StylusPort: Investigating Teleportation using Stylus in VR

Yang Liu
Aarhus University
Aarhus, Denmark
yang.liu@cs.au.dk

Qiushi Zhou
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China
qiushizhou@hkust-gz.edu.cn

Mathias N. Lystbæk
Aarhus University
Aarhus, Denmark
mathiasl@cs.au.dk

Aidan Kehoe
Logitech
Cork, Ireland
akehoe@logitech.com

Mario Gutierrez
Logitech Europe S.A.
Lausanne, Switzerland
mgutierrez1@logitech.com

Hans Gellersen
Lancaster University
Lancaster, U.K.
Aarhus University
Aarhus, Denmark
h.gellersen@lancaster.ac.uk

Ken Pfeuffer
Aarhus University
Aarhus, Denmark
ken@cs.au.dk

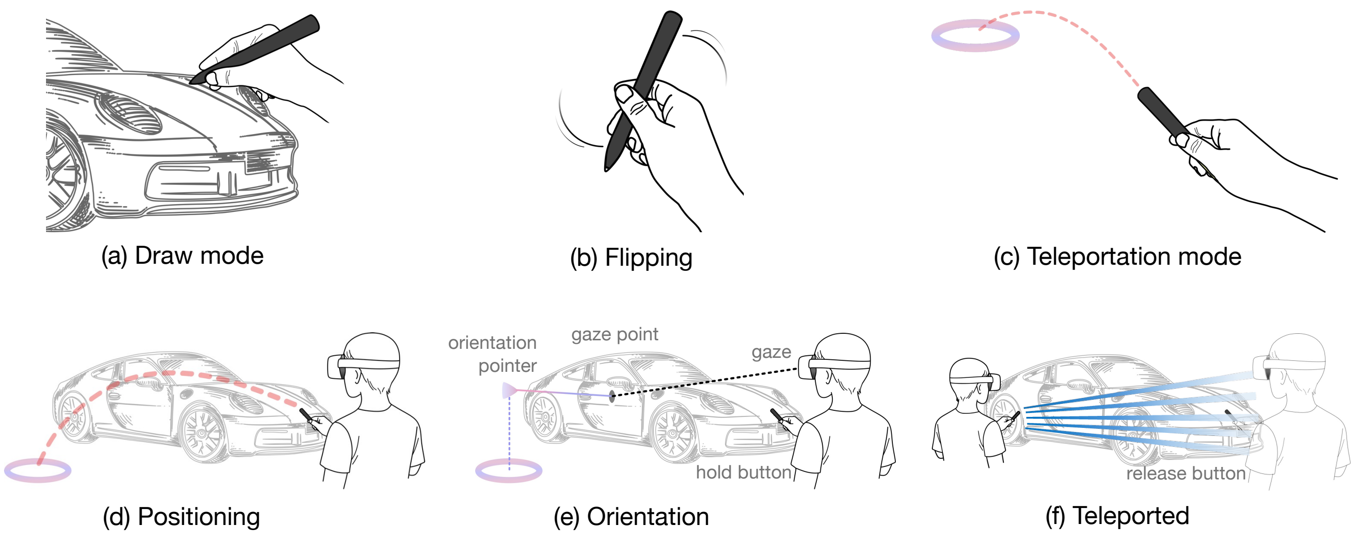


Figure 1: We investigate and propose new ways to teleport with a stylus. For example, by default users sketch (a), but at any time they can flip the stylus to switch to teleport mode (b–c). Users select a target position via raycasting (d). While holding the button, they can look in the direction they want to face after teleporting (e). Releasing the button executes the teleport (f).

Abstract

With a stylus, users can both sweep sketches across models and pinpoint locations with precision. Building on this dual capability, we explore how teleportation can be integrated into stylus interaction without disrupting the flow of common stylus usage. We introduce two key ideas: flipping the stylus as an intuitive mode switch between drawing and teleportation, and using gaze to set orientation while the stylus handles positioning. In a user study that features a teleport-and-orient task, we evaluate six teleportation techniques, covering two mode-switching methods (Button and Flip) and three orientation approaches (StylusRoll, StylusPoint, and GazePoint). The results offer new insights into the relative merits and limitations of each technique. Our work contributes to knowledge about teleportation in VR and fills the gap in seamlessly integrating teleportation with stylus use in 3D.

CHI '26, Barcelona, Spain

ACM ISBN 979-8-4007-2278-3/2026/04
https://doi.org/10.1145/3772318.3793420

CCS Concepts

• **Human-centered computing** → **Interaction techniques**; **Pointing devices**; **User studies**.

Keywords

virtual reality, teleportation, locomotion, stylus, pen, gaze

ACM Reference Format:
Yang Liu, Qiushi Zhou, Mathias N. Lystbæk, Aidan Kehoe, Mario Gutierrez, Hans Gellersen, and Ken Pfeuffer. 2026. StylusPort: Investigating Teleportation using Stylus in VR. In *Proceedings of the 2026 CHI Conference on Human Factors in Computing Systems (CHI '26), April 13–17, 2026, Barcelona, Spain.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3772318.3793420

1 Introduction

Designers can engage in Virtual Reality (VR) to work with 2D and 3D content in ways that go beyond traditional desktop tools–moving around objects, seeing them from different perspectives, editing, drawing, and refining precise models. A key part of this process is navigation. While users can physically walk in VR, they also perform *Point & Teleport* [8, 17, 37] to instantly change position and orientation, particularly when physical movement is constrained, for example, when sitting on a chair or using a tethered headset. Orientation control is especially important when working with large-scale models or navigating large spaces, which is extensively supported in current VR design applications, such as Arkio, EngageXR, and VXRLabs [1, 16, 55]. In a typical workflow of such, a designer may teleport inside a large model to focus on a specific component for sketching and editing. They may also need to maneuver within a room-scale scene containing multiple objects (e.g., furniture) and quickly orient themselves around different items of interest.

This can be performed with a controller that includes a clear division of labour between object manipulation and teleportation capabilities through multiple buttons and joysticks [21, 35]. With a stylus, however, it is more challenging to map the same number of functions to fewer inputs. A stylus usually has few buttons, while at least one is reserved to activate manipulation in 3D space. A direct one-to-one mapping of controller functionality to a stylus would only make it work for the simplest applications [26]. And while it is possible to extend a stylus with additional buttons, widgets, menus, or even non-dominant hand input, they take up space and can be error-prone [11, 56], rather than supporting the use of the stylus in a simple or efficient way.

In this work, we investigate StylusPort: teleportation techniques with stylus-based interaction in VR. Building on prior research on Point & Teleport techniques using hand- or controller-pointing with parabolic rays [8, 17, 37], we explore this approach with VR styluses, and aim to address two research questions:

RQ1 - Mode switching: How can users switch modes between draw mode and teleportation mode? Previous work in 3D stylus input indicates high potential to leverage implicit methods (e.g., gesture, posture, and grip) [11]. We explore a *flip gesture* and compare it to a *button-based mode switch* as a baseline.

RQ2 - Orientation control: How can users specify teleportation orientation with a stylus? We investigate three techniques: StylusRoll as a means to adjust orientation, extending prior controller [17, 37] and 2D stylus [6] techniques; a *dual-pointer approach* StylusPoint, where users specify both position and orientation through pointing [37]; and a *gaze-enabled approach* GazePoint, where users look at the point they wish to face after teleporting, inspired by prior work on gaze-directed flying [24, 36, 54] and multimodal gaze + pen work [41, 44, 58].

To address these questions, we conducted a user study ($N = 18$) with two main independent variables: Mode Switching (Button, Flip) and Orientation Control (StylusRoll, StylusPoint, and GazePoint). In each task trial, participants (1) switched to teleportation mode, (2) specified a teleport position and confirmed with a button press, (3) specified orientation until button release, and (4) switched back to draw mode to complete the trial by drawing a stroke. The target was a blackboard representing a typical drawing canvas, presented at five possible rotations and two distances, which is inspired by common use cases in commercial education applications [16, 55]. Teleport positions were constrained to the ground plane, while both the ground and object surfaces could be used to specify orientation.

Our results show that Flip provided faster mode switching and higher pointing accuracy, thanks to the palm grip, while maintaining an overall task completion time comparable to Button. Regarding orientation control, we primarily see differences in time efficiency, where StylusRoll proved to be the slowest method, while StylusPoint was found to be faster for orientation than GazePoint.

Overall, our contributions are: (1) novel teleportation techniques featuring stylus-flip mode switching and gaze-based orientation specification; (2) empirical insights from a user study that delineate performance and user experience trade-offs of these techniques.

2 Related Work

Our work sits at the intersection of VR teleportation, stylus-based interaction, and gaze-based locomotion.

