

# Snap, Pursuit and Gain: Virtual Reality Viewport Control by Gaze

Hock Siang Lee

Lancaster University  
Lancaster, United Kingdom  
h.s.lee3@lancaster.ac.uk

Ludwig Sidenmark

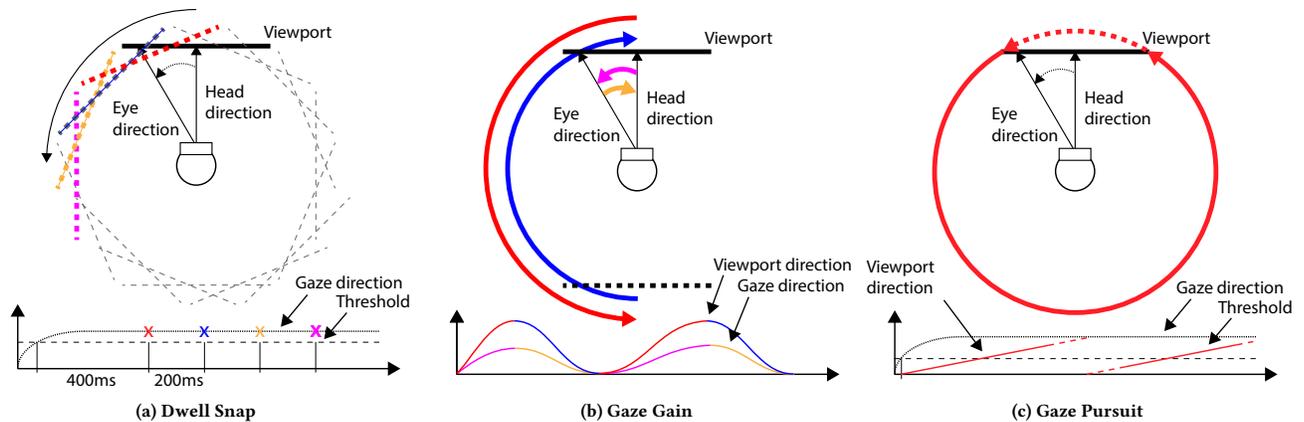
University of Toronto  
Toronto, Ontario, Canada  
lsidenmark@dgp.toronto.edu

Florian Weidner

Lancaster University  
Lancaster, United Kingdom  
f.weidner@lancaster.ac.uk

Hans Gellersen

Lancaster University  
Lancaster, United Kingdom  
Aarhus University  
Aarhus, Denmark  
h.gellersen@lancaster.ac.uk



**Figure 1: Viewport Control with dwell snap, gaze gain, and gaze pursuit.** Dwell Snap is illustrated by Figure (a). It shows the top-down view of a user with the head looking forward and the eyes looking to the left. If the eyes' yaw angle crosses a threshold and stays there for 400 ms (see graph at the bottom), the viewport snaps once in this direction (first "x" in the graph). Follow-up snaps take only 200 ms if the eyes stay in this area (second to fourth "x"). Dashed lines indicate viewport positions. Figure (b) illustrates the principle of Gaze Gain. If the user's eyes yaw into a direction (small magenta arrow), the viewport rotates into this direction with amplified rotation (red arrow). Backward yaw (yellow arrow) leads to the opposite viewport rotation (blue arrow). Gaze and viewport directions are illustrated in the lower part of (b). Figure (c) shows the working principle of Gaze Pursuit. Here, the user's gaze crosses a threshold, and with that, the viewport starts rotating (red arrow) until the gaze returns to the central field of view. The red lines in the graph correspond to the viewport rotation, whereas the black line corresponds to the eye direction

## ABSTRACT

Head-mounted displays let users explore virtual environments through a viewport that is coupled with head movement. In this work, we investigate gaze as an alternative modality for viewport control, enabling exploration of virtual worlds with less head movement. We designed three techniques that leverage gaze based on different eye movements: Dwell Snap for viewport rotation in discrete steps, Gaze Gain for amplified viewport rotation based on

gaze angle, and Gaze Pursuit for central viewport alignment of gaze targets. All three techniques enable 360-degree viewport control through naturally coordinated eye and head movement. We evaluated the techniques in comparison with controller snap and head amplification baselines, for both coarse and precise viewport control, and found them to be as fast and accurate. We observed a high variance in performance which may be attributable to the different degrees to which humans tend to support gaze shifts with head movement.

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

© 2024 Copyright held by the owner/author(s).

This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24)*, May 11–16, 2024, Honolulu, HI, USA, <https://doi.org/10.1145/3613904.3642838>.

## CCS CONCEPTS

• Human-centered computing → Interaction techniques; User studies; Mixed / augmented reality; Virtual reality.

## KEYWORDS

Eye tracking, gaze interaction, eye-head coordination, viewport control, user study, gaze-based interaction, virtual reality

### ACM Reference Format:

Hock Siang Lee, Florian Weidner, Ludwig Sidenmark, and Hans Gellersen. 2024. Snap, Pursuit and Gain: Virtual Reality Viewport Control by Gaze. In *Proceedings of the CHI Conference on Human Factors in Computing Systems (CHI '24)*, May 11–16, 2024, Honolulu, HI, USA. ACM, New York, NY, USA, 14 pages. <https://doi.org/10.1145/3613904.3642838>

## 1 INTRODUCTION

In Virtual Reality (VR), head tracking via head-mounted displays (HMDs) allows users to naturally explore their surroundings by turning their heads. However, users may not always be able to perform such a movement easily or comfortably. This could be due to a posture that prevents them from doing so, for example, lying down [51], due to environmental constraints, for example, in public transport [3], or an injury or disability [53]. While the default viewport control is based on an absolute mapping of head movement to display rotation, alternatives exist. This includes the use of controllers to manually control the viewport, requiring no head movement [27, 35], as well as hands-free techniques that rely on amplification of head movement to move the viewport further with less movement. In this work, we explore gaze as an alternative modality for viewport control.

In considering gaze, we first need to understand how it operates. Gaze represents the direction in which we look and is based on a combination of eye and head movement. Gaze shifts over small amplitudes are possible without head movement, while large gaze shifts necessitate head rotation to bring targets into view. In a range of up to 45° from the head forward direction, gaze shifts are possible without any head movement, although shifts over 15–20° typically involve head movement to keep the eyes within a comfortable eye-in-head position [39]. Any supporting head movement is slower than the movement of the eyes and results in gaze fixations during which the head is still turning toward the target, compensated by vestibulo-ocular reflex (VOR) eye movements in the opposite direction. Other specific eye movement effects occur when gaze fixation is on objects that are in motion in the visual field, triggering smooth pursuit as a closed-loop behaviour to remain on target [54]. For the design of viewport control by gaze, it is important to reflect how natural eye-head coordination translates to HMDs. Gaze shifts are relative to the world and can involve a combination of eye and head movement. As the head is coupled with the display, part of the gaze shift will then result in viewport rotation, and part in a change of gaze angle relative to the display.

We introduce three novel techniques for viewport control by gaze, illustrated in Figure 1: *Dwell Snap*, *Gaze Gain* and *Gaze Pursuit*. All three techniques are designed so that it is possible to control the viewport solely by eye movement, without any head movement. However, head movement is in no way inhibited and can naturally support the gaze. As any head movement supporting gaze is slower than the associated eye movement, this naturally facilitates the use of head movement to refine the viewport orientation, following any larger rotation of the viewport triggered by gaze. Our individual techniques leverage gaze based on different types of eye movement:

- *Dwell Snap* is based on prolonged fixation akin to holding a button down to trigger viewport rotation in discrete steps (Figure 1a).
- *Gaze Gain* is based on saccades and uses a transfer function with gaze angle as input for amplified rotation of the viewport (Figure 1b).
- *Gaze Pursuit* is based on pursuit and rotates the view towards the direction of the user gaze, for the user to follow any target until it becomes centrally aligned in the field of view (FOV) (Figure 1c).

We compared our three new techniques with controller and head amplification baselines on an abstract task adopted from peephole pointing literature [6]. Participants were effective with gaze for viewport control, and performed as efficiently with our three novel techniques as with the existing techniques. The results demonstrate the viability of gaze for viewport control while our techniques provide novel and diverse ways in which surround viewing can be supported. This has practical significance as it enables hands-free camera control with limited or no head movement. Gaze is relevant for accessibility but can also be preferable for viewport control when controllers are available, to focus manual input on other tasks. *Dwell Snap*, *Gaze Gain* and *Gaze Pursuit* each differ in how they enable viewport control, and their respective affordances may support entirely different types of applications, from smooth browsing to fast switching of views.

The core contributions are as follows:

- The implementations of *Dwell Snap*, *Gaze Gain*, and *Gaze Pursuit* as techniques for gaze-based viewport control, including optimizations dealing with characteristics of the human visual system.
- Results demonstrating the applicability and viability of hands-free gaze-based viewport control in abstract tasks, enabling researchers to integrate it into research prototypes and applications immediately.
- Results showing the competitiveness of *Gaze Gain*, *Gaze Pursuit*, and *Dwell Snap* compared to traditional techniques for coarse and precise alignment tasks regarding error rate, viewport alignment time, task load, simulator sickness, and total movement (via eye and head yaw).

## 2 RELATED WORK

This research builds upon previous work on travel in VR, viewport control and gaze-based interaction in head-mounted displays.

