Fly the Moon to Me: Bimanual 3D Locomotion in Virtual Reality By Manipulating the Position of the Destination Object

Futian Zhang
Cheriton School of Computer Science,
University of Waterloo
Canada
futian.zhang@uwaterloo.ca

Jiawen Stefanie Zhu
jiawenz2@uw.edu
University of Waterloo
Waterloo, Ontario, Canada
University of Washington
Seattle, Washington, USA

Edward Lank
Cheriton School of Computer Science
University of Waterloo
Waterloo, ON, Canada
lank@uwaterloo.ca

Keiko Katsuragawa
Cheriton School of Computer Science
University of Waterloo
Waterloo, ON, Canada
National Research Council Canada
Waterloo, ON, Canada
kkatsuragawa@uwaterloo.ca

Jian Zhao
Cheriton School of Computer Science
University of Waterloo
Waterloo, ON, Canada
jianzhao@uwaterloo.ca

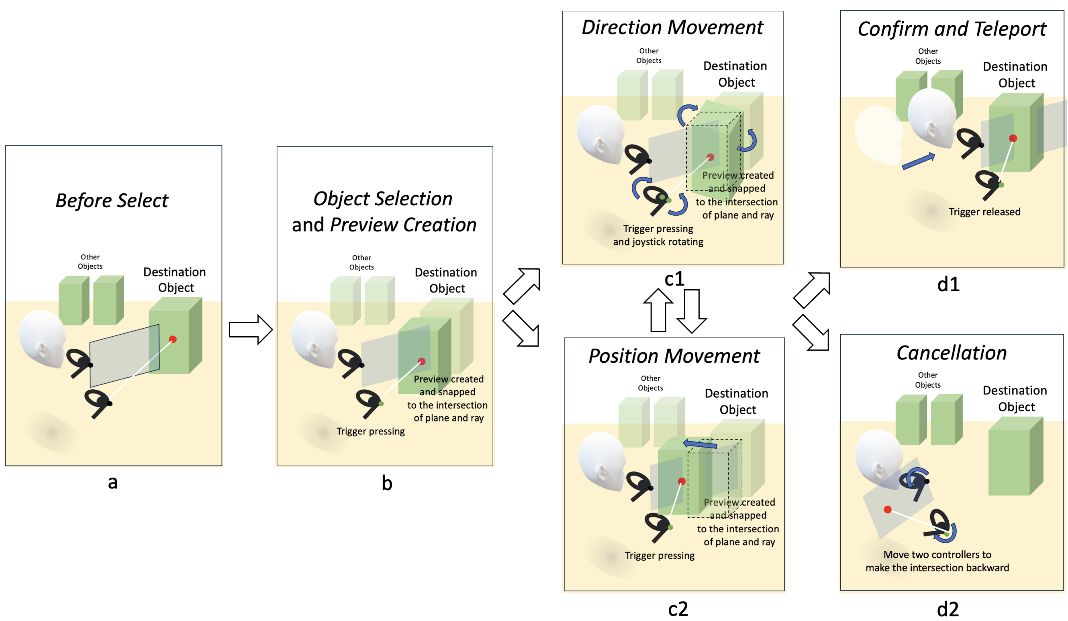

Figure 1: Workflow of Locomoontion, a precise 3D locomotion technique in VR. a) Find the object that the user wants to approach and point with semi-auto Conductor[65] (a 3D cursor technique). b) Press the index trigger on the dominant controller to select the object. Locomoontion will make the world semi-transparent and create a solid copy of the object as a preview of the final teleportation view. The preview will be snapped to the red cursor on intersection of the plane on the non-dominant controller and ray on the dominant controller, as per manual Conductor. c) Users can manipulate the preview by c1) rotating the joystick to rotate the object or c2) moving the object in 3D space with manual Conductor. d) To finalize, users can either d1) release the index trigger, snap the object to the preview, together with the rest of the world, or d2) move the intersection backwards of the ray shooting direction to cancel Locomoontion.

, ,
. 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* .

ABSTRACT

Teleportation - changing the point of view in 3D space by specifying a position - is one of the most common locomotion solutions in VR. However, it currently lacks a mechanism to adjust the height in 3D space, and it is difficult for users to predict the exact final view after the teleportation. Users are relocated to a place without knowing what the final view will look like. As a result, they often need to perform remedial interactions to achieve their ideal position, which can be time-consuming and effort-intensive. In this paper, we present Fly the Moon to Me (Locomoontion), a novel technique that enables users to bring their destination to themselves through object manipulation. Users first create a copy of the object they want to approach as a preview by selecting it, then bring it to an ideal position and direction using existing object manipulation techniques, and then snap the original object to the preview together with the rest of the world. A controlled experiment with 18 participants via a teleportation task reveals that Locomoontion is more effective than the traditional Point&Teleport technique with grabbing the world as a remedy to adjust the final positioning.

CCS CONCEPTS

• **Human-centered computing → Interaction techniques**.

KEYWORDS

Locomotion techniques, virtual reality, interaction technique, controlled experiments.

ACM Reference Format:

Futian Zhang, Jiawen Stefanie Zhu, Edward Lank, Keiko Katsuragawa, and Jian Zhao. . Fly the Moon to Me: Bimanual 3D Locomotion in Virtual Reality By Manipulating the Position of the Destination Object. In *Proceedings of* . ACM, New York, NY, USA, 13 pages.

1 INTRODUCTION

Virtual Reality (VR) is widely used in both recreational and productive domains. Locomotion is one of the elementary interactions in VR [32], which enables users to move in VR and interact with distant virtual objects in a large VR space. Walking by feet, the most immersive [50, 52] and least discomforting [24, 35] VR locomotion technique, could be limited by real-world space constraints, including limited tracking space and physical blockers in the room [21, 60]. Additionally, it is hard to reach different heights of a virtual space by walking only. Several locomotion techniques were designed to mitigate the challenges posed by limited physical space.

Locomotion techniques could be categorised primarily as two types of movement: discrete and continuous [36]. The most popular discrete locomotion in current VR systems is teleportation [7, 10, 17, 36], which allows users to be moved instantly after a destination is selected by pointing to the ground. Other teleportation techniques would involve a preset location, such as portals [14] and gates [23]. It is easy to get hands-on, especially for long-distance movements, but hard to predict post-teleportation appearance. They assumed users could simply walk to their desired location for short-distance adjustments, but these refinement movements can be time-consuming and potentially ergonomically problematic [58]. Such challenges are particularly pronounced for seated users or individuals with mobility impairments. There are precise continuous remedy interactions like Point&Tug [6, 11, 33]. However, this approach may be less efficient, requiring additional physical effort and potentially disrupting workflow. Moreover, traditional point-to-the-ground and teleportation techniques would not support 3D movement at different heights, which requires the remedy interactions to specify the height. With continuous locomotion techniques, users could move in the virtual environment continuously but may experience more motion sickness due to the mismatch of the static physical body and moving world. Therefore, various interactions were designed to mitigate the problem by reducing the visual area to a small area in the centre [4] or asking users to perform some actions such as walking in place [54], flying [38, 47, 67], swimming [25] and cycling [15, 40]. However, they would easily introduce fatigue, thus not being suitable for long-time use. Continuous techniques also tend to require more time for the same locomotion task compared to discrete techniques [58]. Our research question thus is: **How to enable quick and precise 3D discrete locomotion in VR?**

In this work, we present *Locomoontion*, a 3D VR locomotion technique simultaneously enabling location (xy-plane), height (z-axis), and orientation with high precision, fast speed, and low physical effort. Instead of moving the user to the desired position in VR, we employ a novel approach to "bring" the destination object to the user for previewing the teleportation results — fly the moon to me, rather than fly me to the moon. We envision this technique being particularly useful in scenarios that require precise locomotion over visible distances – for example, when the goal of movement is to position oneself for an object manipulation task. The Locomoontion technique comprises the following steps:

- **Selection And Preview:** Users select the target objects that they intend to teleport toward. A copy of the object will be generated as a visual preview, clarifying the object's post-teleportation state.
- **Direction And Position Adjustment:** Users refine the preview's 3D position and orientation to align it with their desired interaction location using Conductor [65], a 3D pointing technique, and direction with the joystick.
- **Teleportation:** Once users finalize the preview's placement, the original object is moved to the preview's confirmed location, followed by the rest of the world.

Locomoontion utilizes Conductor [65], a 3D pointing technique that allows users to point anywhere in the 3D space, to facilitate the 3D position manipulation during the teleportation. With Conductor, the user would control a pointing ray with the dominant hand and a cutting plane attached to the non-dominant hand, and the intersection of the ray and the plane will be the cursor position. For 3D object selection, a Semi-Auto mode was also introduced to facilitate the selection, where the object on the dominant hand ray and closest to the intersection will be selected, and the cursor will be placed in the first occlusion point on the object. In theory, Conductor could reach an arbitrary 3D position in 3D space, but it might be hard to point precisely when the distance is too far. Semi-Auto Conductor has been proven to perform well even with high occlusion.