2.1 Teleportation

Early teleportation prototypes started from viewport control in 2D interfaces for 3D applications. For instance, *Navidget* and *Immersive Navidget* explored novel 2D and 3D viewport control approaches when navigating a virtual environment [18, 28]. These early works indicated that positioning and orientation are two fundamental components of teleportation that need to be addressed separately. This orientation approach has inspired later works. *Anchored Jumping* specifies a ground point to define the facing direction and then selects the teleport destination [7]. As well, *SkyPort* evaluated linear and parabolic pointing and different transition types–instant, interpolated, and continuous– and concluded that linear aiming with instant transitions offers high efficiency and accuracy without increasing sickness [35]. Weissker et al. explored teleportation to mid-air 3D positions by specifying ground position and height either simultaneously or separately. They reported that simultaneous control improves accuracy but leads to a longer time than controlling position and height separately [59].

2D teleportation–selecting a ground position using a parabolic ray from the controller or the user's hand–is a common locomotion method across prevalent devices and applications. For example, Meta Quest devices introduce 2D teleportation as a primary locomotion type implemented for both controller and hand microgesture [21]. These techniques have been iteratively developed and evaluated in HCI research. Bozgeyikli et al. proposed *Point & Teleport* and compared it with walk-in-place and joystick locomotion. They also proposed and evaluated an orientation component that let users set their facing direction by rolling the hand. Whereas Point & Teleport was found fun and user-friendly, the orientation component was considered unintuitive and difficult to control [8]. Similarly, a study on spatial cognition found that teleportation is less error-prone without orientation [13].

For pointing methods of 2D teleportation, Rupp et al. found that linear rays make long-distance teleportation more difficult than parabolic rays [46], while other studies reported that linear rays are faster and more intuitive in certain contexts [39, 48]. Despite that, parabolic rays are now the teleportation standard. Funk et al. investigated parabolic-ray Point & Teleport with several orientation control techniques: *AngleSelect* (touchpad-based), *Curved Teleport* that visualizes an adjustable curved trajectory, and *HPCurved* that combines a parabola with an adjustable curvature. They argued that while these three techniques with orientation control were slower than those without, they reduced post-teleport orientation correction [17]. Mori et al. evaluated four orientation approaches for Point & Teleport inspired by "natural turning", including head-turning, integrated touchpad control, wrist-based orientation, and post-positioning pointing for orientation. They concluded that while orientation is useful–especially in extreme conditions–no single consensus of approach is both intuitive and low-effort [37]. Müller et al. explored undoing teleportation with separate position and orientation components, finding that combining both significantly improves usability [38]. Finally, most commercial VR applications implement orientation as a separate step from positioning [16, 50, 55]. In this work, we address the gap in integrated positioning and orientation with a stylus.

2.2 Stylus-based Interaction

Whereas mid-air stylus input was explored in early desktop VR systems [14], the large virtual spaces afforded by modern VR present new opportunities and challenges for using 3D stylus input in navigation while performing primary tasks such as sketching and manipulation [25, 45]. Early work addressed this by reducing the need to move physically and enabling stylus input at a distance through physical surfaces. For instance, Arora et al. demonstrated the viability of using a physical drawing surface to support free-form mid-air sketching in VR [2, 3]. Other works explored multitouch gestures, such as *VRSketchIn*, which investigated a design space of stylus and tablet interaction for 3D sketching in VR that combines unconstrained 3D mid-air with constrained 2D surface-based sketching [15]. However, 3D sketching without a physical surface is still a major use case, as evidenced by leading VR applications such as ShapesXR, GravitySketch, and VXRLabs [49, 50, 55], broadly used in 3D product design. Other works have explored mid-air 3D interaction with distant objects using a stylus, such as pointing and manipulation. Chen et al. found that VR controllers and styluses yield significantly higher precision than bare hands in a 3D target-tracing task, suggesting that while styluses benefit precise 3D drawing, target selection may be better handled by hand or another modality due to stylus-induced fatigue [12].

Integrating stylus input with other modalities opens new interaction opportunities. Matulic and Vogel explored bimanual pen-and-touch interactions in VR, showing how asymmetric pen-hand coordination supports manipulation and navigation [34]. Wagner et al. investigated combining gaze and stylus input for selecting and translating shape points in 3D modeling, finding that gaze-assisted dragging reduced task time and manual effort with slightly increased errors [58]. We extend the prior multimodal interaction work by a focus on teleportation.

Similar to the use of physical pens, grip postures also affect comfort and performance for different tasks, such as coarse and precise drawing. Batmaz et al. found that a "precision grip" (typical pen grip) significantly improves accuracy in VR [4]. Li et al. similarly found that a rear-end "tripod" precision grip allows the largest range of motion. They specified that while forward-and-downward pointing is easy with a precision grip, forward-and-upward pointing is easier with a palm grip (like holding a wand), which also induces less fatigue during prolonged use [31].

Previous work has explored enriching stylus affordances by using different grip postures for distinct functions and modes. Cami et al. evaluated common variations of precision grips and demonstrated using unoccupied fingers for alternative input modes, such as touch on a tablet [11]. Song et al. proposed *Multi-Touch Pen*, which enables input mode switching through different tapping gestures on a stylus afforded by different grip postures [52]. Cai et al. presented *HPIPainting* that detects different grips to contextualize gesture recognition for triggering commands in VR painting [10]. Li et al. specifically investigated switching between inking and gesturing mode when using a stylus. They explored approaches such as changing stylus tip pressure, holding the stylus still, and pressing a button using the non-dominant hand. They found trade-offs between approaches with no clear winner in speed and accuracy [32]. Other works evaluated alternative approaches, including non-dominant hand gestures and inferring from stylus trajectory for mode switching between inking and other input (e.g., selection), suggesting that mode switching needs more thoughtful design than simply adding gestures or using implicit information [47, 51].

In this work, inspired by previous work on input mode switching for VR [53] and other platforms [43], we explore a novel approach of switching between drawing and teleportation modes using a stylus flip action that transitions between precision grips and palm grips. This exploits the ease of forward-and-upward pointing using the palm grip, which suits parabola-based Point & Teleport techniques [31].

2.3 Gaze-based Locomotion

Gaze input has been envisioned in some of the earliest explorations of VR locomotion. In his seminal work *Virtual environment interaction techniques*, Mine envisioned *gaze-directed flying*, where users are transported towards the direction of their gaze [36]. Similarly, an early exploration of "walking in place" proposed incorporating gaze and head direction—essential natural behaviors during locomotion—as an integrated input modality [54]. Following a similar idea of using gaze for viewport control, Lee et al. explored several gaze-based viewport control techniques combined with head movement, including snapping to gaze-dwell location, using gaze saccades as gain functions to amplify head rotation, and gaze pursuit for aligning the viewport with specific targets [29]. These works suggest a promise of using gaze as an additional modality for teleportation in VR.

Gaze has been explored as a natural and easy-to-use input modality, both implicit and explicit, to specify points of interest in interaction tasks [22, 23]. One particular use case relevant to our work is extending direct input devices like touch, gesture, or pen with additional indirect input modes. Previous works, such as Gaze-Touch [40], Gaze-Shifting [41], and Gaze+Pinch [42], exploit the natural eye-hand coordination so users can access both direct and gaze-assisted indirect input modes, enabling rapid access to distant objects. In the large virtual spaces of VR, this saves time and effort induced by manual pointing while retaining the primary role of the given input modality [33, 57].

These benefits of gaze input have been explored in recent works on VR teleportation. For instance, Kim et al. explored teleportation methods using hand-tracking, eye-tracking, and EEG signals. Their proposed techniques used either hand or gaze for position aiming, and triggered teleportation using hand gestures or EEG. They found that gaze-based positioning was faster and more precise than hand pointing [27]. Lee et al. proposed an approach of specifying teleportation orientation by detecting users' gaze towards directions that fall out of the central area of their view, constructing it as an explicit trigger for orientation change [30]. In this work, we explore the use of gaze for orientation specification in VR teleportation as an explicit pointer that activates after stylus-based positioning.

3 Design of Stylus-based Teleportation

We describe the design of techniques for switching to teleportation mode and specifying orientation with a stylus.

3.1 Mode Switching

We assume typical stylus-based interaction where most of the available buttons are dedicated to default operations (e.g., sketching, pointing, object manipulation), and incorporate a mode-switching button for users to enter teleportation mode. In this mode, users point with the stylus to specify position and press-hold a button to specify orientation, and release to accomplish the teleport.

3.1.1 Button. Our rationale for the first mode-switching technique is as follows. We ruled out assigning teleportation to a simple button press because users need to preview the parabolic teleport ray before performing a teleport, and continuously displaying this ray would interfere with primary actions such as sketching. Showing the ray by pressing and holding the button is also not viable–the button-hold action is already needed for orientation control. Instead, it is reasonable for the secondary button, namely the button not dedicated to primary actions, to act as a mode selector. Our first teleportation method, Button, adopts this approach: pressing the button switches from the primary function into teleportation mode; pressing it again returns to the previous mode. In principle, additional modes could be added to provide more functionality beyond teleportation. Within teleportation mode, the ray visualization gives clear feedback that the user is in the correct mode.

