

# Planning Locomotion Techniques for Virtual Reality Games

Cameron Moore  
cam1111@vt.edu  
Virginia Tech  
Blacksburg, Virginia, USA

Wallace S. Lages  
w.lages@northeastern.edu  
Northeastern University  
Boston, Massachusetts, USA



Figure 1: Continuous Movement Pad Technique - Player placing a movement pad.

## ABSTRACT

Locomotion is a fundamental component in many virtual reality (VR) games. However, few techniques have been designed with game's demands in mind. In this paper, we propose two locomotion techniques for fast-paced VR games: Repeated Short-Range Teleports and Continuous Movement Pads. We conducted a user study with 27 participants using these techniques against Smooth Locomotion and Teleport in a game-like scenario. We found that Movement Pads can be a suitable alternative for games, with competitive performance on various criteria such as time, damage taken, usability, workload, and user preference. On the other hand, Repeated Short-Range Teleport displayed lower usability and higher mental workload.

## CCS CONCEPTS

• **Human-centered computing** → **Virtual reality**; • **Applied computing** → **Computer games**.

## KEYWORDS

Virtual Reality, Games, Locomotion, Movement, Human-Computer Interaction

## ACM Reference Format:

Cameron Moore and Wallace S. Lages. 2023. Planning Locomotion Techniques for Virtual Reality Games. In *29th ACM Symposium on Virtual Reality Software and Technology (VRST 2023)*, October 09–11, 2023, Christchurch, New Zealand. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/3611659.3615711>

## 1 INTRODUCTION

Locomotion techniques are an integral part of modern first-person Virtual Reality (VR) games, as they enable players to explore expansive levels using standard controllers. The most common techniques in games, such as Half-Life: Alyx [25] and Bethesda Games Studio's The Elder Scrolls V: Skyrim VR [21], are Teleportation (Point-and-Teleport) and Continuous Movement (also known as Smooth Locomotion or Steering-based) [2, 9]. Teleportation moves the player instantaneously to a target location indicated by pointing, while Continuous Movement steers the player in the direction indicated by the controller orientation or thumbstick.

Despite their need, locomotion techniques are usually of secondary importance to players, which are primarily concerned with achieving game objectives like gathering items, fighting enemies, or dodging attacks. Because most techniques require controllers, they limit opportunities for environmental interaction. In addition, commonly used locomotion techniques do not support game movement patterns very well, such as moving backward while facing an enemy, moving sideways, or following a specific trajectory.

In this paper, we propose two techniques for fast-paced first-person games inspired by route-planning techniques. Route-planning techniques divide locomotion into a planning phase and an execution phase [17]. This can be advantageous in dynamic and complex scenarios since it moves some of the intricacies of movement to the planning phase and frees the controller for interaction. However,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).  
VRST 2023, October 09–11, 2023, Christchurch, New Zealand

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.  
ACM ISBN 979-8-4007-0328-7/23/10...\$15.00  
<https://doi.org/10.1145/3611659.3615711>

planning can also make the techniques more complex and cumbersome for simple movements, since more steps are required to move.

Compared to Teleportation and steering, techniques with planning elements have been significantly less explored [19]. This research contributes to addressing this gap by answering the following question: What is the effect of planning locomotion techniques on user performance in a game-like environment compared to more standard techniques such as Smooth Locomotion and Teleport.

The contributions of this work include:

- (1) The design of two new techniques: Continuous Movement Pads and Repeated Short-Range Teleport.
- (2) An evaluation with 27 participants of performance and characteristics of the techniques, along with a comparison against Smooth Locomotion and standard teleport techniques.

## 2 RELATED WORK

Here we review prior work most closely related to ours: node-based techniques and planned-movement techniques.

### 2.1 Node-based techniques

Node-based movement is a selection-based technique where the player selects a target to travel to [17]. Rotation can also be adjusted by the system or kept the same as the user's initial viewpoint upon moving from one node to another. Games such as Batman Arkham VR have extended this idea to provide several nodes which the player can select [22]. Recently, Meta, developer of the Quest 2 HMD, completed adding node-based instant movements to all the home environments within the Quest 2 standalone headset [16].

Habgood et al. describe a node-based technique which translates the user quickly and continuously between nodes individually placed by hand within the scene [12]. The technique was compared to teleport [5] and controller-based smooth locomotion and was found to have reduced feelings of motion sickness. The technique was harder to use than Teleport but easier than Smooth Locomotion. No significant difference in Presence scores was observed.

An issue with node-based techniques is how to place nodes at the desired locations efficiently. Li et al. proposed a method for automatically generating teleport graphs by predicting visually desirable positions [18]. The authors observed that positions generated by the method are often more desirable for users. One reason is that, unlike the algorithm, participants did not necessarily know the locations offering the best view before actually going there. This was also reflected in participants doing fewer teleports with the proposed method. Unfortunately, calculating the node positions is expensive, precluding it from being used in real-time: updating the teleport graph takes 30 seconds in an Intel Core i7-9700 CPU with an NVIDIA GeForce RTX 2070 GPU.

### 2.2 Planned-Movement Techniques

The essential feature of route-planning or path-planning is that they have an active and a passive phase [17]. First, the user actively plans their movement, and then the system passively takes them through the path. The path can be defined in a variety of ways. For example, in Bowman et al. route planning technique, the user uses a stylus to mark points on a 3D map [4]. In Igareshi et al., the

path is drawn directly on the screen and projected into 3D space [11]. Route planning can be considered a type of *semi-automated* technique. As with the case of node-based techniques, they have been less explored than techniques based on walking, steering, or manipulation [19].

Depending on the specific implementation, route-planning techniques can take longer than simpler techniques such as teleportation or a controller-based Smooth Locomotion. Igareshi et al. found that drawing took more time than manipulating a 3D avatar with arrow keys in a fast modality and via an animated point-and-click flying technique when animated through the path. However, in a separate modality where the user was instantly teleported to the end of the path, it took significantly less time than using arrow keys but comparable time to the flying technique without animation.

