users decelerated later during upward head movement compared to downward movement, which might be caused by the kinematics of the head, the effect of gravity, the weight of the headset or the more limited view in upward gaze [13]. These differences resulted in cursor offset accumulation upwards from the head vector, causing discomfort as it is more straining to gaze upward relative to the head vector than downward. To counter

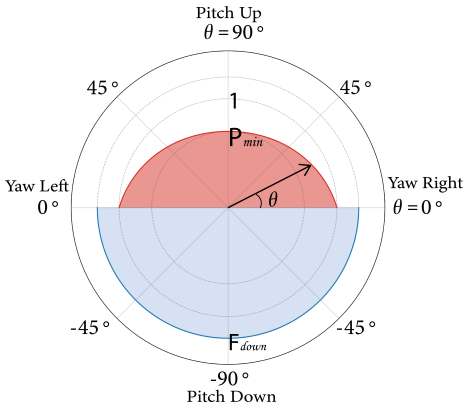


Fig. 3. Adjustment of vertical gain. Head movement downward has a different dynamic compared with upward movement, and cursor gain is adjusted accordingly. For downward movement (blue), gain is scaled at a fixed rate. For upward movement (red), gain is reduced depending on the direction of movement.

the bias in vertical offset, we adjust gain for vertical head movement with a direction-dependent scale factor:

$$F_{vert} = \begin{cases} 1 - (1 - P_{min}) \cdot sin(\theta), & HeadVel_y > 0 \\ F_{down}, & HeadVel_y \leq 0 \end{cases} \tag{3}$$

where $HeadVel_y$ is the head movement velocity along the y-axis in world coordinates (vertical movement, with filtering applied), and a positive $HeadVel_y$ indicates upward head rotation, while a negative means downward head rotation. θ is the angle between the horizontal (xz axis) plane and the direction of movement (filtered). P_{min} ($[0; 1]$) is the minimum proportion of gain applied with upward head movement, and F_{down} ($[1; \infty[$) is the scaling factor for downward head movement. Figure 3 illustrate the dependency of F_{vert} on the direction of head movement. For downward movement, gain is scaled at a fixed rate defined by F_{down}. For upward movement, gain is scaled to a proportion that depends on the angle of head movement, where the scaling factor is lowest when the head rotates straight up and approximates 1 when the movement is near-horizontal.

The vertical adjustment factor is applied to the gain value of the fast mode G_{fast} in pointer acceleration function G_{core} to determine the final gain:

$$G_{final} = \begin{cases} G_{fast}(acc) \times F_{vert}, & \text{if } speed \geq 0.2\ rad/s, \\ & or\ |acc| \geq 1\ rad/s^2 \\ G_{moderate}(speed), & \text{otherwise} \end{cases} \tag{4}$$

Vertical gain adjustment reduces the offset generated when the cursor moves upward and contributes to offset accumulation in the lower part of the bounding area. Note that vertical adjustment of gain is only applied in fast mode. There is no adjustment in moderate mode, to ensure that fine positioning on a target is not biased in any direction.

4 PERFORMANCE EVALUATION

The main objective of our first study was to compare the performance of HeadShift for head pointing in HMDs against the conventional 1:1 mapping of the head vector for input. The conventional technique, in our study, referred to as

Table 1. HeadShift Parameter Values

Technique Variation	Transfer Function								Vertical Correction	
	G_{fast}				$G_{moderate}$				Up	$Down$
	G_{Max}	G_{Min}	P_{inf}	k	G_{Max}	G_{Min}	P_{inf}	k	P_{min}	F_{down}
Low	1.5	0.8	1	1.3	1.2	0.5	0.3	7	0.95	1
Medium	3.85	0.55	4.2	0.4	1.2	0.5	0.3	7	0.9	1.1
High	6.05	0.55	6.5	0.35	1.2	0.5	0.3	7	0.87	1.1

NoGain constitutes the relevant baseline for our work as it is the only available technique that supports head pointing in HMDs generically. We did not consider techniques that rely on target awareness for gain modulation, or use head movement only for refinement. A secondary objective was to evaluate HeadShift performance with different gain levels, to inform the choice of maximum gain for a follow-up study on pointing in different application contexts. For this purpose, we implemented low-, medium-, and high-speed variants of the technique.

We used a 2D Fitts' Law task for our evaluation, and a $4 \times 3 \times 3$ within-subject design with four techniques (NoGain, Low, Medium, High), 3 target amplitudes (7°, 25°, 40°) and 3 target widths (1°, 2°, 4°). Amplitudes were chosen as in prior work on eye-head pointing in HMDs [33], to range from a short distance where targets are in comfortable gaze range before the head movement starts to a larger distance where a head movement has to be underway before the targets come into the comfortable gaze range. Target widths were chosen to include small targets in accordance with prior work on precise pointing in HMDs [19].

The study was conducted with 28 participants (16 male, 12 female) aged 20 – 32 (M = 25, SD = 3.2) recruited from our local university. 17 reported normal vision, and 11 wore glasses or contact lenses. 7 reported no, 20 occasional, and 1 weekly experience with VR. We used an HTC VIVE Pro Eye VR headset for the study, with 110° diagonal FOV, 2880×1600 pixels resolution, and 90 Hz refresh rate. The study task was presented in a VR environment developed with the SteamVR toolkit 2.7.3 in Unity 2020.3 on a computer with an Intel Core i7-8700 CPU, 16 GB RAM, and an NVIDIA GeForce GTX 1080 GPU.

4.1 Techniques

All techniques were based on the head vector of the HMD as input, in the case of NoGain without any transformation. In the HeadShift conditions, filtering was only applied in gain calculation but not to head vector control of the cursor. The implementation of the Low, Medium, and High variants of HeadShift differed in the parameterisation of the fast mode (G_{fast}), while the $G_{moderate}$ was implemented with the same parameters for all three variants (cf. Figure 4 and Table 1). The maximum gain G_{Max} in fast mode was set to 1.5, 3.85, and 6.05, with the lowest value based on recommendations in literature [9] and the largest value based on the maximum we had found usable in pilot testing. G_{Min}, k and P_{inf} were hand-tuned for the fast mode, so that head acceleration in the range from -1 to 1 rad/s^2 would map to output in the range from 0.85 to 1.15 on the slope of the transfer function, which we found optimal for a fast but controlled response of the pointer. Mininum gain G_{Min} for moderate mode was set to 0.5 as in other work on head movement for precise input [19] with G_{Max} set to 1.2 which he found optimal in pilot testing to avoid over- or undershooting. P_{min} and F_{down} were adjusted for each variant so that offsets would tend to accumulate in the lower half of the FOV. See Appendix B for a more detailed description of the parameterisation process.

For all techniques, cursor feedback was implemented as a red dotwith a diameter of 10mm, presented on the task plane at a depth of 1 meter. Accordingly, the cursor size was 0.57° in visual angle.

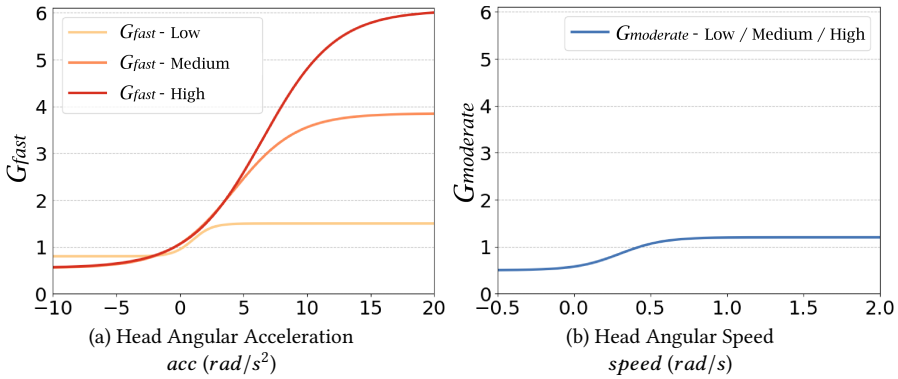


Fig. 4. Low, Medium and High variants of HeadShift. The three variants differ in parameterization of the transfer function in fast mode but use the same transfer function in moderate mode.

4.2 Task

The task was a conventional 2D Fitts' Law pointing task with a circular arrangement of targets, following the ISO 9241-9 standard and adopting presentation detail from prior Fitts' Law work [41]. 11 circular, flat targets were presented one meter away on a dark grey plane in front of the participants. The target plane was fixed in the environment for natural acquisition by gaze. Targets were shown transparent by default, with only one target highlighted at a time, distinguished by a vibrant yellow hue. Participants confirmed the selection of targets with a click on a handheld controller touchpad for a shorter activation distance than the trigger. After participants selected a target, they received visual and audio feedback, and the next target was highlighted in yellow on the opposite side for the next selection. Hovered targets were visually indicated by blue. Successful selections were visually indicated by a change in target colour to green, while a change to red indicated errors. The same audio feedback was given regardless of the success of the selection. The task was completed in one round of 11 selections.

4.3 Procedure

Upon arrival, participants were briefed on the study and completed a consent form and demographics questionnaire. Participants were seated for the study and introduced to the task with a short video. After being fitted with the VR headset, participants calibrated their position relative to the virtual environment.