3.1.2 Flip. For teleportation without relying on a button, we exploit stylus grip semantics. Prior work shows that different stylus grips afford distinct functions and can support mode switching [10, 11, 31, 52]. Building on this, we employ the natural distinction between drawing with the stylus tip and pointing with the stylus tail: teleportation is active whenever the stylus is flipped. This Flip method provides a low-cost, button-free mechanism that leverages finger dexterity and familiar pen usage. A flip action is illustrated in Figure 1a–c. Flip is detected by monitoring the direction of the stylus (from tail to tip) relative to the user's forward view. A flip is valid when the stylus points more than 120° away from the camera's forward vector. The angular threshold was chosen through in-house testing to minimize unintentional flip events.

For both Button and Flip, a parabolic ray cast from the stylus previews the teleport destination. The ray trajectory is defined by the origin (stylus tip or tail, depending on the grip), initial direction (tail-to-tip or tip-to-tail), velocity (10 m/s), gravity (9.81 m/s^2), and a maximum fall time (1.5 s). These yield a maximum distance of about 11.15 m at an optimal pitch angle of 42°. When the parabola hits the ground, a solid circular cursor appears and marks the teleport destination. Users press the primary function button–which acts as the teleport button in teleportation mode–to confirm the destination, and release it to execute the teleport.

To support teleporting behind objects, we allow developers to configure whether specific objects block or permit the parabolic ray. For example, walls block teleportation, while a blackboard may allow the ray to pass. This provides a shortcut for crossing obstacles without detouring. As shown in Figure 3, when the parabolic ray penetrates an object, that object temporarily becomes semi-transparent so users can clearly see the destination.

3.2 Orientation Control

We design techniques where a quick click of the teleport button triggers a teleport without changing the user's orientation, while a long press enables orientation control. This mirrors a two-stage input design, as also seen in Mori et al.'s *P2T*, which differentiates between half- and full-press actions [37]. We set the threshold between a short click and a hold to 200 ms, slightly above the 150 ms average click duration observed in our in-house testing. In the following, we illustrate the design of three orientation methods based on this principle, namely StylusRoll, StylusPoint, and GazePoint.

3.2.1 StylusRoll. StylusRoll builds on prior work [8, 17, 37] that maps stylus roll to relative changes in orientation (cf. Figure 2a), leveraging the unused rotational axis during pointing. When the user holds the button, the initial orientation is set to the user's head direction. Rolling the stylus then rotates this orientation either left or right according to the stylus's rotational direction. The technique supports flexible rolling not only through wrist rotation, but also through in-hand finger movement. To facilitate large orientation angles, we align with Mori et al. [37] and apply a 1.5× scaling factor: 1° rolling maps to 1.5° orientation change. This was determined through pilot testing to balance rolling efficiency and accuracy. The visual feedback is illustrated in Figure 2a. A 3D arrow is displayed above the teleport destination to indicate orientation, appearing 20 cm below the user's eye level. Also, two 1 m horizontal lines extend from the center and the arrow in the direction.

3.2.2 StylusPoint. StylusPoint introduces an orientation cursor. Once the teleport button is held, the position cursor on the ground

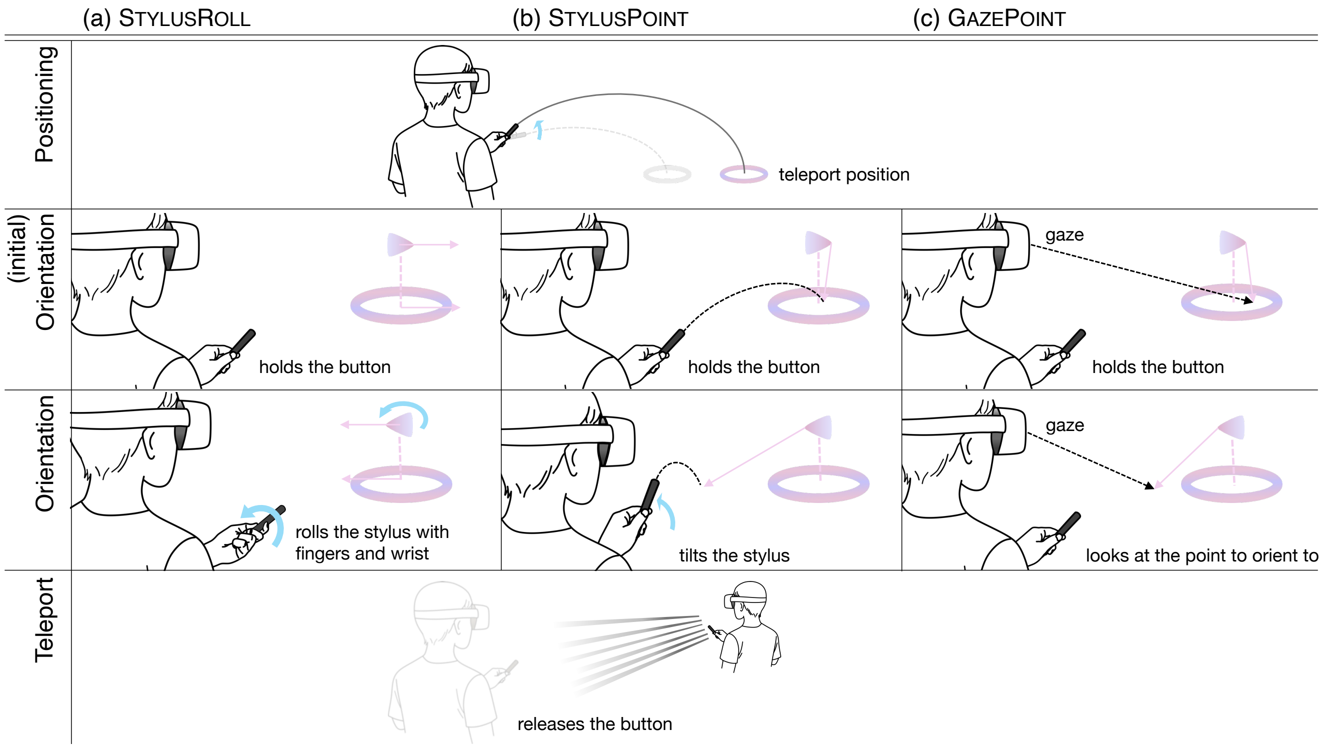


Figure 2: Interaction steps for teleportation with three orientation controls (example with mode switch by stylus flip): StylusRoll, StylusPoint, and GazePoint. After the mode switch, the user first confirms a teleport position by pressing and holding the teleport button. Then, they specify the orientation by either (a) rolling the stylus, (b) pointing the stylus toward the desired facing direction, or (c) directing their gaze toward the desired facing direction. Releasing the button triggers the teleport. Note that the blue arrows are for illustrative purposes and are not displayed to users.

becomes fixed. Then, a second parabolic ray–with the same parameters as the positioning parabola–is cast to define an orientation cursor (cf. Figure 2b). The cursor appears at the intersection point with any surface, including the ground and objects. The orientation is defined by the horizontal vector from the destination to the orientation cursor. Upon button release, users are teleported to the destination and oriented toward the orientation cursor.

As illustrated in Figure 3, when a user aims to teleport behind an object, the object becomes semi-transparent upon being penetrated by the positioning ray, thereby allowing the orientation ray to also pass through. This thereby ensures a consistent behavior between position and orientation pointing.

The visual feedback of StylusPoint is presented as a 3D arrow above the position cursor pointing in the horizontal direction toward the orientation cursor, which is rendered as a solid circle and connected to the arrow with a line (cf. Figure 2b). Since the orientation cursor can be cast on any surface, displaying it along with a connection line facilitates cursor placement and orientation perception. To reduce clutter, the orientation parabolic ray is not rendered.

3.2.3 GazePoint. GazePoint extends StylusPoint with multimodal input by using gaze for orientation, instead of a stylus-based pointing parabolic ray. This method leverages the natural behavior

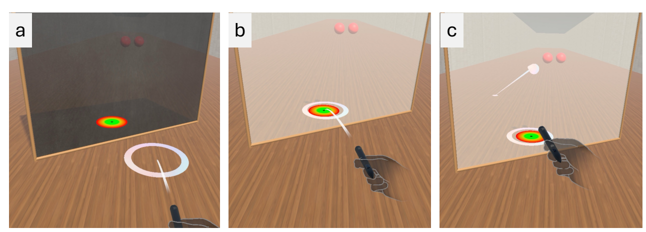


Figure 3: When teleporting behind an object, the user positions the cursor so that the ray intersects the object. The object then becomes semi-transparent, allowing the orientation ray to pass through.

of gaze, as users typically look at where they wish to face before teleporting. After a position is selected via stylus pointing and confirmed by pressing the button, users specify their post-teleport orientation using gaze during button holding. A gaze ray is continuously cast and can intersect with any surface, including the ground and objects, similar to the orientation ray in StylusPoint. The orientation is defined by the horizontal vector from the destination position to the gaze cursor. Upon button release, users are teleported to the selected position and oriented. GazePoint's visuals are identical to StylusPoint, illustrated in Figure 2c.

4 User Study

We evaluate the mode switching and teleportation orientation techniques using a teleport-and-orient task.