To assess the performance of Locomoontion, we conducted a controlled experiment with 18 participants to compare our technique with the most popular point and teleport technique for teleportation tasks. We adopted a within-subjects design involving 3 DISTANCE, 3 HEIGHT, and 4 ANGLE conditions. With the current popular point and teleport technique, we conducted a within-subjects experiment with 12 participants with a box positioning task. The independent variables of the study include 3 DISTANCE, 3 HEIGHT, and 4 ANGLE conditions. The result shows that Locomoontion achieved less task completion time for both overall (2.4 seconds faster) and among all conditions, also with less physical fatigue. We also discuss the limitations of Locomoontion and some potential future extensions such as "Fly The Moon Away From Me".

2 BACKGROUND

In this section, we present related work on VR locomotion, especially 3D teleportation techniques, as well as object manipulation techniques in VR. We also introduce Conductor in detail, which Locomootion is based on.

2.1 VR Locomotion Techniques

A large body of VR Locomotion techniques has been explored in the past. Walking, one of the most intuitive ways to travel, provides a continuous way to navigate in VR. Some walking techniques allow users to walk in the real world but with modifications, such as rescaling the walking space [1] or altering the direction mapping between the real and virtual world [2, 31, 41, 45]. While those techniques reduce the necessary space, they still demand unobstructed areas. In order to enable stationary locomotion in VR, in-place locomotion techniques were developed. One of the most significant issues with in-place locomotion techniques is motion sickness [4, 23, 28] resulting from the visual and vestibular mismatch when the user remains still while the VR experience shows they are moving rapidly through a virtual environment. To mitigate motion sickness, some techniques incorporate physical movements during locomotion to convince the mind that the body is in motion. Walking-In-Place [54] is one of the most popular solutions for this design. Devices like treadmills [13, 22, 48] or cycling [15, 40] were also developed for simulating walking. Other body movements like arm rolling [37], jumping [61], finger simulating walking [29, 63], or swimming [25] were also introduced for the compelling purpose. In contrast to continuous locomotion techniques, teleportation [7], a discrete method of movement in VR, typically allows users to specify a point and then instantly move the user there. Certain teleportation methods allow users to indicate the desired direction [7, 17] after teleportation during the process of selecting the teleportation destination. One common way is Point&Teleport, which is pointing at the ground using a ray, and the user will be moved to the collision point. Apart from being triggered by controllers, other body parts like the eye [44], foot [56], and wrist [10] were created for teleportation as well as for bare-hand VR teleportation. Different proxies like gate [23] or portal [14] were also introduced for teleportation. Bimberg et al. [5] proposed a technique that first selects the object of interest and then the teleportation position, automatically orienting the user to face the selected object. However, it does not allow specification of height and does not provide a preview of the final view after teleportation, which may require small positional adjustments. Buttussi et al. [8] compared teleportation, joystick, and leaning-based locomotion techniques in a VR travel task. The result revealed that joystick and leaning-based techniques elicited similar levels of VR motion sickness, while teleportation techniques had no significant impact on the participants' well-being. Additionally, Frommel et al. [16] also found that teleportation provided the least discomfort and the highest scores for enjoyment, presence, and affective state.

2.2 VR Locomotion in 3D

Certain methods were introduced to enable users to navigate thoroughly through virtual 3D environments by incorporating height adjustments during locomotion. Skyport [36] controls a cursor on a ray shooting from the controller, using a touchpad to control the distance of the cursor, and the user would then be teleported to the cursor's position. A coin collection experiment was conducted to evaluate the performance, where the participants needed to move to somewhere within 1 meter away from the coin. They compared two aiming methods (linear and parabolic) and three transition types (instantaneous, interpolated, and continuous) and found that linear aiming and instant transition performed better. However, it cannot control the direction after teleportation, and the precision is limited. Weissker et al. [57] created a preview of the users' avatar and moved it by specifying the 2D position and elevation above the position using the controller. Specifying the 2D position and elevation could be simultaneous, two-step, or Separate, each with its own advantages and disadvantages. However, the final view from the user's perspective remains unknown, and the direction is not supported. Point-Tug [6, 11, 33] techniques, inspired by the ladder-climbing motion, would allow the user to grab the world and tug to move the world. Originally, it would support the x-z plane but not vertically in the y-axis. In our experiment, we extended it to 3D and treated it as a refining step for Point&Teleport technique, combining them and comparing them with Locomoontion technique. Some approaches also explore how to use existing tools or paradigms for 3D movements, like ladders [26, 30, 49], virtual stairs [30], ramps [12], and elevators [55]. Actions like jumping [61], swimming [25], and flying [38, 47, 67] were introduced as well, with the benefit of reducing VR sickness.

2.3 Object Manipulation Techniques

Virtual Hand [39, 43, 59] techniques are considered one of the most natural solutions for manipulating the location and/or rotation of a distant object in VR. The User controls a 3D cursor in VR with controllers or bare hands and translates the offset to the cursor position. However, these techniques often require a significant amount of time, especially for distant objects, which can lead to increased fatigue. OrthoGaze [34] allows the gaze to manipulate the object along three orthogonal planes in VR. Yu et al. [64] also proposed four designs for gaze-supported object manipulation combined with hand motion. However, they didn't support depth changes beyond the user's hand reach. In contrast, World-In-Miniature techniques [42, 53] involve scaling the world and manipulating the world to access objects. Some 3D selection techniques also offer potential for 3D object manipulation, as they can control a cursor in 3D space, not limited to selection only. One solution involves placing a cursor on a ray and using a controller to control the cursor depth on the ray, for example, using a touchpad [3] or a smartphone [46]. There are also bimanual cursor control techniques that use a ray [27, 62] or a plane [65] on the non-dominant controller to intersect the dominant controller's ray and control the cursor accordingly.

2.4 Conductor

Conductor [65] is a 3D pointing technique on which Locomoontion is based. It involves a ray projecting from the dominant hand and a plane attached to the non-dominant hand. Users can control the plane and ray to employ the intersection of the plane and ray for pointing in 3D space. Conductor offers two modes: manual mode and Semi-Auto mode. In manual mode, the cursor will always be positioned at the intersection, and the object closest to the cursor will be selected. In Semi-Auto mode, the object that collides with the ray and is also closest to the intersection will be selected, and the cursor will be snapped to the first collision point of the ray and the object. The selection is performed by the dominant controller's index trigger. The selection performance of Conductor, particularly in Semi-Auto mode, was proven to be faster than RayCursor [3] even under high occlusion and small target conditions. In Locomoontion, we use Semi-Auto Conductor to facilitate quick object selection among all the objects in the world. Once selected, manual Conductor is utilized to move the object (essentially the copy of the object) along the ray. Note that once the object is selected, it will be attached to the cursor, and the cursor will then be snapped to the intersection together with the object, moving with the intersection.

3 LOCOMOONTION TECHNIQUE

Locomoontion is a VR locomotion technique that leverages Conductor [65] to fulfill 3D position manipulation during the process of teleportation. It consists of the following key steps shown in Figure 1. Our vision is that the preview-based teleportation workflow will help reduce the task time, especially when reaching different heights.

Object Selection and Preview Creation (Figure 1-ab): Locomoontion utilizes Semi-Auto Conductor to select the object the user wants to interact with by pressing the dominant controller's index trigger. It automatically creates a copy of the object as a preview. As mentioned earlier, the preview will then be attached to the red cursor, located at the first collision point of the ray and preview. Subsequently, the cursor, along with the preview, is instantly snapped and attached to the intersection of the plane and the ray for position movement.

Position Movement (Figure 1-c2): Once the preview and cursor are snapped and attached to the intersection, the preview will continually follow the intersection of the plane and ray. Users can adjust the preview's position using manual Conductor. The user keeps pressing the index trigger on the dominant controller to move the copy with Manual Conductor. The copy looks identical to the original object, while the rest of the world, including the original object, will turn semi-transparent.

Direction Movement (Figure 1-c1): Users can also rotate the dominant controller's joystick clockwise or counterclockwise to change the direction of the object. To activate rotation, the user needs to push the joystick to the boundary. The user can then rotate the joystick against the boundary clockwise or counterclockwise. Based on our pilot testing, the Control-Display Gain (CD gain) of the joystick rotation to object rotation is set to 1/3, which means 3 degrees of joystick rotation would lead to one degree of preview rotation.

To keep the preview outline always snapped to the cursor, we developed an algorithm to maintain that relationship as shown in Figure 2. To calculate the rotation angle, we first define how the preview is rotated and moved from the current frame to the next frame. In the current frame, there is a collision point on the dominant controller's ray, which is also the cursor position. Then, calculate the preview rotation $\Delta\alpha$. Draw a line from the object centre to the collision point, and shoot another new invisible ray from the preview centre with an angle difference of $\Delta\alpha$ from the line. The new ray will intersect with the outline of the preview at a new collision point. Then rotate the preview from the centre by $\Delta\alpha$. The new collision point will rotate together with the preview, and then move the preview to the position where the new collision point is at the same position as the cursor (previous collision point). To prevent jitter while pushing the joystick against the boundary, such as accidentally leaving the boundary at some certain angle, the rotation won't be completed until the thumb leaves the joystick.

Confirm and Teleport (Figure 1-d1): The previous two steps allow users to move and rotate the preview freely in 3D space. Once the preview is in the desired position and direction, the user can release the dominant controller's index trigger to confirm. At this point, the original object that the user selected will be relocated to the preview position and direction, along with the rest of the world. By nature, Locomoontion can also facilitate height manipulation. Users can simply select the ideal height, and move it lower or higher and close to the user, given that the object is attached to the cursor.