Participants completed the task with technique order counterbalanced via a balanced Latin square. In each technique condition, participants practised 4 – 8 blocks with 2° of target size and 25° amplitude to familiarise themselves with the cursor speed. Participants were then instructed to perform the task as fast and accurately as possible. Amplitude and width conditions were randomised before each technique, and participants completed the task with two repetitions for each unique combination of width and amplitude. Each task involved 11 selections, of which the first was to initiate the task, with the remaining 10 selections included in the data analysis, resulting in a total of 720 data points per participant. After each technique condition, participants filled in post-condition questionnaires and had a short break before the next technique. At the end of the study, participants completed a post-study questionnaire to rank techniques and comment on their preferences. The study took around 60 minutes.

Performance was measured with the conventional Fitts' Law metrics movement time, throughput, effective width, and error rate (cf. Appendix C for detail on calculation of throughput and effective width). In addition, the amount of head movement was measured as an indicator of effort. The perceived workload was measured with the RAW NASA-TLX questionnaire [8] formatted as a 7 Likert scale. We also measured user's perception of Control, Fatigue, Ease, Precision, and Perceived Offset with a self-defined questionnaire (cf. Appendix D).

4.4 Results

Unless otherwise stated, the analysis was performed with a three-way repeated measures ANOVA (α =.05) with Technique, Amplitude, and Width as independent variables. When the assumption of sphericity was violated, as tested with Mauchly's test, Greenhouse-Geisser corrected values were used in the analysis. Shapiro-Wilk test and QQ plots were used to validate the assumption of normality. ART (Aligned Rank Transform) [42] was applied when normality was violated. Bonferroni-corrected post-hoc tests were used when applicable. Partial eta squares (η_p^2) were used for reporting effect sizes. Statistical results of the three-way repeated measures ANOVA of all dependent variables are shown in Table 6 in Appendix E. Subjective data were analyzed using Friedman tests and Bonferroni-corrected Wilcoxon signed-rank tests in post-hoc tests.

Trials with movement time or trial endpoint distance to the target greater or smaller than 3 standard deviations from the grand mean were discarded as outliers. In total, 391 (1.9%) trials were discarded from the dataset before extracting metric data. After outlier filtering within each evaluation metric during statistical analysis, 74 of the 12,096 cells for the 3-way repeated measures ANOVA were missing and replaced using winsorization.

4.4.1 Movement Time (s). Movement time is defined as the time between the completion of the previous trial and the completion of the current trial. Both correct and erroneous selections are considered (Figure 5). No significant interaction was found between Technique × Amplitude × Width. We found a significant interaction between Technique × Amplitude. Post-hoc analysis showed that NoGain required significantly less movement time than Medium and High (all $p \leq .029$; $\Delta_{\text{NoGain}-\text{Medium}} = -0.07s$, $\Delta_{\text{NoGain}-\text{High}} = -0.07s$) for 7° amplitude. We found a significant interaction between Technique × Width but no significance shown in the post-hoc analysis. We found no significant main effect of Technique, suggesting statistically absence of difference between the techniques regarding overall movement time (M = 1.04, 95% CI [1.02, 1.06]).

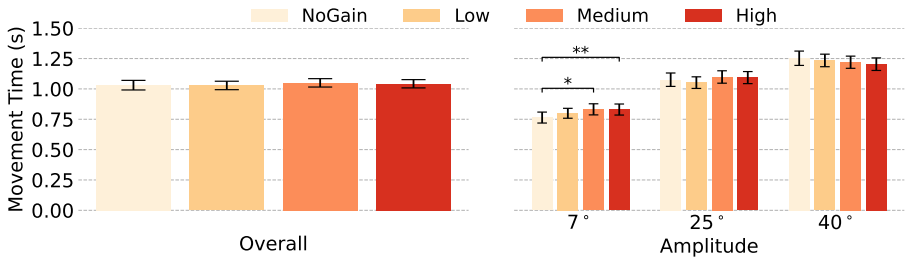


Fig. 5. Movement time in Fitts' Law evaluation. Error bars represent the mean 95% confidence interval.

4.4.2 Throughput ($bits/s$). Throughput is the human motor system transmission rate in the metaphor of Fitts' Law [21] (Figure 6). We found no significant 3-way interaction, but 2-way interactions between Technique × Amplitude and

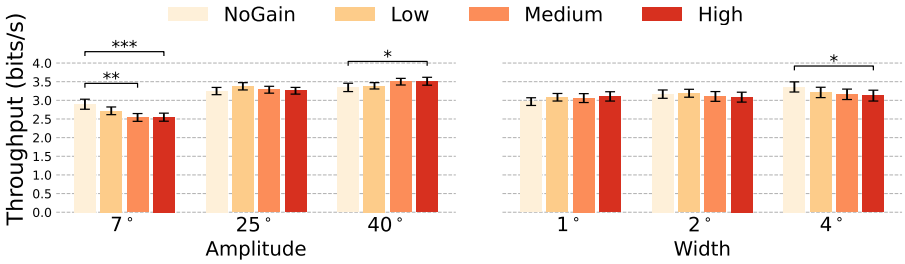


Fig. 6. Throughput in Fitts' Law evaluation. Error bars represent the mean 95% confidence interval.

Technique × Width. Post-hoc tests showed that NoGain had higher throughput than Medium for the target amplitude of 7° (p = .006; $\Delta_{\text{NoGain}-\text{Medium}}$ = 0.36). NoGain also had higher throughput than High for both target amplitude of 7° (p < .001; $\Delta_{\text{NoGain}-\text{High}}$ = 0.35) and target width of 4° (p = .025; $\Delta_{\text{NoGain}-\text{High}}$ = 0.23). On the other hand, High performed higher throughput than NoGain for a target width of 40° (p = .032; $\Delta_{\text{High}-\text{NoGain}}$ = 0.16). However, we found no statistical significant main effect on Techniques (M = 3.13, 95% CI [3.10, 3.17]).

4.4.3 Effective Width (°). The effective target width is the width of the selection endpoint clusters, revealing the pointing precision. The results did not show a 3-way interaction, but a significant 2-way interaction was found for Technique × Amplitude. The estimated means of Medium and High were higher than NoGain and Low for 7° target amplitude (with 0.24° to 0.30° of differences). However, significance only showed between Medium and Low (M_{Low}=2.52, 95% CI [2.25, 2.80]; M_{Medium}=2.81, 95% CI [2.51, 3.11]; p = .026). We did not find a significant 2-way interaction for Technique × Width (cf. Table 2 for complementary data). We found no significant main effects on Technique (M = 2.88, 95% CI [2.80, 2.96]).

Table 2. Mean values and 95% confidence intervals of Effective Width[a] for all target widths.

Technique	Width = 1°	Width = 2°	Width = 4°
NoGain	1.76 (1.64, 1.89)	2.77 (2.60, 2.95)	4.10 (3.85, 4.35)
Low	1.63 (1.48, 1.77)	2.63 (2.44, 2.82)	4.24 (4.05, 4.44)
Medium	1.64 (1.52, 1.76)	2.65 (2.50, 2.81)	4.32 (4.12, 4.51)
High	1.67 (1.54, 1.81)	2.79 (2.63, 2.95)	4.37 (4.18, 4.57)

[a]Mean (95% CI) in degrees (°).

4.4.4 Error Rate. We define an error as the cursor being outside the target when the participant confirms the selection (Figure 7). We found no 3-way interaction. However, the results showed significant 2-way interactions for Technique × Amplitude and Technique × Width. Medium and High had significantly lower error rates than NoGain with relatively large amplitudes (25°, 40°; all $p \leq .007$; cf. Table 3) and relatively small target widths (1°, 2°; all $p \leq .047$; cf. Table 3). At the same time, Low had a significantly lower error rate than NoGain for the medium target amplitude (25°; p = .018) and small target width (1°; p < .001). No significant differences were shown for easier selections: large target size (4°) and small target amplitude (7°). We found a significant main effect of Technique. Low (10.5%), Medium (10.3%), and High (11.1%) had significantly lower error rates than NoGain (14.0%) overall (all $p \leq .002$; M_{NoGain} = 0.14, 95% CI [0.12, 0.16]; M_{Low} = 0.11, 95% CI [0.09, 0.12]; M_{Medium} = 0.10, 95% CI [0.09, 0.12]; M_{High} = 0.11, 95% CI [0.10, 0.12]).

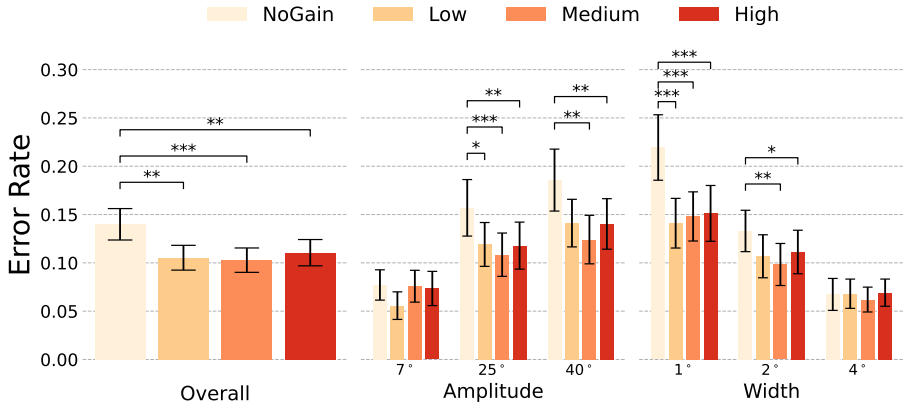


Fig. 7. Error rate in Fitts' Law evaluation. Error bars represent the mean 95% confidence interval.

