2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR) | 979-8-3503-7402-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/VR58804.2024.00111

illumotion: An Optical-illusion-based VR Locomotion Technique for Long-Distance 3D Movement

Zackary P. T. Sin* Ye Jia† Richard Chen Li‡ Hong Va Leong§ Qing Li¶ Peter H. F. Ng‖

The Hong Kong Polytechnic University, HKSAR

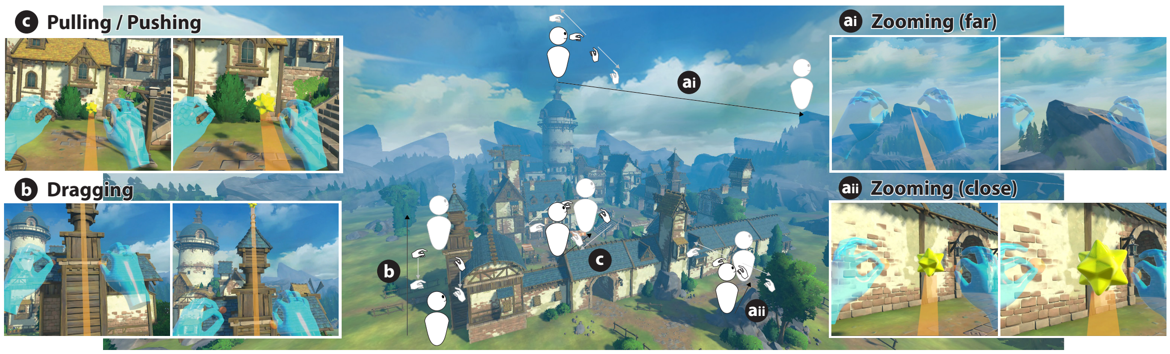

Figure 1: *illumotion* is a locomotion technique based on optical manipulation. The user can use (*a*) zooming, (*b*) dragging and (*c*) pulling/pushing to move around a scene with 3D movement. As an optically-driven method, it tunes the speed based on the target; (*ai*) targets further away will lead to faster movement and vice versa, (*aii*) for closer targets.

Abstract

Locomotion has a marked impact on user experience in VR, but currently, common to-go techniques such as steering and teleportation have their limitations. Particularly, steering is prone to cybersickness, while teleportation trades presence for mitigating cybersickness. Inspired by how we manipulate a picture on a mobile phone, we propose *illumotion*, an optical-illusion-based method that, we believe, can provide an alternative to these two typical techniques. Instead of zooming in a picture by pinching two fingers, we can move forward by "zooming" toward part of the 3D virtual scene with pinched hands. Not only is the proposed technique easy to use, it also seems to minimize cybersickness to some degree.

illumotion relies on the manipulation of optics; as such, it requires solving motion parameters in screen space and a model of how we perceive depth. To evaluate it, a comprehensive user study with 66 users was conducted. Results show that, compared with either teleportation, steering or both, *illumotion* has better performance, presence, usability, user experience and cybersickness alleviation. We believe the result is a clear indication that our novel optically-driven method is a promising candidate for generalized locomotion.

Index Terms: Human-centered computing—Human computer interaction—Interaction paradigms—Virtual reality

1 Introduction

Despite being the golden standards for traveling in VR [13, 14], teleportation and steering are known to have their issues. Among the two, steering can be more immersive with its continuous movement, but it has been known to generate considerable cybersickness; teleportation, on the other hand, alleviates cybersickness by removing continuous movement, but at the cost of presence [14]. Hence, for VR applications, choosing a locomotion technique can become deciding whether to prioritize presence or reducing cybersickness. Of course, using hardware solutions like VR treadmills is possible, but it can be expensive and not readily available.

There is also the issue of traveling large distances via 3D movement. Typical locomotion techniques operate by assuming that the user will only walk on a relatively flat surface. Although this assumption may work well for first-person VR applications, it may not be optimal for applications that require navigation in 3D, such as in-VR scene editors and 3D world explorers. We further put forward that for 3D (mid-air) locomotion, easily traveling large distances is also important as those applications may require the user to travel in a larger world (e.g. an application for viewing city landscape). It seems, however, that discussions on effective 3D locomotion techniques for both long- and short-distance travel are limited.

We propose *illumotion*, a novel optical-illusion-based locomotion technique that (1) enables continuous movement while mitigating cybersickness, (2) naturally works for 3D locomotion, (3) excels at large-distance travel, and (4) is easy to use. The idea of our technique stems from how we resize pictures on mobile phones; by pinching two fingers, users are able to resize a picture by zooming in or out. Further, it has been known that some bimanual actions, such as enlarging, can be habitually performed by humans. Together with this insight, we transfer this iconic and easy-to-use interaction into 3D space such that when the users are performing "zooming" by pinching with their two hands, they will be moving forward and backward instead (Fig. 1).

Our locomotion technique is based on optical manipulation as the "zooming" treats what the user is seeing as if it is a 2D picture. When the user is pinching, the two positions where they are visually pinched at are locked. After the movement, the user will find that the two positions are still locked at where they have been pinching. In contrast to other bimanual techniques for locomotion, the novelty of our work lies in the optically-driven movement, which

*e-mail: zackary-p-t.sin@polyu.edu.hk
†e-mail: aimmeng-ye.jia@polyu.edu.hk
‡e-mail: richard-chen.li@polyu.edu.hk
§e-mail: hong.va.leong@polyu.edu.hk
¶e-mail: qing-prof.li@polyu.edu.hk
‖e-mail: peter.nhf@polyu.edu.hk

979-8-3503-7402-5/24/$31.00 ©2024 IEEE
DOI 10.1109/VR58804.2024.00111

can automatically adjust in speed and direction according to where the user is pinching; if the user is pinching a location far away, the zooming effect will become stronger, which results in faster movement that assists in long-distance travel (Fig. 1*a*). Despite its continuous movement, we believe our method has the potential to limit cybersickness as the hands can provide references to help users visually associate them with the movement. Naturally, our novel locomotion technique's methodology is based on optics; with the 3D movement intrinsically linked with what the user is seeing, there is a need to connect the changes in 2D space to 3D movement via a projective transformation.

To demonstrate the efficacy of *illumotion*, we have conducted a comprehensive user study with 66 participants. As one of our goals is to explore an alternative technique to steering and teleportation, they are part of our comparative study. Results show that our novel optically-based technique is efficient for movement in 3D, particularly long-distance movement. Further, when compared with teleportation and steering, *illumotion* is rated higher in terms of usability, presence and user experience, while being able to mitigate cybersickness to some degree.

In summary, the following are this paper's contributions: • *illumotion*, a novel optically-driven 3D locomotion technique for large-distance travel; • The interactive and methodological details of *illumotion*, along with its theoretical discussion on how it may alleviate cybersickness, and; • A comprehensive user study with 66 participants that compares *illumotion* with steering and teleportation.

2 Related Work

Here, we briefly discuss the landscape of locomotion [1, 17, 33] and bimanual research.

Grounded 2D Locomotion Many VR applications require the user to move from one point to another point in a 3D environment. Usually, it assumes there is a ground or landing spot, which forms the basis of traveling that mostly involves 2D movement (forward-backward and left-right). A very typical locomotion technique is steering. By using a directional input via joysticks, head-tracking [48], gaze-tracking [26] or body-leaning [7], the user can move on the ground of the virtual worlds. A well-known issue with steering is that it will induce significant cybersickness [14]. To circumvent this issue, instead of moving the virtual avatar directly, a preview of the steered movement can be given to the user instead [15].

Teleportation is a well-known alternative to steering. It is the default locomotion technique in many VR applications. Despite its effect in addressing cybersickness, undoubtedly, it also hinders presence [14]. It can be combined with animated motion [37] or walking [32] to improve presence. Perhaps due to the necessity for selection before teleportation, an investigation on how to improve the locomotion efficiency has been made [21]. There is also an investigation of a foot-based technique for teleportation [67].

Ideally, we can navigate by physically walking; this can be done by scaled walking or redirected walking. To cater to limited physical space, scaled walking utilizes scaled movement [4] or visual props [15]. Redirected walking tries to redirect the user's physical movement when they are not noticing. The opportunity for doing so can come in the form of blinking [28], distractions [51] or when the user is occupied [56]. Although both methods can extend walkable areas for virtual scenes, they, regardless, require a reasonably sized physical space to work, which may prohibit home use.

With challenges with real walking, some works have looked into imitating it instead. Walking in place [43, 61], jumping [44, 73], and arm swinging [8, 71] are some of the examples. However, these approaches may require additional tracking to function.

Other solutions rely on additional interfaces in various forms. For example, the user can choose where to go from a miniature representation of the virtual worlds [3, 45]. It is also possible to have a method to navigate between different scales of the virtual scene [12] or the virtual avatar [16, 20]. A physical tablet can also be used as a device to drive movement [39].

Utilizing hardware can effectively enable physical walking for VR. For example, treadmills of different calibrations have undergone varying degrees of investigation [42, 74]. It is also possible to reduce cybersickness by synthetically simulating walking vibration [46]. These hardware solutions come at the cost of extra devices and may require sufficient dedicated space, thereby limiting home use.

Mid-air 3D Locomotion There has been growing interest in mid-air locomotion works. We refer to them as 3D locomotion works as they enable movement in 3D (forward-backward, left-right and up-down). The 3D counterpart of steering is sometimes referred to as flying [17]. In many ways, steering in 3D can be quite similar to its 2D counterpart; the control inputs can come from joysticks or head-tracking [10]. It is also possible to use a pulling mechanism to move in 3D [31]. To fully capture the 3D experience, there is also work that focuses on how to create an immersive flying experience [66]. 3D movement can also be done by deforming the scene and performing a point-to-point movement as if there is a wormhole [9]. Still, cybersickness is still a lingering issue for steering in either dimensionality. Using a flying carpet as a visual proxy to provide a fixed reference for the user has been proposed to address cybersickness [38]. Another approach is to mix steering with teleportation, which can avoid discomfort by limiting speed [54].