Cancellation (Figure 1-d2): When users intend to stop teleportation, they can always initiate cancellation by positioning the intersection point in a manner that contradicts the ray shooting direction, and then release the index trigger to confirm the cancellation. The algorithm is as follows (Figure 3). It first calculates the normal vector of the plane towards the ray side; then performs a dot product between the normal vector and the ray vector. If the dot product is greater than zero, then it is cancelling the teleportation.

4 EXPERIMENT

We designed a controlled experiment to evaluate the performance of Locomoontion by comparing it against the traditional Point&Teleport technique with Bimanual Point&Tug as a baseline because it is the most common teleportation technique in current VR applications and games [18–20, 51]. We designed a 3D navigation task involving positioning a box to assess their respective performance. We hypothesise that:

- **H1:** Locomoontion is faster than Point&Tug, since Locomoontion takes less time for refining the position and is able to adjust height quickly.
- **H2:** Locomoontion requires less physical demand than Point&Tug, because it requires less physical movement when adjusting the height.

4.1 Experimental Task

As shown in Figure 4, the task of the experiment asked users to move to a specific place around an object with those two techniques, which mimics the use case that they want to teleport to and interact with some part of an object. Two distinct boxes were presented: a larger blue box around the object to indicate the target location,

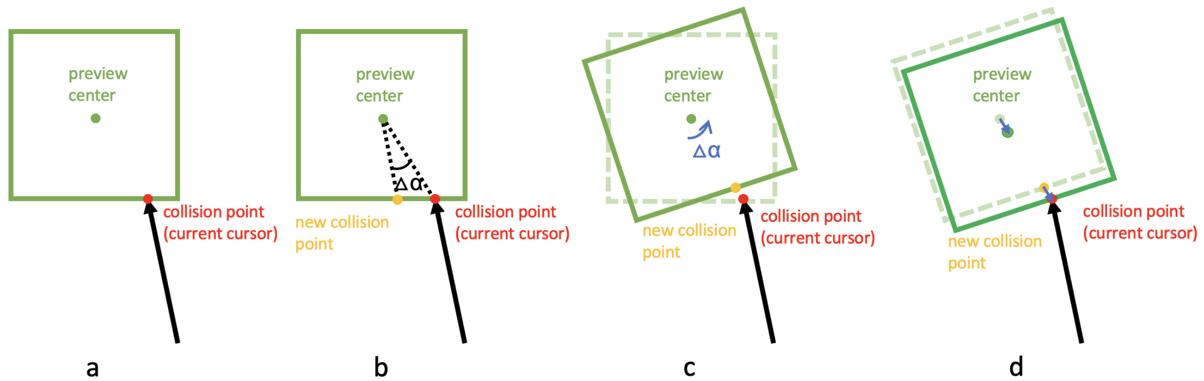

Figure 2: Preview rotation algorithm (a) A collision point (red) on the ray, which is also the cursor position. (b) Then, calculate the preview rotation $\Delta\alpha$. Draw a line from the centre to the collision point, and shoot another new invisible ray from the preview centre with an angle difference of $\Delta\alpha$ from the line. The new ray will intersect with the outline of the preview at a new collision point. (c) Rotate the preview from the centre by $\Delta\alpha$. The new collision point will rotate together with the preview. (d) The new collision point will rotate together with the preview, and then move the preview to the position where the new collision point is at the same position as the cursor (previous collision point).

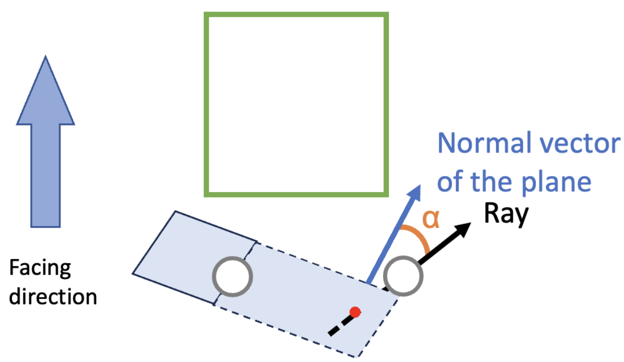

Figure 3: Cancel algorithm: Calculate the dot product of the normal vector of the plane and the ray. If it's larger than zero, then it's in cancellation mode.

and a smaller yellow box attached to the participant to represent their respective location. Users needed to move to an ideal location that could put the yellow box fully inside the blue box.

We selected this task to specifically highlight the refinement aspect of our experiment. While many VR studies focus on long-distance navigation challenges [17, 36, 57] — they often overlook short-distance refinement, assuming users can effortlessly "quickly navigate" to precise targets. However, even minor positional adjustments can become time-consuming, induce ergonomic strain [58] , or introduce usability barriers. This issue is amplified for seated users or individuals with mobility impairments, where subtle movements may require disproportionate physical effort or compromise accessibility.

4.1.1 Task Design. The task is designed to be performed in a seated position. In particular, there would be a white box (referred to as Object) in front of the user in VR, which mimics the object that the user may want to interact with. The white box is 0.3m in width and depth and 2.5m in height (0.30m, 0.30m, 2.50m). The distance between the participant and the object would be 1m, 3m, or 10m. There is also a semi-transparent blue box attached to the object, which indicates the target position that the user needs to move to. The size of the blue box was 0.15m in width and 0.07m in depth and height (0.15m, 0.07m, 0.07m). The blue target could appear in four directions: front, back, left, and right of the object, and also three heights relative to the user, which could be 1m, 0m, and -1m (relative to the user's height).

The purpose of this setting is to simulate the scenario in which the user needs to move upward, stay the same height, and move downward. There would be a yellow box that is smaller than the blue box and always attached to the user, and always in front of and 0.1m lower than the user's eyes. The yellow box indicates the user's position. The size of the yellow box is 0.05m smaller than the blue box in width, depth, and height (0.10m, 0.02m, 0.02m), which restricts both the location and orientation. The goal of the task is to put the yellow box fully inside the blue box. The blue box would change from blue to green when the yellow box is full inside it.

4.1.2 Task Procedure. At the beginning of each trial, the white object turned transparent to allow the user to locate where the blue box target was since it might be behind the white object. Participants could take as much time as they wanted to locate the blue target. Then, participants pressed the A button on the right controller to start the trial. Once A was pressed, the system recorded the start time, and the white object turned solid. The participant then performed the locomotion technique until the yellow box was fully inside the blue box. When they finished the locomotion, the trial ended, and the completion time was recorded.

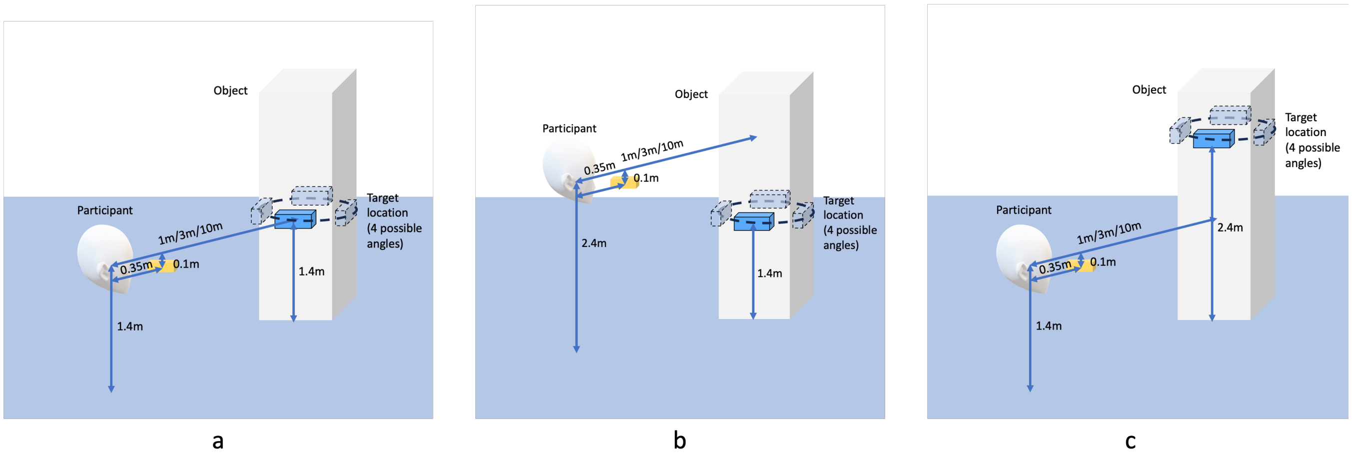

Figure 4: The settings of the experiment. (a) The participant is at the same height as the blue target. (b) The participant is 1m higher than the blue target. (c) The participant is 1m lower than the blue target. The solid blue box is the Front (0°) condition. The semi-transparent boxes are Left (90°), Back (180°), and Right (270°) angle conditions. Only one box would be visible in one angle condition. The distance between the user and the object could be 1m, 3m, or 10m. The yellow box attached to the user is 0.1m lower than the user's head location (VR HMD coordinate) and 0.35m in front.