Table 3. Mean values and 95% confidence intervals of Error Rate[a] for target amplitudes of 25° and 40°, widths of 1° and 2° in Figure 7.

Technique	Amplitude = 25°	Amplitude = 40°	Width = 1°	Width = 2°
NoGain	0.16 (0.13, 0.19)	0.19 (0.15, 0.22)	0.22 (0.19, 0.25)	0.13 (0.11, 0.15)
Low	0.12 (0.10, 0.14)	0.14 (0.12, 0.17)	0.14 (0.12, 0.17)	0.11 (0.08, 0.13)
Medium	0.11 (0.09, 0.13)	0.12 (0.10, 0.15)	0.15 (0.12, 0.17)	0.10 (0.08, 0.12)
High	0.12 (0.09, 0.14)	0.14 (0.11, 0.17)	0.15 (0.12, 0.18)	0.11 (0.09, 0.13)

[a]Mean (95% CI).

4.4.5 Head Movement (°). Head Movement is the accumulated head angular difference in each frame (Figure 8). We found a significant three-way interaction for Technique × Amplitude × Width. Both Medium and High required less head movement than NoGain and Low for target widths of 25° and 40° (all $p < .001$; cf. Table 4). We also observed less head movement with High than Medium for target widths of 1° and 2° with target width of 40°(all $p < .001$; cf. Table 4). Surprisingly, we found Low required more head movement than all the other techniques for the target width of 1° with the target amplitude of 7° (all $p < .001$; $\Delta_{\text{Low}-\text{NoGain}} = 0.98°$, $\Delta_{\text{Low}-\text{Medium}} = 0.89°$, $\Delta_{\text{Low}-\text{High}} = 0.99°$). We found a significant main effect of Technique. Medium (17.73°), and High (17.36°) required significantly less head movement than NoGain (22.44°) and Low (21.87°) overall (all $p < .001$; $M_{\text{NoGain}} = 22.44$, 95% CI [20.67, 24.21]; $M_{\text{Low}} = 21.87$, 95% CI [20.23, 23.51]; $M_{\text{Medium}} = 17.73$, 95% CI [16.47, 18.99]; $M_{\text{High}} = 17.36$, 95% CI [16.15, 18.58]).

Table 4. Mean values and 95% confidence intervals of Head Movement[a] for target amplitudes of 25° and 40° in Figure 8.

Technique	Amplitude = 25°			Amplitude = 40°		
	Width = 1°	Width = 2°	Width = 4°	Width = 1°	Width = 2°	Width = 4°
NoGain	23.19 (21.84, 24.53)	23.56 (22.27, 24.85)	24.39 (23.05, 25.73)	38.36 (36.28, 40.45)	38.55 (36.52, 40.57)	38.92 (36.90, 40.94)
Low	22.32 (21.32, 23.33)	22.74 (21.64, 23.83)	23.14 (22.15, 24.14)	37.46 (35.92, 39.00)	37.00 (35.30, 38.70)	37.13 (35.45, 38.82)
Medium	19.02 (18.06, 19.98)	19.17 (18.25, 20.08)	19.27 (18.32, 20.22)	28.73 (27.46, 30.00)	29.28 (27.90, 30.66)	29.09 (27.84, 30.33)
High	18.89 (17.96, 19.83)	19.02 (17.99, 20.05)	19.53 (18.47, 20.60)	27.57 (26.16, 28.98)	27.54 (26.02, 29.06)	28.63 (27.28, 29.98)

[a]Mean (95% CI) in degrees (°).

4.4.6 Subjective. Friedman tests on RAW NASA-TLX questions did not show significant differences. However, analysis of self-defined usability questions revealed a significant difference in Fatigue ($\chi^2(3) = 8.24, p = .041$). Post-hoc test

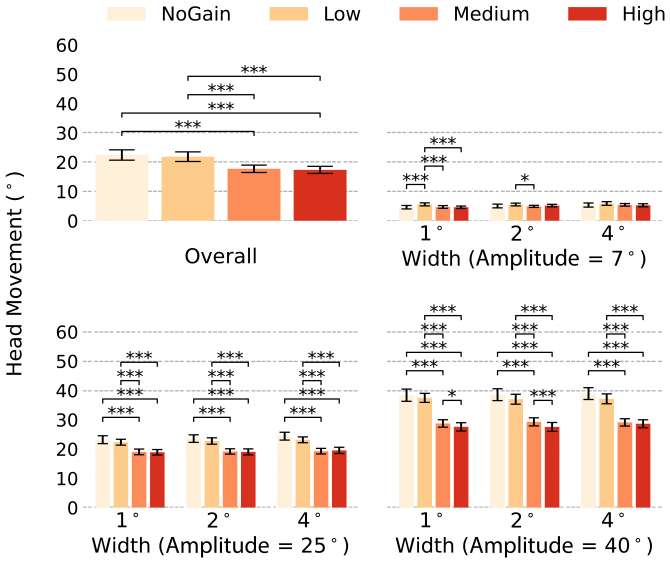


Fig. 8. Head movement in Fitts' Law evaluation. Error bars represent the mean 95% confidence interval.

showed participants rated neck fatigue lower for NoGain (M_{NoGain} = 3.25, 95% CI [2.63, 3.87]) than Low (M_{Low} = 4.04, 95% CI [3.44, 4.63]; p = .038) and High (M_{High} = 3.93, 95% CI [3.23, 4.63]; p = .038). There was no statistically significant preference among techniques. Participants who preferred NoGain and Low felt these techniques were more precise and gave them more control. Other participants preferred High or Medium as they found NoGain and Low too slow, and more tiring as they required more head movement.

4.4.7 *Summary of Key Results.*

- NoGain was faster than Medium and High for short target distances but we found no significant differences in movement time overall.
- High had higher throughput than NoGain for large distance selections, whereas NoGain had higher throughput than Medium and High for selections of short distances and large targets, i.e. tasks with lower difficulty.
- All three variations of HeadShift had a lower error rate than baseline head-pointing. The error rate was lower for selection of smaller targets, and selections over larger distance.
- Medium and High required less head movement than NoGain and Low. High was the best performing of all techniques for selection of small and medium targets over large distance, the two most difficult conditions tested.

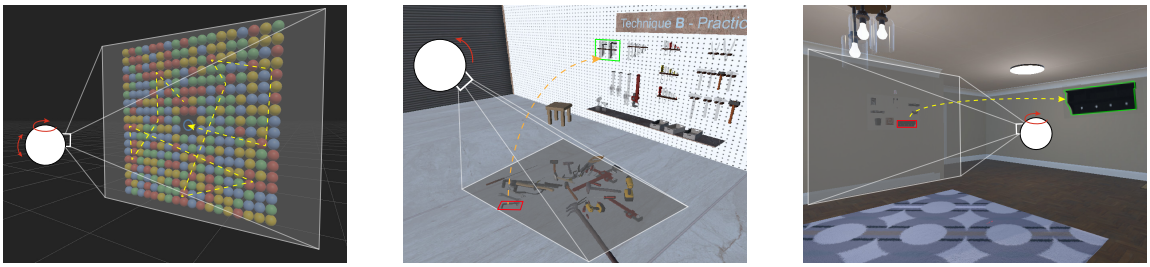
5 USABILITY EVALUATION

We conducted a second study to gain further insight into the usability of HeadShift. The main objective was to compare the usability of High as the most efficient version of our technique with the NoGain baseline on more varied pointing tasks in realistic applications. We also had three specific objectives: (1) to evaluate cursor offset accumulation in free selection on the interface where successive movements are not as regular and predictable as in the Fitts' Law task; (2) to assess the effect of HeadShift in vertical pointing as up and down head movement is comparatively more straining;

and (3) to evaluate usability for precise selection all around, including targets that are not initially in view. We designed three tasks accordingly: Free Selection, Vertical Placement and Horizontal Docking.

This study had 16 participants (8 male, 8 female) aged 20-32 (M = 24.9, SD = 4.43), none of whom participated in the previous study. 8 reported normal vision, and 8 wore glasses or corrective lenses. 3 had no, 12 occasional, and 1 weekly experience of using VR HMDs. We used a Meta Quest Pro (106° × 96°, 90 Hz) with Quest Link. The study environment was developed with the OculusVR toolkit version 53.1 in Unity 2021.3 on a computer with an Intel Core i7-12700 CPU, 16 GB RAM, and an NVIDIA GeForce RTX 3070 Ti GPU.

5.1 Tasks



(a) Free Selection (b) Vertical Placement (c) Horizontal Docking

Fig. 9. Task examples with typical head movement traces are shown in dashed yellow lines. The grey region indicates the FOV. Bubble clustered pseudo-randomly in the grid, which requires free input all over the place with mixed distances and directions (a). Participants pick tools on the ground and position them in the toolbox on the wall, which requires frequent vertical head movements (b). Participants select furniture in the menu and position in the room with a high volume of horizontal head movements (c).