4.1 Study Design

We designed a within-subject experiment with two independent variables: Mode Switching (Button, Flip) and Orientation Control (StylusRoll, StylusPoint, GazePoint). To reduce order effects, we counterbalanced all conditions using a Latin Square. Participants first experienced the Orientation Control conditions in a counterbalanced order, and within each Orientation Control condition, they completed both Mode Switching conditions in counterbalanced order. Each Mode Switching × Orientation Control combination consisted of 10 trials (2 target depths × 5 target orientations). We repeated each trial set twice, resulting in a total of 2 target depths × 5 target orientations × 2 mode switching × 3 orientation controls × 2 repetitions = 120 trials per participant.

4.2 Task

We designed the task in a VR blackboard drawing scenario where stationary, seated participants teleport. Participants are immersed in a long corridor ($70m \times 8m$) with wooden flooring, walls, and ceiling. They press a start button with the stylus to begin a trial. This reveals a transparent blackboard (88% alpha, $2m \times 1.5m$) which appears at two possible distances ($3m$ and $6m$) and five rotation angles (45°, −45°, 90°, −90°, 180°). Two red spheres on the blackboard's front face are positioned $1m$ above the floor and $30cm$ apart from each other. A gradient circle with a black center marks the teleport target $50cm$ in front of the blackboard (cf. Figure 3). For a trial, participants 1) switch mode from draw to teleportation, 2) direct the positioning parabola to the target circle, 3) press the teleport button to activate orientation control, 4) orient towards the midpoint between the two spheres, 5) release the teleport button to teleport, 6) switch back to draw mode and connect both spheres in one stroke (successful trial) or draw in mid-air (failed). After 1.5s, the next trial begins. Figure 4 shows a full trial, Figure 3 shows a 45° trial, and Figure 5 shows a −90° trial.

4.3 Apparatus and Implementation

We implemented with Unity on the Meta Quest 3, which supports the Logitech MX Ink stylus[1]. The stylus is 164 mm long, 18.2 mm in diameter, and weighs 29 g. (cf. Figure 6). Drawing was performed by pressing the front button. With Button, users switched modes by clicking the rear button and teleported with the front button. With Flip, users flipped the stylus to switch modes, and either button teleported. Teleportation was implemented using the Meta XR Interaction SDK (v77) Teleport Interaction. The stylus and the user's hand models were rendered in the scene.

Because Meta Quest 3 does not natively support eye-tracking, we fitted a third-party Neon XR eye tracker from Pupil Labs[2] (100 Hz). Gaze was smoothed with a 10-sample window to reduce noise. The eye tracker is reported to have an accuracy of around 1.3 – 1.8° in a 2D-screen setup (outdoors degrading by 0.2° – 0.4°) [5]. We conducted a separate accuracy test in VR based on [5] using 9 targets spaced 20° apart from the center at 5 depths (0.5–2.5 m), finding an average accuracy of 2.68° (SD = 2.24°). The Neon XR provides offset correction to adjust global positional offset of the estimated gaze. We therefore presented a single-point target at a distance of $6m$ to participants, adjusting their gaze offset until they reported accurate eye-tracking.

4.4 Procedure

Participants were briefed on the study context and completed consent and demographics forms. They were shown a video of the techniques, along with an explanation of the study setup and a demonstration of how to use the stylus. For StylusRoll, participants were shown that they could roll the stylus either by using the fingers or their wrist. Participants were seated on a static chair to fucus on effects of orientation, minimizing effects of physical orientation. They donned the XR headset and held the MX Ink stylus in their dominant hand with a precision grip. Participants were then presented with a training session matching the task design and received assistance as needed. With GazePoint, they completed gaze offset correction. Users completed at least six training trials, and more if deemed needed by the experimenter. After each condition, they were presented with a post-condition NASA-TLX questionnaire. After all conditions, the participant was presented with a final post-study questionnaire, consisting of 7-point preference ratings for all conditions along with a short interview. The study lasted one hour on average.

4.5 Evaluation Metrics

- Switch In Time: time taken from trial start to switching from draw mode to teleportation mode.
- Positioning Time: time taken to specify a teleportation position after entering teleportation mode.
- Orientation Time: time taken to specify an orientation after specifying the position.
- Switch Out Time: time taken to switch back from teleportation mode to draw mode after teleporting.
- Task Completion Time: time taken to complete a task, from pressing the start button to switching back to draw mode after teleport. Drawing time is excluded as not of interest to the research questions.
- Positioning Error: positional offset (meters) between the teleport and the target center positions.
- Orientation Error: angular offset (degrees) between the user's forward orientation and the user direction to the midpoint of the two spheres.
- NASA-TLX [20] in the form of Raw-TLX questionnaires [9, 19].
- Technique ratings on a 7-point Likert scale (1 – least preferred, 7 – most preferred) and a brief interview.

4.6 Participants

We recruited 18 participants (10 male, 7 female, and 1 non-binary) from the local university, mainly Master's students in Computer Science. The ages ranged from 18 to 34, all were right-handed, 8 wore glasses, and 1 wore contact lenses. On a 5-point Likert scale, participants rated themselves as having medium experience

[1]https://www.logitech.com/en-us/products/vr/mx-ink.html
[2]https://pupil-labs.com/products/vr-ar

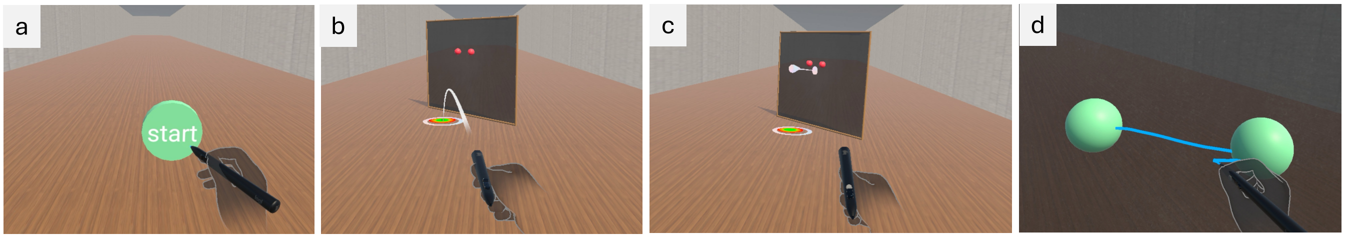


Figure 4: Study example with Flip+StylusPoint: (a) The user taps the button with the stylus to start, (b) they select a teleport destination, (c) they specify orientation while pressing the button, (d) release to teleport and draw a line to finish.

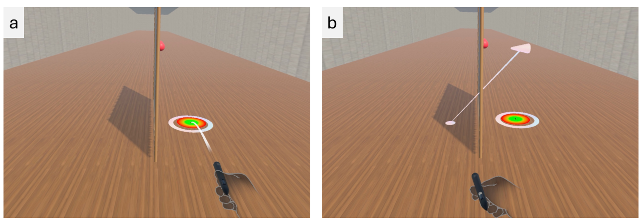


Figure 5: When teleporting to face a blackboard presented from its side, it can be difficult to cast the orientation point onto its thin edge. A practical strategy is to instead place the orientation point on the ground behind the blackboard.

with VR ($M = 3.17, SD = 1.04$), tablets/smartphones styluses ($M = 3, SD = 0.91$), and little experience with VR styluses ($M = 1.39, SD = 0.78$) and teleportation ($M = 1.61, SD = 0.78$).

5 Results

We conducted a three-way (Mode Switching × Orientation Control × Target Depth) Aligned Rank Transform (ART) Repeated Measures ANOVA for data analysis, as measures were non-normally distributed after outlier filtering (Inter-Quartile Range based on Task Completion Time; ≈5.3%), followed by Holm-Bonferroni corrected post hoc tests. For NASA-TLX and preference ratings, we performed Friedman with post hoc Wilcoxon tests (Holm-Bonferroni corrected). Statistical significances in Figure 7-Figure 11 are shown as */**/*** for $p < .05/p < .01/p < .001$ and error bars show 95% confidence intervals.

5.1 Time Measures

On Switch In Time (Figure 7a), we found that participants switched into teleportation mode faster when performing the Flip ($F_{1,187} = 69.25, p < 0.001, \eta_p^2 = 0.270$) than when pressing the Button.

As for Positioning Time (Figure 7a-b), we found that participants were faster with Button compared to Flip ($F_{1,187} = 6.59, p = 0.011, \eta_p^2 = 0.034$). Participants also took longer to position $6m$ away compared with $3m$ ($F_{1,187} = 35.63, p < 0.001, \eta_p^2 = 0.160$).

Regarding Orientation Time (Figure 8), we found that participants specified the orientation faster when using Button ($F_{1,187} = 20.54, p < 0.001, \eta_p^2 = 0.099$). Orientation Control also had an effect ($F_{2,187} = 30.56, p < 0.001, \eta_p^2 = 0.246$), as participants were slowest when using StylusRoll compared with the others (both $p < 0.001$), while participants were faster when using StylusPoint compared with GazePoint ($p = 0.008$). Additionally, a significant interaction was found between Mode Switching × Orientation Control ($F_{2,187} = 4.38, p = 0.014, \eta_p^2 = 0.045$). Specifically, StylusRoll was significantly more impacted by the grip in Flip than both StylusPoint and GazePoint (both $p < 0.034$).