A similar effort to transfer to 3D has also been made for teleportation. A main challenge when designing 3D teleportation techniques is how to select, in 3D, a point to teleport to [34]. In this regard, how far to move in mid-air can be tuned by a controller [18]; alternatively, it can be done by automatically tuning the distance to jump based on nearby objects [63], via evaluating a signed distance field [30], or combining spatial relationships and users' perception of motion [2]. Note that our work, *illumotion*, will automatically adapt the speed based on the distance of where it is pinched. There are exact examples of uplifting teleportation to 3D [34, 70]. One of the works addresses the mid-air selection problem by specifying the distance with the longitudinal axis of a joystick's touchpad [34]. Another work chooses to investigate different multi-stage selection strategies. One example is that the user first selects the direction (just like 2D teleportation) and then selects the height to change; the height is selected by tilting the angle of the joystick controller [70].

Other approaches for 3D locomotion involve a hardware setting, which can be effective in combating cybersickness [25, 52, 59, 69]. Virtual props like an ad-hoc stair [27] can also enable 3D movement.

Hand-based Interaction As a technique that utilizes human hands as input, our work is related to touch-based and bimanual works [64]. Foremost, there are previous works that use single touch [40] or mix with dragging [41] to control 3D navigation. For bimanual research, most works seem to focus on object manipulation (e.g. with handlebar metaphors [60] or physical pens [68]). However, previous works have used bimanual gestures to control movement as well, for example, for mobile VR [24]. Another work, on the other hand, introduces pinching in and out (zooming) for navigation [64], which shows similarity to our work. It is noted, however, that our work's main novelty lies in the introduction of an optically-driven means of locomotion; furthermore, our work is VR-focused.

3 illumotion: Motion via Optical Illusion

As mentioned, *illumotion* is inspired by how we resize pictures on mobile phones (Fig. 2*a*). To resize a picture, we simply perform a pinching gesture on the phone; when we are pinching, essentially, we are also touching the picture with two anchors and these anchors are locked with the picture. As our two fingers move, these anchors will follow where our fingers are. The picture, at the same time, will be resized or moved such that the anchors are still locked at the same places with respect to the picture itself. The intuition of *illumotion* is that if we imagine the picture as a 2D projection of a

3D environment, we are essentially "moving" inward or outward when the picture is resizing. To explain the design, effect and, later, implementation of our method better, we introduce the following terms for clarity.

- **Pinching:** To initiate movement, the user needs to *pinch* both their hands. Specifically, the pinching gesture involves the user touching the index finger with the thumb. With picture resizing as an analogy, pinching is *when* the user decides to "touch the picture".
- **Pinching Position:** When the user is pinching with either hand, where the index finger and the thumb are touching is the *pinching position*. As an analogy, the pinching position is *where* the user is "touching the picture".
- **Projected Anchor:** As a VR environment is not 2D, when the user is pinching, they are not touching a point on a 2D plane. Instead, they are "touching" a point in the 3D scene. We refer to this point as a *projected anchor*, as it is a projection of the pinching position onto the 3D scene. As it is a projection, it is visually aligning with the pinching position such that they *visually overlap*.

3.1 Interaction Design

In order to navigate with *illumotion*, the user simply needs to pinch and move their hands. Then, the user's avatar will be moved in such a way that the projected anchors will be visually overlapping with the new locations of the pinching positions. It is noted here that our method can automatically adjust the speed based on the distance of the pinched position. Specifically, if the projected anchor is far away, it will result in movement that is faster. On the other hand, it has been known that actions such as dragging and enlarging/shrinking (zooming) may be habitually used by humans [23]; based on this previous investigation, we have designed three main gestures to clearly demonstrate how it is possible to do various navigations with *illumotion*. The user can use the following gestures independently or together for 3D locomotion.

- **Zooming (forward/backward):** This gesture mimics how we *resize* a picture on a mobile phone. With *illumotion*, the *zooming* gesture will result in the user's avatar moving forward/backward (Fig. 1*a*). First, the user performs pinching with both hands. Then, the user will pull their hands closer or further away from each other. By pulling the hands away from each other, the two pinching positions will also be further away from each other; this will result in the projected anchors also going further away from each other, visually (not physically). From an optical perspective, this means that the distance between the camera and the projected anchors has been reduced, and thus, this optical change will result in the user's avatar moving forward to get closer to the projected anchors. Vice versa, by pulling the hands closer to each other, the avatar will move backward.
- **Dragging (left/right/up/down):** This gesture, on the other hand, mimics how we *move* a picture on a phone. We can move a picture by using both fingers to touch the picture and drag it to a new location while trying to lock the distance between the two fingers. With *illumotion*, the *dragging* gesture can be used for left/right and up/down movement, similar to how we can move a picture on the 2D screen of a mobile phone (Fig. 1*b*). First, the user performs two-hand pinching. Then, the user will move both hands in the same direction (left/right/up/down). If the user moves their hands downward, optically, the 3D environment will be "moving" downwards as well; hence, this will result in the avatar moving upwards instead.
- **Pulling/Pushing (forward/backward):** This gesture is similar to dragging, but it assists in forward/backward motion. Instead of dragging the 3D scene left/right or up/down, the user will pull or push their pinching hands. When *pulling*, the user will bring their hands closer to themself; this will force the projected anchors to (visually) get closer to the camera (Fig. 1*c*). Thus, the avatar will move forward. Vice versa, when the user is *pushing* their hands further away from themself, the avatar will move backward. Essentially, pulling/pushing is an alternative to *zooming*, and they are suitable to be combined with the *dragging* gesture for more precise short-distance travel.

3.2 Optically-based Movement via the Dominant Eye

As *illumotion* relies on the user's vision (or the visual presented to the user) to manipulate 3D movement, we need to consider the optics from the perspective of human vision and VR head-mounted displays (HMD). A human's vision relies on mixing the visual information from both eyes; however, each human has an eye preference such that "the dominant eye" provides the majority of visual input [47].

As the dominant eye provides most of the visual input that results in what the user actually sees, in this paper, we assume that the dominant eye is a good approximation of the user's true vision. By extension, it means that the display for the dominant eye will provide the majority of the visual input for the user. Hence, our design and implementation of *illumotion* use the display (also the virtual camera) that serves the dominant eye to project the pinching positions into the 3D scene for finding the projected anchors. This also means that *illumotion* requires the user to know whether their left or right eye is the dominant eye, but this can be quickly checked via typical dominant eye tests that can be easily done at home.

3.3 Implication from Postural Instability Theory

Our design is linked with postural instability theory [53] regarding how to mitigate cybersickness. Briefly, the theory rests upon the idea of postural stability, which stipulates that all animals, including humans, try to control their posture in such a way that minimizes uncontrolled movement. As such, our postural control strategy, the set of postures for interacting, is dependent on the environment. A typically used example is how we walked differently on concrete and ice. If the wrong control strategy is used (e.g. walking on ice as if we are walking on concrete), it will result in postural instability. Having entered an unstable state, it is within our nature to attempt to detect and adapt our control strategy to the new environment. If we do so successfully, the state of instability will end, and the person will return to postural stability. The main tenet of postural instability theory is that a prolonged period of postural instability will result in motion sickness. In the context of VR, it means that cybersickness will occur when users are unable to link their postural control to the visual stimulation for a longer period of time [29].

Hence, the key to mitigating cybersickness is to assist the user in quickly learning and adapting a new postural control strategy given a VR environment. The speed of adaptation depends on how quickly the user is able to detect the relationship between interaction (with the system) and the subsequent (visual) stimulation. We believe adaptation is straightforward with *illumotion*. Our optically-based method intrinsically links the change in visual stimulation (optical flow) with the movement of the pinching positions. This clear relationship should be able to provide a strong natural mapping between action and feedback [58], which has been suggested to benefit intuitiveness [35]. Combined with the fact that our method imitates a well-known interaction from mobile phones, this postural control via hand gestures and movement should be easy and quick to learn. Later in our user study, *illumotion* is shown to indeed mitigate cybersickness with some degree of success; that result may be evidence that partially supports the theoretical discussion here.

4 Technical Methodology

In this section, the technical description that realizes *illumotion* is presented. The key idea is to treat what the user is seeing as if it is a 2D image and transfer commonly used bimanual gestures, zooming, dragging and pulling/pushing, to drive our optically-driven locomotion technique. To realize this, the avatar's location will be updated such that the two projected anchors (which are fixed

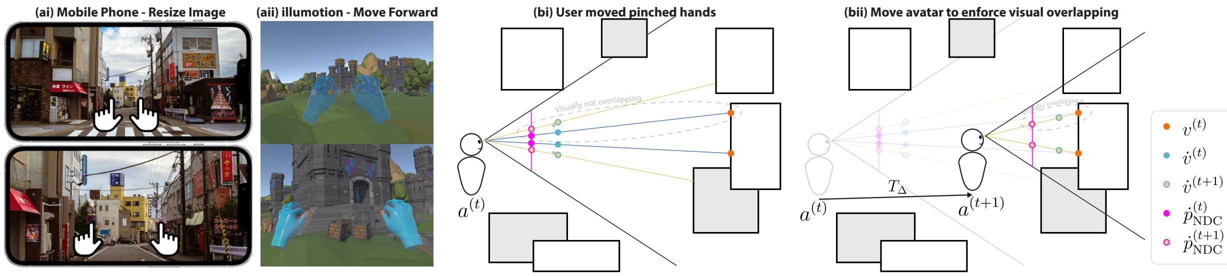

Figure 2: (*ai*) Inspired by how we resize images on mobile phones, (*aii*) *illumotion* transfer the idea for VR locomotion. (*bi*) When the user moves their hands to a new location (i.e. $\dot{v}^{(t+1)}$), it will violate the visual overlapping rule; (*bii*) this can be remedied when motion parameters T_Δ are calculated and the virtual avatar is moved to a new position $a^{(t+1)}$ to enforce visual overlapping.

with the 3D scene) will always be visually overlapped with the pinching positions (Fig. 2*b*), given that the user has moved the two pinching positions. Thus, the direction and speed of movement are dependent on how and where the user is moving their hands. From the user's perspective, the movement's direction and speed can be controlled via the gestures specified in Sect. 3.1. In the following, we first discuss how the anchors can be projected into the 3D scene (Sect. 4.1). Then, we show how to mathematically frame the motion to model the movement triggered by the user's hands (Sect. 4.2). The motion parameter, once applied to the avatar, will result in locomotion. Last, we further discuss how to handle the case when the user is pinching at a location that has no intersection with the 3D scene with perceived depth (Sect. 4.3).