5.1.1 Free Selection. We implemented a bubble shooter game that requires participants to clear a grid of bubbles in front of them. The grid had a size of 18 × 18 bubbles and was placed at 1m depth covering an area of 63° × 63° in the FOV (Figure 9a). Each bubble has one of four colours (red, green, yellow, or blue) and a size of 3.5°. The colour pattern was the same for both techniques (High and NoGain) but mirrored vertically to keep the task difficulty comparable. This task represents free selection across the entire interface as we aimed to evaluate offset accumulation when there is no prescribed order or alternation of movements, such as in the Fitts' Law task.

The gameplay involves two phases. In the first phase, participants generate a bubble with a random colour by pulling the trigger. They then have to find a group of at least two bubbles of the same colour, move the bubble to this group so that they overlap, and release the trigger. With that, the group of bubbles, including the one they had moved, disappears. After 50% of the bubbles are cleared, the game switches to a drag-and-drop phase in which players have to pick up existing bubbles by pulling the trigger and moving them close to a cluster of at least two of the same colour and releasing the trigger to clear the bubbles. When the number of bubbles was below 8, the system allowed participants to generate new bubbles again to help the remaining bubbles. In this situation, the colour of the newly generated bubble will only be chosen from the existing bubbles. Sound feedback was provided every time participants released the trigger.

After every ten bubbles, the application generated a white bubble and played a sound cue to prompt the participant to move the white bubble onto the blue ring displayed in the centre of the grid (Figure 9a), with a tolerance of 1°.

This was done to measure the accumulated offset after every ten interactions. Note that the offset was not reset but continued to accumulate during the entire game.

5.1.2 Vertical Placement. This task is based on a workshop scenario, where participants find 50 tools scattered on the floor that they need to pick up and place on the wall into different areas according to the type of tool (Figure 9b). The tools are positioned in a range of 0.2 to 1.4m away from the participants on the ground, 0.54m wide horizontally. The wall is 2.3m in front of the participant, where tool placement areas are within a vertical range of 0.3m to 1.8m and a horizontal range of 2m. Without CD gain modification, the maximum vertical head range necessary for the task is -78° to +20°, with negative representing downwards and positive representing upwards.

Participants can pick up a tool by pulling and holding the trigger on the controller and moving it to the corresponding area on the wall, highlighted in red. The colour changes to green when the tool is positioned in the right area, and trigger release drops the tool, confirmed by a sound cue. Incorrect placement will reset the tool to its original position. Note the task does not require precision as tools only needed to be aligned with the tool area, not a specific shape.

5.1.3 Horizontal Docking. For this task, we created an application that required participants to arrange objects in a virtual apartment. Unlike the other applications, it required participants to stand and move in the virtual environment, which included movements for search and natural head movements induced by torso turning. Participants were presented with a menu in front of them from which they could pick an object (item of furniture). For most of the objects selected, a designated target area is not initially in view and requires the user to search (Figure 9c). Target locations were distributed all around, covering 360° horizontally.

As in previous tasks, objects could be moved while holding the trigger, but the participant was also free to move in the environment. If participants need to rotate the furniture before placement, they can do so by pressing a button on the controller. Objects needed to be docked accurately, with an alignment tolerance set to 0.1m regarding the object's geometric centre in world coordinates. This required precise movement control close to the docking area, in a range of 0.7° to 3.5°. The outline blinked in green to indicate a match and blinked in red otherwise. Audio feedback indicated successful placement. In total, participants had to place 25 objects.

5.2 Procedure

Upon arrival, participants were briefed on the study and filled out the consent form and demographics questionnaire. After they were fitted with a headset, participants completed the three tasks in series and always in the same order (Free Selection → Vertical Placement → Horizontal Docking), first with one technique and then repeated with the other (techniques were presented in counterbalanced order). The participants practised before each task with the current technique. In Free Selection, participants performed 20 selections during practice, which included 2 times centre alignments (the white bubble). With Vertical Placement and Horizontal Docking, participants practised with 5 objects. After each task, participants completed post-task questionnaires and had a short rest. After all the tasks, we asked the participants to rank the techniques in a post-study questionnaire and explain their choice in a text box. The study took around 60 minutes.

We measured task completion time and cumulative head movement to quantify performance. We also measured accumulated offset during Free Selection to characterise the offset during unconstrained interaction over a large 2D plane. In Vertical Placement, we recorded the range of vertical head movement for each trial as an objective measure of effort. In all tasks, we collected self-reported measures of workload with RAW NASA-TLX [8], exertion with BORG

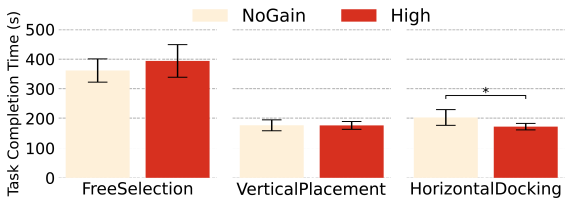


Fig. 10. Task Completion Time in user evaluation. Error bars represent the mean 95% confidence interval.

CR10 [4], and usability with 4 self-defined questions on Control, Comfort, Ease, and Precision (cf. Table 5 in Appendix D), all on a 7-point Likert scales.

5.3 Results

Objective data was analysed with paired-sample t-tests. Shapiro-Wilk tests and QQ plots were used to validate the assumption of normality. When the assumption of normality was violated, Wilcoxon signed-rank tests were used in the analysis. When the distribution of the differences was unsymmetrical, examined with histogram plots, sign tests were used instead. Subjective data was analysed using Wilcoxon signed-rank tests. Sign tests were used instead when the distribution of the differences was unsymmetrical. Standard scores (z-score) are used to report test statistics for Wilcoxon signed-rank and sign tests. While median values were used in Wilcoxon signed-rank or sign tests, we report mean values in all plots for clarity. After outlier filtering (±3 standard deviations of the grand mean per condition) within each metric, 5 of the 224 cells of the objective data and 2 of the 1152 cells of the subjective data were treated with the winsorization procedure.

5.3.1 Task Completion Time. Task completion time is the time interval between the start and completion of the task (Figure 10). The results showed no significance for task completion time in Free Selection and Vertical Placement. However, we found a significant difference in Horizontal Docking, where HIGH was faster than NOGAIN ($t(15) = -2.636, p = .019$; $\Delta_{\text{HIGH}-\text{NOGAIN}} = -30.89s$).

5.3.2 Cumulative Head Movement. Cumulative head movement is the accumulated head angular difference from the task's start to end (Figure 11). We did not find a significant difference for cumulative head movement in Free Selection. However, there was a significant difference between HIGH and NOGAIN in Vertical Placement ($z = -3.516, p < .001$; $\Delta_{\text{HIGH}-\text{NOGAIN}} = -1575.61°$) and Horizontal Docking ($t(15) = -2.309, p = .036$; $\Delta_{\text{HIGH}-\text{NOGAIN}} = -700.79°$). HIGH required less head movement than NOGAIN in these tasks.

5.3.3 Accumulated Offset Distribution. In Free Selection, we measured the cursor offset from the head vector after every ten selections. We did not observe a cumulative pattern in the amount of offset accumulated over time. Figure 12 illustrates the spatial distribution of the combined data points of all participants. Each data point (black dot) represents the cursor's position in the FOV, sampled when the game required alignment of a bubble on the FOV centre (Figure 9a). The transfer function limited offset to a maximum of 15°, but most offsets accumulated well within the bounding area. Along the x-axis, offsets were distributed equally around the centre. Along the y-axis, offsets were clustered in the lower part of the FOV, with only a smaller proportion in the upper part of the FOV.

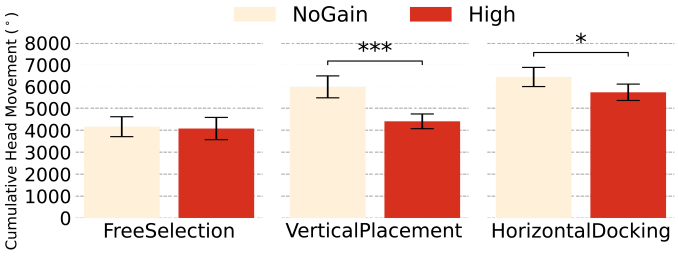


Fig. 11. Head Movement in user evaluation. Error bars represent the mean 95% confidence interval.

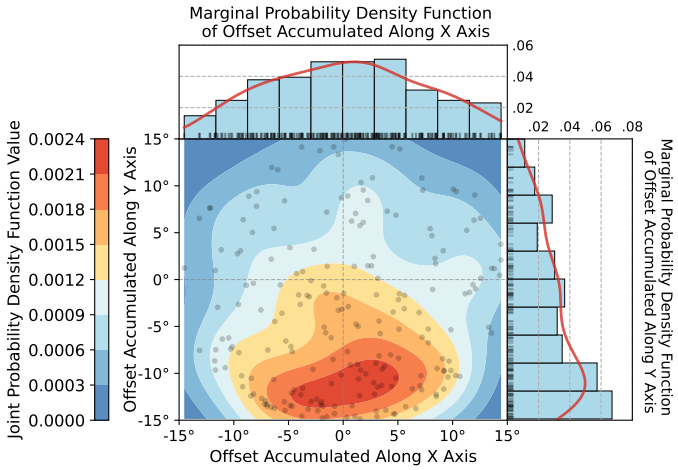


Fig. 12. Distribution of combined accumulated cursor offset from the head vector in the Free Selection task. The offset accumulated centrally with a downward bias, as intended for comfortable viewing. A bounding box prohibited offset from exceeding 15°, but most of the offset accumulated well within the boundary.