### 2.1 Travel in Virtual Reality

Travel refers to the task of moving from the current location to a new target location or in a desired direction and is an essential task in VR [22]. However, since VR is commonly experienced in limited spaces relative to the size of the virtual environment, and the user does not have a view of their physical environment, travel techniques must be intuitive, energy efficient, and efficient in space-constrained settings. As such, developing interactive techniques to address these challenges is an extensive field of research [4, 9, 56]. Travel techniques can vary significantly and have been developed for a variety of modalities, such as controllers [27], head [49], voice [14] or gaze [28]. Techniques that redirect the user into open

physical spaces without the user noticing is also a large field of research to overcome physical restrictions [32, 33, 46].

Travel can be categorized into two actions – *viewport control* and *locomotion* [22]. Viewport control is used to control FOV (in HMDs, commonly by head rotation) to navigate the view in the direction of movement. Locomotion refers to the act of moving in space. Due to the common limitation of space, locomotion has received the most interest from researchers, and it is commonly assumed that users have enough space and the ability to control their view with their heads [9]. However, the range of movement of the head of the user can be limited in certain postures such as sitting [24, 39] or lying [51], or due to other factors such as disabilities or injury [13]. In such cases, users are not able to perform viewport control with the head alone, and support from alternative techniques is needed to give users the full range of motion in the virtual environment. In this work, we investigate how gaze can be leveraged for viewport control with limited head movement.

## 2.2 Viewport Control

Viewport control techniques were originally developed to overcome the small FOV of early head-mounted displays [15]. Early techniques used controllers to allow easy change of the viewport with little head movement [27] and are still popular today to minimise effort and increase navigation efficiency [35]. Hands-free approaches to viewport control have also been proposed and the most common technique is *head amplification* where the movements of the head are amplified to allow the user to reach further with head movement alone than physically possible [15, 21, 30, 34, 48, 57]. Researchers have also investigated systems that automatically control the viewport of the user in storytelling settings [12]. Finally, techniques have been proposed that show the entire environment within the user’s field, either by expanding the user’s field of view with a 360° camera [1], or by overlapping multiple views [36]. While previous techniques have used gaze to steer the direction of locomotion [28], this work is first to investigate eye tracking for viewport control in VR HMDs.

Several works have investigated gaze-based navigation for desktop applications as an accessibility tool or to make navigation more efficient. Researchers have proposed gaze-based one-dimensional scrolling techniques for applications such as websites [18, 25], documents [31, 38] or code navigation [37], and two-dimensional panning techniques and gaze-based panning techniques for 2D applications such as maps [31]. Gaze control has also been proposed for camera control in 3D video games [2, 52, 53]. The most common interaction metaphor for such techniques is that the user gazes at the edge of the screen to move in that direction [18]. Such metaphors are usually also combined with a clutch such as winking [31] to avoid accidental activation. We take inspiration from these desktop-based works in the development of our VR-based techniques.

## 2.3 Gaze Interaction in Head-Mounted Displays

Although novel for viewport control, gaze is widely investigated for pointing in HMDs [19, 29, 40, 47, 55]. Viewport control can be regarded as pointing of the virtual camera, but the tasks differ as pointing is relative to the view for selection on the interface. Gaze pointing at targets that are presented further than 15-20° visual

angle from the display center is typically supported by head movement which in HMDs implicitly aligns the target more centrally in the view [39]. Gaze pointing at targets that are initially outside the display’s FOV presents an interesting case as it requires view panning to reveal the target before it can be acquired, also known as *peephole pointing*. When the same modality is used for both panning and pointing, the task becomes equivalent with pointing of the camera at the target. A peephole pointing task therefore lends itself for evaluation of viewport control, typically with a larger tolerance for alignment in a comfortable viewing area, although more precise control can also be desirable. For this work, we adopt a peephole pointing protocol introduced by Cao et al. [6], to evaluate gaze for coarse as well as precise control of the viewport.

A recent study by Sidenmark et al. compared gaze, head, and hand for peephole pointing in head-mounted displays and is noteworthy as it relied on the same task we use in this work [41]. Their work, however, relied on conventional viewport control by head movement for the initial phase of bringing targets into view. The comparison of gaze, head and hand focused on the second phase of acquiring a target once it has appeared in view, for which gaze and hand outperformed the head due to their independence from display movement. In contrast, we compare gaze, hand and head for viewport control without any separate pointing phase or modality.

## 3 GAZE-BASED VIEWPORT CONTROL TECHNIQUES

In this section, we introduce three new techniques for gaze-based viewport control: Dwell Snap, Gaze Gain, and Gaze Pursuit. In the following, we first outline conventional techniques to control the viewport in VR (*Controller Snap* and *Head Gain*) as the baseline techniques. After that, we explain the three new techniques in detail.

### 3.1 Baseline techniques

**3.1.1 Controller Snap.** Controller Snap is a commonly available technique in many consumer VR experiences. With this technique, a controller rotates the viewport by a fixed amount, triggered via a button press. For example, in SteamVR<sup>1</sup>, the viewport rotates by 45° in the direction of a button press. This rotation happens instantaneously (similar to a teleport). We recreate this technique identically for our study as a baseline technique. Like our techniques, Controller Snap does not require head movement. However, it is not hands-free as ours is.

Dwell Snap is a modified version of Controller Snap. Instead of a button press to trigger a viewport snap, it uses the horizontal eye-gaze angle as a trigger.

**3.1.2 Head Gain.** Head Gain amplifies viewport rotations (cf. Section 2.2). With this technique, the rotation of the user’s head (coupled to the HMD) is amplified so that the visually perceived rotation is larger than the actual head rotation. The amplification factor (gain) is commonly defined as  $g_R := R_{virtual}/R_{real}$  where  $g_R$  is the final gain,  $R_{virtual}$  is the rotation of the virtual camera, and  $R_{real}$  is the rotation of the user’s head (and by that, the HMD) [44]. Contrary to Controller Snap, Head Gain is a hands-free technique.

<sup>1</sup><https://store.steampowered.com/app/250820/SteamVR/>, SteamVR (2023-09-04)

Depending on the gain, it requires relatively small head movements. Still, these movements might not always be possible (e.g., for people with restricted movement or when lying down).

For our study, Head Gain uses a constant gain of 4 in the direction of the HMD rotation. This value was chosen to be consistent with the gain of the Gaze Gain to allow for comparison (cf. Section 3.3).

### 3.2 Dwell Snap

With Dwell Snap (inspired by Controller Snap), the viewport snaps in the direction of the horizontal eye-gaze angle if it is above a fixed angular threshold for a predefined period of time. It keeps snapping as long as the eye gaze remains above the angular threshold. However, the time threshold is reduced for subsequent snaps. Having an initial higher threshold followed by a lower value was added to reduce accidental activation of the snapping while ensuring faster repetition after initial activation. For our study, the horizontal angular threshold for the trigger was set to 25°, as prior work has shown that natural gaze shifts rarely exceed this range [8, 39]. The time threshold for the trigger was set to 400ms for the first snap and 200ms for each subsequent snap if the eye-gaze angle remained above the angular threshold. This means that the user has to look 25° to the left or right for 400ms to activate Dwell Snap and perform the first 22.5° camera rotation. All subsequent 22.5° rotations happen after 200ms each. The camera rotates immediately for each snap, without a smooth transition, providing a similar experience to a teleport. The continuous snapping ends when the participants' eye gaze moves back within the horizontal angular area of [-25°;25°].

When selecting the angle for rotation, we settled on 22.5°. While a larger value is technically possible, we found that with larger values (e.g., 45° as in Controller Snap), users frequently rotated their heads. We suspect this was either the user checking whether they have rotated enough to see the target or to regain orientation. To avoid this issue, we checked 1, 2, 4, 8, and 16 ( $2^n$ ) as divisors of 360° for this issue and selected the fraction of the largest one that did not have this issue ( $360/16=22.5$ ). That means participants need 16 snaps to perform a full 360° viewport rotation if they keep their head still (head movement still contributes to viewport control without gain). As a comparison, Controller Snap requires 8 snaps for a full rotation.

### 3.3 Gaze Gain

Head Gain modifies the rotation of the virtual camera by applying a gain to the horizontal rotation of the user's head (yaw angle). Gaze Gain extends this technique by integrating not only horizontal head rotations but also horizontal eye rotations. Next to that, there is another important difference between Head Gain and Gaze Gain: In Head Gain, the rotation of the viewport is always equal to the rotation calculated via the gain factor and the head rotation. In Gaze Gain, the rotation angle of the viewport is modified by the combined eye and head rotation, a gain factor that amplifies it, and a variable angular velocity.

The introduction of a variable angular velocity was necessary when introducing eye rotation into the technique. This is to deal with three characteristics of the human visual system: Firstly, the extreme difficulty of canceling a saccade mid-movement [17]. Secondly, the fact that humans can only smooth pursuit up to a certain

speed [26]. Thirdly, the reflexive nature of the eyes to saccade, fixate, and follow points of interest (POI) before saccading back to their original position as the viewport rotates (similar to an optokinetic response [20]). Without the variable angular velocity limiting the viewport velocity, the scene will visually move too rapidly. This causes saccades to miss their landing frequently, as the landing spot of the saccade would have visually moved. In this case, the human eye would have to catch up during smooth pursuit, introducing additional saccades ("catch-up saccades"). Additionally, the third point would lead to the viewport movement being uncontrollable and jerky (or perceived stuttering of the viewport), aggravating cybersickness and disorientation [42, 43].