4.1 Projecting the Anchor

In order to treat what the user is seeing as if it is a 2D image, we need first to map where the user is pinching to the 3D scene. As mentioned in Sect. 3, the projected anchors are used as references to visually lock the 3D scene with the pinching positions. As the dominant eye provides the majority of the visual information, we assume that the HMD display for the dominant eye is a sufficient approximation of what the user is actually seeing. We call the virtual camera that serves the display for the dominant eye as the dominant camera. Hence, we say that the pinching position and the projected anchor are visually overlapping when their 2D projections on the screen of the dominant camera are spatially identical. Formally, let that clip coordinate as v_{clip} such that we have the projection formula $v_{\text{clip}} = P_c \cdot T_c^{-1} \cdot v$, where P_c is the projection matrix of the dominant camera c, T_c^{-1} is the inverse transformation matrix of c and v is the world position, when the pinching position $\dot{v}$ and the projected anchor v are *visually overlapping*, the following will hold true,

$$[\dot{x}_{\text{NDC}}, \dot{y}_{\text{NDC}}]^T = \dot{p}_{\text{NDC}} = p_{\text{NDC}} = [x_{\text{NDC}}, y_{\text{NDC}}]^T, \quad (1)$$

where $v_{\text{NDC}} = (x_{\text{NDC}}, y_{\text{NDC}}, z_{\text{NDC}})^T$, $v_{\text{NDC}} = \frac{v_{\text{clip}}}{w_{\text{clip}}}$ is the normalized device coordinate and $x_{\text{NDC}} \in [-1, 1]$. Note that we use p to indicate a coordinate on screen, v to indicate a Cartesian or a homogeneous coordinate and x is the first component of p or v. To find the position of a projected anchor v, here, we use raycasting; given the dominant camera position c, we shoot a ray r to find the intersection with the 3D scene v in world space such that

$$v = r(\dot{v}) = c + \dot{n} \cdot d \;\text{ where }\; \dot{n} = (\dot{v} - c)/|\dot{v} - c|, \quad (2)$$

where $\dot{n}$ is the direction of the ray derived from the pinching position $\dot{v}$ in world space and d is the distance to the closest point that intersects with the 3D scene that depends on where $\dot{v}$ is.

4.2 Solving the Motion Parameters

With the projected anchors, we can use them as references to derive how much movement is needed when the user is moving their pinched hands. The main mechanism that drives *illumotion* is to perform a 3D movement such that when the pinching positions are moved, the projected anchors remain visually overlapping with them; this is needed because when the user moves their pinching hands, the two projected anchors will no longer visually overlap with their pinching position counterparts if the avatar's location is not adjusted (Fig. 2*bi*). Hence, a 3D movement is applied to the avatar such that they will be visually overlapping again. We will show how to compute the movement in forward-right direction and then up direction separately. The calculated magnitudes will determine the speed and direction of movement. We will then show how to apply the movement to the avatar.

More concretely, given that the pinching position $\dot{v}$ has moved to a new location at a new time frame $t+1$ such that $\dot{v}^{(t+1)} = \dot{v}^{(t)} + \Delta\dot{v}^{(t+1)}$, where $\dot{v}^{(t)}$ is the pinching position at time frame t and $\Delta\dot{v}^{(t+1)}$ is the movement of the pinching position at time frame $t+1$, we embed the motion parameters T_Δ into the projection formula with the following

$$v^{(t+1)}_{\text{clip}} = P_c \cdot T_c^{-1} \cdot T_\Delta \cdot v^{(t)}, \quad (3)$$

and the goal is to solve T_Δ while ensuring Equation 1 holds true. Note that with $v^{(t+1)} = T_\Delta \cdot v^{(t)}$, the motion parameter "moves" the projected anchors at time frame $t+1$; we will later show how to move the avatar instead.

As the movement is 3D, we allow the avatar to be moved in three orthogonal directions based on the direction of the camera. Specifically, these directions are the forward f, right r and up u direction of the dominant camera and their magnitudes of movement are Δf, Δr and Δu, respectively. To simplify, the calculation for the magnitudes of forward and rightward movement is separated from that of upward movement.

Forward-Rightward Movement First, we present the calculation for forward and rightward movement's magnitudes, Δf and Δr. As the movement is simply a translation transformation, for brevity, let the projected anchor at frame t as $v = (x_{\text{world}}, y_{\text{world}}, z_{\text{world}}, 1)^T$ we will have $T_{\Delta fr} \cdot v = [x_{\text{world}} + f_x\Delta f + r_x\Delta r,\;\; y_{\text{world}} + f_y\Delta f + r_y\Delta r,\;\; z_{\text{world}} + f_z\Delta f + r_z\Delta r,\;\; 1]^T$ where $T_{\Delta fr}$ is the motion parameters for forward and rightward movement, which is interchangeable with T_Δ, and f_x is the x-component of f. For the dominant camera's projection matrix P_c and inverse transformation matrix T_c^{-1}, the following common generic forms for a left-handed coordinate system are used,

$$P_c = \begin{bmatrix} \frac{n_P}{r_P} & 0 & 0 & 0 \\ 0 & \frac{n_P}{t_P} & 0 & 0 \\ 0 & 0 & \frac{f_P + n_P}{f_P - n_P} & \frac{-2 f_P n_P}{f_P - n_P} \\ 0 & 0 & 1 & 0 \end{bmatrix}, T_c^{-1} = \begin{bmatrix} c_{00} & \dots & c_{03} \\ \vdots & & \vdots \\ c_{20} & \dots & c_{23} \\ 0 & \dots & 1 \end{bmatrix},$$

where, respectively, n_P, f_P, r_P and t_P are the near, far, right and top boundary for the projection, and c_{ij} is the element at index i, j which is aggregated from translation and rotation transformations. From the above, we will have the following components for the eye coordinate $v_{\text{eye}} = T_c^{-1} \cdot T_\Delta \cdot v_{\text{world}}$,

$$\begin{aligned} x_{\text{eye}} &= \Delta f(c_{00}f_x + c_{01}f_y + c_{02}f_z) + \Delta r(c_{00}r_x + c_{01}r_y + c_{02}r_z) \\ &\quad + c_{00}x_{\text{world}} + c_{01}y_{\text{world}} + c_{02}z_{\text{world}} + c_{03} \\ z_{\text{eye}} &= \Delta f(c_{20}f_x + c_{21}f_y + c_{22}f_z) + \Delta r(c_{20}r_x + c_{21}r_y + c_{22}r_z) \\ &\quad + c_{20}x_{\text{world}} + c_{21}y_{\text{world}} + c_{22}z_{\text{world}} + c_{23}. \end{aligned} \tag{4}$$

With the visual overlapping rule set by Equation 1, we translate the eye coordinate to an NDC coordinate and get,

$$\dot{x}_{\text{NDC}} = x_{\text{NDC}} = (n_P/r_P) \cdot (x_{\text{eye}}/z_{\text{eye}}). \tag{5}$$

Let that $a_0 = c_{00}f_x + c_{01}f_y + c_{02}f_z$, $b_0 = c_{00}r_x + c_{01}r_y + c_{02}r_z$ $c_0 = c_{00}x_{\text{world}} + c_{01}y_{\text{world}} + c_{02}z_{\text{world}} + c_{03}$, $a_1 = c_{20}f_x + c_{21}f_y + c_{22}f_z$, $b_1 = c_{20}r_x + c_{21}r_y + c_{22}r_z$ and $c_1 = c_{20}x_{\text{world}} + c_{21}y_{\text{world}} + c_{22}z_{\text{world}} + c_{23}$, combining with Equation 5, we have $\dot{x}_{\text{NDC}} = \frac{n_P}{r_P} \cdot \frac{a_0\Delta f + b_0\Delta r + c_0}{a_1\Delta f + b_1\Delta r + c_1}$, which can be rewritten into

$$\left(\frac{n_P}{r_P}a_0 - a_1\dot{x}_{\text{NDC}}\right)\Delta f + \left(\frac{n_P}{r_P}b_0 - b_1\dot{x}_{\text{NDC}}\right)\Delta r + \frac{n_P}{r_P}c_0 - c_1\dot{x}_{\text{NDC}} = 0 .$$

The above equation only has Δf and Δr as variables. Recall that since there are two pairs of pinching position and projected anchor (from the left and the right hand), we actually have two equations from Equation 5, $\dot{\overleftarrow{x}}_{\text{NDC}} = \overleftarrow{x}_{\text{NDC}}$ and $\dot{\overrightarrow{x}}_{\text{NDC}} = \overrightarrow{x}_{\text{NDC}}$, where $\dot{\overleftarrow{x}}$ ($\dot{\overrightarrow{x}}$) is the left (right) pinching position. As there are two conditions, Δf and Δr can be solved by Gaussian elimination. Once the computed Δf and Δr are applied to the avatar, visual overlapping will hold again.

Upward Movement The magnitude for upward movement Δu can be similarly computed as above. Foremost, the moved projected anchor is $T_{\Delta u} \cdot v = [x_{\text{world}} + u_x\Delta u, y_{\text{world}} + u_y\Delta u, z_{\text{world}} + u_z\Delta u, 1]^T$ which will lead to the eye coordinate v_{eye}. Then from Equation 1, we translate the eye coordinate to an NDC coordinate with the y-component this time, getting $\dot{y}_{\text{NDC}} = y_{\text{NDC}} = \frac{n_P}{t_P} \cdot \frac{y_{\text{eye}}}{z_{\text{eye}}}$. As Δu is the only variable, it can be directly computed.