5.3.4 Vertical Head Range. In Vertical Placement, we measured the vertical head movement range. Positive values indicate upward tilt relative to the neutral position, and negative values indicate downward tilt (Figure 13). HIGH was effective for task completion within a smaller head movement range. Participants tilted their heads significantly further down when they completed the task with NOGAIN ($p < .001$; $\Delta_{\text{NoGain}-\text{High}} = -14.29°$). There was no significant difference in the upward head angle.

5.3.5 Subjective. RAW NASA-TLX metrics showed significant differences in perceived workload in the Vertical Placement and Horizontal Docking tasks (Figure 14). For Vertical Placement, participants reported lower Effort with HIGH than with NOGAIN ($z = 2.667, p = .004$; $\Delta_{\text{High}-\text{NoGain}} = -0.82$). For Horizontal Docking, Effort ($\Delta_{\text{High}-\text{NoGain}} = -1.56$), Frustration ($\Delta_{\text{High}-\text{NoGain}} = -1.06$), and Overall Score ($\Delta_{\text{High}-\text{NoGain}} = -0.87$) were lower with HIGH (all $z \leq -1.98, p \leq .047$). We found no significance for the Free Selection. While effort was reduced in two tasks, no significant difference was found in the exertion rating (M = 4.09, 95% CI [3.06, 5.13]). The additional usability scales showed significant differences only for Horizontal Docking, where HIGH was rated higher for Control ($\Delta_{\text{High}-\text{NoGain}} = 1.18$) and Ease ($\Delta_{\text{High}-\text{NoGain}} = 1.00$; both $z = 2.10, p = .036$).

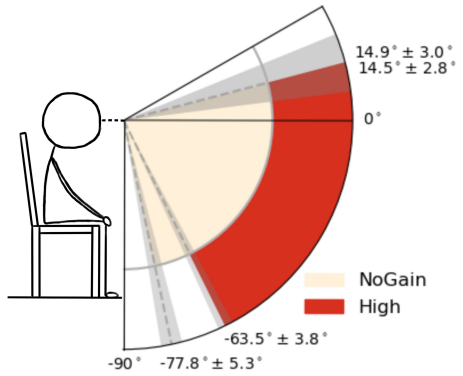


Fig. 13. Average vertical head angle in Vertical Placement with grey region representing 95% confidence interval ranges.

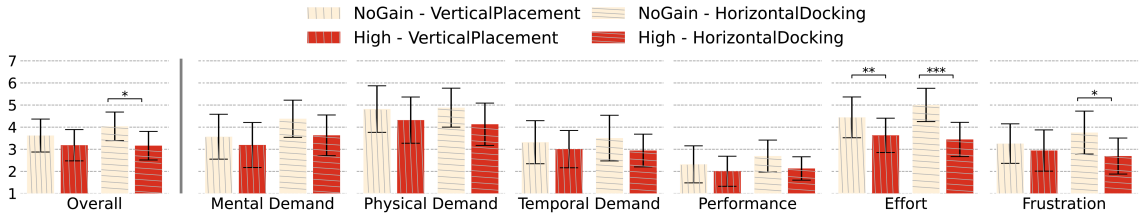


Fig. 14. RAW NASA-TLX scores for Vertical Placement and Horizontal Docking. Error bars representing mean 95% confidence interval ranges.

Overall, 10 out of 16 participants selected High as their preferred technique, most citing that it was easier to use and control. Participants also noted less neck strain (P3) and that "the cursor moves as fast as my eyes" (P15) with High. All participants who preferred NoGain also gave ease and control as reasons. The differences in perceived control may reflect better predictability of the cursor's movement with NoGain versus better control for precise input with High.

5.3.6 *Summary of Key Results.*

- HeadShift was more efficient than conventional head pointing for Horizontal Docking where pointing tasks required high precision, and also rated higher for Ease and Control in these tasks.
- HeadShift resulted in less head movement and effort for tasks that were dominated by movement along one dimension, but this was not the case in Free Selection where successive movements varied more in direction.
- Cursor offsets induced by HeadShift tended to accumulate centrally with a downward bias, as intended by design to maintain the cursor within comfortable viewing positions.
- HeadShift was effective in reducing vertical head movement to a more ergonomic range with less extreme downward movement for the selection of targets on the ground.

6 DISCUSSION

Head movement is so well established for input in HMDs, with a 1:1 mapping in accordance with viewport control, that any suggestion of a different mapping requires careful grounding and analysis. Historically, head movement has been motivated in utilitarian terms, providing a sweet spot for pointing without hands but with more control than

afforded by eye-tracking. In this work, we are strongly influenced by recent work on eye-head coordination of pointing, which has shown that where we point with our heads is not the same as where we naturally look [33]. While head movement approximates gaze, it rarely fully aligns. As a result, we have proposed a dynamic mapping where the cursor is no longer fixed to the head vector but can accelerate ahead toward a target, such that targets can be acquired at a natural offset from the head vector. The design of our HeadShift technique leverages well-known concepts for pointer acceleration, and factors in that eye and head are coupled and that a cursor must be maintained within an ergonomic viewing range.

6.1 Trade-offs in Head Pointing with Dynamic Gain

The comparison of Headshift and baseline head pointing in our two studies provides insight into trade-offs in adopting a dynamic gain transfer function. The clearest benefit of our dynamic gain approach is added precision, as the transfer function ensures that the cursor will be slower than the head at the end of any pointing movement. This increases the target size in motor space irrespective of whether the movement is ballistic or corrective, and provides the user with more fine-grained control. Accordingly, the error rate was lower with HeadShift than with 1:1 head pointing, and throughput was higher for pointing tasks that are more difficult, i.e. when targets are small or more distant. A gain function is typically also expected to reduce movement time by employing high gain in the ballistic phase but this effect is limited by bounding of the cursor for offset control in our transfer function. On the other hand, we think this limitation contributes to limiting overshooting, especially over larger distances. In our studies, users were faster with conventional head pointing when targets were at a short distance, but faster with HeadShift in the Horizontal Docking application which required larger movements and more precise pointing. The cursor bounding majorly affects the ballistic phase and normally presents no limitation to fine positioning where the speed of the cursor is below head.

Cursor offset presents a main factor in the usability of head pointing with gain. From a neutral position with head and cursor aligned, it is natural to acquire a target with cursor gain as long as the offset is bounded to a comfortable viewing range. However, in successive pointing, head and cursor will not remain aligned. In HeadShift, the offset can never exceed 15° and will typically be lower as it becomes reduced during fine positioning with a gain below 1. The offset presents in the direction of the target and if the next target is in the same direction, the cursor will already be closer and require less movement. If the next target is in the opposite direction, the cursor has further to travel but can also accelerate over a larger range before becoming bounded. However, if the next target presents in an orthogonal direction, there will be a discrepancy between the cursor direction versus head direction to the target. With cursor feedback, users are able to compensate for the difference between cursor and viewport movement to a target but this will likely affect performance.

The adoption of gain led to a significant reduction in head movement and perceived effort in two of the applications we tested, Vertical Placement and Horizontal Docking, but not in Free Selection. This is in line with our consideration of cursor offset dynamics, as movements in Vertical Placement and Horizontal Docking were largely along one dimension whereas Free Selection involved pointing in different directions that we presume induced more corrective movements. Gain has the strongest effect when head, cursor and target are aligned in the same direction, and when movements extend over larger ranges. The reduction in head movement presents not only a reduction in effort but also an ergonomic benefit as it facilitates target acquisition with less strain on the neck, in particular for reaching more extreme positions. However, in the Fitts' Law study, subjective rating of neck fatigue was lower for 1:1 pointing than for HeadShift. Participants may have perceived HeadShift as more fatiguing as the cursor movement was less predictable, while all movements occurred within a range where neck strain is comparatively low (±20° from the neutral head position). In

contrast, we saw a clear ergonomic benefit in Vertical Placement, where HeadShift led to selection of objects on the ground from a less straining neck position.

6.2 Implications for Design and Application

Head pointing presents a particular challenge in HMDs as the display itself is head-referenced. A key implication of our work is that a pointer need not be tied rigidly to the display centre, as in existing implementations. Our work demonstrates advantages of a dynamic mapping to increase precision, reduce effort and improve ergonomics. However, adoption of a dynamic mapping introduces a tension with viewport control, as any pointing movement will still shift the viewport but not at the same speed [26]. HeadShift manages this by maintaining a coupling between viewport and pointer, where the relationship is defined by the transfer function. This allows the pointer to be faster than the viewport for coarse positioning and slower for refinement while targets become selectable in a comfortable focal area around the display centre. For larger ballistic pointing movements, HeadShift takes advantage of the head's natural tendency to follow gaze toward the target, consistent with viewport movement.