To deal with these challenges while also keeping the time needed for a viewport alignment short, we can formulate three objectives for Gaze Gain:

- (1) To minimize the magnitude of viewport motion during saccades.
- (2) To minimize the jerkiness of the viewport motion.
- (3) To minimize the duration of viewport rotation.

We approach these challenges by modifying the angular velocity and integrating it into calculating the new viewport angle. To do this, we calculate a new velocity  $\omega$  for every frame as follows:

$$\omega = \theta \times (v_{near} + (v_{far} \times v_{near}) \times \max(0, \min(1, \frac{|\theta| - I_{near}}{I_{far} - I_{near}}))) \div \delta t \quad (1)$$

where:

- $\theta$  = unmodified new viewport angle  $\alpha$  – current viewport angle  $\beta$
- $I$  = tuning parameter determining inflexion points
- $v$  = tuning parameter determining the rotational speed at the respective inflexion point
- $\delta t$  = time elapsed between the previous and current frame

The unmodified absolute new viewport angle  $\alpha$  is the viewport angle we would get if we did not consider viewport velocity, calculated with:

$$\alpha = gain \times (yaw_{head} + yaw_{eye}) \quad (2)$$

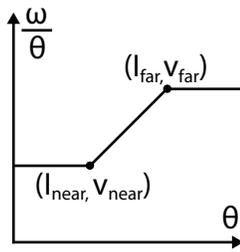
Regarding  $\theta$ , the current viewport angle  $\beta$  is the direction in which the viewport is currently oriented. With that,  $\theta$  is the viewport displacement without velocity modification.

The new  $\omega$  is then used together to calculate the final new viewport angle per frame:

$$yaw_{viewport} += \omega \times \delta t \quad (3)$$

where  $\delta t$  is the time elapsed since the last frame.

Equation 1 and the meaning of the tuning parameters are visualized in Figure 2.  $I_{near}$  and  $I_{far}$  are the angular thresholds in which  $\theta$  would lead to an angle "in view" or "out of view" (note:  $\theta$  is the difference between the current and the unmodified new viewport angle; "view" means the current visual FOV). If  $\theta$  is smaller than  $I_{near}$ , we consider it to be in view. Vice versa, if  $\theta$  is larger than  $I_{far}$ , we consider it to be out of view.  $v_{near}$  and  $v_{far}$  are the angular speeds when  $\theta$  is "in view" and "out of view", respectively. For all values between  $I_{near}$  and  $I_{far}$ , we use linear interpolation between  $v_{near}$  and  $v_{far}$ , to reduce jerkiness.



**Figure 2: Schematic illustration of the angular velocity for Gaze Gain.**

With this, we can achieve our objectives (1) – (3): When  $\theta$  is “in view”, viewport rotation is slow but upper bounded by smooth pursuit speed  $v_{near}$ . However, when the  $\theta$  is “out of view”, the camera may rotate as fast as it needs, using  $v_{far}$ , minimizing the duration of the rotation. In between, we calculate  $v$  by linearly interpolating between  $v_{near}$  and  $v_{far}$ , which reduces perceived jerkiness. That is because the minimal angular scene displacement detectable by humans increases linearly with saccade length (cf. Li and Matin [23]), and during development, we found it to be similar to perceived jerkiness.

Informal pretests during development showed that  $0.25s^{-1}$ ,  $1.6s^{-1}$ ,  $60^\circ$  and  $100^\circ$  for  $v_{near}$ ,  $v_{far}$ ,  $I_{near}$ , and  $I_{far}$  lead to good performance, good user experience, and low cybersickness.

Finally, the gain was chosen to be 4, as it is the smallest integer that allows looking directly behind without any head movement. Note, to allow for a proper comparison, we used the same gain value in Head Gain (cf. Section 3.1.2).

Together, this means that on every frame, we calculate a viewport velocity based on the remaining amount the viewport has to rotate to reach 4 times the combined head and eye yaw angle.

### 3.4 Gaze Pursuit

Gaze Pursuit, inspired by Zhang et al. [59], continuously rotates the viewport in the direction of the eye in the horizontal direction (yaw). When an object comes into view, and the user wants to stop (a point of interest), they naturally fixate on the point and smooth pursuit to the centre. In simple terms, that means that if the user looks with their eye to the left, the viewport rotates to the right until the user stops looking to the left but at the centre.

To avoid accidental rotations, we implemented a deadzone of  $[-5^\circ; +5^\circ]$  in the centre of the FOV (to cover the paracentral field of vision,  $5^\circ - 9^\circ$  [45]). This concept is similar to analogue joystick deadzones<sup>2</sup> and also similar to the threshold in Dwell Snap (cf. Section 3.2). Within this area, eye movements do not lead to viewport rotations. The deadzone serves two primary purposes: First, to establish a neutral position that induces no viewport rotation and allows for a scene exploration. Second, to deal with the inaccuracy of the eye-tracker, which causes, despite calibration, a small offset between where the user is actually looking and where the eye-tracking believes the user is looking.

Similar to Dwell Snap, head movements still contribute to viewport control but without gain. Similar to Gaze Gain, the velocity of the viewport rotation is adapted to every frame and then used to calculate the new viewport angle. Inspired by previous work [59], we chose a linear velocity function to calculate the viewport velocity  $\omega$ , calculated per frame as follows:

$$\omega = k \times (|\theta| - d) \times \text{sgn}(\theta) \div \delta t \quad (4)$$

where:

- $\omega$  = viewport angular velocity
- $k$  = tuning parameter for gradient
- $\theta$  =  $\text{yaw}_{eye} - \text{yaw}_{head}$
- $d$  = dead-zone radius
- $\delta t$  = time between the previous and current frame

With that, the larger the difference between the eye’s and head’s yaw angles per frame, the larger the velocity<sup>3</sup>.

For our study,  $k$  was set to 5. During development, we found that higher values for  $k$  increase perceived jerkiness, whereas smaller movements lead to viewport movements that are too slow to be comfortable.

### 3.5 Implementation details

Current eye-tracking technology in consumer-available HMDs outputs noisy data — especially while looking near the edges of the HMD or during blinks. In such cases, the eye tracker will often experience decreased accuracy or lost-of-tracking. To remedy this, all eye-tracking data is de-noised via a 1€ Filter [7] before being used by any technique. Following the tuning method by Casiez et al. [7], the derivative’s filter frequency,  $f_{min}$ , beta, and cutoff were set at 90, 0.05, 10, and 1, respectively. Temporary loss-of-tracking is handled by using the last known position of the eye. If eye-tracking were lost for an extended period of time ( $>2s$  in the study), all techniques assume whichever yaw angle of the eye reduces camera motion induced by the eye.

## 4 EVALUATION

Our study aims to measure how fast and accurately a participant may align the viewport to point in a particular direction with our techniques. We are also interested in the performance of our techniques compared to existing baselines. This includes errors during alignment, cumulative eye and head angle, and task completion time.

### 4.1 Study Design

To accommodate the large variety of scenarios that require viewport alignment, we use an abstract task that entails fine and coarse viewport alignment towards both an initially known and an unknown direction, separated by a range of possible angles (or amplitudes). To do this, we modified the task of Cao et al. [6]. In our task, participants sit in an empty, plain room (in VR) and have to rotate the viewport to bring a pillar (the target) into view. In our case, for coarse alignment, they have to rotate the viewport to bring the pillar within  $20^\circ$  of the FOV’s centre. For precise alignment, they

<sup>2</sup><https://www.playstation.com/en-gb/support/hardware/set-up-edge-controller/#deadzone>, 2023-09-13

<sup>3</sup> $\text{sgn}(x)$  is the sign function. If  $x < 0$ , it returns -1. If  $x = 0$ , it returns 0 and if  $x > 0$ , it returns 1.

have to rotate the viewport to align the pillar precisely with the centre of the viewport (cf. detail in Section 4.2).

With this task, we performed a 4-factor within-subjects study. The independent variables (IV) are as follows:

- IV1: Viewport-Control Technique: {Controller Snap, Head Gain, Dwell Snap, Gaze Gain, Gaze Pursuit}
- IV2: Level of Control: {Fine-grained, Coarse-grained}
- IV3: Prior Knowledge (PK) of target: {Yes (PK), No (NoPK)}
- IV4: Amplitude between targets: {15°, 30°, 45°, 60°, 120°, 180°}

For each viewport control technique (IV1), participants experience a set of trials. A trial contains two alignments (one without prior knowledge and one with prior knowledge). Each set contains 6 amplitudes between targets  $\times$  2 directions = 12 trials, and each participant will experience 2 levels of control  $\times$  5 viewport-control techniques = 10 sets. This results in a total of 10 sets  $\times$  12 trials in each set = 120 alignments. These 120 alignments form one block. The whole experiment contains 3 blocks, requiring 360 alignments per participant.