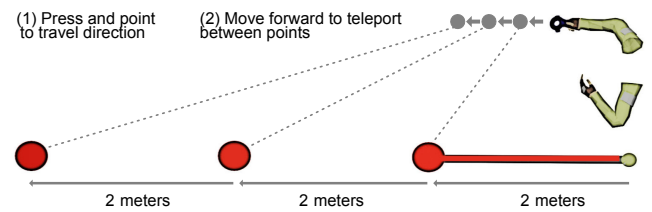
A limitation of traditional route-planning techniques is that once the plan is done, the execution is automatic. While this is reasonable for fly-throughs, it becomes an issue in dynamic scenarios where a previously planned path can suddenly become undesirable because of obstructions, enemies, or the appearance of better options. Another related issue is that only one path can be created. Even if path execution is paused, is not clear where to go next.

## 3 TECHNIQUES DESIGN

Our goal was to design techniques that provide a better balance between planning and execution. Since movement is a secondary task, visualizing future positions could result in more precise movement and less cognitive load. We explored and implemented seven techniques for visualizing and defining intermediate points for locomotion. We looked for designs that allowed players to:

- (1) Plan two or more future positions;
- (2) Quickly replan during runtime;
- (3) Execute simple movements with little overhead.

After initial pilot testing, we selected two techniques for further investigation: Repeated Short-Range Teleports (RSRT) and Continuous Movement Pads.



**Figure 2: Repeated Short-Range Teleports (RSRT) diagram showing movement estimator dots that change positions based on the controller's direction and movement and a line showing the current movement up to 2 meters based on the user's controller movement. Upon the line reaching two meters or releasing the trigger button, the user will be moved to wherever the line ends. The movement will then be repeated as long as the user keeps moving their controller.**

Repeated Short-Range Teleports (RSRT) is a gesture-based extension of Teleport. To initiate the technique, the user points the

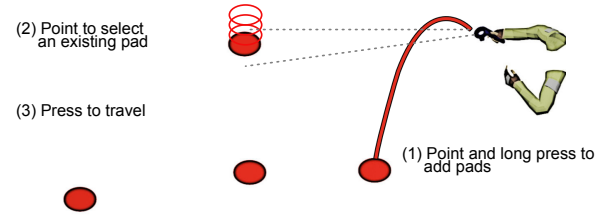
controller in the desired travel direction and presses the trigger button. Instead of just showing the destination (as in a regular teleport), Repeated Short-Range Teleports indicates the *next three teleport locations* along the controller direction. The user can then teleport to each point sequentially by moving the controller forward or backward. Similar to regular teleport, the player can reorient the controller at any point and modify the location of the next teleport points at any time (Figure 2).

The insight behind this technique is that once a teleport destination is “locked”, the player can quickly move between two known protected locations (origin and destination) without the need to aim again. This also allows the player to look forward while moving backwards or sideways, knowing exactly where the movement will end. In addition, the player can use the middle point to interact with the environment and stay there as little time as required to shoot, press a button, or engage with the environment (e.g., grab a pickup on the way to safehouse). The technique allows fast planning, with minimal overhead when compared to a regular teleport. While intermediate points can be adjusted to fall in strategic places (behind a cover, close to an item, etc.), RSRT offers less degrees of freedom than techniques where each point can be specified independently.

The distance between each teleport location was empirically set to 2 meters. This can be represented by the equation:  $\text{movement\_per\_frame} = \max(\Delta \text{hand\_position} \cdot 15 \cdot 10\text{m/s}, 10\text{m/s}) \cdot \text{frame\_time}$ . Teleport is initiated when the scaled hand has passed two meters. This is done by tracking how much the player has moved their hand between the current and previous teleport locations. This allows most users to move over three dots in a single arm extension.

Continuous Movement Pads (CMP) is a hybrid node-based technique that allows new nodes to be placed at any time. Players can place pads on the ground by holding the trigger button for 0.5 second, pointing a parabolic arc to place the pad (maximum range of  $\sim 15$  meters), and then releasing the trigger button. Once at least one pad is placed, players can select a pad to move by pointing the controller towards it. The pad closest to the user within a specified angle (60 deg) is then highlighted. To initiate the movement the player quickly presses the trigger button ( $< 0.5$  second). Any number of pads can be placed before movement. To keep the environment clean, we limit the number of simultaneous pads to 10. If this number is exceeded, the oldest placed pad is deleted (Figure 3). Since the movement between pads is continuous, we use a navigation mesh to go around any obstacles between them.

The rationale behind CMP is to give players the ability to execute complex or simple planning with the same technique. Unlike RSRT, the points don’t need to be in a single line, and unlike traditional route-planning, the points don’t need to form a single path. In addition, even when not explicitly planning, the last pads function as a breadcrumb that allows players to quickly trace back their steps if they need to retreat. Unless pads are very close to each other, the 60 degrees selection angle allows players to select a pad by roughly pointing in its direction (e.g., backward). As with RSRT, players may deviate from the planned path mid-execution by setting a new destination. Finally, the pace can be entirely managed by the player.



**Figure 3: Continuous Movement Pads (CMP) in the planning (1) and selection (2, 3) phases. In the planning phase, the user holds the trigger button for at least 0.5 seconds and releases it to place a pad at the end of a parabolic arc starting from the user’s controller. In the selection phase, the user points to an existing pad and then quickly presses and releases to move to the selected pad continuously.**