We designed HeadShift to address the challenges of head pointing in HMDs, but the technique is generic and in principle usable in any context where the head vector is tracked for input on a display. In a conventional display setting, the display does not move with the head, and selections can occur anywhere on the display. Head pointing with gain provides the same advantages with display fixed in the world, for selection with less effort, higher precision and reduced neck rotation range. Use of dynamic gain while ensuring that the cursor remains within comfortable viewing range is generalisable as design principle for head pointing.

HeadShift is not limited in application and in principle usable in any context in which hands-free input is required or desirable, including for accessibility, input at a distance without a controller, or just convenience. Our study results suggest that applications that involve larger movements or require more precision will benefit more from dynamic gain than applications where movements are shorter and targets larger. For applications in 3D, it is significant that HeadShift makes targets attainable with a reduced neck rotation range, for example for use of VR in a seated position, or interaction with wall-size displays in the real world. We saw considerable variation in user performance and preference in our studies, and would expect scope for improvement through personalization of the gain factor in HeadShift. Therefore, for practical application, we would envisage that sensitivity may be adjustable to personal preference, in the same manner as, for example, the mouse in desktop operating systems. For this, we provide recommendations on tuning the transfer function parameters in Appendix B.

6.3 Limitations and Future Work

The performance of HeadShift depends on a range of parameters that we tuned in iterative development of the technique. Our first study provides some insight into the effect of different gain ranges but the contribution of specific parameter choices is harder to assess. A description in Appendix B provides further insight into the parameterisation, and open source code is available for experimentation (https://doi.org/10.5281/zenodo.11368509).

The results reported for the evaluation of HeadShift are specific to the particular parameterisation of the technique. We used different headsets between our two studies and consider our results transferable across devices, although headset weight might influence performance. As noted above, we would not expect our choice of parameters to be a universal fit and the technique would need to be adaptable to individual users to maximise benefit.

We based our first study on the ISO 9241-9 standard. This has a limitation as the cost of missing targets is not accounted for in movement time. The effective movement time is larger when errors occur, which is more likely with

smaller targets [7]. The error rate on small targets was significantly higher with 1:1 pointing than with HeadShift but we did not observe a significant difference in effective target width between the techniques. Our second study provides insight into different applications and includes a variation of sitting versus standing and moving in space but is still limited to a controlled experiment, leaving scope for further study of user experience on less constrained tasks.

The motivation for our development of HeadShift was to overcome disadvantages of the standard 1:1 head pointing technique, while still relying solely on head tracking for input. Our evaluation was therefore limited to comparison with 1:1 head pointing as baseline. However, as we reflect eye-hand coordination in the design of our technique, an alternative is to use eye-tracking as input for coarse positioning and head tracking for refinement [15, 19, 35]. HeadShift has the advantage that it does not require any mode switch for precise input but a gaze+head technique would leverage the natural gain the eyes have over the head.

7 CONCLUSION

Head pointing is compelling for hands-free interaction, in particular with head-worn displays that have head tracking built-in for viewport control. This work presents an investigation of head pointing with dynamic control-display gain to address usability limitations of the conventional use of head input in a 1:1 mapping. The main contribution of the work is HeadShift as a novel technique that supports head pointing with dynamic gain while ensuring that the cursor remains within a comfortable viewing range around the head vector.

The main conclusions from the work are: (1) The main performance benefit is in increased precision of input; this is significant for interaction in 3D where raycast input is limited in accuracy. (2) The technique makes targets selectable at a natural offset from the head; this reduces effort and avoids the exaggerated movement for alignment of the viewport with the target. (3) The technique reduces the overall range of neck rotation required for reaching targets; this improves ergonomics and can help avoid neck strain while increasing the interaction range, for example in seated interaction with 3D environments. In generalisation from the specifics of our technique, we propose use of dynamic gain with cursor control in accordance with natural eye-head coordination as a design principle for head pointing.

ACKNOWLEDGEMENTS

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).

REFERENCES

[1] Rowel Atienza, Ryan Blonna, Maria Isabel Saludares, Joel Casimiro, and Vivencio Fuentes. 2016. Interaction techniques using head gaze for virtual reality. In *2016 IEEE Region 10 Symposium (TENSYMP)*. 110–114. https://doi.org/10.1109/TENCONSpring.2016.7519387

[2] Richard Bates and Howell O Istance. 2003. Why are eye mice unpopular? A detailed comparison of head and eye controlled assistive technology pointing devices. *Universal Access in the Information Society* 2 (2003), 280–290. https://doi.org/10.1007/s10209-003-0053-y

[3] Jonas Blattgerste, Patrick Renner, and Thies Pfeiffer. 2018. Advantages of Eye-gaze over Head-gaze-based Selection in Virtual and Augmented Reality Under Varying Field of Views. In *Proceedings of the Workshop on Communication by Gaze Interaction* (Warsaw, Poland) *(COGAIN '18)*. ACM, New York, NY, USA, Article 1, 9 pages. https://doi.org/10.1145/3206343.3206349

[4] GA Borg. 1982. Psychophysical bases of perceived exertion. *Medicine and science in sports and exercise* 14, 5 (1982), 377—381. http://europepmc.org/abstract/MED/7154893

[5] Géry Casiez, Nicolas Roussel, and Daniel Vogel. 2012. 1 € Filter: A Simple Speed-Based Low-Pass Filter for Noisy Input in Interactive Systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Austin, Texas, USA) *(CHI '12)*. Association for Computing Machinery, New York, NY, USA, 2527–2530. https://doi.org/10.1145/2207676.2208639

[6] Géry Casiez, Daniel Vogel, Ravin Balakrishnan, and Andy Cockburn. 2008. The Impact of Control-Display Gain on User Performance in Pointing Tasks. *Human–Computer Interaction* 23, 3 (2008), 215–250. https://doi.org/10.1080/07370020802278163

[7] Olivier Chapuis and Pierre Dragicevic. 2011. Effects of motor scale, visual scale, and quantization on small target acquisition difficulty. *ACM Trans. Comput.-Hum. Interact.* 18, 3, Article 13 (aug 2011), 32 pages. https://doi.org/10.1145/1993060.1993063

[8] Lacey Colligan, Henry W.W. Potts, Chelsea T. Finn, and Robert A. Sinkin. 2015. Cognitive workload changes for nurses transitioning from a legacy system with paper documentation to a commercial electronic health record. *International Journal of Medical Informatics* 84, 7 (2015), 469–476. https://doi.org/10.1016/j.ijmedinf.2015.03.003

[9] Cheng-Long Deng, Lei Sun, Chu Zhou, and Shu-Guang Kuai. 2023. Dual-Gain Mode of Head-Gaze Interaction Improves the Efficiency of Object Positioning in a 3D Virtual Environment. *International Journal of Human–Computer Interaction* 0, 0 (2023), 1–16. https://doi.org/10.1080/10447318.2023.2223861

[10] Canadian Centre for Occupational Health and Safety (CCOHS). 2022. Office Ergonomics - Positioning the Monitor. https://www.ccohs.ca/oshanswers/ergonomics/office/monitor_positioning.html [Accessed: 2024-02-09].

[11] S. Frees and G.D. Kessler. 2005. Precise and rapid interaction through scaled manipulation in immersive virtual environments. In *IEEE Proceedings. VR 2005. Virtual Reality, 2005.* 99–106. https://doi.org/10.1109/VR.2005.1492759

[12] Luigi Gallo and Aniello Minutolo. 2012. Design and comparative evaluation of Smoothed Pointing: A velocity-oriented remote pointing enhancement technique. *International Journal of Human-Computer Studies* 70, 4 (2012), 287–300. https://doi.org/10.1016/j.ijhcs.2011.12.001

[13] Uwe Gruenefeld, Dag Ennenga, Abdallah El Ali, Wilko Heuten, and Susanne Boll. 2017. EyeSee360: Designing a Visualization Technique for out-of-View Objects in Head-Mounted Augmented Reality. In *Proceedings of the 5th Symposium on Spatial User Interaction* (Brighton, United Kingdom) *(SUI '17)*. Association for Computing Machinery, New York, NY, USA, 109–118. https://doi.org/10.1145/3131277.3132175

[14] 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* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 253, 14 pages. https://doi.org/10.1145/3544548.3581201

[15] 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. *Proc. ACM Hum.-Comput. Interact.* 8, ETRA, Article 227 (may 2024), 20 pages. https://doi.org/10.1145/3655601

[16] Richard J. Jagacinski and Donald L. Monk. 1985. Fitts' Law in Two Dimensions with Hand and Head Movements Movements. *Journal of Motor Behavior* 17, 1 (1985), 77–95. https://doi.org/10.1080/00222895.1985.10735338 PMID: 15140699.