## 4.2 Task description

Figure 3 outlines the procedure of one trial for the coarse alignment task (IV2). Before each trial, the participant will align their torso, head gaze, and eye gaze towards a target positioned in front of them (Figure 3A). This is so the data gathered from each trial have the same initial starting direction. Then, for each trial, the participant will perform three consecutive viewport alignments, with the second and third being used for data analysis. First, the participant searches for the starting target placed clockwise or counter-clockwise of the initial forward direction (Figure 3B) and aligns the viewport so the target is within  $[+20^\circ; -20^\circ]$  of the centre of the FOV and confirms alignment with a button press on the controller. If the viewport is aligned correctly, the target is highlighted with a white border shortly (Figure 3C) and disappears. If the alignment is wrong and the button is pressed, the target is highlighted red, and the participant has to realign until the alignment is correct. Next, the participant has to search for the second target in the opposite direction of the start target and align the viewport (Figure 3D; no prior knowledge of the target location, IV3). Finally, participants have to return to the *same* position of the start target, which completes the trial (Figure 3E, with prior knowledge of the target location, IV3). Each viewport alignment (Figure 3A, C, D, and E) is confirmed by pressing a button on the controller. The angular distance between the start target and the second target (red and green in Figure 3C) is defined by our IV4 *amplitude*. All targets are positioned 3 meters away from the user and appear as cylinders with a width of 2° visual angle. The targets start at the floor and are 8 m tall (same as the room). The participant is positioned so that the virtual camera is at a height of 4m.

For the precise alignment task, the second level of IV2, the level of control, the procedure is the same as in the coarse alignment task. Here, the difference is that the alignment is counted as correct if the centre of the FOV is over the pillar.

## 4.3 Procedure

At the start of the study, the experimenter welcomed the participant and explained the purpose and procedure of the study. After providing consent, participants filled out a demographics questionnaire asking for age, gender, vision, experience with video games (never, rarely, monthly, weekly, daily), experience with VR (never, rarely, monthly, weekly, daily), and experience with eye tracking (never, rarely, monthly, weekly, daily). For the study, participants were seated on a stationary seat (no wheels, no rotation) and adjusted the VR headset before completing an eye-tracking calibration. As the study commenced, participants performed all task conditions with one technique at a time before moving to the next technique. The order of techniques each participant experienced was counter-balanced via a balanced Latin square.

Before each technique, participants received an explanation of the technique from the experimenter and were allowed to practice the technique by performing a set of 4 coarse alignments as many times as they wished. After each set of 4, the experimenter prompted the participant if they wished to continue practising or start the experiment. Most participants chose not to continue practising, with a rare few choosing to practice an additional time. No participants chose to continue practising more than once. After each set of selection sequences, participants took off the VR HMD and completed questionnaires. Before starting the next selection sequence, participants were asked if they wished to have a break. After completing all task conditions with one technique participants were asked if they had any opinions about the technique. The study lasted under 90 minutes.

## 4.4 Apparatus

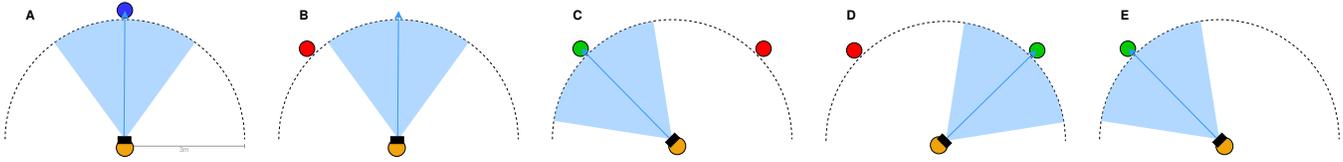
We implemented the scenarios and the individual techniques with Unity 2021.3.14f1. We used an HTC VIVE Pro Eye VR headset for the study, with 110° diagonal FOV, 2880 $\times$ 1600 pixels resolution, and 90 Hz refresh rate on a computer with an Intel Core i7-12700 CPU, 16 GB RAM, and an NVIDIA GeForce RTX 3070 Ti GPU.

## 4.5 Measures

With our study, we are primarily interested in how well participants are able to control the viewport in fine and coarse-grained alignment tasks. To quantify the effectiveness of our technique, we measure the following:

- Cumulative head yaw angle per alignment
- Cumulative eye yaw angle per alignment
- Task completion time: The time between the start of an alignment and the final successful confirmation of the alignment.
- Error rate: Participants must align the viewport and confirm alignment. If they confirm without correct alignment, we record this trial as an error.

These measures provide us with objective results about the effectiveness and accuracy of our techniques. In addition, we ask participants to fill out the RAW NASA-TLX [5] to get subjective data on the taskload. Finally, we measure cybersickness using the simulator sickness questionnaire (SSQ) [16].



**Figure 3: Schematic overview of the alignment task. A: Participants first perform an initial alignment. B: The participant searches for the start target that is randomly located at their left. C: The participant has aligned the viewport and confirms. D: The participant selects the second target. E: The participant selects the first target again, which ends the task sequence. The target in E is at the same location as the target in C and the user has to rotate the viewport to select it again.**

## 4.6 Participants

We recruited 21 participants. 1 dropped out early. In the end,  $N = 20$  participants participated in our study (10 identified as male and 10 as female). All were recruited from the local university. On average, participants were  $M = 27.2$  years old ( $SD = 7.6$ ;  $min = 19$ ,  $max = 53$ ). 10 had normal vision, 2 had corrected to normal vision with contact lenses, and 8 with glasses. 9 stated to play video games very rarely, 3 each reported to play monthly and weekly, 5 play video games daily, 2 reported no experience, 15 rare, 3 monthly usage of VR. Regarding eye tracking experience, 11 reported no, 7 rare, 2 monthly usage. Participants received a 10 GBP Amazon voucher as compensation for their participation. One participant reported slight astigmatism but could perceive everything in VR clearly and one participant self-reported a diagnosis of autism. The university's Institutional Review Board approved the study.

## 5 RESULTS

Unless otherwise stated, the analysis was performed with a four-way repeated measures ANOVA ( $\alpha = .05$ ) with Technique, Prior Knowledge, Level of Control, and Amplitude 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) [58] was applied when normality was violated. Bonferroni-corrected post-hoc tests were used when applicable. Partial eta squares ( $\eta_p^2$ ) were used to report effect sizes. For error rate, data points were classified as outliers if they were larger or smaller than  $2 \times SD$  of the group mean and corrected using winsorization [10]. For cumulative head yaw angle, cumulative eye yaw angle, and task completion time, we performed outlier removal based on amplitude groups following the same procedure as larger amplitudes naturally lead to longer travel time, which increases all three measures systematically. Subjective data was analyzed with Friedman tests and Bonferroni-corrected Wilcoxon signed-rank tests as post-hoc tests.

### 5.1 Error Rate

Figure 4 shows an overview of error rate grouped by Technique, Amplitude, and Task, and Table 1 detail for each technique by level of control. Mean error rates were low for coarse control, including with gaze-based techniques (2.1%, 1.0% and 0.1% with Dwell, Gain, and Pursuit), and moderately higher for precise control (mean error 6.4%, 11.4% and 6.1% with Dwell, Gain and Pursuit), however in

Technique	Coarse	Precise
Controller Snap	$M=0.031$ ( $SD=0.172$ )	$M=0.081$ ( $SD=0.272$ )
Head Gain	$M=0.014$ ( $SD=0.117$ )	$M=0.167$ ( $SD=0.373$ )
Dwell Snap	$M=0.021$ ( $SD=0.143$ )	$M=0.064$ ( $SD=0.245$ )
Gaze Gain	$M=0.010$ ( $SD=0.098$ )	$M=0.114$ ( $SD=0.318$ )
Gaze Pursuit	$M=0.001$ ( $SD=0.037$ )	$M=0.061$ ( $SD=0.240$ )

**Table 1: Mean error rates and standard deviation of each technique by level of control.**

all cases with high variance. Analysis shows a significant four-way interaction  $F(8,389,159.386) = 5.443$ ,  $p < 0.001$ ,  $\eta_p^2 = 0.223$ ). In pairwise comparisons for each technique, we found higher error rates for precise than for coarse control for specific conditions: with Controller Snap, for tasks with prior knowledge and amplitudes  $30^\circ$  and  $60^\circ$  (all  $p < 0.043$ ); With Head Gain, for all conditions ( $p < 0.049$ ) except the  $15^\circ$  NoPK ( $p = 0.096$ ) and  $60^\circ$  NoPK ( $p = 0.065$ ); with Dwell Snap for low amplitude conditions without prior knowledge ( $15^\circ$ ,  $30^\circ$ , and  $45^\circ$ ) (all  $p = 0.021$ ); with Gaze Gain, for the high ( $180^\circ$ ) and the low amplitudes ( $15^\circ$ ,  $30^\circ$ ) regardless of prior knowledge (all  $p < 0.042$ ); and with Gaze Pursuit, for the larger amplitude conditions ( $120^\circ$ ,  $180^\circ$ ) irrespective of prior knowledge (all  $p < 0.043$ ).