## 4 TESTBED DESIGN

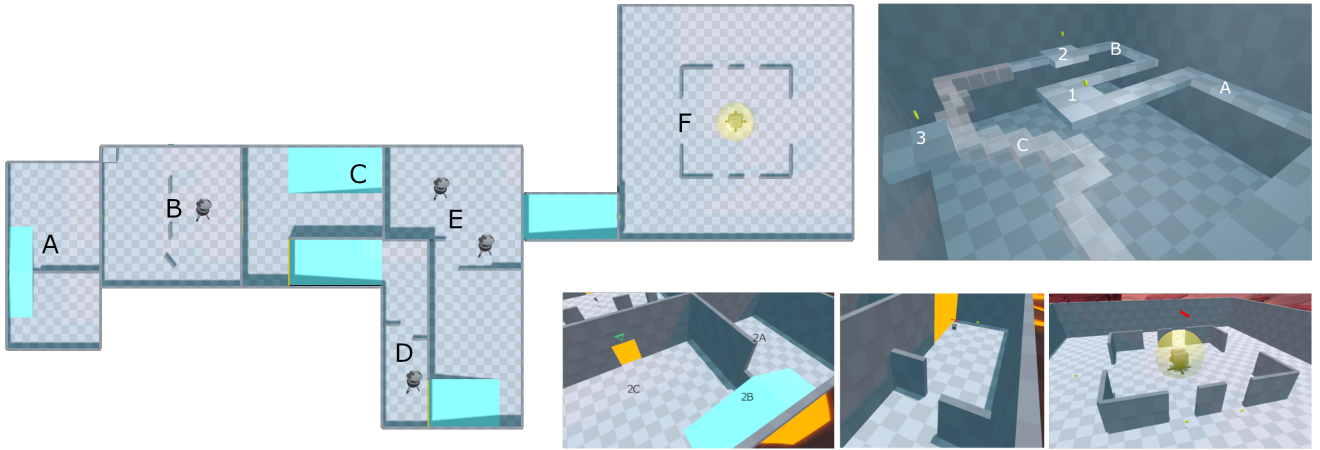
To evaluate the different techniques regarding their general usability and performance, we designed a first-person shooter game level, based on the Unity FPS Microgame [23]. We extended the demo to include common features found in these games, including multiple rooms, moving targets, cover walls, pickup items, and ramps.

The level was laid out in six key checkpoints in the following order: the starting room (A), a room with a single enemy with three walls for cover (B), a maze area (C), a corridor with two covering walls along with a single enemy guarding a coin (D), a room with two enemies guarding a single coin (E), and a final boss room (F) (Figure 4). Participants use one controller to move around with a locomotion technique and the other to grab the items in the level by clicking the grip button on the Vive controller.

The starting room (A) was designed to acclimate the player to the selected technique. It included flat areas with no enemies and two ramps. Participants were guided in this room until they were comfortable moving around and up-down the ramp. The timing started after grabbing a coin near the exit of the room.

The intermediate rooms (B, D, and E) were designed to give players a sense of urgency and danger. They were set up with different configurations of enemies and hiding surfaces. Enemies followed the player at 2.5 meters per second once they were inside a detection range of 20 meters. When they are within 10 meters of the player, they start shooting at 1 shot per second. The bullets are shot in random directions within 5 degrees from the player’s center to allow them a chance to dodge the shot. Each enemy had a starting health of 150 and could take up to 10 shots from the player before dying. Players also started with 150 health units, and each enemy shot would do 10 damage. They could check their remaining time and health by looking at their wrist.

To evaluate locomotion accuracy and precision, we included a maze leading from room C that mimics a platformer-type game. In this maze, three coins were placed on platforms for the player to grab. To get each coin, the player would have to traverse the maze without falling off the edges of the walkways. Players had to collect the coin in the middle of the room and could optionally collect the other two. Players were advised to leave this area when they reached



**Figure 4: Layout of the testbed. Left - A) starting room, B) single enemy room with three walls for cover, C) ramp to the maze area, D) corridor with two covering walls and a single enemy, E) room with two enemies and a coin, F) final boss room. Blue areas are ramps connecting different rooms. Right Bottom - Overview of areas A, D, and F. Right Top - Maze area. A) 1.5m wide walkway, B) 0.75m wide walkway, C) zigzag walkway made of cubes 1x1 meters that broke 3.5s after stepping on them.**

120 seconds left on the in-game timer so they could complete the rest of the level. To provide variation in getting the coins, the walkways had three separate variations. The first variation was a wider walkway (MA) of about 1.5 meters, the second variation was a walkway (MB) half that size at about 0.75 meters wide, and the third variation was a zigzag-like glass walkway (MC) made of cubes 1x1 meters that broke after 3.5 seconds after the player has stood on them. Players could fall by moving too close to the edge during real walking, Smooth Locomotion, or RSRT. They could also fall by placing a pad off the edge and moving to it with CMP. If the player fell off the walkway and into the area below, they would be taken back to the beginning of the maze area. No damage would be taken, and the only thing lost to the player would be time to complete the rest of the level.

The last room (F), forces the player to change movement patterns by going through a methodical, extended fight against a much stronger enemy. The turret in this room has a detection range of 30 meters, 500 health units, and can shoot at the player at a rate of 2 shots per second with a bullet spread of 5 degrees. The turret is protected by a shield, that when active makes it invulnerable. To start attacking, the player must first collect six coins placed along the sides of the walls around the boss. Upon collecting the sixth coin, the shield would drop for five seconds giving the player time to shoot at it. After five seconds, the shield would come back up. The shield would stay up at this point until the turret rotates fifteen degrees. The shield would drop for five seconds. So in order to shoot at the enemy again, the player has to expose themselves to danger in order to make it rotate and have the shield drop again. Defeating the turret requires players to keep moving to complete the level. This mimics boss battles in games, providing a bigger challenge to the player and a reason to understand and explore the features of each technique in a fun and strategic way. Upon killing the boss turret, the player would complete the level.

#### 4.1 Control Techniques

Since Smooth Locomotion and Teleport are the most commonly used techniques in games, we included them as controls.

Our Smooth Locomotion technique uses the controller’s analog directional to move. The user points the joystick in the direction they would like to move. The controller used in the study uses a touchpad. The touchpad controls the translation speed, from 0 m/s (at the center) to 5 m/s at the edge of the touchpad (along the direction of the controller). The speed change from the center to the edge is linear. If the user lets go of the touchpad, they decelerate to 0 m/s.

The Teleport implementation was a simple point-and-click teleport. The user holds the trigger button to enable a parabolic arc. The arc can then be used to indicate where they wish to move. After releasing the trigger button, the player is instantly moved to the newly designated position. To attain a comparable average speed of 5 m/s, a delay of 3 seconds was included between two consecutive uses of the technique (cooldown). The max length of the teleport was about 15 meters, so placing the technique on cooldown after a teleport would lower the max speed to about 5 m/s.

To balance the techniques relative to each other, we adjusted them to an average moving speed of 5 m/s, half of the speed used by Bhandari et al. [1]. The value was fast enough to traverse a large environment comfortably, yet slow enough for somewhat precise movements depending on the technique. Our implementation did not incorporate control over vertical movement, but the player camera automatically moves up and down to follow the level elevation (e.g., when traversing stairs). That was achieved by raycasting the level surface to find the correct height at each point.