When looking at Switch Out Time (Figure 9a-c), we found that participants were significantly faster when using Flip compared with Button ($F_{1,187} = 195.62, p < 0.001, \eta_p^2 = 0.511$). Target Depth also impacted Switch Out Time as participants were faster at switching out at the closer distance of $3m$ ($F_{1,187} = 4.01, p = 0.047, \eta_p^2 = 0.021$).

Further, we found significant interactions between Mode Switching × Target Depth ($F_{1,187} = 6.67, p = 0.011, \eta_p^2 = 0.034$). Specifically, Button was significantly more impacted by depth than Flip ($p = 0.011$).

Finally, on Task Completion Time (Figure 9d-e), we found that Orientation Control had a significant effect ($F_{2,187} = 9.34, p < 0.001, \eta_p^2 = 0.091$), as StylusRoll was significantly slower overall than the others (both $p < 0.018$). Participants were also slower overall when teleporting to $6m$ away ($F_{1,187} = 14.95, p < 0.001, \eta_p^2 = 0.074$).

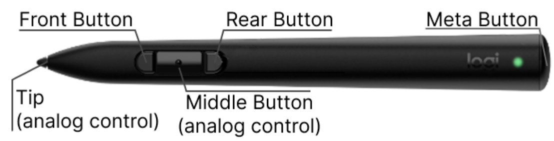
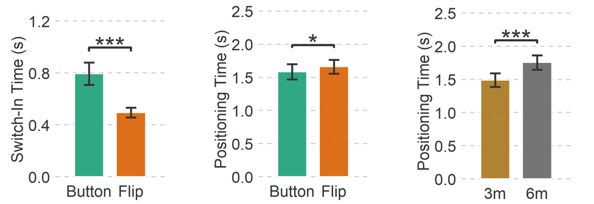


Figure 6: Logitech MX Ink stylus controls. Figure adapted with permission from [26].

5.2 Error Measures

On Positioning Error (Figure 10), we found that participants were more accurate with Flip ($F_{1,187} = 28.86, p < 0.001, \eta_p^2 = 0.134$). Target Depth also impacted Positioning Error, as the performance was more accurate at the close depth of $3m$ ($F_{1,187} = 25.87, p < 0.001, \eta_p^2 = 0.122$). Furthermore, we found significant interactions between Mode Switching × Orientation Control ($F_{2,187} = 4.33, p < 0.015, \eta_p^2 = 0.044$), Mode Switching × Target Depth ($F_{1,187} = 14.50, p < 0.001, \eta_p^2 = 0.072$), Orientation Control × Target Depth ($F_{2,187} = 4.015, p < 0.020, \eta_p^2 = 0.041$), and Mode Switching × Orientation Control × Target Depth

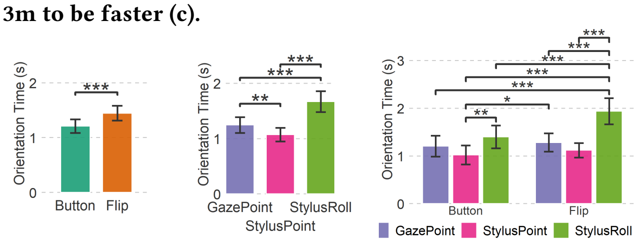


(a) Mode Switching on Switch In Time.

(b) Mode Switching on Positioning Time.

(c) Target Depth on Positioning Time.

Figure 7: Results of on Switch In Time (a) and Positioning Time (b-c). Notably, main effects show Flip to switch in faster (a) while being slower at positioning (b) with positioning at 3m to be faster (c).

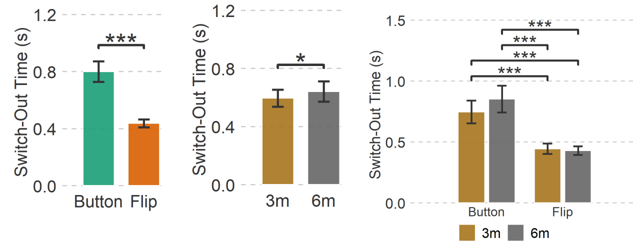


(a) Mode Switching.

(b) Orientation Control.

(c) Mode Switching × Orientation Control.

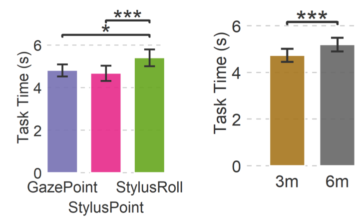
Figure 8: Results of Orientation Time. Notably, main effects show Button to be faster (a). Meanwhile, StylusPoint is faster than both GazePoint and StylusRoll, with StylusRoll being the slowest (b).



(a) Mode Switching on Switch Out Time.

(b) Target Depth on Switch Out Time.

(c) Mode Switching × Target Depth on Switch Out Time.



(d) Orientation Control on Task Completion Time.

(e) Target Depth on Task Completion Time.

Figure 9: Results of Switch Out Time (a-c) and Task Completion Time (d-e). Notably, main effects show Flip (a) and 3m (b) to switch out faster, while showing StylusRoll to be the slowest overall (a) and teleporting at 3m to be faster overall (c).

($F_{2,187} = 3.97, p < 0.021, \eta_p^2 = 0.041$). Specifically, StylusRoll was more impacted by Button than both StylusPoint and GazePoint ($p < 0.041$). Positioning error at $6m$ was also significantly more impacted by Button ($p < 0.001$), while StylusRoll was more impacted by Target Depth than GazePoint ($p = 0.017$). Finally, from the three-way interaction, GazePoint was more accurate than StylusRoll when using Flip while teleporting $3m$ away ($p = 0.019$).

Regarding Orientation Error (Figure 11a–c), we found that orientation was more accurate with Flip than Button ($F_{1,187} = 28.02, p < 0.001, \eta_p^2 = 0.130$) and at $3m$ compared with $6m$ ($F_{1,187} = 54.54, p < 0.001, \eta_p^2 = 0.189$). Additionally, we found significant interactions between Mode Switching × Orientation Control ($F_{2,187} = 3.22, p = 0.042, \eta_p^2 = 0.033$) and Mode Switching × Target Depth ($F_{1,187} = 6.25, p = 0.013, \eta_p^2 = 0.032$). However, post hoc interaction analyses found no significant differences in the Mode Switching × Orientation Control interaction (all $p > 0.078$). Instead, participants were more impacted by the difference in Target Depth when using Button than when using Flip ($p < 0.013$).

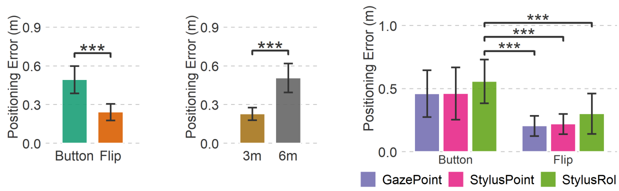
5.3 NASA-TLX, Preferences and User Feedback

The only significant NASA-TLX result was that GazePoint+Button was perceived less physically demanding ($\chi^2(5) = 15.625, p = 0.008, W = 0.935$) than StylusRoll+Flip ($p = 0.044$).

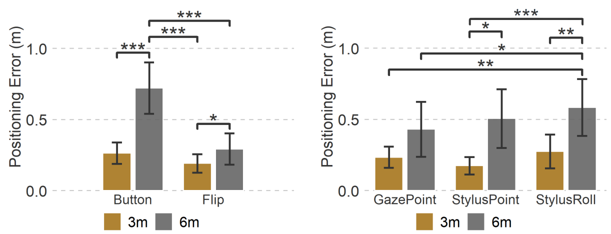
The results on preference (Figure 12) indicated that participants preferred StylusPoint (with both Button and Flip) over StylusRoll+Button (both $p < 0.037$). The feedback showed that they found all techniques viable for completing the task, but with trade-offs between Mode Switching and Orientation Control.

5.3.1 Button vs. Flip. The two Mode Switching methods divided opinions, reflecting a trade-off between intuitiveness and reliability. Flip was praised for its intuitiveness and reduced mental load since there was *"no need to memorize [the] current mode"* (P3) and *"I used less time [than] figuring out what the two buttons do"* (P5). However, some participants found Flip suffered from *"awkward grips"* (P17) and made buttons *"not very handy after reversing"* (P11). P7 even worried about the stylus *"falling down"* when performing flips. In contrast, Button was seen as *"more stable"* (P1, P15, P18) and provided *"button feedback (click) so I know the mode has been switched"* (P7).

5.3.2 StylusRoll. StylusRoll was perceived as *"precise"* (P3, P11), but most participants (10 out of 18) criticized its physical demands. The most frequently mentioned problem was the strain and discomfort in their wrists, especially with *"half circle (180°)"* rotations (P1, P8, P13), or with *"clockwise rotations"* (P8). Although StylusRoll only maps roll-axis rotation to orientation, some participants still rotated the stylus freely along all axes to overcome the limited wrist rotational range. P9 noted that *"I had to give up the precision for some large angles because my wrist was limited"*.