[17] Rick Kjeldsen. 2001. Head gestures for computer control. In *Proceedings IEEE ICCV Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems.* 61–67. https://doi.org/10.1109/RATFG.2001.938911

[18] Werner A. König, Jens Gerken, Stefan Dierdorf, and Harald Reiterer. 2009. Adaptive Pointing – Design and Evaluation of a Precision Enhancing Technique for Absolute Pointing Devices. In *Human-Computer Interaction – INTERACT 2009*, Tom Gross, Jan Gulliksen, Paula Kotzé, Lars Oestreicher, Philippe Palanque, Raquel Oliveira Prates, and Marco Winckler (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 658–671. https://link.springer.com/chapter/10.1007/978-3-642-03655-2_73

[19] 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*. ACM, Montreal QC Canada, 1–14. https://doi.org/10.1145/3173574.3173655

[20] Mei Li Lin, Robert G. Radwin, and Gregg C. Vanderheiden. 1992. Gain effects on performance using a head-controlled computer input device. *Ergonomics* 35, 2 (1992), 159–175. https://doi.org/10.1080/00140139208967804 PMID: 1628609.

[21] I. Scott MacKenzie. 2018. *Fitts' Law.* John Wiley & Sons, Ltd, Chapter 17, 347–370. https://doi.org/10.1002/9781118976005.ch17

[22] Ian Scott MacKenzie et al. 2013. A note on the validity of the Shannon formulation for Fitts' index of difficulty. *Open Journal of Applied Sciences* 3, 06 (2013), 360. https://doi.org/10.4236/ojapps.2013.36046

[23] David E Meyer, Richard A Abrams, Sylvan Kornblum, Charles E Wright, and JE Keith Smith. 1988. Optimality in human motor performance: ideal control of rapid aimed movements. *Psychological review* 95, 3 (1988), 340. https://doi.org/doi/10.1037/0033-295X.95.3.340

[24] Katsumi Minakata, John Paulin Hansen, I. Scott MacKenzie, Per Bækgaard, and Vijay Rajanna. 2019. Pointing by gaze, head, and foot in a head-mounted display. In *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications*. ACM, Denver Colorado, 1–9. https://doi.org/10.1145/3317956.3318150

[25] Mark R Mine. 1995. Virtual environment interaction techniques. *UNC Chapel Hill CS Dept* (1995). https://citeseerx.ist.psu.edu/documentrepid=rep1&type=pdf&doi=69ff1367d0221357a806d3c05df2b787ed90bdb7

[26] Pedro Monteiro, Guilherme Gonçalves, Hugo Coelho, Miguel Melo, and Maximino Bessa. 2021. Hands-free interaction in immersive virtual reality: A systematic review. *IEEE Transactions on Visualization and Computer Graphics* 27, 5 (2021), 2702–2713. https://doi.org/10.1109/TVCG.2021.3067687

[27] Mathieu Nancel, Olivier Chapuis, Emmanuel Pietriga, Xing-Dong Yang, Pourang P. Irani, and Michel Beaudouin-Lafon. 2013. High-Precision Pointing on Large Wall Displays Using Small Handheld Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Paris, France) *(CHI '13)*. Association for Computing Machinery, New York, NY, USA, 831–840. https://doi.org/10.1145/2470654.2470773

[28] Yuan Yuan Qian and Robert J. Teather. 2017. The Eyes Don't Have It: An Empirical Comparison of Head-Based and Eye-Based Selection in Virtual Reality. In *Proceedings of the 5th Symposium on Spatial User Interaction* (Brighton, United Kingdom) *(SUI '17)*. Association for Computing Machinery, New York, NY, USA, 91–98. https://doi.org/10.1145/3131277.3132182

[29] Robert G. Radwin, Gregg C. Vanderheiden, and Mei-Li Lin. 1990. A Method for Evaluating Head-Controlled Computer Input Devices Using Fitts' Law. *Human Factors* 32, 4 (Aug. 1990), 423–438. https://doi.org/10.1177/001872089003200405

[30] Joseph D Rutledge. 1990. Force-to-motion functions for pointing. In *Proc. INTERACT'90: The IFIP Conf. on Human Computer Interaction*.

[31] John A. Schaab, Robert G. Radwin, Gregg C. Vanderheiden, and Per Krogh Hansen. 1996. A Comparison of Two Control-Display Gain Measures for Head-Controlled Computer Input Devices. *Human Factors* 38, 3 (1996), 390–403. https://doi.org/10.1518/001872096778702042 PMID: 8865765.

[32] Marcos Serrano, Barrett Ens, Xing-Dong Yang, and Pourang Irani. 2015. Gluey: Developing a Head-Worn Display Interface to Unify the Interaction Experience in Distributed Display Environments. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services* (Copenhagen, Denmark) *(MobileHCI '15)*. Association for Computing Machinery, New York, NY, USA, 161–171. https://doi.org/10.1145/2785830.2785838

[33] Ludwig Sidenmark and Hans Gellersen. 2019. Eye, Head and Torso Coordination During Gaze Shifts in Virtual Reality. *ACM Trans. Comput.-Hum. Interact.* 27, 1, Article 4 (dec 2019), 40 pages. https://doi.org/10.1145/3361218

[34] 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* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 1161–1174. https://doi.org/10.1145/3332165.3347921

[35] 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* (Stuttgart, Germany) *(ETRA '20 Full Papers)*. Association for Computing Machinery, New York, NY, USA, Article 8, 9 pages. https://doi.org/10.1145/3379155.3391312

[36] 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* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 740, 11 pages. https://doi.org/10.1145/3411764.3445697

[37] Ludwig Sidenmark, Franziska Prummer, Joshua Newn, and Hans Gellersen. 2023. Comparing Gaze, Head and Controller Selection of Dynamically Revealed Targets in Head-Mounted Displays. *IEEE Transactions on Visualization and Computer Graphics* 29, 11 (2023), 4740–4750. https://doi.org/10.1109/TVCG.2023.3320235

[38] John S. Stahl. 2001. Eye-head coordination and the variation of eye-movement accuracy with orbital eccentricity. *Experimental Brain Research* 136, 2 (1 2001), 200–210. https://doi.org/10.1007/s002210000593

[39] Miguel A. Velasco, Alejandro Clemotte, Rafael Raya, Ramón Ceres, and Eduardo Rocon. 2017. A Novel Head Cursor Facilitation Technique for Cerebral Palsy: Functional and Clinical Implications. *Interacting with Computers* 29, 5 (07 2017), 755–766. https://doi.org/10.1093/iwc/iwx009

[40] Simon Voelker, Sebastian Hueber, Christian Corsten, and Christian Remy. 2020. HeadReach: Using Head Tracking to Increase Reachability on Mobile Touch Devices. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376868

[41] Uta Wagner, Mathias N. Lystbæk, Pavel Manakhov, Jens Emil Sloth Grønbæk, Ken Pfeuffer, and Hans Gellersen. 2023. A Fitts' Law Study of Gaze-Hand Alignment for Selection in 3D User Interfaces. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 252, 15 pages. https://doi.org/10.1145/3544548.3581423

[42] Jacob O. Wobbrock, Leah Findlater, Darren Gergle, and James J. Higgins. 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* (Vancouver, BC, Canada) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 143–146. https://doi.org/10.1145/1978942.1978963

[43] Robert C. Zeleznik, Andrew S. Forsberg, and Jürgen P. Schulze. 2005. *Look-That-There: Exploiting Gaze in Virtual Reality Interactions*. Technical Report. Brown University. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=60dc5c21863a73546d0bd980fe9efb140b8c01fa

[44] Shumin Zhai, Carlos Morimoto, and Steven Ihde. 1999. Manual and Gaze Input Cascaded (MAGIC) Pointing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Pittsburgh, Pennsylvania, USA) *(CHI '99)*. ACM, New York, NY, USA, 246–253. https://doi.org/10.1145/302979.303053

[45] Yunxiang Zhang, Kenneth Chen, and Qi Sun. 2023. Toward Optimized VR/AR Ergonomics: Modeling and Predicting User Neck Muscle Contraction. In *ACM SIGGRAPH 2023 Conference Proceedings (SIGGRAPH '23)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3588432.3591495

A MOVEMENT PARSING

Movement parsing is a data-prepossessing process adopted from [23] to separate the movement process into primary and secondary submovements with 1€ Filter smoothed head speed and acceleration magnitudes for testing and development. To determine the start of the primary submovement, we applied a sliding window with size of 5 data frames from the first smoothed speed data frame toward the end in one trial. In each iteration, the mean of the first two values (frames) is calculated and multiplied by 1.1. If all the other three values in the sliding window are greater than this value, the third (middle) frame of the sliding window is regarded as a *candidate frame* for the start of the primary submovement. After iterating throughout all frames in the entire trial, the latest candidate start frame is taken as the starting frame of the primary submovement. If there is no matched candidate frame, the first frame in the trial is selected as the start of the primary submovement.

The end of the primary submovement, which is also the start of the secondary submovement, is calculated by a sliding window of 7 applied to smoothed acceleration data. The first frame of the sliding window is added to the candidate frame list if it is the only negative value in the window. After iteration throughout the trial, all the candidate frames that are located before the first frame of the primary submovement or before the frame of the minimum acceleration value in the trial are dropped from the candidate list. If no candidate is in the list after dropping, the last frame is regarded as the end of the primary submovement and no secondary submovement in this trial. Otherwise, we take the first item in the candidate list and use the second frame after it as the start frame of the secondary submovement. This is because the first frame after is where a positive acceleration occurs, and the second frame after is the frame where the positive acceleration is applied. The start frame of the primary submovement was only used for candidate dropping mentioned above. During the pilot, we always assumed the trial's first frame as the primary submovement's starting frame for simplicity.