## 5 EVALUATION

We conducted a testbed evaluation to assess the techniques. In general, we expected that planning techniques would allow players to



follow safer movement patterns, leading to fewer hits by the enemies, more precise movement, and increased level completion rates. However, since additional planning complexity when compared to quicker techniques like Teleport and Smooth Locomotion, we expected an increase in the time necessary to complete the level. Finally, because they are improvements to existing techniques, we did not expect them to rate much worse regarding usability, comfort, or mental workload. Formally our hypotheses are that our techniques will:

- **H1** - result in higher level completion than (a) Teleport and (b) Smooth Locomotion
- **H2** - result in a higher completion time than (a) Teleport and (b) Smooth Locomotion
- **H3** - result in more precise paths than (a) Teleport and (b) Smooth Locomotion
- **H4** - not differ in mental workload from (a) Teleport and (b) Smooth Locomotion
- **H5** - be as comfortable as (a) Teleport and more comfortable than (b) Smooth Locomotion
- **H6** - not differ in usability from (a) Teleport and (b) Smooth Locomotion

## 5.1 Study Design

The study followed a within-subjects design with four conditions: Continuous Movement Pads, Repeated Short-Range Teleports, Teleport, and Smooth Locomotion. We chose a within-subjects design so that participants could directly contrast one technique to another and reduce the effect of individual differences. The order of the techniques was counterbalanced using a Latin square design. On average, the study took 60 minutes per participant.

During each trial, we measured the following dependent variables: damage taken, completion time, number of falls in the maze area, and deaths while using a technique. The damage taken was based on how many overall shots the participant took from enemies and the final boss turret. The time to complete the level is the time in seconds after collecting the first coin in the starting area until the participant either loses or completes the level. The number of falls in the maze area is how many times the participant went off the maze platform and landed on the platform beneath it.

After each technique we applied the NASA Task Load Index (TLX) questionnaire without the weighting portion [10], the System Usability Scale (SUS) questionnaire [6], and the Simulator Sickness questionnaire (SSQ) [14] as an additional metric for comfort. Upon completion of all the techniques, we conducted an open interview. We did not include a Presence questionnaire since it was not the goal of this study.

## 5.2 Apparatus

We used an HTC Vive Pro Eye with matching controllers and the Lighthouse tracking system. The testbed ran on an Intel i7 8700K computer, with an NVIDIA GeForce RTX 2080 and 32 gigabytes of RAM. All techniques were implemented within Unity 3D [24], version 2019.4.8f1.

## 5.3 Participants

Participants were sampled by convenience from the local university population and screened by four main criteria: 18 years or older, ability to follow instructions in English, perfect or corrected vision, and no known vestibular or neurological dysfunction.

We recruited 31 participants, 4 of which were excluded because of technical problems or for not completing all the techniques. The age of the remaining participants was between 18 and 32 years old (median=21). Overall, 22 participants had experience with First-Person Shooter (FPS) games and 7 participants had experience with VR FPS games. Most participants (23) had experience with platform games, but only 3 people had experience with VR platform games. Regarding VR use, 12 reported using VR once every few months, 3 reported using VR once a week, 2 reported using VR once a month, 1 reported using VR every day, and 9 reported never using VR.

## 5.4 Procedures

Upon meeting with a participant, they were welcomed and given a copy of the information sheet to read. After, they were asked to fill out the background questionnaire. Next, the experimenter demonstrated the use of the HTC Vive controller and headset, including the procedure to adjust the interpupillary distance. After donning the headset, the participant would be given the controllers and asked if they were ready to start the experiment. After receiving a positive acknowledgment, the testbed application would be started and the participant would be loaded into the virtual environment.

Next, we provided the participants with basic information about the testbed. This included the player's health bar and the timer on the participant's right wrist. Participants were instructed to look at their wrist to see the remaining health value, which started at 150. The player was able to shoot as fast as they could click the trigger button or could hold the trigger button to shoot every 300 milliseconds. It also included the ability to shoot by clicking the controller assigned to the gun hand's trigger button. Before each technique's run, the technique would be explained to the participant and the participant would be given instructions on how to use it while in the starting area of the level. In order to provide a sense of urgency to complete the level, each technique run was limited to 300 seconds (5 minutes).

Before the 5 minutes started counting down for the actual trials, participants could use the technique for up to 5 minutes in the starting area to get a better understanding of it. Once they were done practicing, they were asked to grab the starting coin to begin. On the first technique's run, the player was directed through each section and how to move on to the next by either grabbing a coin or killing enemies. Upon reaching the boss fight part of the level, two requirements were explained. The first required action was to collect all six coins around the turret. Doing so would cause the shield to disappear and allow the participant to shoot at the turret. The second required action that was explained was how to cause the shield to drop again upon regeneration of the shield (make the turret rotate fifteen degrees. On subsequent technique runs, the participant was directed if they seemed confused about what they were doing still or if they asked about what to do next.

After either killing the turret enemy, running out of time, or running out of health; the participant would then be asked to remove

the headset and fill out the NASA TLX, SUS, and SSQ questionnaires. The procedure was repeated until all techniques were completed. Upon completion of all the techniques, we did a semi-structured interview. After the interview, participants were thanked for their time and then asked if they had any questions. The research protocol was reviewed by the university Institutional Review Board.

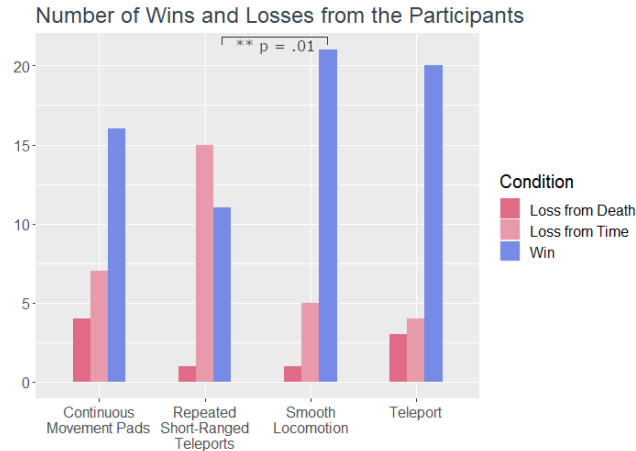
## 6 RESULTS AND ANALYSIS

For data analysis of non normal data and questionnaires we used Noguchi et al.'s nonparametric multiple contrast test procedure (MCTP) [20] using Dunnett's test for contrast. We used the R package nparcomp, version 3.0 [15]. We compared the proposed technique to Teleport and Smooth Locomotion at 95% significance level.