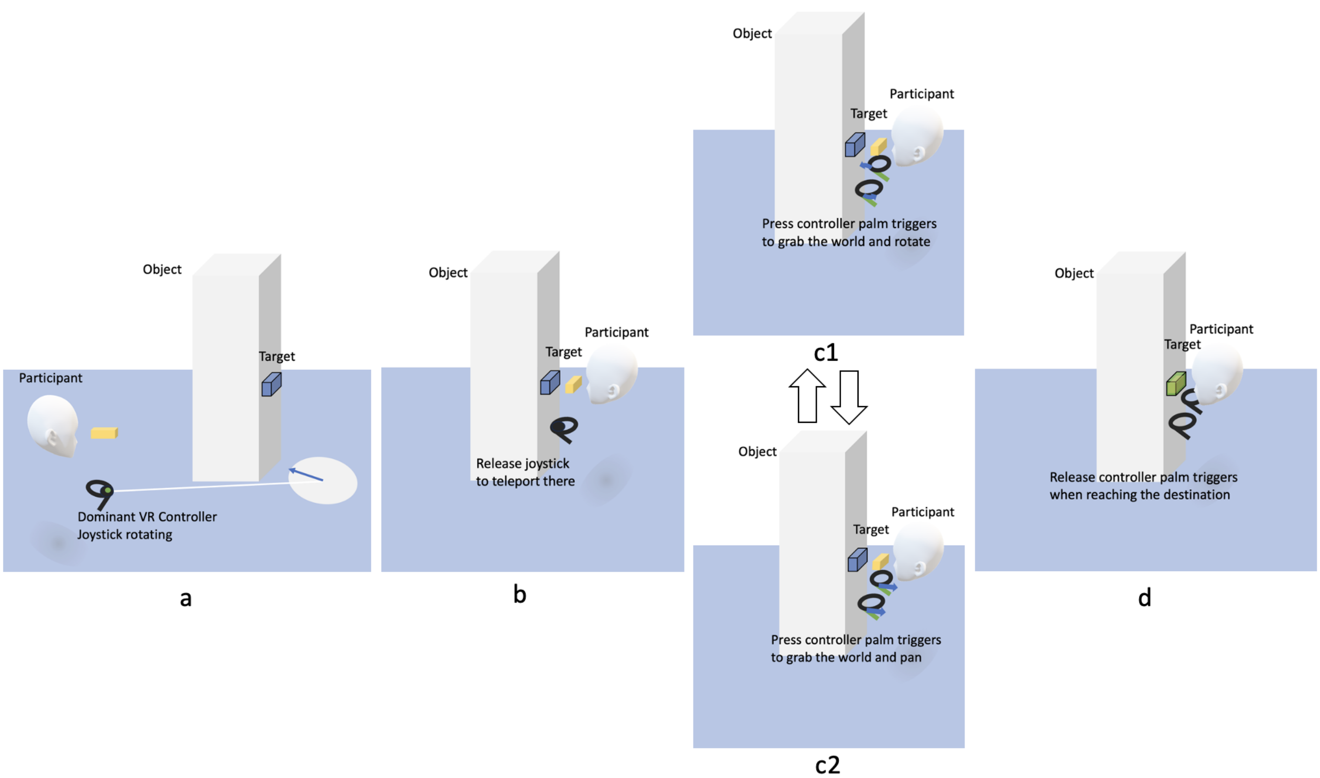

Figure 5: The implementation of the Baseline technique. The participant is going to reach the target position. (a) The participant is rotating the joystick to shoot a ray pointing at the ground. It would be somewhere close to the destination. (b) The participant releases the joystick to teleport there with the same height. (c1) The participant presses the two-palm trigger on the controller to grab and rotate the world. (c2) The participant presses the two-palm trigger on the controller to grab and pan the world. (d) The destination is reached.

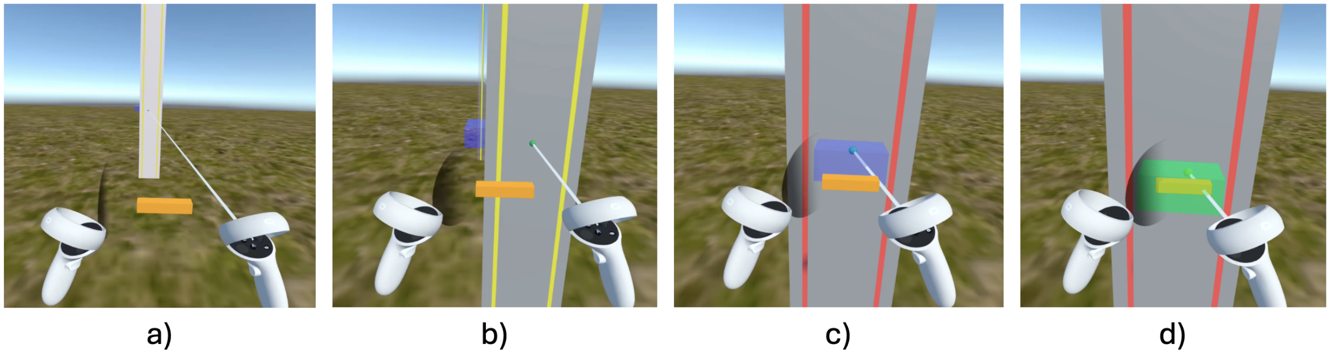

Figure 6: Screenshot of Locomoontion technique. a) Selecting the object with Conductor. b) Moving preview closer with Conductor. c) Rotating the preview using the joystick. d) Fine-tuning the position until the yellow box is fully inside the blue box (turn green).

4.2 Baseline: Point&Teleport and Bimanual Point&Tug

We chose the Point&Teleport and Bimanual Point&Tug, the most popular 3D teleportation technique, as the baseline (Figure 5). This combination has been adopted by several VR applications and games in current VR applications and games [18–20, 51]. Point&Teleport allows users to rotate the dominant hand joystick to shoot a ray pointing on the ground. The pointing position represents the location of the teleportation, and the joystick direction represents the direction that the user will be facing. Once the joystick is released, the user will be teleported to that position with the same height. In order to refine the position and direction, users could also use Bimanual Point&Tug technique. Users can press the two palm triggers to grab the whole world, and pan or rotate it in any direction, which is similar to picture manipulation on the touchscreen.

4.3 Participants and Apparatus

We recruited 18 participants from a local university via mailing list and word-of-mouth, six women and 12 men, ages 22 to 35. All of the participants were right-handed. Five of them are familiar with VR. Participants received $25 Amazon Gift card as compensation in local currency.

We developed an experiment platform with C# in Unity 3D 2021.3.27. We use Oculus Quest 2 as the VR Headset, and the experiment platform is running as a standalone app installed inside the Oculus Quest 2.

4.4 Design

We employed a within-subjects design with three primary independent variables:

- TECHNIQUE with 2 levels (Baseline, Locomoontion);
- DISTANCE with 3 levels (1m, 3m, 10m);
- HEIGHT with 3 levels (-1m, 0m, 1m).

And a random factor DIRECTION with 4 levels (Front (0°), Left (90°), Back (180°), Right (270°)).

The order of TECHNIQUES was counterbalanced with a Latin square. Participants first finished all the trials for one TECHNIQUE and then started the other. Within each TECHNIQUE, DIRECTIONS were combined into one block, which means one block would contain four trials with all four DIRECTIONS in a random order and the same DISTANCE and HEIGHT conditions in a block. The combination of DISTANCE and HEIGHT was also executed in random order. Once all the conditions were performed within a TECHNIQUE, we repeated the same sequence 2 times, thus 3 repetitions in total.

The dependent variable of our experiment was task completion time, which was computed from the time when participants confirmed finding the blue box position by pressing button A in the right controller, to the time when the locomotion technique finished and the yellow box was fully inside the blue box.

In summary, we collected: 2 TECHNIQUES × 3 HEIGHTS × 3 DISTANCES × 4 ANGLES × 3 REPETITIONS = 216 data points per participant.

4.5 Procedure

The researcher first went through the experimental procedure, making sure that the participants understood the experiment, and then instructed the first technique to the participants. The research took 1 block (4 trials per block) to show how the technique works, and then the participants tried it themselves for another 2 blocks. They had 1 repetition to practice and 3 repetitions for the real experiment. Participants were allowed to take a break between blocks. When they finished the first technique, they performed the same procedure again for the second technique. After they finished all the techniques, they were asked to complete a 21-point (0–20) NASA-TLX questionnaire and invited to a semi-structured interview. The overall experiment took around 75 minutes for each participant.

5 RESULTS

For each combination of participant, TECHNIQUE, DISTANCE, HEIGHT, and ANGLE, the task completion times more than 3 standard deviations from the mean time were excluded as outliers. In total, 40 trials (1.5%) were removed.