However, an issue is that either of the user's hands can provide a reference for what Δu should be. Although previously, in our interaction design, we assume that the user will perform dragging with two hands moving in synchronization, we cannot make the assumption that the user can do so perfectly. Thus, it is proposed that the upward magnitude should be an average $\bar{\Delta u} = \frac{1}{2} \cdot (\overleftarrow{\Delta u} + \overrightarrow{\Delta u})$ where $\overleftarrow{\Delta u}$ ($\overrightarrow{\Delta u}$) is the upward movement magnitude computed as $\dot{\Delta u}$ with the left (right) hand as the reference.

Moving the Avatar Finally, with the magnitude for forward, rightward and upward movement computed, the avatar is moved with the following (Fig. 2*bii*), $a^{(t+1)} = a^{(t)} - f\Delta f - r\Delta r - u(\overline{\Delta u} - u \cdot (f\Delta f + r\Delta r))$, where a is the world position of the avatar of the user. Note that the avatar's movement needs to be the inverse of the movement derived from Equation 3. In practice, we use the projected anchor at the frame when pinching starts to calculate the motion parameter such that $v_{\text{clip}}^{(t+k)} = P_c \cdot T_c^{-1} \cdot T_\Delta \cdot v^{(k)}$, where the current time frame is $t+k$ for the sake of numerical stability.

4.3 Perceived Depth

The discussion in the previous section assumes that there exists two projected anchors that can be retrieved by raycasting; this will require the rays to have intersections with the 3D scene, as described in Equation 2. This assumption cannot be guaranteed as there will always be views that are not fully covered by parts of the 3D scene. An obvious example is that when the user is looking at the sky, there will be many pinching positions that lead to raycasting towards the skybox, which does not result in intersections.

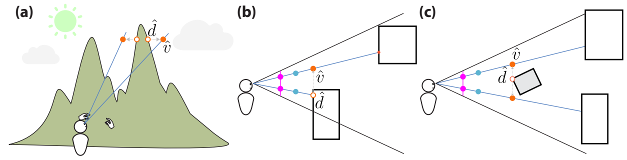

Figure 3: Perceived depth comes from (*a*) nearby objects, (*b*) reference from the other projected anchor and (*c*) an in-between obstacle.

To handle the case where projected anchors cannot be retrieved due to a lack of intersections with the 3D scene, we introduce *perceived depth*. Its main idea is that a pinching position is always associated with a depth perceived by the user. Here, we make an assumption that, given the intent of the user, this perceived depth can always be inferred by nearby objects. For example, given the situation where the user is pinching at two positions with the skybox as the background, if there is a tree in the middle, the user is trying to use the tree as the anchor for navigation. Hence, the two pinching positions can use the distance of the tree to infer the projected anchors. The purpose of the perceived depth is to produce a projected anchor that enables movement that is perceived by the user as natural, even if there is no physical location to visually lock with the 3D scene. In the coming subsections, we discuss three cases of perceived depth.

Perceived Depth from Nearby Objects When a pinching position cannot produce a projected anchor via raycasting, a perceived projected anchor can be produced instead. It relies on finding the perceived depth at $\dot{p}_{\text{NDC}}$, which is where the pinching position is visually at. For this case, our assumption for perceived depth is that it can be referenced from the nearest object as the user most likely takes a nearby object as a reference for the 3D environment (Fig. 3*a*). As the z-buffer from the rendering pipeline already provides a depth map, we propose that the perceived depth is the nearest depth value from the z-buffer. Formally, given a z-buffer map Z, the perceived depth is $\hat{d} = P_z(Z(j)) = 2f_P n_P/(f_P + n_P - Z(j) \cdot (f_P - n_P))$, given $j = \text{argmin}_i\{\ ||\dot{p} - p_i||_2 \mid Z(i) < \inf \ \wedge \ i \in I\}$, where i is a pixel index, I is the hypothetical image sized $W \times H$ rendered by the dominant camera, p_i is a pixel position of pixel i and $\dot{p}$ is the pixel position of where the pinching is, that is $\dot{p} = [\frac{W}{2}(1 + \dot{x}_{\text{NDC}}), \frac{H}{2}(1 + \dot{y}_{\text{NDC}})]^T$. Here, for brevity, when there is a fragment from the 3D scene, the z-value is $Z(\cdot) \in [-1, 1]$, but when there is no fragment, the z-value is set as infinity. With a variant of Equation 2, a projected anchor derived from perceived depth can then be produced with

$$\hat{v} = c + \dot{n} \cdot d, \text{ where } d = \hat{d}/(f \cdot \dot{n}), \tag{6}$$

where d, in this case, is the anchor's distance from the camera. Note that since $\hat{d}$ is retrieved from the depth buffer, it needs to be adjusted for manipulation in world space.

Adjusting Imbalanced Projected Anchors When pinching, it is possible for the user to pinch a close position and a faraway position at the same time. Although this will enable both slow and fast movement, we observe that users generally perceived the closer object as the main reference for movement and may become confused when the movement is larger than their expectation (Fig. 3*b*). To cater to this user preference, we designed a balancing mechanism that adjusts the anchor further away if its distance is significantly different than the closer anchor. Specifically, given that the original further anchor's distance is d_{further} and the original closer anchor's distance is d_{close}, and that their difference exceeds a threshold such that $f \cdot d_{\text{further}} > f \cdot d_{\text{close}} \cdot \varepsilon_{\text{imbalance}}$, we balance the further anchor such that it has the following as the perceived depth, $\hat{d} = f \cdot d_{\text{close}}$. This adjusting $\hat{d}$ is used in Equation 6 to provide a new projected anchor. Empirically, we set $\varepsilon_{\text{imblance}} = 2$.

In-between Visual Obstacle as Perceived Depth Another common user behavior we observed is that given there is an object that the user wishes to move closer to, they may simply pinch at positions that flank the left and right side of the object without considering the pinched part of the 3D scene. In most cases, this will not be an issue as usually nearby positions on the screen are also physically nearby. However, imagine a case where there is a flat plane with a tree and far away, there is a large mountain. As the tree is quite thin, the user may tend to pinch the left and right sides of the tree. Since the left and right sides are where the mountain is at and it is far away, this pinching will result in fast movement, which is suitable for large-distance travel, but not for short-distance movement, which helps with moving towards the tree.

To cater to this user tendency, we propose a step that checks for any visual obstacle between the two pinching positions in screen space (Fig. 3*c*). Specifically, we check for an obstacle that is significantly closer to the user compared to the projected anchors. For a pair of left ($\overleftarrow{p}$) and right pinching position ($\overrightarrow{p}$) in screen space, the following candidate perceived depth is calculated $\hat{d} = \min\{P_z(Z(k)) \mid (||\Delta p||_2^2 - (\Delta p \cdot n_{\text{line}})^2)^{\frac{1}{2}} < \varepsilon_{\text{line}} \wedge k \in I\}$, where $\Delta p = p_k - \overleftarrow{p}$ and $n_{\text{line}} = (\overrightarrow{p} - \overleftarrow{p})/||\overrightarrow{p} - \overleftarrow{p}||_2$. Essentially, $\hat{d}$ is the minimum depth value from the z-buffer between the two pinching positions in screen space. The search area follows a line with its width bounded by $\varepsilon_{\text{line}}$, which we empirically set $\varepsilon_{\text{line}} = 5$.

The candidate perceived depth should only be used if it shows that the visual obstacle is significantly closer to the user compared with the projected anchors such that $\hat{d} \cdot \varepsilon_{\text{inbetween}} < f \cdot d_{\text{close}}$, where d_{close} is the distance of the closer projected anchor and $\varepsilon_{\text{inbetween}}$ is an empirical threshold. We set $\varepsilon_{\text{inbetween}} = 2.5$. If the condition is satisfied, the calculated perceived depth will be the perceived depth for both projected anchors and re-calibrated with Equation 6.

5 User Study

As *illumotion* is designed to be a general locomotion technique that can act as an alternative to teleportation and steering, our user study compares all three in a VR experiment with fixed sets of trials. A Latin square design is used to fairly evaluate the three techniques.

As mentioned, our belief is that *illumotion* is a faster method for (long-distance) travel that is easy to use, immersive, and less inducive to cybersickness. In part to the just mentioned benefits, we also believe our method can provide a better overall user experience. Thus, our user study is designed based on the following hypotheses: (H_1) *illumotion* is more efficient for movement, (H_2) particularly, more efficient for long-distance movement; (H_3) *illumotion* is more usable compared with teleportation; (H_4) *illumotion* has better presence compared with teleportation; (H_5) *illumotion* induce less cybersickness compared with steering; and, (H_6) *illumotion* is more enjoyable (or has better user experience).

5.1 Experimental Techniques

Our user study aims to investigate *illumotion* with comparisons to standard locomotion techniques; the following are their details:

• ***illumotion* (hand):** The proposed optically-driven technique used the interaction specified in Sect. 3 (Fig. 4*ai*). Our implementation will be released in a code repository[1].

• **3D point-and-teleport (hand):** To the best of our ability, we implemented a 3D point-and-teleport technique based on a recent work [70] (Fig. 4*aii*). Originally, a joystick is used, but here, the hand is used to select a direction to move towards to and the height

[1] https://github.com/zackarysin/illumotion

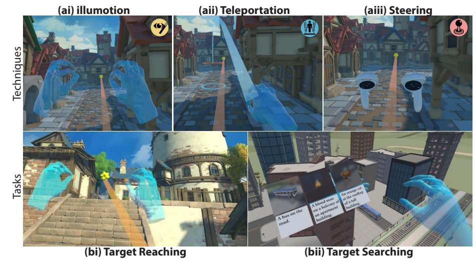

Figure 4: The user study used (*ai*) *illumotion*, (*aii*) teleportation and (*aii*) steering as experimental techniques; and (*bi*) target reaching and (*bii*) target searching as experimental tasks.

selection can be done by tilting the hand thereafter. As specified in that paper, the tilt specifies how quickly the height selection will be tuned. However, no portal window is used as it is not a common standard for typical teleportation techniques. It is implemented based on Meta's Unity Oculus integration package.