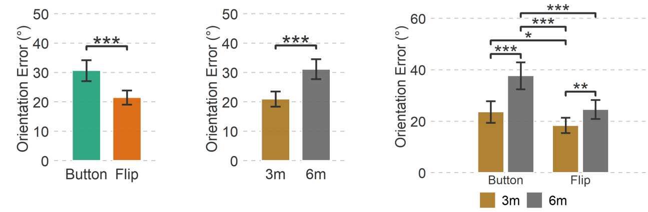


(a) Mode Switching. (b) Target Depth. (c) Mode Switching × Orientation Control.



(d) Mode Switching × Target Depth. (e) Orientation Control × Target Depth.

Figure 10: Results of Positioning Error. Notably, main effects show Flip (a) and teleporting at 3m (b) to be more accurate.



(a) Mode Switching (b) Target Depth (c) Mode Switching × Target Depth.

Figure 11: Results of Orientation Error (a-c). Notably, main effects show that Flip (a) and 3m (b) resulted in higher accuracy.

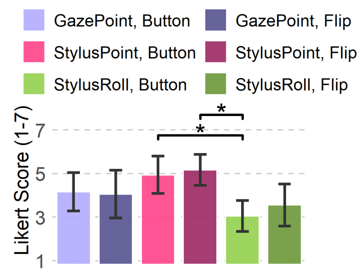


Figure 12: Preference results. Notably, main effects show that the StylusPoint variants are preferred by users over Button+StylusRoll.

In addition, holding the teleport button while rolling the stylus sometimes caused accidental releases (P9, P16, P17).

5.3.3 StylusPoint. Controlling orientation by pointing with the stylus was considered the most *"natural"*, *"intuitive"*, and to provide *"great control"* (P3, P6, P7, P9, P10, P12, P13, P15, P18). Participants reported that it was *"much easier to hit the right rotation"* (P15). Compared with StylusRoll, participants found that StylusPoint caused less physical strain, as it was *"much more efficient when the blackboard was reversed"* (P7, P16) and *"became much easier to line up the orientation when being able to use all axes of the stylus"* (P15).

5.3.4 GazePoint. The gaze-based orientation method received mixed comments. Some praised it as *"the fastest"* (P1, P3, P7), involving *"less physical strain"* for the hand and being *"effortless"* (P7, P13, P14, P16, P17). Others found it *"fun"* (P4, P12, P13, P14), and *"very intuitive"* (P3, P4, P13, P16, P17). However, several struggled with *"eye-tracking [that] wasn't very accurate"* (P5, P9, P10, P13, P16, P17). Participants with glasses especially mentioned *"eye fatigue"* (P8, P10, P12) and that *"the eye tracker was pressing on my glasses uncomfortably"* (P17).

5.4 Observations

When switching modes via Flip, some participants reported struggling to change between the precision and palm grips. Though we anticipated a fluid transition that exploits finger dexterity, Flip was limited by individual differences in hand size and motor ability across participants. To prevent the stylus from slipping, some participants maintained the precision grip but rotated their wrist largely to point with the stylus tail, or even tried using their other hand to assist.

We also noted the hardware limitations of the Neon eye tracker. As a mounted component between the lenses and the user, it pressed against participants' glasses and caused discomfort. Though some participants who wore glasses reported eye tracking inaccuracy in GazePoint, statistical analyses (ANOVA for normal, Kruskal-Wallis for non-normal) did not indicate significant differences in any performance metrics between the groups of participants with or without glasses. Therefore, we claim that despite discomfort, Neon XR provides reliable eye tracking quality for our study.

6 Discussion

In summary, we can address **RQ1**: Flip is faster to perform and results in higher accuracy in subsequent teleportation. To address **RQ2**, we can conclude that StylusRoll is the most challenging for orientation control, while StylusPoint was most preferred by participants and was more efficient at orientation than GazePoint.

As guidelines for designing teleportation in stylus-based VR interfaces, we suggest assigning frequently-used functions to grip-based methods like Flip, which enables rapid switching, while binding less frequent functions to Button. Among orientation methods, StylusRoll is the least recommended, while StylusPoint becomes the most suitable choice for the system we tested. GazePoint achieves comparable task performance to StylusPoint, demonstrating its potential as an alternative orientation method.

Potentially, a higher-accuracy eye-tracker can improve user performance with this method.

In the following sections, we elaborate on these findings and discuss the study's limitations and avenues for future work.

6.1 Mode Switching Techniques

A qualitative advantage of FLIP over BUTTON is that it does not rely on extra buttons. In contrast, the BUTTON approach requires an extra mode-switch button, which may take away space for other operations. Yet, a button press is simple and efficient in principle compared with gestures. Interestingly, our results show that switching between modes through the FLIP was faster. We attribute this to intuitive action of flipping, as a natural behavior that is similar to pen use in physical contexts, compared with the arbitrary task of memorizing the current mode along with locating and pressing a specific button using a digital stylus, which requires dedicated visual feedback.

Furthermore, while the act of positioning and orientation were slower using FLIP, it achieved higher accuracy with comparable overall task time. The longer teleportation time occurs because with FLIP, participants needed to switch grips and adapt to the palm grip–e.g., locating the teleport button–while BUTTON maintained a static precision grip. The higher accuracy, however, corroborates Li et al.'s findings on stylus grips in VR: while pointing forward and downward is easier with a precision grip, pointing forward and upward is easier with a palm grip [31]. Since parabola-based Point & Teleport requires raising the stylus higher for distant targets, it becomes easier to use the palm grip after FLIP than continuing with the precision grip in BUTTON. This is further supported by that BUTTON was more impacted by depth, with slower switching-out and lower orientation accuracy for far targets This indicates that the difficulty of holding the stylus higher with a precision grip reduces users' ability to finely control orientation.

6.2 Orientation Control

For the different orientation control methods, our results show that participants took longer to complete the compound teleport-and-orient task with STYLUSROLL than with the other two methods. This difference mainly occurs when orienting, where STYLUSROLL was the slowest for orientation and STYLUSPOINT was the fastest. We interpret these results as evidence that rolling the stylus to specify orientation is harder to perform because it requires fine hand dexterity to manipulate the stylus efficiently or effectively while holding it. This difficulty is augmented by the need to simultaneously press the teleport button, which restricts in-hand finger rotation and lead to more reliance on wrist rotation. It is in line with other findings indicating its inherent difficulty [8].

In contrast, pointing via the stylus or gaze offers larger movement spaces and greater freedom, making orientation easier. Similarly, results for positioning errors indicate that STYLUSROLL produced worse accuracy when used together with BUTTON, suggesting an additional challenge of rolling a stylus in a precision grip, which offers more limited rolling motion space than a palm grip.

Regarding GAZEPOINT, although six participants reported eye-tracking inaccuracy, the orientation task itself does not require high-precision pointing, since users can physically fine-tune their orientation afterwards. Consequently, GAZEPOINT outperformed STYLUSROLL and showed no significant differences from STYLUSPOINT in overall task time. Looking one level deeper, we attribute this robustness to our design, which allows the orientation ray to intersect any surface rather than only the ground. By moving away from the strict assumption that teleport direction must be specified on the ground—and instead allowing orientation to be defined via arbitrary spatial objects—we leverage natural gaze behaviour: users tend to look toward the objects they want to inspect or interact with.

6.3 Limitations and Future Work

Our explorations are grounded in the common Point & Teleport approach, which defines teleportation position through hand (stylus) pointing. However, it is also possible to use gaze pointing for position specification [27], and it remains to be studied whether both position and orientation could be specified with gaze. We conducted the study using a Meta Quest 3 with a Neon XR eye tracker, as it was the most appropriate hardware setup available for a stylus. It caused some discomfort and can be improved with better mounts or future devices that integrate eye-tracking and stylus. Some of our proposed techniques support setting orientation through pointing at object surfaces in 3D space, which diverges from most prior work that focused solely on ground-based teleportation, making it less comparable to prior approaches. Our study was conducted in a controlled laboratory setting, where participants could use only a limited set of surfaces for orientation, such as the blackboard, ground, and walls. Future work is needed to examine how these techniques translate to more realistic uses of a stylus involving more interactive objects and crowded environments. Also as FLIP relies on stylus orientation, occasional unintentional flips can occur, e.g., when drawing on tables. Context-aware cues such as disabling FLIP near surfaces have potential to address this. The study focused on two subtasks of positioning and orientation, though in many cases, users may only want to specify the position. It is plausible that the findings for the two mode-switching methods extend to purely teleportation positioning, which could be validated by future work. Further, the study task required participants to specify a single teleport position and orientation per trial. While this sufficiently captures the fundamental teleportation task and likely extends to scenarios involving multiple successive teleports, further studies are needed to validate this assumption and generate additional insights.