To analyse the data, we used a repeated measures ANOVA with TECHNIQUE, DISTANCE, and HEIGHT as the independent variables, and ANGLE being grouped. Holm-Bonferroni corrected post-hoc pairwise t-tests were used for further analysis. Normality was

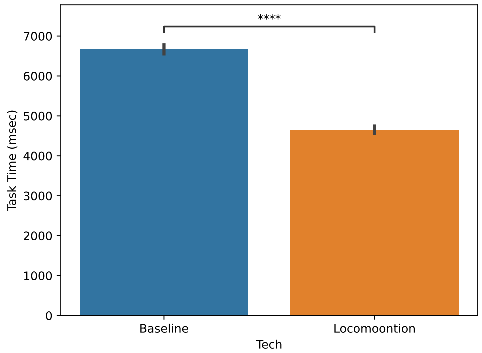

Figure 7: Task completion time for locomotion task. Error bars are 95% CI.

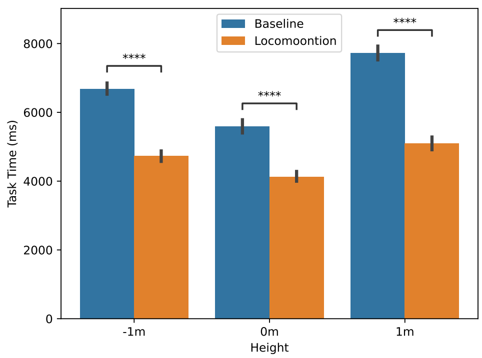

Figure 8: Task completion time for different heights. Error bars are 95% CI.

corrected using a Box-Cox transform. When sphericity was violated, degrees of freedom were corrected using Greenhouse-Geisser.

5.1 Task Completion Time

Figure 13 shows the task completion time for the overall locomotion task. A repeated measures ANOVA indicated a significant main effect for TECHNIQUE on task completion time ($F_{1,17} = 32.17$, $p < .0001$, $\eta_G^2 = .38$). Overall, Locomoontion (Mean = 4652.8 ms) was faster than Baseline (Mean = 6669.3 ms).

Moreover, there were significant main effects for HEIGHT ($F_{2,34} = 17.13$, $p < .0001$, $\eta_G^2 = .03$) and DISTANCE ($F_{2,34} = 74.25$, $p < .0001$, $\eta_G^2 = .11$) on task completion times as well.

Figure 8 shows the task completion times for different height conditions, and Figure 9 shows the task completion times for different distance conditions.

Post-hoc comparisons indicate that Locomoontion was significantly faster than Baseline in all conditions for both height (Table 1) and distance (Table 2). Locomoontion also had a stable time at around 4000 ms across conditions.

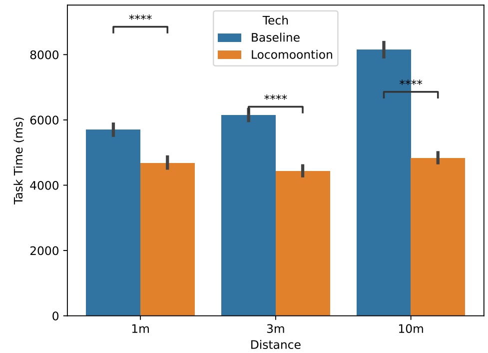

Figure 9: Task completion time for different distances. Error bars are 95% CI.

Table 1: Task time (ms) for different height conditions

Height	Locomoontion (ms)	Baseline (ms)
-1m	4727.6	6684.4
0m	4131.4	5594.6
1m	5097.2	7726.2

Table 2: Task completion time (ms) for different distance conditions

Distance	Locomoontion (ms)	Baseline (ms)
1m	4684.6	5700.6
3m	4434.8	6156.6
10m	4838.6	8149.9

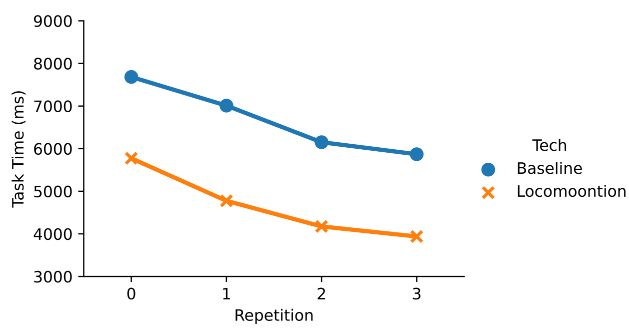

Figure 10: Task completion time over repetitions.

5.2 Task Completion Time Over Repetitions

Looking at task time over repetitions (Figure 10), we notice that both techniques showed a learning curve at some level. Even after one repetition of practicing for both techniques, the Baseline had a steeper learning curve than Locomoontion. For Baseline, the third repetition was 1277 ms faster than the first repetition; however, for Locomoontion, the difference was 625ms. This result suggests that Locomoontion could achieve a more stable performance under the same amount of training than Baseline. In general, the results support H1.

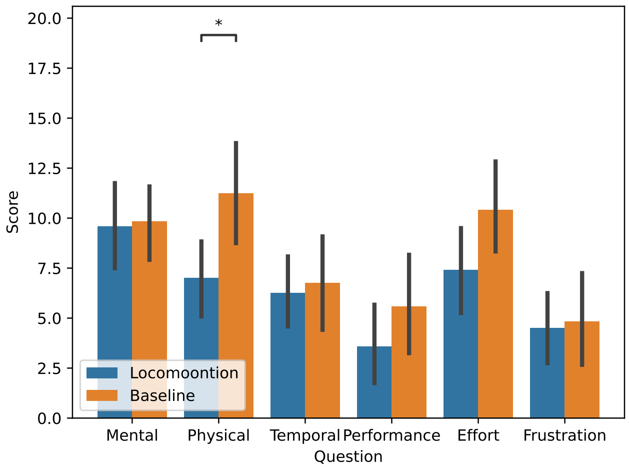

Figure 11: NASA-TLX results on a 21-point scale (the lower is better).

5.3 Questionnaire Results

We also analysed participants' perceived workload using the NASA-TLX questionnaire on a 21-point scale (the lower, the better). Figure 11 shows the results, and Locomoontion (Mean = 7.00) significantly outperformed Baseline (Mean = 11.25) in Physical Demand, which supports H2. This indicates that users may be less fatigued using Locomoontion to perform the teleportation task.

While there were no significant differences for all other dimensions, the average ratings of Locomotion were lower than those of Baseline.

5.4 Interview Results

All participants agreed that Locomoontion was easy to adjust the height. Most participants found Locomoontion was faster once they got familiar with how the technique worked and less physical movement (P1, P3, P4, P5, P7, P9, P10, P11, P12, P13, P14, P15, P18), although it might take more time to learn (P1, P6, P8). Some participants mentioned that Locomoontion was more intuitive and natural, as they could see how the object was moving, and they were controlling the object (P1, P3, P7, P12). Other participants found that the Baseline was intuitive and easy to learn, especially those with some experience with VR (P4, P5, P6, P10).

For direction control, participants said that the joystick controlling the direction could be a bit challenging in Locomoontion in terms of small precise control (P5, P8), but it got some praise for its overall performance from P4 and P10. However, for Baseline, participants commented that direction was hard to identify in far distances since the arrow was small, or when the participant was close to the ground since the point of view was almost parallel to the ground (P10, P12).

As for different distance conditions, participants reported that it was hard to locate the teleportation position with Baseline. For example, pointing more than once and teleportation with the first time just approached the object closer (P7, P11). But Locomoontion performed similarly in all conditions (P7). Motion sickness was only reported by P9, who found Baseline would lead to minor dizziness.

6 DISCUSSION

In this section, we discuss several insights obtained from our design and evaluation of the Locomoontion technique.

6.1 Task Time Over All Conditions.

Locomoontion interestingly performs very similarly in all conditions with around 4 seconds to complete the task, since the distance and height did not affect much once after selecting the object with Semi-Auto Conductor. We only set 1m higher or lower than the user for height conditions, but Locomoontion would likely outperform much more if the height variance is larger. We look forward to the future usage of the technique in VR games, 3D authoring tools, and other applications. One of the reasons why Locomoontion is faster could be that Locomoontion has a simpler workflow compared with Baseline. Locomoontion requires locating the position, selecting the object, rotating the object, and moving the object, which is, in general, more continuous in the mindset. However, Baseline needs to locate the position, select the teleporting point, teleport and check the current place, locate the ideal position, and refine the position with rotation or height adjustment. Teleporting and checking the new place would cause a break in the workflow.

6.2 Task Time Over Repetitions.

As mentioned before, we found that Locomoontion had a smooth learning curve compared with Baseline in terms of task completion time. We asked participants to get hands-on and practice one repetition that was not included in Figure 10, given the fact that participants may want to explore the technique within the practice repetition in a casual way. However, even if we took the practice repetition data as an informal way to compare, we still found Baseline (Mean = 8036ms) was slower than Locomoontion (Mean = 5387 ms). This result could again provide some confidence that Locomoontion is easy enough to acquire.

6.3 Fine Tuning

During the experiment, we observed that participants could quickly move the preview close to themselves but then took additional time to make fine adjustments, especially when the direction was significantly misaligned. This could be due to the challenging nature of the task we designed, which required precise direction changes. We aimed for a highly accurate locomotion technique, resulting in a high constraint: placing a 0.1m × 0.02m × 0.02m box entirely inside a 0.15m × 0.07m × 0.07m box. This level of precision is more demanding than the techniques in previous studies discussed in the "VR Locomotion in 3D" section, which may have had less functionality than our approach.