• **3D steering (joystick):** Following most steering schemes, our steering comparison also uses joysticks to control (Fig. 4*aiii*). The right controller's joystick specifies ground movement: the vertical axis controls the forward-backward movement while the horizontal axis controls the left-right rotation. The left controller's joystick only specifies the upward-downward movement via the vertical axis. The movement speed is 6 m/s, which seems to be a typical speed of a self-balancing scooter and is similar but faster than some previous work [62]. The rotation speed is set to $30°/s$, which has been used in earlier works [62] and is believed to mitigate cybersickness for steering [55].

5.2 Experimental Tasks

Similar to another 3D locomotion work [70], we evaluated the three techniques in two tasks with different goals: a target-reaching task, focused on evaluating a technique's efficiency to move from one point to another; and, a target-searching task, focused on evaluating a technique's ability in exploring a virtual scene. For both tasks, we have set a maximum time limit of 10 minutes.

• **Target Reaching:** In the target-reaching task (Fig. 4*bi*), participants need to reach from one point to another. Each target, which we also call *subgoal*, is visualized in the form of a yellow star. The subgoals are scattered in the virtual scene so as to evaluate how well a technique can be used for traveling in one. The subgoals are distributed in such a way they can be used to evaluate the efficiency of shorter and longer traveling. Once a target is reached, it will disappear, and a confirmation sound will be played; this affirmation effect is also used in the next task.

• **Target Searching:** The target-searching task requires the user to find three targets in an urban neighborhood sized virtually 90m × 75m (Fig. 4*bii*); they are hidden at ground level, on a balcony and on a rooftop. To avoid memorization, there are three variants of the neighborhood and its unique targets. To inform the user what their targets are, a visual hint panel can be accessed. For a hand-tracking technique, it is hidden within the left palm of the user such that it will be revealed when the user flips their hand. For the joystick, the user can point it towards themself to reveal the hint.

5.3 Measures

The following measures, with the arrow ↑ (↓) means the higher (lower) the better, are used to evaluate the three techniques: • **Task Completion Time** (↓) is the total duration needed by the user to complete the target-reaching/searching task, measured in seconds; • **Subgoal Time Interval** (↓) represents the period, in seconds, between traveling two subgoals during target reaching; • **Usability** (↑)

is measured by the System Usability Scale (SUS) [6] in 5-point Likert scale for assessing the usability of the locomotion techniques; • **Presence** (↑) is measured by the Igroup Presence Questionnaire (IPQ) [50] in 7-point Likert scale, which is used to evaluate the participants' sense of presence; • **Cybersickness** (↓) is measured by a one-question 0-to-10 discomfort scale (similar to [49, 70]), which has been used to quickly measure general discomfort, mainly cybersickness [5, 70]; • **Workload** (↓) uses the NASA Task Load Index (NASA-TLX) [22], presented with a 100-points range; • **Preference** from users is extracted from a preference rank question where the user will be asked to rank their preference on technique from most favorite (1^{st} choice) to least favorite (3^{rd} choice); and, • **Overall User Experience** (↑) is measured via the short version of the user experience questionnaire (UEQ-S) [57] in 7-point Likert scale, which captures practical and enjoyment feedback.

5.4 Setting

All users will try the three (technique) by two (task) experimental pairs. Meta Quest 2 is the VR HMD used here. First, the user will perform a dominant eye test[2]. For each experimental pair, a practice session with tutorial videos is provided to help the user familiarize themself with the task and the technique. They will then complete SUS, IPQ, NASA-TLX, discomfort scale and preference ranking after each task with a technique. The UEQ-S questionnaire is administered at the very end of the entire experiment. For safety, the experiments are conducted while seated.

To recruit for participants, the user study had been advertised openly on the university campus; students, staff members and visitors were all welcome to participate. Thus, the 74 participants that were recruited for the user study consisted of many non-students, and a large number of them were from disciplines outside of computer science. Whenever during the user study, they are allowed to stop the experiment if they feel uncomfortable due to cybersickness. There were eight participants who stopped the experiment early. As such, a total of 66 participants (self-reported: 21 male, 38 female, 7 not specified; between 18 and 58 years old with a median of 28) had completed the study, and unless otherwise specified, they serve as the samples of our analysis. The study has been approved by our institutional ethical review board.

6 Result

In this section, the result from the user study is presented. The objective and subjective measures are discussed separately. Note that when reporting a measure's mean (SD) for the techniques, we always report in the order of *illumotion*, teleportation and steering.

For all measures, Shapiro-Wilk test is first conducted to check for data normality. ANOVA is then used to check for the main effect of technique, and if applicable, the main effect of task and the interaction effect between technique and task. If there is a main effect for technique, post-hoc analysis with Bonferroni correction will be performed. For each measure, if the data conforms to data normality, typical ANOVA with Student's t-tests are used. Otherwise, aligned rank transform (ART) [72] is performed on the data for non-parametric ANOVA and Wilcoxon signed-rank (rank sum) tests are used for post-hoc analysis of paired (independent) samples. Note that we are only interested in comparing *illumotion* with teleportation and steering with the post-hoc tests. Other specifics for a measure will be mentioned later.

6.1 Task Performance

Time is used as the main measure of the users' task performance. As the two tasks' objectives and expected completion time are clearly different, the two tasks' performance are evaluated separately. To avoid task memorization, only the first trials of both tasks are measured for the task completion time; as such, the samples are independent and one-pair independent sample ANOVA is used.

For the target-rearching task, there are 20 targets or subgoals to reach. This time interval for reaching each subgoal is used to perform correlation analysis. Note that data from all three trials (techniques) for each user are used.

Task Completion Time (H_1) The mean (SD) time needed to complete the target-reaching task for the techniques (*illumotion*/teleportation/steering), in seconds, are 194.923 (87.328), 382.323 (118.677) and 272.332 (82.131). The technique's main effect is statistically significant ($F(2,63) = 26.096$, $p < 0.001$, $\eta_p^2 = 0.372$). Post-hoc tests revealed that *illumotion* is statistically significantly faster to complete the task compared to teleportation ($p < 0.001$, $d = -1.799$) and steering ($p < 0.001$, $d = -0.931$).

For the target-searching task, the completion time are 247.167 (181.470), 412.644 (172.308) and 332.657 (192.181). Similar to target reaching, there is also a main effect by technique ($F(2,63) = 6.453$, $p = 0.003$, $\eta_p^2 = 0.097$). However, post-hoc analysis shows that *illumotion* is only faster compared to teleportation ($p = 0.003$, $d = -0.935$). Comparison with steering shows no significance ($p = 0.137$, $d = -0.457$). Fig. 5*ai* shows the result for both tasks.

Distance - Time Correlation (H_2) Each subgoal has a different distance in the target-reaching task. These distances and the time intervals users need to reach these subgoals are used for correlation analysis. *illumotion* shows a weak correlation between time and distance ($s = 0.028$, $r = 0.134$, $p = 0.004$) while teleportation ($s = 0.110$, $r = 0.413$, $p < 0.001$) and steering ($s = 0.174$, $r = 0.704$, $p < 0.001$) shows moderate and strong correlation, respectively. Correlation results indicate that the increase in distance will lead to less increase in travel time for *illumotion* (Fig. 5*aii*).

6.2 User Feedback

The following subjective measures (except overall user experience) have undergone a 3×2 factorial repeated-measures ANOVA to test for main and interaction effects. Data from all six trials (techniques-task pair) for each user are used. All measures have no interaction effect, hence, we combine both tasks for post-hoc analysis. Only the presence measure shows data normality.

Usability (H_3) For target reaching, the aggregated mean (SD) SUS scores (Fig. 5*bi*) for *illumotion*, teleportation and steering, respectively, are 3.635 (0.748), 3.286 (0.855) and 3.773 (0.699). For the searching task, the SUS scores are 3.858 (0.638), 3.408 (0.788) and 3.833 (0.815), respectively. There is a significant main effect for both technique ($F(2,130) = 16.722$, $p < 0.001$, $\eta_p^2 = 0.062$) and task ($F(1,65) = 6.735$, $p = 0.0102$, $\eta_p^2 = 0.005$), but no interaction effect ($F(2,130) = 0.133$, $p = 0.985$, $\eta_p^2 = -0.012$). Thus, the paired test combined the scores from both tasks. The combined SUS scores are 3.746 (0.702), 3.347 (0.821) and 3.803 (0.757), respectively. *illumotion*'s comparison with teleportation shows statistical significance ($p < 0.001$, $d = 0.523$), while there is none with steering ($p = 0.763$, $d = -0.078$).

Presence (H_4) The IPQ scores (Fig. 5*bii*) for target reaching are 4.908 (0.719), 4.654 (0.890), and 5.119 (0.768), and for target searching are 5.157 (0.742), 4.740 (0.733) and 5.135 (0.832) respectively. There is main effect for technique ($F(2,130) = 9.847$, $p < 0.001$, $\eta_p^2 = 0.0479$) but not for task ($F(1,65) = 3.463$, $p = 0.067$, $\eta_p^2 = 0.003$). The interaction effect also shows no significance ($F(2,130) = 2.546$, $p = 0.082$, $\eta_p^2 = 0.050$). Combining both tasks, the IPQ scores are 5.032 (0.738), 4.697 (0.813) and 5.127 (0.798) for the three techniques. The pair test shows significance between *illumotion* and teleportation ($p < 0.001$, $d = 0.432$), but not for *illumotion* and steering ($p = 0.370$, $d = -0.123$).