7 Conclusion

In this paper, we investigate teleportation for stylus-based interaction in VR through two mode-switching (BUTTON and FLIP gesture) and three orientation techniques (STYLUSROLL, STYLUSPOINT, and GAZEPOINT). We evaluated these techniques in a compound point, orient, and teleport task. Our key takeaways are: (1) flipping the stylus to enter teleport mode is intuitive, makes users switch modes faster, and induces more accurate though slightly slower teleporting; (2) setting the landing orientation by pointing in the desired facing direction is easy to understand and more efficient than adjusting orientation via stylus roll. Pointing works with both a stylus-ray and gaze; these were on par in our tests, and more accurate eye-tracking could improve gaze-based orientation.

The interaction concepts can extend beyond styluses to controller- and bare-hand-based interaction. Controller teleportation already benefits from multiple buttons, but flip gestures and pointing for teleport orientation may offer additional flexibility. Bare-hand-based teleportation is underexplored, and opportunities may lie in leveraging flip-like gestural mode-switching and multimodal integration with gaze. This hints at a broader design space for advancing how users navigate virtual environments with intuitive multimodal inputs to explore in the future.

Acknowledgments

This work was supported by the Independent Research Fund Denmark (grant number 5254-00080B), a Google Research Gift award ("Multimodal and Gaze + Gesture Interactions in XR"), and the Danish National Research Foundation under the Pioneer Centre for AI in Denmark (DNRF grant P1).

References

[1] Arkio. 2025. Experience the unbuilt, together. https://www.arkio.is/
[2] Rahul Arora, Rubaiat Habib Kazi, Tovi Grossman, George Fitzmaurice, and Karan Singh. 2018. SymbiosisSketch: Combining 2D & 3D Sketching for Designing Detailed 3D Objects in Situ. 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, 1–15. doi:10.1145/3173574.3173759
[3] Rahul Arora, Rubaiat Habib Kazi, Fraser Anderson, Tovi Grossman, Karan Singh, and George Fitzmaurice. 2017. Experimental Evaluation of Sketching on Surfaces in VR. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 5643–5654. doi:10.1145/3025453.3025474
[4] Anil Ufuk Batmaz, Aunnoy K Mutasim, and Wolfgang Stuerzlinger. 2020. Precision vs. Power Grip: A Comparison of Pen Grip Styles for Selection in Virtual Reality. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. 23–28. doi:10.1109/VRW50115.2020.00012
[5] Chris Baumann and Kai Dierkes. 2023. Neon Accuracy Test Report. doi:10.5281/zenodo.10420388
[6] Xiaojun Bi, Tomer Moscovich, Gonzalo Ramos, Ravin Balakrishnan, and Ken Hinckley. 2008. An exploration of pen rolling for pen-based interaction. In *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology* (Monterey, CA, USA) *(UIST '08)*. Association for Computing Machinery, New York, NY, USA, 191–200. doi:10.1145/1449715.1449745
[7] Pauline Bimberg, Tim Weissker, Alexander Kulik, and Bernd Froehlich. 2021. Virtual Rotations for Maneuvering in Immersive Virtual Environments. In *Proceedings of the 27th ACM Symposium on Virtual Reality Software and Technology (VRST '21)*. Association for Computing Machinery, New York, NY, USA, 1–10. doi:10.1145/3489849.3489893
[8] Evren Bozgeyikli, Andrew Raij, Srinivas Katkoori, and Rajiv Dubey. 2016. Point & Teleport Locomotion Technique for Virtual Reality. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play (CHI PLAY '16)*. Association for Computing Machinery, New York, NY, USA, 205–216. doi:10.1145/2967934.2968105
[9] Ernesto A. Bustamante and Randall D. Spain. 2008. Measurement Invariance of the Nasa TLX. *http://dx.doi.org/10.1177/154193120805201946* 3 (9 2008), 1522–1526. doi:10.1177/154193120805201946
[10] Ang Cai, Chao Yao, Hongjun Liu, Changsheng Li, Yongyue Zhang, Yu Guo, Xiaokun Wang, and Xiaojuan Ban. 2025. HPIPainting: A Hand-Pen Interaction for VR Painting. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 9, 3, Article 71 (Sept. 2025), 26 pages. doi:10.1145/3749538
[11] Drini Cami, Fabrice Matulic, Richard G. Calland, Brian Vogel, and Daniel Vogel. 2018. Unimanual Pen+Touch Input Using Variations of Precision Grip Postures. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (Berlin, Germany) *(UIST '18)*. Association for Computing Machinery, New York, NY, USA, 825–837. doi:10.1145/3242587.3242652
[12] Chen Chen, Matin Yarmand, Zhuoqun Xu, Varun Singh, Yang Zhang, and Nadir Weibel. 2022. Investigating Input Modality and Task Geometry on Precision-first 3D Drawing in Virtual Reality. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 384–393. doi:10.1109/ISMAR55827.2022.00054
[13] Lucia A. Cherep, Alex F. Lim, Jonathan W. Kelly, Devi Acharya, Alfredo Velasco, Emanuel Bustamante, Alec G. Ostrander, and Stephen B. Gilbert. 2020. Spatial cognitive implications of teleporting through virtual environments. *Journal of Experimental Psychology: Applied* 26, 3 (2020), 480–492. doi:10.1037/xap0000263
[14] Michael F. Deering. 1995. HoloSketch: a virtual reality sketching/animation tool. *ACM Trans. Comput.-Hum. Interact.* 2, 3 (Sept. 1995), 220–238. doi:10.1145/210079.210087
[15] Tobias Drey, Jan Gugenheimer, Julian Karlbauer, Maximilian Milo, and Enrico Rukzio. 2020. VRSketchIn: Exploring the Design Space of Pen and Tablet Interaction for 3D Sketching in Virtual Reality. 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, 1–14. doi:10.1145/3313831.3376628
[16] ENGAGE. 2025. Spatial Computing Artificial Intelligence - ENGAGE XR. https://engagevr.io/
[17] Markus Funk, Florian Müller, Marco Fendrich, Megan Shene, Moritz Kolvenbach, Niclas Dobbertin, Sebastian Günther, and Max Mühlhäuser. 2019. Assessing the Accuracy of Point & Teleport Locomotion with Orientation Indication for Virtual Reality using Curved Trajectories. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, Glasgow Scotland Uk, 1–12. doi:10.1145/3290605.3300377
[18] Martin Hachet, Fabrice Decle, Sebastian Knodel, and Pascal Guitton. 2008. Navidget for Easy 3D Camera Positioning from 2D Inputs. In *2008 IEEE Symposium on 3D User Interfaces*. 83–89. doi:10.1109/3DUI.2008.4476596
[19] Sandra G. Hart. 2006. Nasa-Task Load Index (NASA-TLX); 20 Years Later. *http://dx.doi.org/10.1177/154193120605000909* (10 2006), 904–908. doi:10.1177/154193120605000909
[20] Sandra G. Hart and Lowell E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. *Advances in Psychology* 52, C (1 1988), 139–183. doi:10.1016/S0166-4115(08)62386-9
[21] Meta Horizon. 2025. Locomotion types. https://developers.meta.com/horizon/design/locomotion-types/
[22] Rob Jacob and Sophie Stellmach. 2016. What you look at is what you get: gaze-based user interfaces. *Interactions* 23, 5 (Aug. 2016), 62–65. doi:10.1145/2978577
[23] Robert J. K. Jacob. 1990. What You Look at is What You Get: Eye Movement-Based Interaction Techniques. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Seattle, Washington, USA) *(CHI '90)*. Association for Computing Machinery, New York, NY, USA, 11–18. doi:10.1145/97243.97246
[24] Sei Kang, Jaejoon Jeong, Gun A. Lee, Soo-Hyung Kim, Hyung-Jeong Yang, and Seungwon Kim. 2024. The RayHand Navigation: A Virtual Navigation Method with Relative Position between Hand and Gaze-Ray. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 634, 15 pages. doi:10.1145/3613904.3642147
[25] Daniel Keefe, Robert Zeleznik, and David Laidlaw. 2007. Drawing on Air: Input Techniques for Controlled 3D Line Illustration. *IEEE Transactions on Visualization and Computer Graphics* 13, 5 (2007), 1067–1081. doi:10.1109/TVCG.2007.1060
[26] Aidan Kehoe, Mario Gutierrez, Yena Ahn, and Vadim Kogan. 2025. Logitech MX Ink: Extending the Meta Quest Platform with a 6DoF Stylus. In *2025 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. 338–342. doi:10.1109/VRW66409.2025.00080
[27] Jinwook Kim, Hyunyoung Jang, Dooyoung Kim, and Jeongmi Lee. 2023. Exploration of the Virtual Reality Teleportation Methods Using Hand-Tracking, Eye-Tracking, and EEG. *International Journal of Human–Computer Interaction* 39, 20 (Dec. 2023), 4112–4125. doi:10.1080/10447318.2022.2109248
[28] Sebastian Knödel, Martin Hachet, and Pascal Guitton. 2008. Navidget for immersive virtual environments. In *Proceedings of the 2008 ACM symposium on Virtual reality software and technology (VRST '08)*. Association for Computing Machinery, New York, NY, USA, 47–50. doi:10.1145/1450579.1450589
[29] 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 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 375, 14 pages. doi:10.1145/3613904.3642838
[30] Jaeyoon Lee, Hanseob Kim, Yechan Yang, and Gerard Jounghyun Kim. 2024. RPG: Rotation Technique in VR Locomotion using Peripheral Gaze. *Proc. ACM Hum.-Comput. Interact.* 8, ETRA, Article 235 (May 2024), 19 pages. doi:10.1145/3655609
[31] Nianlong Li, Teng Han, Feng Tian, Jin Huang, Minghui Sun, Pourang Irani, and Jason Alexander. 2020. Get a Grip: Evaluating Grip Gestures for VR Input using a Lightweight Pen. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. doi:10.1145/3313831.3376698
[32] Yang Li, Ken Hinckley, Zhiwei Guan, and James A. Landay. 2005. Experimental analysis of mode switching techniques in pen-based user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Portland, Oregon, USA) *(CHI '05)*. Association for Computing Machinery, New York, NY, USA, 461–470. doi:10.1145/1054972.1055036
[33] Mathias N. Lystbæk, Thorbjørn Mikkelsen, Roland Krisztandl, Eric J Gonzalez, Mar Gonzalez-Franco, Hans Gellersen, and Ken Pfeuffer. 2024. Hands-on, Hands-off: Gaze-Assisted Bimanual 3D Interaction. In *Proceedings of the 37th Annual*