For the baseline condition, direction changes could be easily and intuitively accomplished with the Bimanual Point & Tug method, similar to grabbing and rotating an object in the real world. However, rotating the joystick requires precise thumb control and careful visual monitoring of the direction, particularly given the small margin for error. Once the direction is confirmed, participants need to adjust the preview's position, and the threshold for how deeply the yellow box can be placed within the blue box is significantly impacted. As illustrated in Figure 12, when the direction deviates

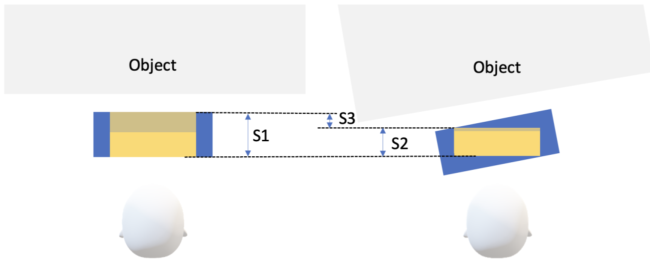

Figure 12: The depth of the yellow box can be put into the blue box under distinct direction differences. If the direction varies a lot from the object orientation, the room for the yellow box to be placed inside the blue box (right, denoted by S2) is less than that with the parallel orientation (left, denoted by S1) by S1 - S2 = S3.

considerably from the object's orientation (right), the space available for the yellow box inside the blue box diminishes compared to when the orientations are parallel (left). This mechanism was intentionally designed as a penalty for directional inaccuracy.

However, in scenarios where orientation is less restricted, we anticipate that Locomoontion would demand less mental effort and be preferred over the baseline. Interestingly, some participants adjusted the direction by rotating the joystick while simultaneously modifying the position by pressing the dominant controller's index trigger. This strategy allowed them to avoid restarting the entire procedure of joystick rotation, thereby completing the task more quickly. This approach could be included as a tip for novice users in future instructions.

6.4 Tutorial for Locomoontion.

Locomoontion and Conductor are relatively new techniques compared with Baseline. Therefore, it is necessary for designers or the developers to come up with a systematic and intuitive tutorial system to help future users onboard the technique rapidly. For example, the way to introduce Conductor technique, since it's a relatively new technique for users. Designers could show the large plane on the non-dominant controller during the tutorial, which could help users understand how Locomoontion works. Then, in the real-world use case, it may replace it with a small plane or line as a visual indicator without blocking the view. Another tip that could be provided in the tutorial is to select at a similar height to the destination, which could bring the object up or down quickly.

7 LIMITATIONS AND FUTURE WORK

We designed the Locomoontion technique with scenarios that require moving to precise positions over visible distances in mind, such as navigating a confined space to manipulate objects. Our experiment focused on evaluating this process of precise movement, especially with regards to the position refinement process. We acknowledge that Locomoontion may not be suitable for all scenarios, and our study did not comprehensively explore possible use case. In this section, we discuss the limitations of the technique and experiment, and suggest directions for future work to further improve upon it.

7.1 Locomoontion Technique

We identify ways in which the locomotion technique can be further optimized.

Direction Control: As mentioned in the previous section, we found some participants complained about the precision for direction control in Locomoontion. The current implementation is a direct 1/3 CD gain mapping, but the performance might be improved with other CD gain functions [9, 66]. It's worthwhile to explore the best CD gain function for joystick usage.

Scaling Factor for Different Sizes of Object and World: Due to the limited independent variable that could be analysed by ANOVA and total experiment time, SCALE condition is not included in this experiment. We would also be interested to know how Locomoontion would integrate and perform when scaling the scene is enabled. Furthermore, the scene in the current experiment setting contains only one object with one size. It is also worth exploring when the scene is more complex and the ideal object size varies. The virtual environment could be complicated, and we can only evaluate limited conditions, and we are curious about whether Locomoontion could keep a stable performance in the other conditions. For small and far objects, Locomoontion might be challenging as it's harder to select the object. However, it's also possible to optimise it by enlarging the Collider of the small objects or grouping small objects together to make the selection easier.

Valid Teleporting Location: We did not set any constraints for the object, which means it is still possible for users to "enter" the object, which sounds unreasonable. However, we emphasise that in our experiment, we focus on investigating general teleportation tasks to one object, as the first attempt of validating our new technique. Moreover, in some scenarios, like 3D editing, constraints on users' location are not necessary when they are editing the objects. We do acknowledge that the teleporting location constrained by physical objects is important in some cases, such as gaming. It can be easily resolved by adding location checking in our current implementation. Additionally, Locomoontion requires at least one existing object in the scene as an anchor for teleportation. In sparse environments such as open outdoor spaces, Locomoontion would be ineffective on its own due to lack of available anchors. In such cases, it would need to be combined with other position-based techniques to support movement across diverse environments.

Fly The Moon Away From Me: Conductor has a high precision for close position pointing, which makes it reliable when moving the preview from far to close in Locomoontion technique. However, if the user wants to move away from the object and have a bird's-eye view of the world, the precision might be limited. That is because when the intersection is far, the angle of the plane and ray could be very small, thus, a minor change of the angle or controller position would lead to a large distance difference. One way to mitigate this issue is to remap the real plane when the plane-ray angle is large (Figure 13). In addition, the Locomoontion technique assumes that the users want to face the selected object after teleportation. This limitation may mitigated in dense environments, where numerous alternative anchors are available. However, in other cases, additional refinement steps may be required to adjust the orientation.

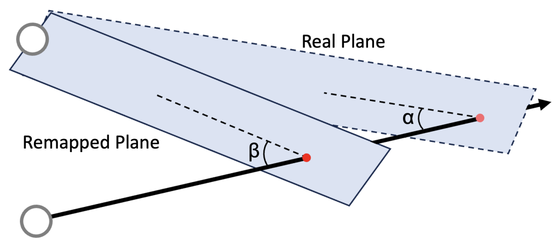

Figure 13: Remapping the plane to mitigate the far distance imprecise issue. The plane with a dashed line is the real plane with a small plane-ray angle α. The plane with a solid line is the remapped visible plane with a large plane-ray β.

7.2 Study Design

We outlines aspects of our study design that may restrict the applicability of our results and point out avenues for future work.

Baseline Selection: The Locomoontion technique allows for simultaneous adjustment of location (xy-plane), height (z-axis), and orientation. To the best of our knowledge, no other such technique existed at the time of the study. Thus when selecting the baseline for comparison, we had to combine and/or enhance existing techniques to ensure they also supported manipulation across all three dimensions. We chose Point&Teleport and Bimanual Point&Tug due to their wide-spread usage [18–20, 51]. While we believe this is a reasonable baseline that allows us to understand the utility of object-based teleportation techniques represented by Locomoontion, we do recognize that other options exist. One alternative is to modify the Point&Teleport variant with height adjustment outlined in [57] to allow orientation adjustment as well. The advantage of this option is that it would be a fully discrete teleportation technique, the same as Locomoontion, unlike the current baseline which includes the continuous Point&Tug technique. On the other hand, the shortcoming is that this technique already reserves the joystick for height adjustment. Adding orientation control would require either overloading the joystick (e.g., through mode switching) or utilizing another input mechanism – both of which could signifcantly increase cognitive load for users. Overall, we believe there is no single best baseline, and trade-offs must be considered. Future work could focus on comparing Locomoontion with a variety of techniques to evaluate it more thoroughly.

Task Selection: The user study is designed to investigate the effectiveness of Locomoontion in scenarios where locomotion is used to position oneself for object manipulation, a type of use case often overlooked in existing locomotion studies [17]. We chose the simple task of moving to the correct position in the correct orientation in a space with a single object, to comprehensively evaluate the utility of Locomoontion during refinement stages, across various relative positions to the target object. However, we recognize that our approach involves a trade-off between thoroughly evaluating simple settings and exploring more complex scenarios and tasks. To fully understand Locomoontion's practical utility, more complex tasks – such as navigating through a labyrinth – would be necessary; this remains an avenue for future work.

Another point of interest is participants' perception and acceptance of manipulating certain objects – such as moving a building within an outdoor city environment. While such interactions are possible through Locomoontion, it remains unknown how participants would feel about this rather unorthodox operation. Future work should take into account users' psychological responses and expectations, and explore adaptations of the technique to better align with user intuition and comfort, as needed.

8 CONCLUSION

We present a novel VR locomotion technique, Locomoontion, allowing users to select an object, create a preview of the selected object to provide a clear understanding of the virtual world after teleportation, and employ Conductor to manipulate the position and orientation of the preview with high precision. The experiment conducted with 18 participants demonstrated the superior performance of Locomoontion over the traditional Point&Teleport technique. Users experienced reduced task completion times and reported less physical fatigue, indicating a more efficient and comfortable locomotion experience with Locomoontion.

In conclusion, Locomoontion shows promising potential to enhance locomotion experiences in VR, offering users the ability to navigate and interact in the virtual environment with greater ease, precision, and comfort. Further research and improvements on Locomoontion could open new avenues for immersive and seamless locomotion in VR applications.

ACKNOWLEDGMENTS

This work is supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) Discovery Grant and the University of Waterloo.