[2]https://www.performancevisioninc.com/blog/32/dominant-eye-test-determining-which-eye-is-your-dominant-eye/

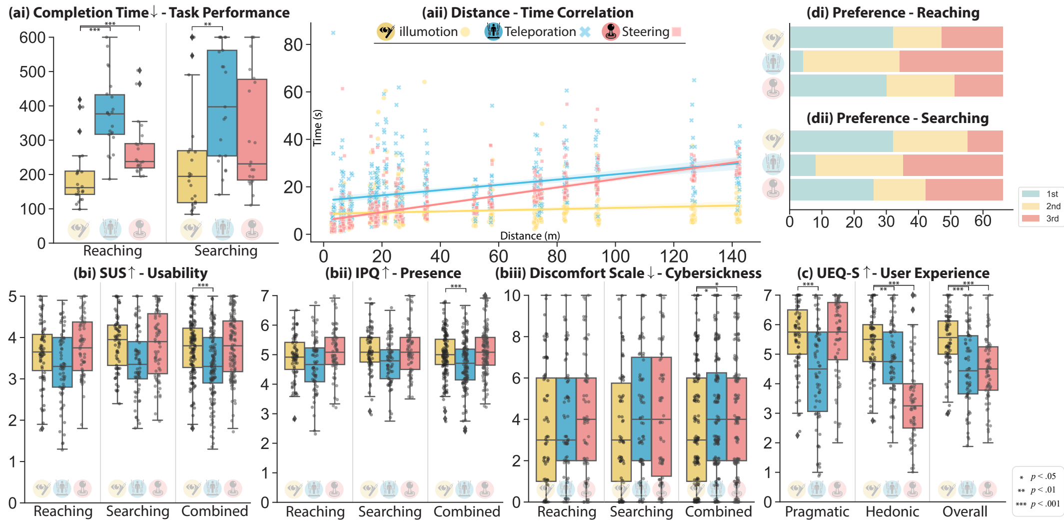

Figure 5: (*ai*) The task performance in time for the reaching and searching task. (*aii*) The correlations between distance and time; as *illumotion*'s speed scales based on the target's distance, it is least affected by distance. The user feedback for (*bi*) usability (SUS), (*bii*) presence (IPQ), (*biii*) cybersickness (discomfort scale) and (*c*) user experience (UEQ-S). The preference ranking for the (*di*) reaching and (*dii*) searching task.

Cybersickness (H_5) The scores for the discomfort scale (Fig. 5*biii*) are, respectively, 3.712 (2.944), 4.106 (2.774) and 4.197 (2.973) for target reaching, and 3.439 (2.888), 4.439 (2.920) and 4.197 (3.240) for target searching. There is main effect for technique ($F(2,130) = 4.226$, $p = 0.016$, $\eta_p^2 = 0.006$) but not for task ($F(1,65) = 0.093$, $p = 0.760$, $\eta_p^2 = -0.003$), and there are no interaction effect ($F(2,130) = 0.500$, $p = 0.776$, $\eta_p^2 = -0.009$). Combined, the scores are 3.576 (2.909), 4.273 (2.842) and 4.197 (3.098). Comparing *illumotion* with teleportation ($p = 0.037$, $d = -0.242$) and steering ($p = 0.039$, $d = -0.207$) both show statistical significance. In addition, it is important to note that eight users had terminated the experiment during a task session with steering.

Workload The mean NASA-TLX scores for target reaching are 4.965 (1.413), 5.144 (1.488) and 4.735 (1.576), and for target searching, they are 4.778 (1.507), 4.783 (1.678) and 4.965 (1.556), respectively. There is no significant main effect for both technique ($F(2,130) = 0.294$, $p = 0.746$, $\eta_p^2 = -0.004$) and task ($F(1,65) = 0.614$, $p = 0.434$, $\eta_p^2 = -0.001$).

Preference *illumotion* is, generally, the most preferred technique (Fig. 5*d*). For target reaching, 32, 4 and 30 users pick *illumotion*, teleportation and steering as the first choice, respectively. For target searching, the first choice's distribution is 32, 8 and 26.

Overall User Experience (H_6) The UEQ-S provides a user experience measure on the pragmatic and hedonic quality. Pragmatic quality is a meta-measure for efficiency, perspicuity and dependability while hedonic quality is for stimulation and novelty. Combining the two qualities' scores, UEQ-S also produces an aggregated user experience score. We deployed UEQ-S at the end of the experiment to get the overall feedback from users on the locomotion techniques. For pragmatic quality, the scores are 5.595 (1.120), 4.375 (1.698) and 5.572 (1.319), respectively (Fig. 5*c*). One-way repeated-measure ANOVA shows that there is a main effect by technique ($F(2,65) = 15.680$, $p < 0.001$, $\eta_p^2 = 0.135$). *illumotion*, when compared with teleportation, shows statistical significance ($p < 0.001$, $d = 0.848$), while there is no significance with steering ($p = 1.000$, $d = 0.019$). For hedonic quality, the scores, respectively, are 5.402 (1.023), 4.807 (1.241) and 3.466 (1.435); there is a main effect ($F(2,65) = 45.080$, $p < 0.001$, $\eta_p^2 = 0.292$). Post-hoc analysis shows *illumotion* has statistically significant improvements over both teleportation ($p = 0.005$, $d = 0.523$) and steering ($p < 0.001$, $d = 1.553$). Finally, the overall UEQ-S scores (Fig. 5*c*) are 5.498 (0.936), 4.591 (1.283) and 4.519 (1.120), respectively; again, it also has a main effect ($F(2,65) = 19.173$, $p < 0.001$, $\eta_p^2 = 0.129$). Post-hoc analysis shows *illumotion* has significant improvement over teleportation ($p < 0.001$, $d = 0.808$) and steering ($p < 0.001$, $d = 0.949$) as well.

7 Discussion

Movement A property of *illumotion* is that the speed of travel scales according to where the user is trying to reach. If the user is targeting somewhere further away, based on the optically-based algorithm, it can result in faster movement. As presented in the result, *illumotion* has a significant improvement for the target-reaching task. Further, our correlation analysis seems to show that our method can achieve significantly faster speed when traveling longer distances. On the other hand, this also means that shorter distances do not seem to particularly benefit with our method; this may be related to the fact that decision-making processes take up a significant amount of time for shorter traveling.

For the target-searching task, our expectation is that our method's ability to efficiently travel large distances should assist in exploration like the target-searching task. However, the result indicates that our method has significantly improved over teleportation, but not over steering. We believe this result implies that continuous movement, which has been known to benefit spatial awareness [21], is also an important factor for exploration. However, we believe that if the exploration environment is larger, our method may have a stronger effect due to a greater need for long-distance movement. Overall, the generally faster completion time demonstrated by *illumotion* in the two tasks, along with the correlation analysis, seems to support that our method is an effective locomotion technique that is usable for short-distance while efficient for long-distance movement.

Usability As most users are already familiar with how to scale a picture on a mobile phone, we expect this gesture should be relatively easy to transfer to VR for the purpose of locomotion. Thus, it is believed that *illumotion* may be relatively easy to use. This is in contrast to teleportation, which may be less well-known to typical users who are not familiar with VR. Not to mention, most teleportation techniques assume 2D movement while we investigate 3D movement. So, even for typical VR users, 3D movement with teleportation may result in an extra layer of complexity. For steering, however, due to the prevalence of joysticks, we do not expect our method to be particularly more usable in comparison to it.

We applied SUS after each technique-task pair to better understand the usability of a technique on a particular task. The result shows that the task does not significantly change in terms of usability. Hence, the key to both reaching and searching lies in whether the locomotion technique is usable or not. The result indicates that *illumotion* seems significantly more usable compared to teleportation while similarly usable compared to steering.

Presence As mentioned, a continuous movement technique like steering is beneficial in inducing a sense of presence in the user [14]. *illumotion* should have a similar effect as it enables continuous movement as well. Further, it has been suggested that interface transparency is linked with presence, as minimizing visual interfaces may help users focus on the content [11, 36]. In contrast to teleportation, which requires a teleportation portal to specify the movement, our method only requires a minimal visual interface indicating where the anchors are. Note that visualizing the anchors is not necessary. As such, we expect *illumotion* to perform better in presence compared to teleportation, but not necessarily for steering. This expectation seems to be verified as the IPQ shows that *illumotion* performed better in terms of presence when compared to teleportation, but at the same time, our method does not have a better IPQ score compared with steering.

Cybersickness A goal of *illumotion* is to enable continuous movement while limiting cybersickness. Our belief in its ability to do so is related to the postural instability theory (Sect. 3.3); as our method should be able to have an intuitive connection between action and subsequent visual feedback, it is believed that our method may be able to reduce uncontrolled or unexpected movement and thus limit cybersickness. The discomfort scale result seems to show that our method has significantly less cybersickness compared to both steering and teleportation, albeit with only a small effect size. However, it should be noted that eight of our participants ($> 10\%$ of our total participants) had terminated the experiment during a steering trial, while there was no terminated user for both *illumotion* and teleportation trials. Thus, there is some indication that the cybersickness occurring in steering seems to have not occurred with our method or teleportation. Overall, we believe that the result seems to support that *illumotion* can enable continuous movement like steering while able to limit cybersickness to a certain degree.

Overall User Experience In addition to its efficacy, we believe *illumotion* is also "fun" to use as the user can quickly navigate around the map. It has been suggested that speed is tied with perceived funness [65]. Regardless, we want to evaluate the overall user experience of the users. First, the ranking provides us with insight into the preference users have when conducting a task. Later, at the end of the experiment, UEQ-S was administered to help us understand their preferences.

As discussed previously, *illumotion* has performed well over several factors (i.e. speed, presence, usability and limited cybersickness); this seems to be reflected by the ranking as our method is most preferred in both tasks. Particularly, users seem to prefer our method more for the searching task. This user preference for our method in the second task may indicate that the users implicitly agree that our method's ability to quickly move the environment does assist in exploration. Further, the UEQ-S shows that our method's practical quality is significantly better than teleportation for the tasks. This result aligns with our previous discussion on how our method is more usable and efficient compared to teleportation. For the hedonic quality, however, *illumotion* is significantly better compared with both other techniques; this gain in this meta-measure seems to indicate that our method is likely to indeed be more fun for the users. Funness should not be considered a trivial effect, as it results in our method having a higher overall user experience score.