### 6.1 Level Completion Rates (H1)

Participants would lose by running out of health after taking enough hits from the enemies or running out of time after 300 seconds. Here we look into the impact of the techniques on level completion.

In terms of level completion rates (Figure 5), Smooth Locomotion and Teleport techniques resulted in the highest number of wins with 21 wins and 20 wins respectively. Continuous Movement Pads (CMP) resulted in fewer wins than Smooth Locomotion or Teleport (16). Repeated Short-Range Teleports (RSRT) resulted in only 11 total wins and was the only technique with less wins than losses.



**Figure 5: Number of wins and the number of losses from either participant death or running out of time.**

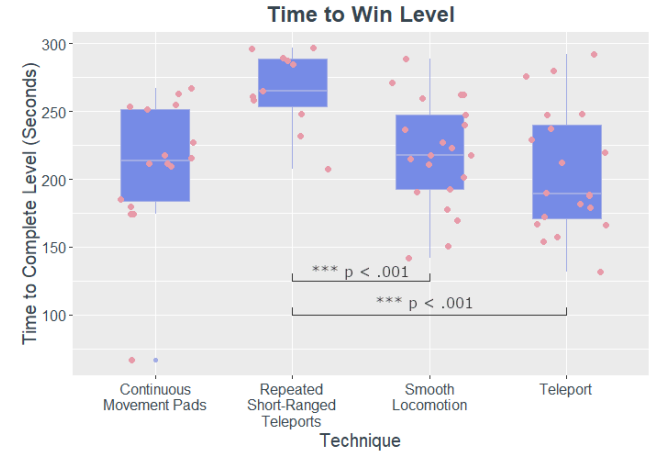
In comparison to Teleport (*wins* = 20), neither CMP (*wins* = 16,  $T^g = -1.158$ ,  $p = .578$ ) nor RSRT (*wins* = 11,  $T^g = -2.317$ ,  $p = .085$ ) were significantly different. In comparison to Smooth Locomotion (*wins* = 21), RSRT (*wins* = 11,  $T^g = -3.330$ ,  $p = .010$ ) was significantly different. CMP (*wins* = 16,  $T^g = -1.542$ ,  $p = .393$ ) was not significantly different from Smooth Locomotion.

In terms of number of deaths, in comparison to Teleport ( $M=0.111$ ), we found no significant differences between Continuous Movement Pads ( $M = 0.148$ ,  $T^g = 0.4$ ,  $p = .981$ ) or Repeated Short-Range Teleports ( $M = 0.037$ ,  $T^g = -1.0$ ,  $p = .658$ ). Similarly, relative to Smooth Locomotion ( $M=0.037$ ), we found no significant differences between

Continuous Movement Pads ( $M = 0.148$ ,  $T^g = 1.4$ ,  $p = .484$ ) or Repeated Short-Range Teleports ( $M = 0.037$ ,  $T^g = 0.0$ ,  $p = 1.000$ ).

In summary, we did not find evidence for difference in completion rate for CMP when compared to Smooth Locomotion and Teleport. However, participants had fewer wins with RSRT, primarily due to exceeding completion time, rather than getting hits.

### 6.2 Level Completion Time (H2)



**Figure 6: The times to successfully win the level. This includes only trials that resulted in wins.**

Level completion time (Figure 6) was computer-measured from the end of the starting area to killing the boss turret. Total time was capped at 300 seconds (loss from Time). MCTP analysis showed that compared to Teleport ( $M=215.2$ ), use of RSRT resulted in a significantly higher time to complete ( $M = 281.9$ ,  $T^g = 6.1$ ,  $p < .001$ ). We did not find significant differences between CMP and Teleport ( $M = 237.8$ ,  $T^g = 1.9$ ,  $p = .207$ ). Compared to Smooth Locomotion ( $M=236.7$ ), RSRT resulted in a significantly higher time to complete ( $M = 281.9$ ,  $T^g = 4.5$ ,  $p < .001$ ). Again, Smooth Locomotion time to complete the level did not differ significantly from CMP ( $M = 237.8$ ,  $T^g = 0.5$ ,  $p = .975$ ).

In summary, our hypothesis that our new planning techniques would lead to higher completion time was only partially supported with only RSRT taking more time than the control techniques.

### 6.3 Path Precision (H3)

To evaluate path precision, we looked into the number of falls in the maze area (Figure 7). We found that CMP number of falls ( $N = 3$ ) was not significantly different from Teleport ( $N = 7$ ,  $T^g = -1.44$ ,  $p = .445$ ) and significantly fewer than Smooth Locomotion ( $N = 70$ ,  $T^g = -7.19$ ,  $p < .001$ ). RSRT ( $N = 108$ ), on the other hand, was significantly worse than both Teleport ( $N = 7$ ,  $T^g = 9.70$ ,  $p < .001$ ) and Smooth Locomotion ( $N = 70$ ,  $T^g = 2.76$ ,  $p = .028$ ).

In summary, contrary to our hypothesis, CMP was as precise as teleport and RSRT was worse. On the other hand, CMP was more precise than Smooth Locomotion but RSRT was not, partially supporting H3b.

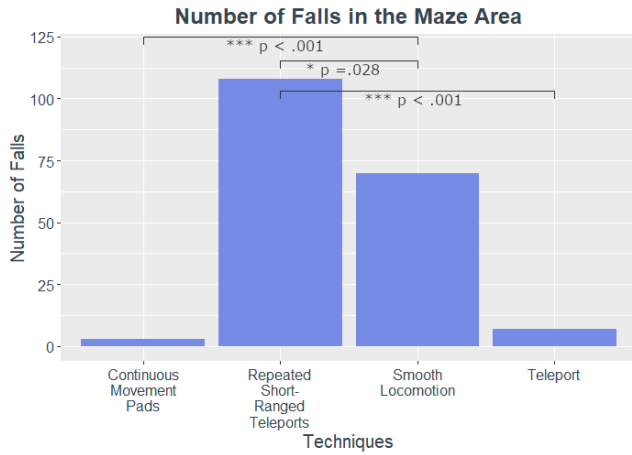


Figure 7: Number of Falls in the Maze Area

#### 6.4 Mental Workload (H4)

We used the NASA Task Load Index Questionnaire to assess the overall workload score from 6 different subscales: “Mental Demands, Physical Demands, Temporal Demands, Own Performance, Effort, and Frustration” [10]. Usually, there are weights to these different scales; however, in this study we used the Raw TLX version.