We used movement time and amount of head movement during primary and secondary submovement, endpoint distribution and effective width at the end of primary submovement, correction rate (proportion of the trials having secondary submovement) as submovement evaluation metrics in addition to the overall movement evaluation metrics in the study for iterating and fine-tuning the transfer function.

B PARAMETERISATION PROCESS

B.1 Tuning of HeadShift

Here, we share our experience in the pilot testing to help readers and practitioners adapt HeadShift to their needs and circumstances. For tuning G_{fast}, we started with setting G_{Max} values for the overall cursor speed of Low and Medium, and then hand-tuned G_{min}, P_{inf}, and k for optimized gain response and pointing experience in the 2D Fitts' Law task. Specifically, we logged cursor movement data and separated the movement process into primary submovement and secondary submovement with movement parsing (Appendix A). We then analysed the endpoint distribution of the primary submovement for quick evaluation stability and accuracy of G_{fast}. With the endpoint distribution, we derived and optimized the effective target width of the primary submovement, over- and under-shooting rate as our major metrics, trying to keep them at a minimum. We then found tuning the G_{fast} so that the sigmoid output is approximately 0.85 with the input of -1 rad/s^2 and 1.15 with the input of 1 rad/s^2 gave the best balance between the three metrics. With this experience, we tuned G_{fast} of High as our highest technique speed variation.

To make the refinement comparable, we used the same $G_{moderate}$ for all three variations. We tuned with $G_{moderate}$ with a 5-degree selection task, where the cursor is reset to 5° away from a 1° target each trial. We then tuned $G_{moderate}$ for balancing the observed overshoot and undershooting with fast mode deactivated.

In the pilot study, we observed that lateral deceleration in upward head movement moves the cursor 10% more than downward head movement on average. This effect is then amplified by high gain values, which gives an upward bias to the cursor movement within the bounding box, making the eyes fatiguing to see. For balancing the bias, we set the downward gain factor F_{down} to 1.1 in Medium and High. However, we kept F_{down} in Low to 1 because we found the gain output is not high enough to amplify the bias to a significant observable effect.

As participants frequently reported upward eye strain during the pilot study, we also considered applying an upward correction P_{min} to reduce the upward bias caused by the upward lateral deceleration. For this, we slightly further reduced P_{min} from Low to High in accordance with the maximum amplification.

B.2 Tuning Recommendations

In general, we recommend the readers start from on the current HeadShift parameters and only tune the fast mode for cursor speed, and key parameters needed to be tuned are G_{Max}, P_{inf}, and k. First, set G_{Max} for maximum cursor speed with a value higher than 2 to reduce head movement while also keeping it below 7 to remain in control. Higher values are recommended to minimize head movement and potentially achieve faster selection speeds with practice. With G_{Max} settled, tune P_{inf}, and k so that the gain ranges approximately 1 ± 0.15 with head acceleration input from -1 rad/s^2 to 1 rad/s^2, where we found to be the balanced sensitivity for the transition from low to high state in the fast state that will not be either "overreaction" (unexpected speed up, thus introducing more mental load to control) or "impassive" at the initiation of cursor movement (inviting overshoot). G_{Min} can possibly be tuned down for a sharper slowing down process, but still needs to follow the aforementioned tuning rule.

P_{min} and F_{down} should be adjusted for each variant so that offsets would tend to accumulate in the lower half of the FOV. We suggest setting the default values as $P_{min} = 0.9$ and $F_{down} = 1.1$, which should be suitable for general cases. Set F_{down} to 1 if relative low G_{Max} is used (cf. Low) and to 1.1 if relative relative high G_{Max} is used (cf. Medium and High). If eye strain is severely experienced from looking up, or neck strain by frequently pointing with a biased downward head position, we recommend decreasing P_{min} to decrease the cursor upward bias.

When tuning the moderate mode, it's important to consider the potential limitation in our technique, that the theoretical gain range was limited to approximately 0.5 - 0.73, resembling a fixed gain. In this case, we suggest using the same parameter sets for the slow state as we did or simply replacing it with a fixed low gain. As the default parameters of 1€ Filter already showed decent stability in our experiment, we recommend no tuning is needed for filtering in this case.

If the reader considers implementing HeadShift as a customer-tuneable product, we suggest providing two hyper-parameters to the user for tuning the model in commercial devices for head pointing. 1) G_{Max} for cursor speed, and then the system adjusts the fast state sigmoid with suitable P_{inf}, k, and G_{Min} values. 2) a vertical adjustment parameter P_{vert} that controls both P_{min} and F_{down} at the same time for ergonomics, indicating how much downward bias should be added to the cursor.

C FITTS' LAW METRICS CALCULATION

According to the ISO 9241-9 standard, the "Shannon formulation" equation [22] was used for throughput in the performance evaluation study, calculated per block (i.e. 10 selections):

$$TP = \frac{ID_e}{MT} \tag{5}$$

where MT refers to the mean movement time for the block. ID_e is the effective index of difficulty derived by:

$$ID_e = log_2(\frac{A_e}{W_e + 1}) \tag{6}$$

where A_e is the effective movement amplitude from the previous selection point to the centre of the current target. Since we're using serial responses in this task, we calculate A_e by adding the dx values (i.e., the distance between the aiming target centre and the projection of the hit point on the task axis) from both the current trial and the previous trial for all trials after the first (c.f. [21], Figure 17.8). W_e is the effective width, derived with the standard deviation (SD_x) of dx along the task axis in the selection coordinates:

$$W_e = 4.133 \times SD_x \tag{7}$$

where 4.133 is a coefficient for adjusting the target width so that 96% of the hits fall within the target with the assumption of normally distributed selection coordinates [24]. We report W_e in visual angle, converted with:

$$W_e(^\circ) = 2 \cdot \tan^{-1}(\frac{\frac{W_e(m)}{2}}{dist}) \tag{8}$$

where $W_e(m)$ is the effective width in meters calculated with Equation 7 on the target plane, $dist$ is the distance from the headset to the plane at the moment selection is confirmed.

D USABILITY QUESTIONS

Table 5. Usability Questions

Metric	Performance Evaluation (Fitts' Law)	Usability Evaluation (Application)
Control	I felt in control of the cursor.	I felt in control of the cursor.
Fatigue/Comfort	I felt neck discomfort while using the cursor.	I felt comfortable using the cursor.
Ease	It was easy to select targets.	It was easy to use the cursor.
Precision	I could select targets precisely.	I was able to precisely select objects.
Perceived Offset	I could select targets without uncomfortable eye and head positions.	-

E INFERENTIAL STATISTICS

Table 6. Repeated Measures ANOVA Statistical Analysis on the Dependent Variables of the Fitts' Law Study

Variable	ANOVA				Variable	ANOVA			
	Effect	***F* value**	***p***	η_p^2		**Effect**	***F* value**	***p***	η_p^2
Movement Time	T x A x W	F(12, 324) = 1.14	.329	.040	Throughput	T x A x W	F(12, 324) = .93	.520	.033
	T x A	F(6, 162) = 14.74	< .001	.353		T x A	F(6, 162) = 14.61	< .001	.351
	T x W	F(4.24, 114.57) = 3.71	.006	.121		T x W	F(4.14, 111.68) = 3.60	< .001	.175
	A x W	F(4, 108) = 17.99	< .001	.400		A x W	F(4, 108) = 18.04	< .001	.401
	T	F(3, 81) = .56	.640	.020		T	F(2.30, 62.18) = .56	.60	.020
	A	F(1.44, 38.85) = 1309.10	< .001	.980		A	F(2, 54) = 280.85	< .001	.912
	W	F(1.18, 31.94) = 723.59	< .001	.964		W	F(1.34, 36.14) = 10.08	.001	.272
Effective Width	T x A x W	F(12, 324) = 1.32	.208	.046	Error Rate	T x A x W	F(12, 324) = 1.63	.081	.057
	T x A	F(6, 162) = 3.30	.004	.109		T x A	F(6,162) = 4.03	< .001	.130
	T x W	F(4.10, 110.59) = 1.71	.151	.060		T x W	F(6,162) = 8.05	< .001	.230
	A x W	F(4, 108) = 2.12	.083	.073		A x W	F(4,108) = 16.80	< .001	.384
	T	F(2.36, 63.76) = .58	.59	.021		T	F(3, 81) = 11.22	< .001	.293
	A	F(2, 54) = 36.49	< .001	.575		A	F(2, 54) = 100.00	< .001	.787
	W	F(1.30, 34.96) = 922.59	< .001	.972		W	F(1.59, 42.85) = 87.92	< .001	.765
Head Movement	T x A x W	F(6.69, 180.69) = 2.89	.008	.097					
	T	F(1.48, 39.91) = 78.64	< .001	.744					
	A	F(1.04, 28.18) = 2601.32	< .001	.990					
	W	F(1.47, 39.56) = 21.77	< .001	.446					

T = Technique, A = Amplitude, W = Width.

Received 14 March 2024; revised 14 May 2024; accepted 6 August 2024