8 Limitation and Future Works

We discuss the limitation in two contexts. First, regarding the user study, hand-tracking and joysticks are simultaneously compared; this mixture introduced an additional variable, but we believe it is justified as our focus is to compare with well-established locomotion techniques. Also, the one-question discomfort scale [5] (which can encompass factors other than cybersickness) and UEQ-S (a short version of UEQ) are used to alleviate the users' survey workload; future work can apply more precise tools for in-depth analysis. In addition, there can be various implementations of teleportation and steering. For our user study, we have picked a curved pointing ray for teleportation and a fixed speed for steering. Although these configurations seem to us best reflect how they are used in practical applications now, nonetheless, they and their different maximum speeds incur an inherent bias to the study that can favor *illumotion* in terms of usability (with teleportation) and completion time (with teleportation and steering). A future study may target comparing *illumotion* with other locomotion variants. Second, the proposed technique has the following limitations for future works:

- **Dominant Eye Check:** A main limitation of *illumotion* is the requirement of the dominant eye. Not only does this compel the user to conduct an optical test before usage (if they are not aware which eye is dominant), but it also places an implicit assumption that the dominant eye's visual feedback is a sufficient representation of the visual perceived by the user. Although the result shows that this assumption is adequate for providing an effective optically-driven solution, it is known that the true vision perceived by the user is a weighted mixture of both eyes with different weightings for different people [19]; we believe that could pave the way to improvement.
- **Locking 2D Movement:** As of this moment, we show *illumotion* in its 3D form for three-dimensional exploration. Although it can be argued that 3D movement is a more generalized form of 2D movement and that *illumotion* can already enable 2D-only movement via removing the upward motion parameter, 2D movement is the most common form of navigation that warrants separate work to optimize and investigate for an optically-driven method. Particularly, it is not clear how *illumotion* can handle an empty room with only a few optical cues or an uneven terrain for 2D movement. In the future, it may also be an interesting direction to mix natural walking with *illumotion* to benefit each other.
- **Unimanual Control:** Although *illumotion* is presented as a bimanual technique here, it is noted that it is possible to convert it into a unimanual technique by adding constraints and assumptions to the movement, as the core novelty lies in its optically-driven movement. Unimanual control will require additional investigation on human behavior and subsequent interaction design, but it will enable other hand interactions to mix with *illumotion*.

9 Conclusion

Inspired by how mobile phones resize images and bimanual gestures from the literature, *illumotion* transfers the resizing interaction for in-VR 3D locomotion. As discussed, the task performance of users clearly shows that *illumotion* is very efficient for long-distance travel as our method scales the speed based on the distance of the target. Reflected by the SUS, IPQ, UEQ-S and discomfort scale, *illumotion* also performed better in terms of usability, presence, overall user experience and cybersickness alleviation when compared with either teleportation, steering or both. We believe our method is a promising candidate for general locomotion that can be used for a variety of current and future VR applications, particularly for large-distance 3D movement.

REFERENCES

[1] M. Al Zayer, P. MacNeilage, and E. Folmer. Virtual locomotion: a survey. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 26(6):2315–2334, 2018.

[2] F. Argelaguet. Adaptive navigation for virtual environments. In *IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 123–126. IEEE, 2014.

[3] L. Berger and K. Wolf. Wim: fast locomotion in virtual reality with spatial orientation gain & without motion sickness. In *International Conference on Mobile and Ubiquitous Multimedia*, pp. 19–24. ACM, 2018.

[4] J. Bhandari, S. Tregillus, and E. Folmer. Legomotion: Scalable walking-based virtual locomotion. In *ACM Symposium on Virtual Reality Software and Technology (VRST)*, pp. 1–8. ACM, 2017.

[5] P. Bimberg, T. Weissker, and A. Kulik. On the usage of the simulator sickness questionnaire for virtual reality research. In *IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pp. 464–467. IEEE, 2020.

[6] J. Brooke. Sus: a "quick and dirty'usability. *Usability Evaluation in Industry*, 189(3):189–194, 1996.

[7] F. Buttussi and L. Chittaro. Locomotion in place in virtual reality: A comparative evaluation of joystick, teleport, and leaning. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 27(1):125–136, 2019.

[8] D. Calandra, F. Lamberti, and M. Migliorini. On the usability of consumer locomotion techniques in serious games: Comparing arm swinging, treadmills and walk-in-place. In *International Conference on Consumer Electronics*, pp. 348–352. IEEE, 2019.

[9] S. Chen, F. Miranda, N. Ferreira, M. Lage, H. Doraiswamy, C. Brenner, C. Defanti, M. Koutsoubis, L. Wilson, K. Perlin, et al. UrbanRama: Navigating cities in virtual reality. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 28(12):4685–4699, 2021.

[10] W. Chen, A. Plancoulaine, N. Férey, D. Touraine, J. Nelson, and P. Bourdot. 6dof navigation in virtual worlds: comparison of joystick-based and head-controlled paradigms. In *ACM Symposium on Virtual Reality Software and Technology (VRST)*, pp. 111–114. ACM, 2013.

[11] E. Cherry and C. Latulipe. Quantifying the creativity support of digital tools through the creativity support index. *ACM Transactions on Computer-Human Interaction*, 21(4):1–25, 2014.

[12] I. Cho, J. Li, and Z. Wartell. Multi-scale 7dof view adjustment. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 24(3):1331–1344, 2017.

[13] J. Clifton and S. Palmisano. Comfortable locomotion in vr: teleportation is not a complete solution. In *ACM Symposium on Virtual Reality Software and Technology (VRST)*, pp. 1–2. ACM, 2019.

[14] J. Clifton and S. Palmisano. Effects of steering locomotion and teleporting on cybersickness and presence in HMD-based virtual reality. *Virtual Reality*, 24(3):453–468, 2020.

[15] S. Cmentowski, F. Kievelitz, and J. H. Krueger. Outpace reality: A novel augmented-walking technique for virtual reality games. *Proceedings of the ACM on Human-Computer Interaction*, 6(CHI PLAY):1–24, 2022.

[16] S. Cmentowski, A. Krekhov, and J. Krüger. Outstanding: A multi-perspective travel approach for virtual reality games. In *ACM Annual Symposium on Computer-Human Interaction in Play (CHI PLAY)*, pp. 287–299. ACM, 2019.

[17] M. Di Luca, H. Seifi, S. Egan, and M. Gonzalez-Franco. Locomotion vault: the extra mile in analyzing vr locomotion techniques. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pp. 1–10. ACM, 2021.

[18] A. Drogemuller, A. Cunningham, J. Walsh, M. Cordeil, W. Ross, and B. Thomas. Evaluating navigation techniques for 3d graph visualizations in virtual reality. In *International Symposium on Big Data Visual and Immersive Analytics (BDVA)*, pp. 1–10. IEEE, 2018.

[19] W. H. Ehrenstein, B. E. Arnold-Schulz-Gahmen, and W. Jaschinski. Eye preference within the context of binocular functions. *Graefe's Archive for Clinical and Experimental Ophthalmology*, 243:926–932, 2005.

[20] N. N. Griffin and E. Folmer. Out-of-body locomotion: Vectionless navigation with a continuous avatar representation. In *ACM Symposium on Virtual Reality Software and Technology (VRST)*, pp. 1–8. ACM, 2019.

[21] M. J. Habgood, D. Moore, D. Wilson, and S. Alapont. Rapid, continuous movement between nodes as an accessible virtual reality locomotion technique. In *IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 371–378. IEEE, 2018.

[22] S. G. Hart. Nasa-task load index (NASA-TLX); 20 years later. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 50, pp. 904–908. Sage publications Sage CA: Los Angeles, CA, 2006.

[23] U. Hinrichs and S. Carpendale. Gestures in the wild: studying multi-touch gesture sequences on interactive tabletop exhibits. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pp. 3023–3032. ACM, 2011.

[24] Y. R. Kim, H. Choi, M. Chang, and G. J. Kim. Applying touchscreen based navigation techniques to mobile virtual reality with open clip-on lenses. *Electronics*, 9(9):1448, 2020.

[25] D. Krupke, P. Lubos, L. Demski, J. Brinkhoff, G. Weber, F. Willke, and F. Steinicke. Evaluation of control methods in a supernatural zero-gravity flight simulator. In *Proceedings of the GI-Workshop VR/AR*, 2015.

[26] C. Lai, X. Hu, A. Segismundo, A. Phadke, and R. P. McMahan. The comfort benefits of gaze-directed steering. In *IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMARW)*, pp. 102–103. IEEE, 2020.

[27] C. Lai, R. P. McMahan, and J. Hall. March-and-reach: A realistic ladder climbing technique. In *IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 15–18. IEEE, 2015.

[28] E. Langbehn, F. Steinicke, M. Lappe, G. F. Welch, and G. Bruder. In the blink of an eye: leveraging blink-induced suppression for imperceptible position and orientation redirection in virtual reality. *ACM Transactions on Graphics*, 37(4):1–11, 2018.

[29] J. J. LaViola Jr. A discussion of cybersickness in virtual environments. *ACM SIGCHI Bulletin*, 32(1):47–56, 2000.

[30] J.-I. Lee, P. Asente, B. Kim, Y. Kim, and W. Stuerzlinger. Evaluating automatic parameter control methods for locomotion in multiscale virtual environments. In *ACM Symposium on Virtual Reality Software and Technology (VRST)*, pp. 1–10. ACM, 2020.

[31] D. Lim, S. Shirai, J. Orlosky, P. Ratsamee, Y. Uranishi, and H. Takemura. Evaluation of user interfaces for three-dimensional locomotion in virtual reality. In *ACM Symposium on Spatial User Interaction (SUI)*, pp. 1–9. ACM, 2022.