For the raw TLX scores (Figure 8), MCTP analysis showed that compared to Teleport ( $M=4.52$ ), RSRT ( $M = 8.90, T^g = 5.1, p < .001$ ) scored significantly higher. We did not find significant differences between CMP and Teleport ( $M = 4.90, T^g = 0.5, p = .954$ ). In comparison to Smooth Locomotion ( $M=4.31$ ), we found significantly higher scores for RSRT ( $M = 8.90, T^g = 5.9, p < .001$ ). Again, Smooth Locomotion’s TLX did not differ significantly from CMP ( $M = 4.90, T^g = 0.9, p = .745$ ).

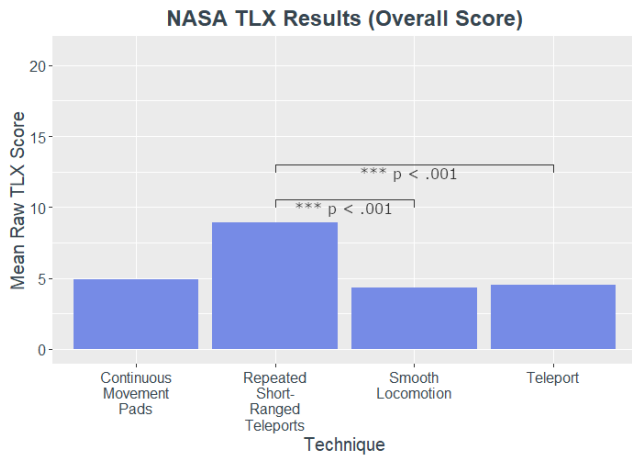


Figure 8: Overall raw workload score for each technique.

In summary, we did not find evidence that CMP induces higher mental workload than the control techniques. However, we found evidence that RSRT can be worse than Smooth Locomotion in this respect.

#### 6.5 Overall System Usability and Simulator Sickness (H5, H6)

To be of practical impact, techniques need a high level of usability and be as comfortable as existing techniques.

For the standard composite score from the System Usability Scale Questionnaire (Figure 9), MCTP analysis showed that RSRT ( $M = 40.2$ ) scored significantly lower than Teleport ( $M = 83.6, T^g = -10.7, p < .001$ ) and Smooth Locomotion ( $M = 83.6, T^g = -9.5, p < .001$ ). We did not find significant differences in usability between CMP (79.9) and Teleport ( $M = 83.6, T^g = -1.1, p = .564$ ) or CMP and Smooth Locomotion ( $M = 83.6, T^g = -1.2, p = .488$ ).

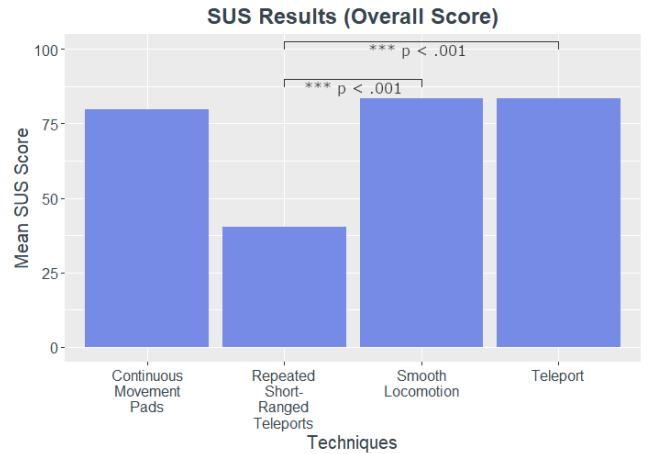


Figure 9: Mean usability scores of the SUS responses.

Regarding the Total Severity Score from the Simulator Sickness Questionnaire (Figure 10), we did not find any significant differences between the proposed techniques and controls. When compared to Teleport ( $M=1.8$ ), we found no significant difference for CMP ( $M = 5.9, T^g = .36, p = .992$ ) or RSRT ( $M = 3.7, T^g = 0.9, p < .813$ ). We also did not find any significant difference between Smooth Locomotion ( $M=6.51$ ) against CMP ( $M = 5.9, T^g = -1.2, p = .483$ ) or RSRT ( $M = 3.7, T^g = -1.9, p < .172$ ).

In summary, we did not find evidence that the proposed techniques induce significant higher cybersickness than Teleport or Smooth Locomotion. Symptoms for all techniques can be considered negligible [13]. We also did not find evidence for better or worse usability with CMP, however, usability for RSRT was lower than Smooth Locomotion.

#### 6.6 Participant Impressions

After completing the experiment, we asked participants to rank the techniques and conducted a semi-structured interview about the applicability of the techniques to a variety of scenarios.

**6.6.1 Preferences.** Participants were asked to rank techniques in order of preference. We assigned 4 points for the first-placed technique and 0 for the last. The results ( $n=27$ ) are shown in Table 1. Of the four techniques, CMP was the most preferred and had the highest overall ranking with 76 points total. Smooth Locomotion is very close with only a one-point difference at 75 points in the

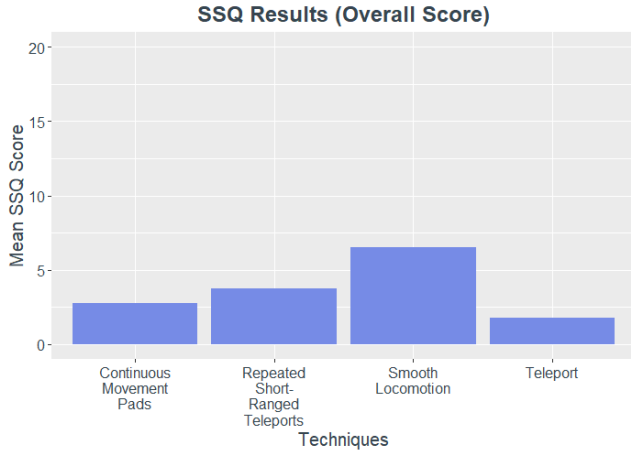


Figure 10: Mean total severity scores for the SSQ responses.

overall ranking score. Teleport is also not far behind, with only a nine-point difference between it and CMP at 66 points. RSRT was ranked least preferred fifteen times out of twenty-seven and had the lowest ranking score overall with 18 points.