### 5.2 Cumulative Head Yaw Angle

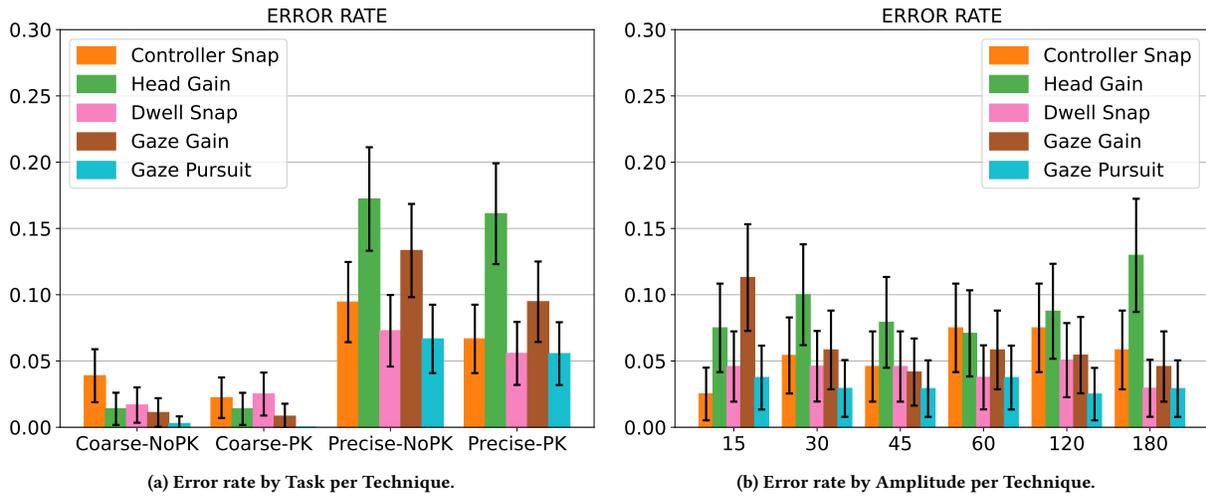
We found no significant interaction effects on the cumulative head yaw angle (all tests,  $p > .171$ ,  $\eta_p^2 < .077$ ), indicating that the head yaw angle did not differ between factor combinations. However, we found a significant main effect of amplitude ( $F(1,323,25.135) = 6.060$ ,  $p = .014$ ,  $\eta_p^2 = .242$ ). Post-hoc tests indicated that every combination of amplitudes significantly differs ( $p < 0.01$ , Figure 5).

### 5.3 Cumulative Eye Yaw Angle

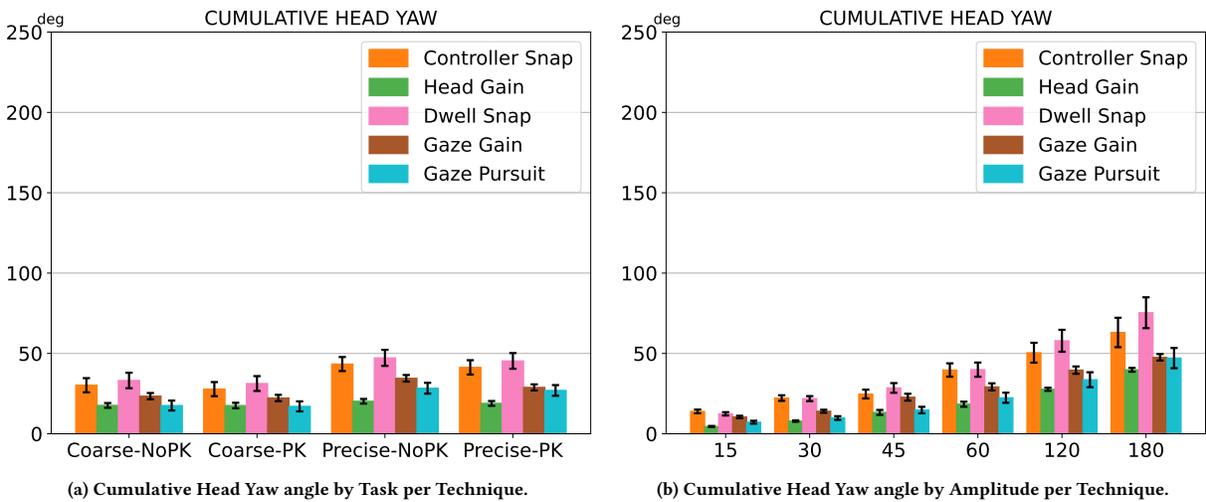
Analyzing the results of cumulative eye yaw angle, we did not find any significant interactions (all tests,  $p > .314$ ,  $\eta_p^2 < .053$ ). We did find a significant main effect of amplitude ( $F(5,95) = 2.046$ ,  $p = .079$ ,  $\eta_p^2 = .097$ ); cf. Figure 6). Like cumulative head yaw, posthoc tests indicate that every combination of amplitudes significantly differs ( $p < 0.01$ , cf. Figure 6).

### 5.4 Task Completion Time

Figure 7 shows the average task completion time per technique. Analysis with a four-way repeated measures ANOVA on aligned



**Figure 4: Error rates by Task (a) and by Amplitude (b) per technique. Error bars represent the 95% confidence interval. There was a significant four-way interaction and a relevant significant main effect of Level of Control (precise led to more errors than coarse).**



**Figure 5: Cumulative Head Yaw angle by Task (a) and by Amplitude (b) per Technique. Error bars represent the 95% confidence interval. There were no significant interactions. There was a significant main effect of amplitude: cumulative head yaw angles of every amplitude are significantly different to all other amplitudes.**

rank-transformed data shows no significant main or interaction effects between factors ( $p > .086$ ,  $\eta_p^2 < .087$ ).

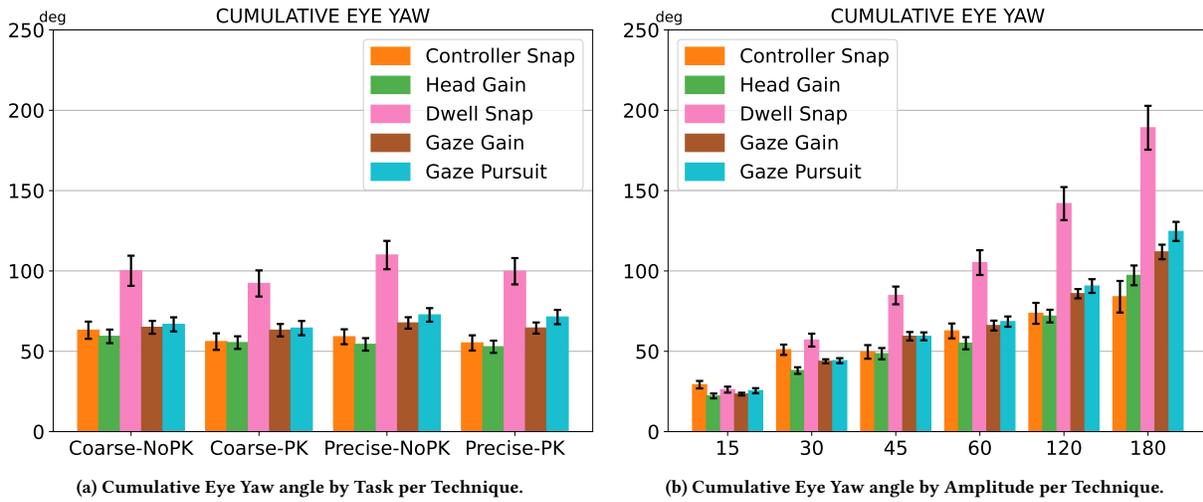
## 5.5 Taskload (RAW NASA TLX)

Figure 8 illustrates the results by Technique and Level of Control.

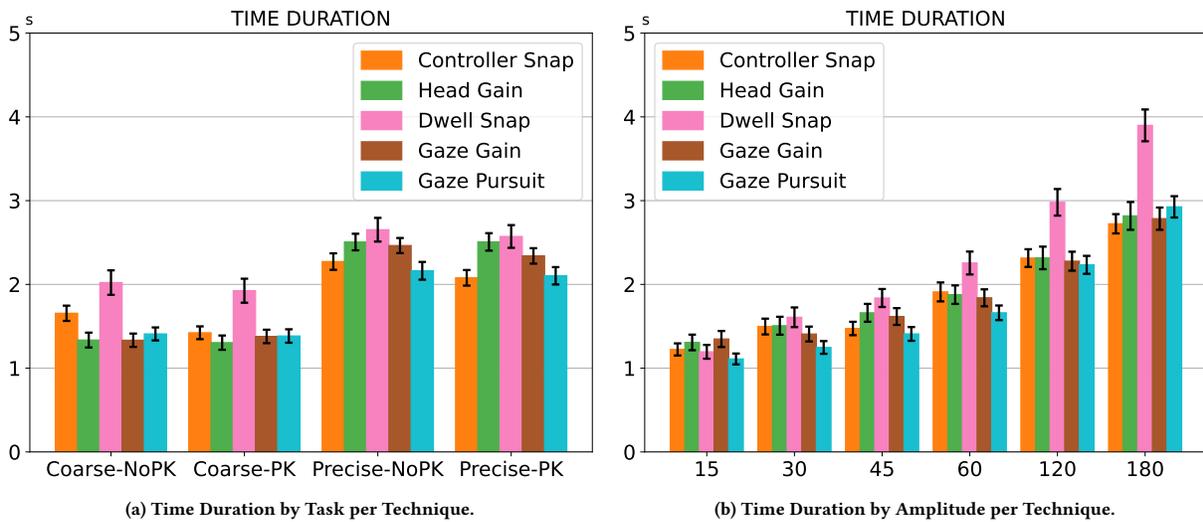
**5.5.1 Effort.** For effort, Technique had a significant impact on the coarse alignment task ( $\chi^2(4) = 16.226$ ,  $p = .003$ ). Here, we found a significant difference between Controller Snap and Dwell Snap ( $p = .001$ ). The technique also had a significant effect on effort for

the precise alignment task ( $\chi^2(4) = 12.985$ ,  $p = .011$ ). However, we found no significant difference with post-hoc tests. Additionally, only Head Gain ( $z = 11.842$ ,  $p = .001$ ) had a significant difference between tasks with higher effort for the precise task.