[32] J. Liu, H. Parekh, M. Al-Zayer, and E. Folmer. Increasing walking in vr using redirected teleportation. In *ACM Symposium on User Interface Software and Technology (UIST)*, pp. 521–529. ACM, 2018.

[33] E. S. Martinez, A. S. Wu, and R. P. McMahan. Research trends in virtual reality locomotion techniques. In *IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 270–280. IEEE, 2022.

[34] A. Matviienko, F. Müller, M. Schmitz, M. Fendrich, and M. Mühlhäuser. Skyport: Investigating 3d teleportation methods in virtual environments. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pp. 1–11. ACM, 2022.

[35] M. W. McEwan, A. L. Blackler, D. M. Johnson, and P. A. Wyeth. Natural mapping and intuitive interaction in videogames. In *ACM Annual Symposium on Computer-Human Interaction in Play (CHI PLAY)*, pp. 191–200. ACM, 2014.

[36] A. McMahan. Immersion, engagement and presence: a method for analyzing 3-d video games. *The Video Game Theory Reader*, pp. 67–86, 2003.

[37] D. Medeiros, E. Cordeiro, D. Mendes, M. Sousa, A. Raposo, A. Ferreira, and J. Jorge. Effects of speed and transitions on target-based travel techniques. In *ACM Symposium on Virtual Reality Software and Technology (VRST)*, pp. 327–328. ACM, 2016.

[38] D. Medeiros, M. Sousa, A. Raposo, and J. Jorge. Magic carpet: Interaction fidelity for flying in vr. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 26(9):2793–2804, 2019.

[39] D. Medeiros, L. Teixeira, F. Carvalho, I. Santos, and A. Raposo. A tablet-based 3d interaction tool for virtual engineering environments. In *ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*, pp. 211–218, 2013.

[40] D. Mendes, M. Sousa, A. Ferreira, and J. Jorge. Thumbcam: Returning to single touch interactions to explore 3d virtual environments. In *ACM International Conference on Interactive Tabletops and Surfaces (ISS)*, pp. 403–408. ACM, 2014.
[41] C. Moerman, D. Marchal, and L. Grisoni. Drag'n go: Simple and fast navigation in virtual environment. In *IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 15–18. IEEE, 2012.
[42] M. Nabiyouni, A. Saktheeswaran, D. A. Bowman, and A. Karanth. Comparing the performance of natural, semi-natural, and non-natural locomotion techniques in virtual reality. In *IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 3–10. IEEE, 2015.
[43] N. C. Nilsson, S. Serafin, M. H. Laursen, K. S. Pedersen, E. Sikström, and R. Nordahl. Tapping-in-place: Increasing the naturalness of immersive walking-in-place locomotion through novel gestural input. In *IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 31–38. IEEE, 2013.
[44] K. Ogawa, K. Fujita, K. Takashima, and Y. Kitamura. Pseudojumpon: Jumping onto steps in virtual reality. In *IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 635–643. IEEE, 2022.
[45] R. Pausch, T. Burnette, D. Brockway, and M. E. Weiblen. Navigation and locomotion in virtual worlds via flight into hand-held miniatures. In *Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pp. 399–400. ACM, 1995.
[46] Y.-H. Peng, C. Yu, S.-H. Liu, C.-W. Wang, P. Taele, N.-H. Yu, and M. Y. Chen. Walkingvibe: Reducing virtual reality sickness and improving realism while walking in vr using unobtrusive head-mounted vibrotactile feedback. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pp. 718:1–12. ACM, 2020.
[47] C. Porac and S. Coren. The dominant eye. *Psychological bulletin*, 83(5):880, 1976.
[48] Y. Y. Qian and R. J. Teather. Look to go: An empirical evaluation of eye-based travel in virtual reality. In *ACM Symposium on Spatial User Interaction (SUI)*, pp. 130–140. ACM, 2018.
[49] L. Rebenitsch and C. Owen. Individual variation in susceptibility to cybersickness. In *ACM Symposium on User Interface Software and Technology (UIST)*, pp. 309–317. ACM, 2014.
[50] H. Regenbrecht and T. Schubert. Real and illusory interactions enhance presence in virtual environments. *Presence: Teleoperators & Virtual Environments*, 11(4):425–434, 2002.
[51] N. Rewkowski, A. Rungta, M. Whitton, and M. Lin. Evaluating the effectiveness of redirected walking with auditory distractors for navigation in virtual environments. In *IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 395–404. IEEE, 2019.
[52] M. Rheiner. Birdly an attempt to fly. In *ACM SIGGRAPH 2014 Emerging Technologies*, pp. 1–1. 2014.
[53] G. E. Riccio and T. A. Stoffregen. An ecological theory of motion sickness and postural instability. *Ecological Psychology*, 3(3):195–240, 1991.
[54] B. E. Riecke, D. Clement, A. Adhikari, D. Quesnel, D. Zielasko, and M. von der Heyde. Hyperjumping in virtual vancouver: Combating motion sickness by merging teleporting and continuous vr locomotion in an embodied hands-free vr flying paradigm. In *ACM SIGGRAPH 2022 Immersive Pavilion*, pp. 1–2. 2022.
[55] S. P. Sargunam and E. D. Ragan. Evaluating joystick control for view rotation in virtual reality with continuous turning, discrete turning, and field-of-view reduction. In *3rd International Workshop on Interactive and Spatial Computing*, pp. 74–79, 2018.
[56] T. Schmelter, L. Hernadi, M. A. Störmer, F. Steinicke, and K. Hildebrand. Interaction based redirected walking. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 4(1):1–16, 2021.
[57] M. Schrepp, A. Hinderks, and J. Thomaschewski. Design and evaluation of a short version of the user experience questionnaire (UEQ-S). *International Journal of Interactive Multimedia and Artificial Intelligence, 4 (6), 103-108.*, 2017.
[58] P. Skalski, R. Tamborini, A. Shelton, M. Buncher, and P. Lindmark. Mapping the road to fun: Natural video game controllers, presence, and game enjoyment. *New Media & Society*, 13(2):224–242, 2011.
[59] L. P. Soares, L. Nomura, M. C. Cabral, M. Nagamura, R. de Deus Lopes, and M. K. Zuffo. Virtual hang-gliding over rio de janeiro. In *SIGGRAPH Emerging Technologies*, p. 29, 2005.
[60] P. Song, W. B. Goh, W. Hutama, C.-W. Fu, and X. Liu. A handle bar metaphor for virtual object manipulation with mid-air interaction. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pp. 1297–1306. ACM, 2012.
[61] C. T. Tan, L. C. Foo, A. Yeo, J. S. A. Lee, E. Wan, X.-F. K. Kok, and M. Rajendran. Understanding user experiences across vr walking-in-place locomotion methods. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pp. 1–13. ACM, 2022.
[62] D. S. Tan, D. Gergle, P. G. Scupelli, and R. Pausch. Physically large displays improve path integration in 3d virtual navigation tasks. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pp. 439–446. ACM, 2004.
[63] H. Taunay, D. Medeiros, and A. Raposo. A distributed approach for automatic speed adjustment during navigation in 3d multiscale virtual environments. In *Symposium on Virtual and Augmented Reality (SVR)*, pp. 140–146. IEEE, 2019.
[64] C. Telkenaroglu and T. Capin. Dual-finger 3d interaction techniques for mobile devices. *Personal and ubiquitous computing*, 17:1551–1572, 2013.
[65] J. Togelius, R. De Nardi, and S. M. Lucas. Making racing fun through player modeling and track evolution. In *Proceedings of the SAB'06 Workshop on Adaptive Approaches for Optimizing Player Satisfaction in Computer and Physical Games*, 2006.
[66] X. Tong, A. Kitson, M. Salimi, D. Fracchia, D. Gromala, and B. E. Riecke. Exploring embodied experience of flying in a virtual reality game with kinect. In *2016 IEEE International Workshop on Mixed Reality Art (MRA)*, pp. 5–6. IEEE, 2016.
[67] J. Von Willich, M. Schmitz, F. Müller, D. Schmitt, and M. Mühlhäuser. Podoportation: Foot-based locomotion in virtual reality. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pp. 1–14. ACM, 2020.
[68] P. Wacker, O. Nowak, S. Voelker, and J. Borchers. Arpen: Mid-air object manipulation techniques for a bimanual ar system with pen & smartphone. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pp. 1–12. ACM, 2019.
[69] J. Wang and R. Lindeman. Leaning-based travel interfaces revisited: frontal versus sidewise stances for flying in 3d virtual spaces. In *ACM Symposium on Virtual Reality Software and Technology (VRST)*, pp. 121–128. ACM, 2012.
[70] T. Weissker, P. Bimberg, A. S. Gokhale, T. Kuhlen, and B. Froehlich. Gaining the high ground: Teleportation to mid-air targets in immersive virtual environments. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 29(5):2467–2477, 2023.
[71] P. T. Wilson, W. Kalescky, A. MacLaughlin, and B. Williams. Vr locomotion: walking¿ walking in place¿ arm swinging. In *ACM SIGGRAPH Conference on Virtual-Reality Continuum and Its Applications in Industry-Volume 1*, pp. 243–249, 2016.
[72] J. O. Wobbrock, L. Findlater, D. Gergle, and J. J. Higgins. The aligned rank transform for nonparametric factorial analyses using only anova procedures. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pp. 143–146. ACM, 2011.
[73] D. Wolf, K. Rogers, C. Kunder, and E. Rukzio. Jumpvr: Jump-based locomotion augmentation for virtual reality. In *ACM Conference on Human Factors in Computing Systems (CHI)*, pp. 1–12. ACM, 2020.
[74] J. Yoon, A. Manurung, and G.-S. Kim. Impedance control of a small treadmill with sonar sensors for automatic speed adaptation. *International Journal of Control, Automation and Systems*, 12:1323–1335, 2014.