Table 1: Ranks of the 4 techniques based on participants' preferences.

	CMP	Smooth	Teleport	RSRT
Overall Rank	1	2	3	4
Ranking Score	76	75	66	18
Most Preferred	10	9	5	0
Least Preferred	3	0	1	15

We asked participants whether they would prefer using a specific technique in different types of in-game environments. In a fast-paced environment, Smooth Locomotion was mostly ranked first because of how it mimics movement in traditional games. In a leisurely environment, CMP was most preferred because the pads could be set up and used at their own pace. One response also compared it to Google Maps in how it moves. In a stealthy environment, participants mainly picked CMP because of its ability to plan where you go and easily move and stop between them. Teleport was also preferred, although slightly less than CMP, because you could go over gaps or around things and the lack of considerable overall movement.

For both fast-paced action and leisure movement, Smooth Locomotion was preferred, with CMP slightly behind. Participants ranked Smooth Locomotion as best because it allowed them to react more quickly and better control the overall pace. Many participants also mentioned CMP because they could choose where to go with precise movements and control the pace to be fast or slow.

**6.6.2 Learning Curve.** We asked participants to list unintuitive parts of the techniques. For RSRT, participants commented that it was hard to see the guiding line and the position indicators of future movement. They also found the need for hand movement "weird" or "obnoxious". Other responses listed RSRT as getting easier to use once they realized they could swing their arms to

use it, a motion characterized by moving the controller in an arc around the shoulder joint or elbow joint, or that it eventually got easier on them physically. From observations during the study, most participants used a jabbing motion, a movement characterized by linearly moving the controller forwards and backwards. Some participants listed that RSRT made the coins harder to grab and interact with or that they felt they had a more difficult time in the narrow spaces (such as the maze section). RSRT was consistently the most disliked. Answers related to RSRT ranged from not being able to move quickly enough, to feeling hard to control, inconsistent, or being physically draining.

We asked participants about the learning curve of the techniques. In general, CMP was rated as easy to learn (Figure 11). Participants mentioned that CMP became easier when they learned they could place them beforehand, how far they could use the pads, or once they learned that they could place multiple and move between them. On the other hand, most participants disagreed that RSRT as easy to learn. Even though some users indicated that it became easier when they realized they could swing their arms.

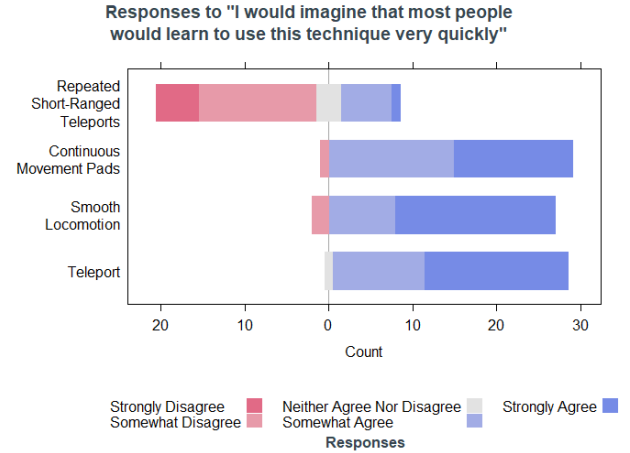


Figure 11: Answers to the questions: I believe most people would learn how to use this technique easily.

Some participants mentioned feeling sick, with many responses attributing Smooth Locomotion as the cause of their sickness in the game. Other responses included that falling within the maze area felt most sickening or that sudden movements, small movements (such as from RSRT), or delays could cause them to feel sick.

## 7 DISCUSSION

In this section, we revisit our larger research questions, and discuss the tradeoffs of the techniques and implications for design.

### 7.1 General Performance

Our techniques were designed to help players plan and preview movements before committing to them. RSRT does that by showing the next three teleport points along the controller direction, while CMP requires players to mark the pads and then select one to move to. In this sense, RSRT is a simpler and more powerful technique than CMP. RSRT's preview and execution are integrated



into a single button press, movement, and release. CMP on the other hand, requires six button presses for the same three teleports. Furthermore, RSRT allows players to walk backwards or towards a previous teleport location, while CMP requires players to either turn around or try to locate the pad without looking.

Despite the theoretical advantages, RSRT did not perform as well as we expected. Participants were slow with the technique and often exceeded the allotted time. We hypothesize that one of the reasons is that although users could stop at the first, second, or third location, teleportation was initiated by hand movement. Because of the scaled-down mapping, users need to move slowly to stop once the intermediate or final destination is reached (otherwise a new sequence of 3 points would be activated). CMP, however, could be used quickly and without fear of error.

On the other hand, Continuous Movement Pads exceeded our expectations, doing fairly well when compared to Teleport and Smooth Locomotion in terms of general usability, completion time, damage taken, and overall effort. In terms of participant preference, it ranked first, ahead of Smooth Locomotion and Teleport. In the maze area, it exceeded the performance of Smooth Locomotion regarding the number of falls, since the waypoints in connection with the navigation mesh created a very precise path.

Another reason for Continuous Movement Pads' (CMP) good performance is the way it *optionally* extended Teleport's basic functionality. If planning is unnecessary (e.g., the path is free of danger), players can use it as a regular teleport, placing one pad at a time and moving to it continuously. Once a more complex situation arises, they can seamlessly add multiple pads or use previously laid-out pads. In other words, Continuous Movement Pads can fall back to being used just like Teleport, but with little overhead in operation and effort. Our test environment also did not require participants to use planning or revisit locations, so participants would often plan a single point at a time. This indicates that planning capabilities were only used to a partial extent. Further studies are needed to explore more thoroughly the use of this technique.