ACM Symposium on User Interface Software and Technology (Pittsburgh, PA, USA) *(UIST '24)*. Association for Computing Machinery, New York, NY, USA, Article 80, 12 pages. doi:10.1145/3654777.3676331

[34] Fabrice Matulic and Daniel Vogel. 2023. Pen+Touch+Midair: Cross-Space Hybrid Bimanual Interaction on Horizontal Surfaces in Virtual Reality. https://api.semanticscholar.org/CorpusID:259300598

[35] Andrii Matviienko, Florian Müller, Martin Schmitz, Marco Fendrich, and Max Mühlhäuser. 2022. SkyPort: Investigating 3D Teleportation Methods in Virtual Environments. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22)*. Association for Computing Machinery, New York, NY, USA, 1–11. doi:10.1145/3491102.3501983

[36] Mark R Mine. 1995. Virtual environment interaction techniques. *UNC Chapel Hill CS Dept* (1995).

[37] Shohei Mori, Satoshi Hashiguchi, Fumihisa Shibata, and Asako Kimura. 2023. Point & Teleport with Orientation Specification, Revisited: Is Natural Turning Always Superior? *Journal of Information Processing* 31 (2023), 392–403.

[38] Florian Müller, Arantxa Ye, Dominik Schön, and Julian Rasch. 2023. UndoPort: Exploring the Influence of Undo-Actions for Locomotion in Virtual Reality on the Efficiency, Spatial Understanding and User Experience. 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 234, 15 pages. doi:10.1145/3544548.3581557

[39] Bakdauren Narbayev, A K M Amanat Ullah, Jaisie Sin, Patricia Lasserre, and Khalad Hasan. 2025. Exploring Pointing and Confirmation Techniques for Teleportation Across Varying Elevations in Virtual Reality. In *2025 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 1671–1681. doi:10.1109/ISMAR67309.2025.00170

[40] Ken Pfeuffer, Jason Alexander, Ming Ki Chong, and Hans Gellersen. 2014. Gaze-touch: combining gaze with multi-touch for interaction on the same surface. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (Honolulu, Hawaii, USA) *(UIST '14)*. Association for Computing Machinery, New York, NY, USA, 509–518. doi:10.1145/2642918.2647397

[41] Ken Pfeuffer, Jason Alexander, Ming Ki Chong, Yanxia Zhang, and Hans Gellersen. 2015. Gaze-Shifting: Direct-Indirect Input with Pen and Touch Modulated by Gaze. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (Charlotte, NC, USA) *(UIST '15)*. Association for Computing Machinery, New York, NY, USA, 373–383. doi:10.1145/2807442.2807460

[42] Ken Pfeuffer, Benedikt Mayer, Diako Mardanbegi, and Hans Gellersen. 2017. Gaze + pinch interaction 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, 99–108. doi:10.1145/3131277.3132180

[43] Ken Pfeuffer, Lukas Mecke, Sarah Delgado Rodriguez, Mariam Hassib, Hannah Maier, and Florian Alt. 2020. Empirical Evaluation of Gaze-Enhanced Menus in Virtual Reality. In *26th ACM Symposium on Virtual Reality Software and Technology* (Virtual Event, Canada) *(VRST '20)*. Association for Computing Machinery, New York, NY, USA, Article 20, 11 pages. doi:10.1145/3385956.3418962

[44] Leping Qiu, Erin Seongyoon Kim, Sangho Suh, Ludwig Sidenmark, and Tovi Grossman. 2025. MaRginalia: Enabling In-person Lecture Capturing and Note-taking Through Mixed Reality. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems (CHI '25)*. Association for Computing Machinery, New York, NY, USA, Article 141, 15 pages. doi:10.1145/3706598.3714065

[45] Hugo Romat, Andreas Fender, Manuel Meier, and Christian Holz. 2021. Flashpen: A High-Fidelity and High-Precision Multi-Surface Pen for Virtual Reality. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*. 306–315. doi:10.1109/VR50410.2021.00053

[46] Daniel Rupp, Tim Weissker, Matthias Wölwer, Torsten W. Kuhlen, and Daniel Zielasko. 2025. How Far is Too Far? the Trade-Off Between Selection Distance and Accuracy During Teleportation in Immersive Virtual Reality. *IEEE Transactions on Visualization and Computer Graphics* (2025), 1–15. doi:10.1109/TVCG.2025.3632345

[47] Eric Saund and Edward Lank. 2003. Stylus input and editing without prior selection of mode. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology* (Vancouver, Canada) *(UIST '03)*. Association for Computing Machinery, New York, NY, USA, 213–216. doi:10.1145/964696.964720

[48] John Sermarini, Joseph T. KiderJr., Joseph J. LaViolaJr., and Daniel S. McConnell. 2021. A Kinematic Evaluation of Linear and Parabolic Pointing in Virtual Reality. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 65, 1 (2021), 86–90. doi:10.1177/1071181321651131

[49] ShapesXR. 2025. Bring ideas to life in 3D. https://www.shapesxr.com/

[50] Gravity Sketch. 2025. Gravity Sketch. https://gravitysketch.com/

[51] Jesse Smith, Isaac Wang, Winston Wei, Julia Woodward, and Jaime Ruiz. 2020. Evaluating the Scalability of Non-Preferred Hand Mode Switching in Augmented Reality. In *Proceedings of the 2020 International Conference on Advanced Visual Interfaces* (Salerno, Italy) *(AVI '20)*. Association for Computing Machinery, New York, NY, USA, Article 19, 9 pages. doi:10.1145/3399715.3399850

[52] Hyunyoung Song, Hrvoje Benko, Francois Guimbretiere, Shahram Izadi, Xiang Cao, and Ken Hinckley. 2011. Grips and gestures on a multi-touch pen. 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, 1323–1332. doi:10.1145/1978942.1979138

[53] Hemant Bhaskar Surale, Fabrice Matulic, and Daniel Vogel. 2019. Experimental Analysis of Barehand Mid-air Mode-Switching Techniques in Virtual Reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–14. doi:10.1145/3290605.3300426

[54] James N. Templeman, Patricia S. Denbrook, and Linda E. Sibert. 1999. Virtual Locomotion: Walking in Place through Virtual Environments. *Presence: Teleoper. Virtual Environ.* 8, 6 (Dec. 1999), 598–617. doi:10.1162/105474699566512

[55] VictoryXR. 2025. VictoryXR. https://www.victoryxr.com/

[56] Daniel Vogel and Ravin Balakrishnan. 2010. Direct pen interaction with a conventional graphical user interface. *Human–Computer Interaction* 25, 4 (2010), 324–388.

[57] Uta Wagner, Andreas Asferg Jacobsen, Tiare Feuchtner, Hans Gellersen, and Ken Pfeuffer. 2024. Eye-Hand Movement of Objects in Near Space Extended Reality. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. 1–13.

[58] Uta Wagner, Jinwook Kim, Zhikun Wu, Qiushi Zhou, Mario Romero, Alessandro Iop, Tiare Feuchtner, and Ken Pfeuffer. 2025. A Study of Multimodal Pen+ Gaze Interaction Techniques for Shape Point Translation in Extended Reality. In *Proceedings - 2025 IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2025*. doi:10.1109/ISMAR67309.2025.00056

[59] Tim Weissker, Pauline Bimberg, Aalok Shashidhar Gokhale, Torsten Kuhlen, and Bernd Froehlich. 2023. Gaining the High Ground: Teleportation to Mid-Air Targets in Immersive Virtual Environments. *IEEE Transactions on Visualization and Computer Graphics* 29, 5 (May 2023), 2467–2477. doi:10.1109/TVCG.2023.3247114