**5.5.2 Frustration.** The technique also had a significant effect on frustration for the coarse alignment task ( $\chi^2(4) = 11.401$ ,  $p = .022$ ). Even so, post-hoc tests showed no significant differences. Technique also had a significant effect on frustration for the precise task ( $\chi^2(4) = 15.078$ ,  $p = .005$ ). Post-hoc tests showed that Controller



**Figure 6: Cumulative Eye Yaw angle by Task (a) and by Amplitude (b) per Technique. Error bars represent the 95% confidence interval. There were no significant interactions. There was a significant main effect of amplitude. Overall, cumulative eye yaw angles of every amplitude are significantly different to all other amplitudes.**



**Figure 7: Average Trial Duration by Task (a) and by Amplitude (b) per Technique. Error bars represent the 95% confidence interval. There were no significant differences.**

Snap had significantly lower frustration than Dwell Snap ( $p=.044$ ) and Head Gain ( $p=.023$ ). Only Head Gain ( $z=4.571, p=.033$ ) and Gaze Gain ( $z=5.400, p=.020$ ) showed significantly different levels of frustration between tasks, with it being higher for the precise task.

**5.5.3 Mental Demand.** We found no significant differences between techniques in mental demand for the coarse alignment task. However, we did find that technique had a significant effect on mental demand for the precise task ( $\chi^2(4) = 13.081, p = .011$ ). However, post-hoc tests showed no significant differences. Finally, looking at

the difference between techniques, Gaze Gain ( $z=6.368, p=.012$ ) had a significant difference between tasks with a higher mental demand for the precise task.

**5.5.4 Physical Demand.** In the coarse alignment task, Technique had a significant effect on physical demand ( $\chi^2(4) = 10.049, p = .040$ ). However, post-hoc tests were not significant. Technique also had a significant effect on physical demand for the precise alignment task ( $\chi^2(4) = 15.735, p = .003$ ). Here, post-hoc tests showed that Dwell Snap had significantly higher physical demand than

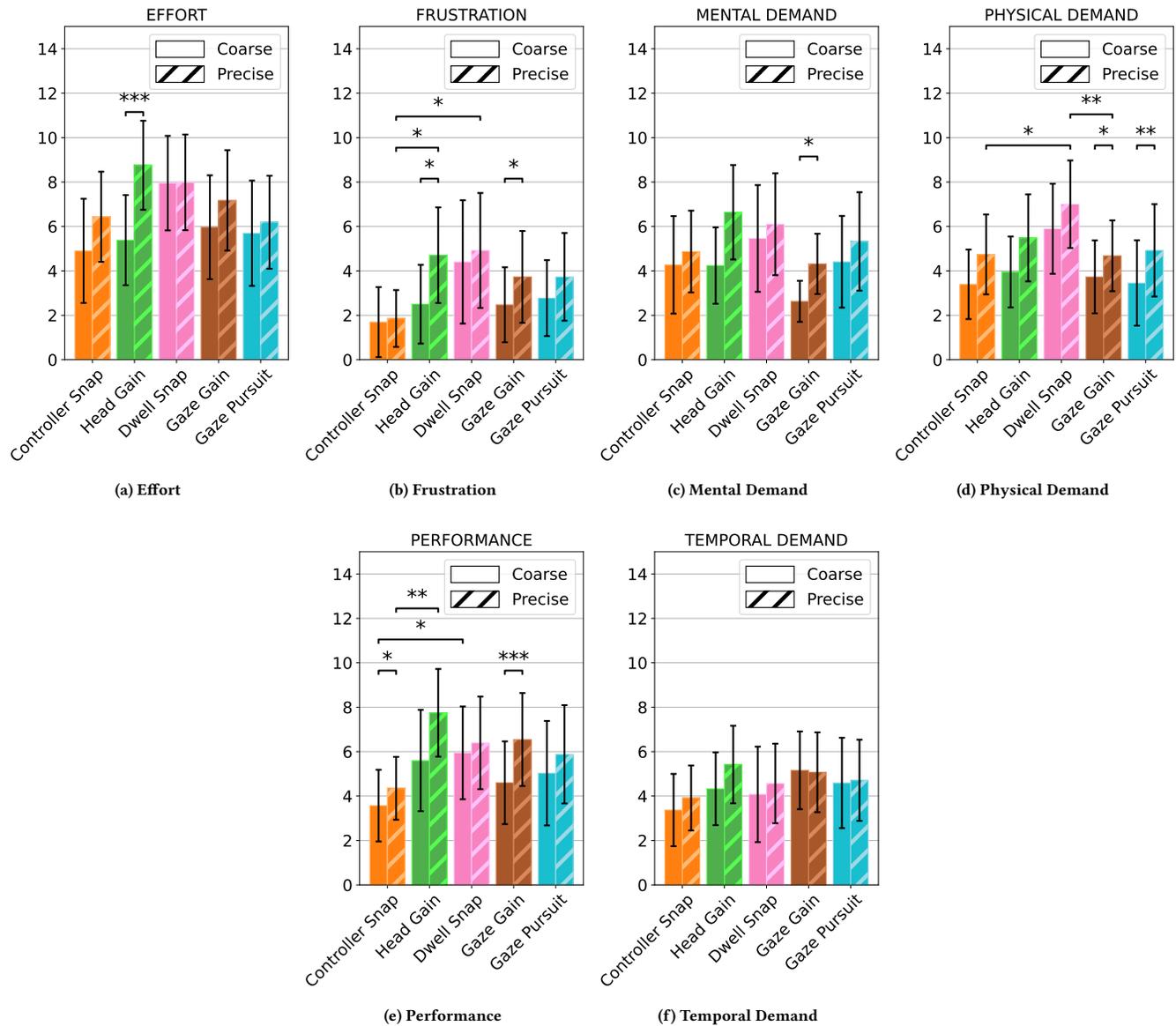


Figure 8: Results of the raw NASA TLX by Technique and Level of Control. Error bars represent the 95% confidence interval.

Controller Snap ( $p=.014$ ) and Gaze Gain ( $p=.007$ ). Only Gaze Gain ( $z=5.556$ ,  $p=.018$ ) and Gaze Pursuit ( $z=8.895$ ,  $p=.003$ ) had a significant difference between tasks for physical demand, with it being higher in the precise task.

**5.5.5 Performance.** Technique had a significant effect on performance for the coarse task ( $\chi^2(4) = 10.699$ ,  $p = .030$ ). Controller Snap was significantly more performant than Dwell Snap ( $p=.037$ ). The technique also had a significant effect on performance for precise tasks ( $\chi^2(4) = 15.264$ ,  $p = .004$ ). Here, Controller Snap was significantly more performant than Head Gain ( $p=.006$ ). Only Controller Snap ( $z=5.556$ ,  $p=.018$ ), and Gaze Gain ( $z=11.842$ ,  $p=.001$ ) were significantly more performant in the coarse task.

**5.5.6 Temporal Demand.** We found no significant differences in the temporal demand for technique in any of the tasks ( $\chi^2(4) < 1.471$ ,  $p > .225$ ) or between the tasks for any technique ( $\chi^2(4) < 8.198$ ,  $p > .085$ ).

**5.5.7 Overall.** Overall workload was affected by technique in the coarse alignment task ( $\chi^2(4) = 13.051$ ,  $p = .011$ ). Dwell Snap has a significantly higher workload than Controller Snap ( $p=.014$ ) and Gaze Gain ( $p=.037$ ). Technique also had a significant effect on the overall workload for the precise alignment task ( $\chi^2(4) = 13.720$ ,  $p = .008$ ). Controller Snap had significantly less workload than Head Gain ( $p=.019$ ). Head Gain ( $z=6.368$ ,  $p=.012$ ), Gaze Pursuit ( $z=5.000$ ,

$p=.025$ ), and Gaze Gain ( $z=9.800$ ,  $p=.002$ ) had a significantly higher workload for the precise task.

## 5.6 Simulator Sickness Score

**5.6.1 Disorientation.** Technique had no effect on disorientation, neither for the coarse nor for the precise alignment task. There was also no effect on disorientation of the task itself (all  $p>.05$ ).

**5.6.2 Nausea.** Technique did not have a significant effect on nausea for the coarse alignment task. However, Technique had a significant effect on nausea for the precise task ( $\chi^2(4) = 9.589$ ,  $p = .048$ ). Post-hoc tests showed no significant differences. Only Head Gain ( $z=5.333$ ,  $p=.021$ ) had a significant difference between tasks, with a higher nausea for the precise task.

**5.6.3 Oculomotor symptoms.** Technique had no effect on the oculomotor symptoms for any task. However, Head Gain ( $z=4.571$ ,  $p=.033$ ), Gaze Gain ( $z=7.143$ ,  $p=.008$ ), and Gaze Pursuit ( $z=4.000$ ,  $p=.046$ ) had a significant difference between tasks, with oculomotor symptoms being higher for the precise task.

**5.6.4 Total Score.** Technique had no effect on the total score for any task. Similar to the oculomotor symptoms, Head Gain ( $z=4.765$ ,  $p=.029$ ), Gaze Gain ( $z=9.800$ ,  $p=.002$ ), and Gaze Pursuit ( $z=4.000$ ,  $p=.046$ ) had a significantly higher total score for the precise task than for the coarse task.