REFERENCES

[1] Parastoo Abtahi, Mar Gonzalez-Franco, Eyal Ofek, and Anthony Steed. 2019. I'm a Giant: Walking in Large Virtual Environments at High Speed Gains. 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, 1–13. https://doi.org/10.1145/3290605.3300752

[2] Jonas Auda, Max Pascher, and Stefan Schneegass. 2019. Around the (Virtual) World: Infinite Walking in Virtual Reality Using Electrical Muscle Stimulation. 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, 1–8. https://doi.org/10.1145/3290605.3300661

[3] Marc Baloup, Thomas Pietrzak, and Géry Casiez. 2019. RayCursor: A 3D Pointing Facilitation Technique Based on Raycasting. 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, 1–12. https://doi.org/10.1145/3290605.3300331

[4] Jiwan Bhandari, Paul MacNeilage, and Eelke Folmer. 2018. Teleportation without Spatial Disorientation Using Optical Flow Cues. In *Proceedings of Graphics Interface 2018* (Toronto, Ontario) *(GI 2018)*. Canadian Human-Computer Communications Society / Société canadienne du dialogue humain-machine, 162 – 167. https://doi.org/10.20380/GI2018.22

[5] 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* (Osaka, Japan) *(VRST '21)*. Association for Computing Machinery, New York, NY, USA, Article 21, 10 pages. https://doi.org/10.1145/3489849.3489893

[6] David Bond and Madelein Nyblom. 2019. Evaluation of four different virtual locomotion techniques in an interactive environment.

[7] 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* (Austin, Texas, USA) *(CHI PLAY '16)*. Association for Computing Machinery, New York, NY, USA, 205–216. https://doi.org/10.1145/2967934.2968105

[8] Fabio Buttussi and Luca Chittaro. 2019. Locomotion in place in virtual reality: A comparative evaluation of joystick, teleport, and leaning. *IEEE transactions on visualization and computer graphics* 27, 1 (2019), 125–136.

[9] Géry Casiez, Daniel Vogel, Ravin Balakrishnan, and Andy Cockburn. 2008. The impact of control-display gain on user performance in pointing tasks. *Human–computer interaction* 23, 3 (2008), 215–250.

[10] S. A. Chowdhury, A K M Amanat Ullah, Nathan Bruce Pelmore, Pourang Irani, and Khalad Hasan. 2022. WriArm: Leveraging Wrist Movement to Design Wrist+Arm Based Teleportation in VR. *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (2022), 317–325. https://api.semanticscholar.org/CorpusID:255187676

[11] Noah Coomer, Sadler Bullard, William Clinton, and Betsy Williams-Sanders. 2018. Evaluating the effects of four VR locomotion methods: joystick, arm-cycling, point-tugging, and teleporting. In *Proceedings of the 15th ACM symposium on applied perception.* 1–8.

[12] Jose L. Dorado and Pablo A. Figueroa. 2014. Ramps are better than stairs to reduce cybersickness in applications based on a HMD and a Gamepad. In *2014 IEEE Symposium on 3D User Interfaces (3DUI).* 47–50. https://doi.org/10.1109/3DUI.2014.6798841

[13] Jeff Feasel, Mary C Whitton, Laura Kassler, Frederick P Brooks, and Michael D Lewek. 2011. The integrated virtual environment rehabilitation treadmill system. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 19, 3 (2011), 290–297.

[14] Sebastian Freitag, Dominik Rausch, and Torsten Kuhlen. 2014. Reorientation in virtual environments using interactive portals. In *2014 IEEE Symposium on 3D User Interfaces (3DUI).* 119–122. https://doi.org/10.1109/3DUI.2014.6798852

[15] Jann Philipp Freiwald, Oscar Ariza, Omar Janeh, and Frank Steinicke. 2020. Walking by Cycling: A Novel In-Place Locomotion User Interface for Seated Virtual Reality Experiences.. In *CHI.* 1–12.

[16] Julian Frommel, Sven Sonntag, and Michael Weber. 2017. Effects of Controller-Based Locomotion on Player Experience in a Virtual Reality Exploration Game. In *Proceedings of the 12th International Conference on the Foundations of Digital Games* (Hyannis, Massachusetts) *(FDG '17).* Association for Computing Machinery, New York, NY, USA, Article 30, 6 pages. https://doi.org/10.1145/3102071.3102082

[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* (Glasgow, Scotland Uk) *(CHI '19).* Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300377

[18] Google VR. 2016. Tilt Brush: Painting in Virtual Reality. https://github.com/googlevr/tilt-brush. Software repository.

[19] Gravity Sketch. 2016. Gravity Sketch: Collaborative 3D Design Platform. https://gravitysketch.com/. 3D design and collaboration software.

[20] HalfLife Alyx. 2020. https://store.steampowered.com/app/546560/HalfLife_Alyx/

[21] Jeremy Hartmann, Christian Holz, Eyal Ofek, and Andrew D Wilson. 2019. Realitycheck: Blending virtual environments with situated physical reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems.* 1–12.

[22] Hiroo Iwata. 1999. Walking about virtual environments on an infinite floor. In *Proceedings IEEE Virtual Reality (Cat. No. 99CB36316).* IEEE, 286–293.

[23] M. P. Jacob Habgood, David Moore, David Wilson, and Sergio Alapont. 2018. Rapid, Continuous Movement Between Nodes as an Accessible Virtual Reality Locomotion Technique. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR).* 371–378. https://doi.org/10.1109/VR.2018.8446130

[24] Beverly K Jaeger and Ronald R Mourant. 2001. Comparison of simulator sickness using static and dynamic walking simulators. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 45. SAGE Publications Sage CA: Los Angeles, CA, 1896–1900.

[25] Dhruv Jain, Misha Sra, Jingru Guo, Rodrigo Marques, Raymond Wu, Justin Chiu, and Chris Schmandt. 2016. Immersive Terrestrial Scuba Diving Using Virtual Reality. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI EA '16).* Association for Computing Machinery, New York, NY, USA, 1563–1569. https://doi.org/10.1145/2851581.2892503

[26] Vineet Kamboj, Tuhin Bhuyan, and Jayesh S. Pillai. 2019. Vertical Locomotion in VR Using Full Body Gestures. In *Proceedings of the 25th ACM Symposium on Virtual Reality Software and Technology* (Parramatta, NSW, Australia) *(VRST '19).* Association for Computing Machinery, New York, NY, USA, Article 98, 2 pages. https://doi.org/10.1145/3359996.3364770

[27] DongHoon Kim, Dongyun Han, Siyeon Bak, and Isaac Cho. 2024. Crossing Rays: Evaluation of Bimanual Mid-air Selection Techniques in an Immersive Environment . In *2024 IEEE International Symposium on Mixed and Augmented Reality (ISMAR).* IEEE Computer Society, Los Alamitos, CA, USA, 329–338. https://doi.org/10.1109/ISMAR62088.2024.00047

[28] Hyun K Kim, Jaehyun Park, Yeongcheol Choi, and Mungyeong Choe. 2018. Virtual reality sickness questionnaire (VRSQ): Motion sickness measurement index in a virtual reality environment. *Applied ergonomics* 69 (2018), 66–73.

[29] Ji-Sun Kim, Denis Gračanin, Krešimir Matković, and Francis Quek. 2008. Finger Walking in Place (FWIP): A Traveling Technique in Virtual Environments. In *Proceedings of the 9th International Symposium on Smart Graphics* (Rennes, France) *(SG '08).* Springer-Verlag, Berlin, Heidelberg, 58–69. https://doi.org/10.1007/978-3-540-85412-8_6

[30] Chengyuan Lai, Ryan P. McMahan, and James Hall. 2015. March-and-Reach: A realistic ladder climbing technique. In *2015 IEEE Symposium on 3D User Interfaces (3DUI).* 15–18. https://doi.org/10.1109/3DUI.2015.7131719

[31] Eike Langbehn, Paul Lubos, Gerd Bruder, and Frank Steinicke. 2017. Bending the Curve: Sensitivity to Bending of Curved Paths and Application in Room-Scale VR. *IEEE Transactions on Visualization and Computer Graphics* 23, 4 (2017), 1389–1398. https://doi.org/10.1109/TVCG.2017.2657220

[32] 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.

[33] Donghae Lim, Shizuka Shirai, Jason Orlosky, Photchara Ratsamee, Yuki Uranishi, and Haruo Takemura. 2022. Evaluation of User Interfaces for Three-Dimensional Locomotion in Virtual Reality. In *Proceedings of the 2022 ACM Symposium on Spatial User Interaction.* 1–9.

[34] Chang Liu, Alexander Plopski, and Jason Orlosky. 2020. OrthoGaze: Gaze-based three-dimensional object manipulation using orthogonal planes. *Computers & Graphics* 89 (2020), 1–10. https://doi.org/10.1016/j.cag.2020.04.005

[35] Gerard Llorach, Alun Evans, and Josep Blat. 2014. Simulator sickness and presence using HMDs: comparing use of a game controller and a position estimation system. In *Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology.* 137–140.

[36] Andrii Matviienko, Florian Müller, Martin Schmitz, Marco Fendrich, and Max Mühlhäuser. 2022. SkyPort: Investigating 3D Teleportation Methods in Virtual Environments. *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (2022).

[37] Morgan McCullough, Hong Xu, Joel Michelson, Matthew Jackoski, Wyatt Pease, William Cobb, William Kalescky, Joshua Ladd, and Betsy Williams. 2015. Myo Arm: Swinging to Explore a VE. In *Proceedings of the ACM SIGGRAPH Symposium on Applied Perception* (Tübingen, Germany) *(SAP '15).* Association for Computing Machinery, New York, NY, USA, 107–113. https://doi.org/10.1145/2804408.2804416

[38] Daniel Medeiros, Maurıcio Sousa, Alberto Raposo, and Joaquim Jorge. 2020. Magic Carpet: Interaction Fidelity for Flying in VR. *IEEE Transactions on Visualization and Computer Graphics* 26, 9 (2020), 2793–2804. https://doi.org/10.1109/TVCG.2019.2905200

[39] Daniel Mendes, Fabio Marco Caputo, Andrea Giachetti, Alfredo Ferreira, and Joaquim Jorge. 2019. A survey on 3d virtual object manipulation: From the desktop to immersive virtual environments. In *Computer graphics forum*, Vol. 38. Wiley Online Library, 21–45.

[40] Yann Moullec, Mélanie Cogné, Justine Saint-Aubert, and Anatole Lécuyer. 2023. Assisted walking-in-place: Introducing assisted motion to walking-by-cycling in embodied virtual reality. *IEEE Transactions on Visualization and Computer Graphics* 29, 5 (2023), 2796–2805.

[41] Niels Christian Nilsson, Tabitha Peck, Gerd Bruder, Eri Hodgson, Stefania Serafin, Mary Whitton, Frank Steinicke, and Evan Suma Rosenberg. 2018. 15 Years of Research on Redirected Walking in Immersive Virtual Environments. *IEEE Computer Graphics and Applications* 38, 2 (2018), 44–56. https://doi.org/10.1109/MCG.2018.111125628

[42] Jeffrey S. Pierce, Brian C. Stearns, and Randy Pausch. 1999. Voodoo Dolls: Seamless Interaction at Multiple Scales in Virtual Environments. In *Proceedings of the 1999 Symposium on Interactive 3D Graphics* (Atlanta, Georgia, USA) *(I3D '99).* Association for Computing Machinery, New York, NY, USA, 141–145. https://doi.org/10.1145/300523.300540

[43] Ivan Poupyrev, Mark Billinghurst, Suzanne Weghorst, and Tadao Ichikawa. 1996. The Go-Go Interaction Technique: Non-Linear Mapping for Direct Manipulation in VR. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology* (Seattle, Washington, USA) *(UIST '96).* Association for Computing Machinery, New York, NY, USA, 79–80. https://doi.org/10.1145/237091.237102

[44] Yuan Yuan Qian and Robert J. Teather. 2018. Look to Go: An Empirical Evaluation of Eye-Based Travel in Virtual Reality. In *Proceedings of the 2018 ACM Symposium on Spatial User Interaction* (Berlin, Germany) *(SUI '18).* Association for Computing Machinery, New York, NY, USA, 130–140. https://doi.org/10.1145/3267782.3267798

[45] Sharif Razzaque. 2005. *Redirected Walking.* Ph. D. Dissertation. USA. Advisor(s) Brooks, Fredrick P. AAI3190299.

[46] Hyocheol Ro, Seungho Chae, Inhwan Kim, Junghyun Byun, Yoonsik Yang, Yoonjung Park, and Tackdon Han. 2017. A dynamic depth-variable ray-casting interface for object manipulation in ar environments. In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC).* 2873–2878. https://doi.org/10.1109/SMC.2017.8123063

[47] Erik Sikström, Amalia de Götzen, and Stefania Serafin. 2015. Wings and flying in immersive VR — Controller type, sound effects and experienced ownership and agency. In *2015 IEEE Virtual Reality (VR)*. 281–282. https://doi.org/10.1109/VR.2015.7223405

[48] Nancy A Skopp, Derek J Smolenski, Melinda J Metzger-Abamukong, Albert A Rizzo, and Greg M Reger. 2014. A pilot study of the virtusphere as a virtual reality enhancement. *International Journal of Human-Computer Interaction* 30, 1 (2014), 24–31.

[49] Mel Slater, Martin Usoh, and Anthony Steed. 1994. Steps and Ladders in Virtual Reality. In *Proceedings of the Conference on Virtual Reality Software and Technology* (Singapore, Singapore) *(VRST '94)*. World Scientific Publishing Co., Inc., USA, 45–54.

[50] Mel Slater, Martin Usoh, and Anthony Steed. 1995. Taking steps: the influence of a walking technique on presence in virtual reality. *ACM Transactions on Computer-Human Interaction (TOCHI)* 2, 3 (1995), 201–219.

[51] Steam VR Home. 2020. https://store.steampowered.com/news/app/250820/view/2898585530113860152

[52] Frank Steinicke, Yon Visell, Jennifer Campos, and Anatole Lécuyer. 2013. *Human walking in virtual environments*. Vol. 56. Springer.

[53] Richard Stoakley, Matthew J. Conway, and Randy Pausch. 1995. Virtual Reality on a WIM: Interactive Worlds in Miniature. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '95)*. ACM Press/Addison-Wesley Publishing Co., USA, 265–272. https://doi.org/10.1145/223904.223938

[54] James N. Templeman, Patricia S. Denbrook, and Linda E. Sibert. 1999. Virtual Locomotion: Walking in Place through Virtual Environments. *Presence: Teleoperators and Virtual Environments* 8, 6 (12 1999), 598–617. https://doi.org/10.1162/105474699566512 arXiv:https://direct.mit.edu/pvar/article-pdf/8/6/598/1623326/105474699566512.pdf

[55] Khrystyna Vasylevska and Hannes Kaufmann. 2014. Influence of Metaphors for Vertical Navigation on Presence.

[56] Julius von Willich, Martin Schmitz, Florian Müller, Daniel Schmitt, and M. Mühlhäuser. 2020. Podoportation: Foot-Based Locomotion in Virtual Reality. *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (2020). https://api.semanticscholar.org/CorpusID:211562653

[57] 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 (2023), 2467–2477. https://doi.org/10.1109/TVCG.2023.3247114

[58] Johann Wentzel, Greg d'Eon, and Daniel Vogel. 2020. Improving Virtual Reality Ergonomics Through Reach-Bounded Non-Linear Input Amplification. *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (2020). https://api.semanticscholar.org/CorpusID:211528762

[59] Johann Wentzel, Greg d'Eon, and Daniel Vogel. 2020. Improving Virtual Reality Ergonomics Through Reach-Bounded Non-Linear Input Amplification. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376687

[60] Betsy Williams, Gayathri Narasimham, Bjoern Rump, Timothy P McNamara, Thomas H Carr, John Rieser, and Bobby Bodenheimer. 2007. Exploring large virtual environments with an HMD when physical space is limited. In *Proceedings of the 4th symposium on Applied perception in graphics and visualization*. 41–48.

[61] Dennis Wolf, Katja Rogers, Christoph Kunder, and Enrico Rukzio. 2020. JumpVR: Jump-Based Locomotion Augmentation for 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, 1–12. https://doi.org/10.1145/3313831.3376243

[62] Hans Peter Wyss, R. Blach, and M. Bues. 2006. iSith - Intersection-based Spatial Interaction for Two Hands. *3D User Interfaces (3DUI'06)* (2006), 59–61.

[63] Zhixin Yan, Robert W. Lindeman, and Arindam Dey. 2016. Let your fingers do the walking: A unified approach for efficient short-, medium-, and long-distance travel in VR. In *2016 IEEE Symposium on 3D User Interfaces (3DUI)*. 27–30. https://doi.org/10.1109/3DUI.2016.7460027

[64] Difeng Yu, Xueshi Lu, Rongkai Shi, Hai-Ning Liang, Tilman Dingler, Eduardo Velloso, and Jorge Goncalves. 2021. Gaze-Supported 3D Object Manipulation in Virtual Reality. 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 734, 13 pages. https://doi.org/10.1145/3411764.3445343

[65] Futian Zhang, Keiko Katsuragawa, and Edward Lank. 2022. Conductor: Intersection-Based Bimanual Pointing in Augmented and Virtual Reality. *Proc. ACM Hum.-Comput. Interact.* 6, ISS, Article 560 (nov 2022), 15 pages. https://doi.org/10.1145/3567713

[66] Futian Zhang, Sachi Mizobuchi, Wei Zhou, Taslim Arefin Khan, Wei Li, and Edward Lank. 2021. Leveraging CD Gain for Precise Barehand Video Timeline Browsing on Smart Displays. In *Human-Computer Interaction–INTERACT 2021: 18th IFIP TC 13 International Conference, Bari, Italy, August 30–September 3, 2021, Proceedings, Part IV 18*. Springer, 72–91.

[67] Yaying Zhang, Bernhard E. Riecke, Thecla Schiphorst, and Carman Neustaedter. 2019. Perch to Fly: Embodied Virtual Reality Flying Locomotion with a Flexible Perching Stance. In *Proceedings of the 2019 on Designing Interactive Systems Conference* (San Diego, CA, USA) *(DIS '19)*. Association for Computing Machinery, New York, NY, USA, 253–264. https://doi.org/10.1145/3322276.3322357