## 5.7 Summary of Results

In summary, the main insights for the evaluation of our techniques are:

- Gaze is effective for viewport control. Participants were able to complete tasks successfully, with low error rates and completion times comparable to manual control. This demonstrates the viability of gaze for viewport control.
- The study did not reveal significant differences in performance, task load, or simulator sickness between our novel gaze-based techniques and established baselines. This is encouraging as it positions gaze as a genuine alternative to methods that depend entirely on the head or hand.
- We found no significant effects of Technique on the amount of eye versus head movement. This is surprising as techniques differ in reliance on eye movement relative to the head. However, technique differences appear masked by individual differences in the tendency to support gaze shifts with head movement [11].
- Technique differences showed only in specific conditions with higher error rates, pointing to different challenges depending on technique behavior.

## 6 DISCUSSION

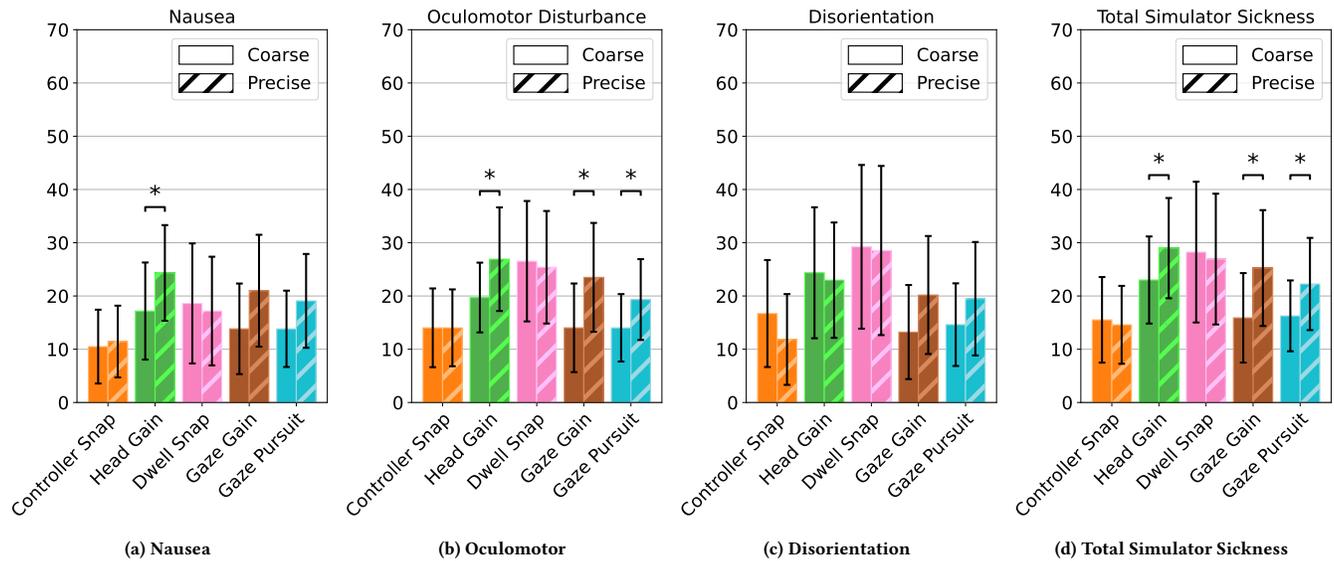
This work is the first to demonstrate gaze for viewport control. Gaze is not straightforward to harness for control as we rely on the movement of our eyes primarily for our visual sense. Viewport control presents a particular challenge in this respect, as the task involves searching through the viewport and aligning the camera towards a target – both inherently visual subtasks that a control method needs to accommodate. Our work shows how viewport

control by gaze can be realized in principally different ways, as seen in the implementation of Dwell Snap, Gaze Gain, and Gaze Pursuit. Our techniques enable horizontal 360° viewport exploration with only a user’s gaze as input. As a headline result from our evaluation, we found all three techniques effective for the task, and hence, gaze viable as a modality for viewport control.

The motivation of our work was to facilitate surround viewing in HMDs with less movement than required in the default mode, where the view is directly coupled with the head. We, therefore, compared our techniques with Controller Snap and Head Gain as existing methods designed for the same purpose. Our study found no significant difference among techniques on key measures. Participants were as efficient with our new techniques and reported comparable task load and simulator sickness compared to the baselines. These positions gaze as an alternative that is on par with existing methods in performance and usability, yet not reliant on hand or head movement.

Hands-free viewport control with less or no head movement has wide-ranging practical relevance. Gaze can be of importance for accessibility and especially for users who have limited or no head or hand movement due to injury or disability. Gaze can also facilitate viewport control in any situation that constrains other movement – for example, in crowded areas, sitting or standing in a tight space or close to others, or when relaxing or lying down in a comfortable posture. Gaze-based viewport control would be preferable when more conspicuous movement is socially awkward – for instance, on public transport. Gaze-based techniques are also a useful alternative to Controller Snap when applications do not rely on controllers for other input, for example, when the interaction is based on hand tracking or when viewport control constitutes the main interaction, such as in cinematic VR. However, our techniques are equally relevant when controllers are available to focus controller use on other input tasks, in the same way, that manual control is separate from viewport control in the default head-coupled mode.

Dwell Snap, Gaze Gain, and Gaze Pursuit differ substantially in how they enable viewport control, each with distinct affordances that may cater to different applications. The techniques are based on different types of eye movement (fixations, saccades, and smooth pursuit) and control the viewport differently in discrete steps, amplified rotation, or continuous motion. Gaze behavior is affected by the nature of the content presented, for instance, motion in the scene, and the user’s task, for instance, focused attention versus casual browsing. We consider Gaze Pursuit best suited for video-like experiences such as 360°-movies, as Gaze Pursuit would leverage the eyes’ natural ability to follow the moving content presented. Gaze Gain is distinct among our techniques (and also different from Controller Snap) in preserving a sense of direction, where the user always returns to their initial view when they gaze straight ahead in a face forward position. This makes the technique preferable for experiences that benefit from a sense of direction or spatial awareness, such as virtual galleries or first-person exploration games. Gaze Dwell, in turn, may excel in applications that require fast switching between different areas and workspaces, such as during work-like and information-dense scenes, multi-screen setups, data analysis, or immersive teleconferences. The technique affords a stable view that can be visually explored with gaze in a natural eye-in-head movement range of up to  $\pm 25^\circ$  without moving the



**Figure 9: Simulator sickness questionnaire by Technique and Level of Control. Error bars represent the 95% confidence interval.**

viewport while supporting fast switching by fixation beyond that range. Naturally, there will also be application contexts in which our techniques may function less well. For example, fast-paced applications such as VR fighting games require rapid reactive gaze to events across the scene, which might lead to inadvertent viewport effects. However, our techniques showcase a range of concepts and behaviors with scope for adaptation and tailoring to different application requirements.

Gaze is complex not only in how it affords both vision and control but also in how it combines eye and head movement. Our techniques allow viewport control to become possible without head movement while not prohibiting or disregarding any natural head movement contribution to gaze. Gaze Gain relied on saccades, which we expected to be naturally supported by head movement, to the effect of head movement seamlessly aiding with the refinement of the viewport orientation. In fact, we observed most errors with Gaze Gain for short gaze shifts ( $15^\circ$ ) that did not implicitly trigger head movement and had participants rely on less precise eye saccades for camera alignment. Dwell Snap required eye movement to more eccentric angles within the HMD, with head movement of no utility for control. Here, we also observed more errors for shorter amplitudes, with participants moving too quickly and overshooting. Gaze Pursuit relied on eye rotation to an offset from the head to trigger rotation of the scene through the viewport, which the head could simultaneously move to reach the target faster. Here, we saw more error at larger amplitudes when the head was at maximum rotation, which hampered the head during the refinement of camera alignment.

In spite of the differences in design, we did not find any significant differences in cumulative eye and head movement between any of the techniques studied. However, despite a sample size of  $N = 20$  and 3 blocks, we found standard deviations high (also for other measures). Here, we believe that fundamental differences between

humans rather than technique and task are the reason for these relatively large variances. Research in eye-head coordination has found significant idiosyncratic variations in people’s tendency to support gaze with head movement and suggested that people fall into groups of “head-movers” and “non-head-movers” [11, 50]. Specifically, in ranges up to  $45^\circ$  from the central position, gaze targets can be acquired completely without head movement through gaze at more eccentric angles, or by using both, eye and head movement. In our study, we found, for example, that 40% of the participants using Gaze Pursuit moved their head less than  $15^\circ$  across all coarse alignment trials, c.f. Figure 5) while others made more extensive use of head movement to complement eye-controlled view rotation.

Our work comes with several limitations. We evaluated our techniques on an abstract task in a plain virtual environment to establish baseline performance. The abstract task followed an established protocol but did not include distractions that might influence performance in more realistic application contexts. Our techniques have the potential to support a wide range of applications on account of their different affordances, but further work will be needed to analyze the design space and evaluate techniques for various applications and use cases. We conducted our study with participants in a seated position as this is a common setting that limits the viewing range in a head-coupled display. Our techniques also aim to facilitate viewport control in other poses that may constrain head movement further, but these were not tested. We designed our techniques to be usable without any contribution of head movement but chose not to constrain head movement artificially, as we aimed for gaze control to be in natural coordination of eye and head movement. As a result, differences between techniques may have become masked by individual differences in the tendency to support gaze with head movement. This suggests further work to understand better and support individual differences. The use of

gaze, of course, also has practical limitations as it relies on eye-tracking. However, compared with other eye-tracking tasks, camera control will typically not require as much precision and be more readily feasible within the limits of state-of-the-art eye tracking.

## 7 CONCLUSION

In this work, we presented and evaluated three gaze-based viewport-control techniques: Dwell Snap, Gaze Gain, and Gaze Pursuit — each leverages a different type of eye movement. Compared with head amplification and controller snap as baselines, we could show that all three of our techniques achieve competitive performance during coarse and precise viewport alignment. Participants did not systematically make more errors with our techniques, did not need more time to align the viewport, and eye and head movements stayed normal. Our results also highlight large variations in performance, possibly because of varying characteristics of how individuals' eyes and head work together. Together, we could show for the first time that all three, gaze, natural head and eye movements, and natural eye-head coordination, can be integrated into viewport-control techniques. This enables radically new interaction techniques for various applications like 360° video and scenarios such as VR on public transport, when lying down, or for people with injuries or disabilities. Further, our work can inspire novel interaction techniques next to viewport control by enhancing gaze-based selection in virtual and augmented reality.

## ACKNOWLEDGMENTS

We would like to thank Haopeng Wang for his help with statistics. This work was supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (Grant No. 101021229 GEMINI: Gaze and Eye Movement in Interaction).

## REFERENCES

- [1] Jérôme Ardouin, Anatole Lécuyer, Maud Marchal, Clément Riant, and Eric Marchand. 2012. FlyVIZ: A Novel Display Device to Provide Humans with 360° Vision by Coupling Catadioptric Camera with Hmd. In *Proceedings of the 18th ACM Symposium on Virtual Reality Software and Technology* (Toronto, Ontario, Canada) (VRST '12). Association for Computing Machinery, New York, NY, USA, 41–44. <https://doi.org/10.1145/2407336.2407344>
- [2] Jeremy B. Badler and Alessandro Canossa. 2015. Anticipatory Gaze Shifts during Navigation in a Naturalistic Virtual Environment. In *Proceedings of the 2015 Annual Symposium on Computer-Human Interaction in Play* (London, United Kingdom) (CHI PLAY '15). Association for Computing Machinery, New York, NY, USA, 277–283. <https://doi.org/10.1145/2793107.2793136>
- [3] Laura Bajorunaite, Stephen Brewster, and Julie R. Williamson. 2021. Virtual Reality in transit: how acceptable is VR use on public transport?. In *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. 432–433. <https://doi.org/10.1109/VRW52623.2021.00098>
- [4] Costas Boletis. 2017. The New Era of Virtual Reality Locomotion: A Systematic Literature Review of Techniques and a Proposed Typology. *Multimodal Technologies and Interaction* 1, 4 (2017). <https://doi.org/10.3390/mti1040024>
- [5] James C Byers and Susan G Hill. 1989. Traditional and raw task load index (TLX) correlations: Are paired comparisons necessary. (1989), 481,485.
- [6] Xiang Cao, Jacky Jie Li, and Bavin Balakrishnan. 2008. Peephole Pointing: Modeling Acquisition of Dynamically Revealed Targets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Florence, Italy) (CHI '08). Association for Computing Machinery, New York, NY, USA, 1699–1708. <https://doi.org/10.1145/1357054.1357320>
- [7] 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>
- [8] Myungguen Choi, Daisuke Sakamoto, and Tetsuo Ono. 2022. Kuiper Belt: Utilizing the “Out-of-Natural Angle” Region in the Eye-Gaze Interaction for Virtual Reality. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) (CHI '22). Association for Computing Machinery, New York, NY, USA, Article 357, 17 pages. <https://doi.org/10.1145/3491102.3517725>
- [9] Massimiliano Di Luca, Hasti Seifi, Simon Egan, and Mar Gonzalez-Franco. 2021. Locomotion Vault: The Extra Mile in Analyzing VR Locomotion Techniques. 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 128, 10 pages. <https://doi.org/10.1145/3411764.3445319>
- [10] W. J. Dixon and K. K. Yuen. 1974. Trimming and winsorization: A review. *Statistische Hefte* 15, 2 (01 Jun 1974), 157–170. <https://doi.org/10.1007/BF02922904>
- [11] James H. Fuller. 1992. Head movement propensity. *Experimental Brain Research* 92, 1 (12 1992), 152–164. <https://doi.org/10.1007/BF00230391>
- [12] Jan Gugenheimer, Dennis Wolf, Gabriel Haas, Sebastian Krebs, and Enrico Rukzio. 2016. SwiVRChair: A Motorized Swivel Chair to Nudge Users' Orientation for 360 Degree Storytelling in Virtual Reality. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) (CHI '16). Association for Computing Machinery, New York, NY, USA, 1996–2000. <https://doi.org/10.1145/2858036.2858040>
- [13] John Paulin Hansen, Astrid Kofod Trudslev, Sara Amdid Harild, Alexandre Alapetite, and Katsumi Minakata. 2019. Providing Access to VR Through a Wheelchair. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland UK) (CHI EA '19). Association for Computing Machinery, New York, NY, USA, 1–8. <https://doi.org/10.1145/3290607.3299048>
- [14] Jan Hombeck, Henrik Voigt, Timo Heggemann, Rabi R. Datta, and Kai Lawonn. 2023. Tell Me Where To Go: Voice-Controlled Hands-Free Locomotion for Virtual Reality Systems. In *2023 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*. 123–134. <https://doi.org/10.1109/VR51514.2023.00028>
- [15] Caroline Jay and Roger Hubbard. 2003. Amplifying Head Movements with Head-Mounted Displays. *Presence: Teleoperators and Virtual Environments* 12, 3 (06 2003), 268–276. <https://doi.org/10.1162/105474603765879521> arXiv:<https://direct.mit.edu/pvar/article-pdf/12/3/268/1623881/105474603765879521.pdf>
- [16] Robert S. Kennedy, Norman E. Lane, Kevin S. Berbaum, and Michael G. Lilienthal. 1993. Simulator Sickness Questionnaire: An Enhanced Method for Quantifying Simulator Sickness. *The International Journal of Aviation Psychology* 3, 3 (1993), 203–220. [https://doi.org/10.1207/s15327108ijap0303\\_3](https://doi.org/10.1207/s15327108ijap0303_3) arXiv:[https://doi.org/10.1207/s15327108ijap0303\\_3](https://doi.org/10.1207/s15327108ijap0303_3)
- [17] Krista Kornlyo, Natalie Dill, Melissa Saenz, and Richard J. Krauzlis. 2003. Canceling of Pursuit and Saccadic Eye Movements in Humans and Monkeys. *Journal of Neurophysiology* 89, 6 (2003), 2984–2999. <https://doi.org/10.1152/jn.00859.2002> arXiv:<https://doi.org/10.1152/jn.00859.2002> PMID: 12783947
- [18] Manu Kumar, Terry Winograd, and Andreas Paepcke. 2007. Gaze-Enhanced Scrolling Techniques. In *CHI '07 Extended Abstracts on Human Factors in Computing Systems* (San Jose, CA, USA) (CHI EA '07). Association for Computing Machinery, New York, NY, USA, 2531–2536. <https://doi.org/10.1145/1240866.1241036>
- [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* (Montreal QC, Canada) (CHI '18). Association for Computing Machinery, New York, NY, USA, 1–14. <https://doi.org/10.1145/3173574.3173655>
- [20] Michael Land. 2019. Eye movements in man and other animals. *Vision Research* 162 (2019), 1–7. <https://doi.org/10.1016/j.visres.2019.06.004>
- [21] Eike Langbehn, Joel Wittig, Nikolaos Katzakis, and Frank Steinicke. 2019. Turn Your Head Half Round: VR Rotation Techniques for Situations With Physically Limited Turning Angle. In *Proceedings of Mensch Und Computer 2019* (Hamburg, Germany) (MuC '19). Association for Computing Machinery, New York, NY, USA, 235–243. <https://doi.org/10.1145/3340764.3340778>
- [22] Joseph J LaViola Jr, Ernst Kruijff, Ryan P McMahan, Doug Bowman, and Ivan P Poupyrev. 2017. *3D user interfaces: theory and practice*. Addison-Wesley Professional.
- [23] Wenxun Li and Leonard Matin. 1990. The influence of saccade length on the saccadic suppression of displacement detection. *Perception & Psychophysics* 48, 5 (01 Sep 1990), 453–458. <https://doi.org/10.3758/BF03211589>
- [24] Mark McGill, Aidan Kehoe, Euan Freeman, and Stephen Brewster. 2020. Expanding the Bounds of Seated Virtual Workspaces. *ACM Trans. Comput.-Hum. Interact.* 27, 3, Article 13 (may 2020), 40 pages. <https://doi.org/10.1145/3380959>
- [25] Raphael Menges, Chandan Kumar, Daniel Müller, and Korok Sengupta. 2017. GazeTheWeb: A Gaze-Controlled Web Browser. In *Proceedings of the 14th International Web for All Conference* (Perth, Western Australia, Australia) (W4A '17). Association for Computing Machinery, New York, NY, USA, Article 25, 2 pages. <https://doi.org/10.1145/3058555.3058582>
- [26] Craig H. Meyer, Adrian G. Lasker, and David A. Robinson. 1985. The upper limit of human smooth pursuit velocity. *Vision Research* 25, 4 (1985), 561–563. [https://doi.org/10.1016/0042-6989\(85\)90160-9](https://doi.org/10.1016/0042-6989(85)90160-9)
- [27] Mark R Mine. 1995. Virtual environment interaction techniques. *UNC Chapel Hill CS Dept* (1995).