Although CMP performed on par with the control techniques, repeatedly laying down the pads could become bothersome if one tries to move in a "point-and-click" mode. A dedicated button that places a pad and continuously moves to it upon button release may be an improvement. RSRT could also be improved by requiring users to restart the technique after reaching the target or implementing a "snap to point" to stabilize hand movement.

## 7.2 Choosing Between Techniques

Smooth Locomotion, as shown in this study and others, can induce more cybersickness [8, 26]. It is also less precise than Teleport and CMP and offers no planning capabilities. We do not recommend for applications requiring planning or precise path control. Anecdotally, players prefer Smooth Locomotion because it is less disorienting and feels more realistic, which could be covered by CMP.

Teleport has some qualities of planning techniques since players can visualize and adjust the position before committing to move. As such, it can be used when precise positioning is required. Teleport is known for producing lower levels of simulator sickness; however, it can also be more disorienting [3, 8]. We think Teleport is still a

good choice for applications where locations are rarely revisited or where little planning is required.

Continuous Movement Pads allows complex planning with little more complexity than Teleport. Because of its continuous movement, we expect CMP to be attractive to Smooth Locomotion users. In our view, CMP is a good choice for dynamic applications where complex planning or precise path following may be required, which is the case for most games.

RSRT is comparable in terms of simulator sickness with Teleport since it is a discrete technique that minimizesvection-induced cybersickness [7]. However, since it was slower, less precise, and rated as more challenging to understand, we do not recommend its use in its current form.

## 7.3 Limitations

Each technique has several parameters that can modify their behavior. Since we balanced all the techniques regarding average traversal time, our findings may not be applicable when there is no limitations on average speed. For example, a version of Teleport without cooldown may rank higher than Smooth Locomotion. It is also worth noting that although the major difference between Teleport and CMP is the route planning, they also differ on how they move between two points (instantaneously vs continuously). This aspect may have also contributed to the observed difference between the techniques.

Our testbed presented three challenges in the same game-like order: maze with no enemies, mobile enemies, and a final boss. Although we counterbalanced the techniques to distribute their influence across the conditions, order effects for the level design may remain. Our sample also had participants with different levels of VR game experience and possibly familiarity with each challenge. We only partially addressed the time constraint by reminding participants that were staying too long in the first section. Studies interested in the interaction between level design and technique should look into counterbalancing challenge presentation or evaluating them separately.

## 8 CONCLUSION

In this paper, we introduced two locomotion techniques inspired by planning techniques. We found that Continuous Movement Pads (CMP) provided similar performance metrics of time, damage taken, usability, and overall workload compared to Smooth Locomotion and Teleport. Conversely, we found that Repeated Short-Range Teleports (RSRT) was significantly worse regarding the overall number of victories, time, and falls. It was also harder to use and had a higher overall workload than Smooth Locomotion and Teleport. Participants preferred CMP the most, due to its ability to reuse the pads and perceived precision. RSRT was the least preferred of all the techniques. This shows that planning can improve game locomotion, but techniques must be carefully designed to avoid increased complexity and discomfort.

## REFERENCES

- [1] Jiwan Bhandari, Paul R MacNeilage, and Eelke Folmer. 2018. Teleportation without Spatial Disorientation Using Optical Flow Cues.. In *Graphics interface*. 162–167.

- [2] Costas Boletsis. 2017. The New Era of Virtual Reality Locomotion: A Systematic Literature Review of Techniques and a Proposed Typology. *Multimodal Technologies and Interaction* 1, 4 (2017). <https://doi.org/10.3390/mti1040024>
- [3] D.A. Bowman, D. Koller, and L.F. Hodges. 1997. Travel in immersive virtual environments: an evaluation of viewpoint motion control techniques. In *Proceedings of IEEE 1997 Annual International Symposium on Virtual Reality*. 45–52. <https://doi.org/10.1109/VRRAIS.1997.583043>
- [4] Doug A Bowman, Elizabeth T Davis, Larry F Hodges, and Albert N Badre. 1999. Maintaining spatial orientation during travel in an immersive virtual environment. *Presence* 8, 6 (1999), 618–631.
- [5] 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*. 205–216.
- [6] John Brooke. 1996. Sus: a 'quick and dirty' usability scale. *Usability evaluation in industry* 189, 3 (1996).
- [7] Jean-Rémy Chardonnet, Mohammad Ali Mirzaei, and Frédéric Merienne. 2021. Influence of navigation parameters on cybersickness in virtual reality. *Virtual Reality* 25, 3 (2021), 565–574.
- [8] Chris G Christou and Poppy Aristidou. 2017. Steering versus teleport locomotion for head mounted displays. In *International conference on augmented reality, virtual reality and computer graphics*. Springer, 431–446.
- [9] MP Jacob Habgood, David Wilson, David Moore, and Sergio Alapont. 2017. HCI lessons from PlayStation VR. In *Extended abstracts publication of the annual symposium on computer-human interaction in play*. 125–135.
- [10] Sandra Hart. 1988. *NASA Task Load Index*.
- [11] Takeo Igarashi, Rieko Kadobayashi, Kenji Mase, and Hidehiko Tanaka. 1998. Path drawing for 3D walkthrough. In *Proceedings of the 11th annual ACM symposium on User interface software and technology*. 173–174.
- [12] 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>
- [13] R Kennedy, J Drexler, D Compton, K Stanney, D Lanham, and D Harm. 2003. Configural scoring of simulator sickness, cybersickness and space adaptation syndrome: similarities and differences. *Virtual and adaptive environments: Applications, implications, and human performance issues* (2003), 247.
- [14] Robert S Kennedy, Norman E Lane, Kevin S Berbaum, and Michael G Lillenthal. 1993. Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness. *The international journal of aviation psychology* 3, 3 (1993), 203–220.
- [15] Frank Konietzschke, Marius Placzek, Frank Schaarschmidt, and Ludwig A. Hothorn. 2015. nparcomp: An R Software Package for Nonparametric Multiple Comparisons and Simultaneous Confidence Intervals. *Journal of Statistical Software* 64, 9 (2015), 1–17. <http://www.jstatsoft.org/v64/i09/>
- [16] Ben Lang. 2022. Quest home spaces get node-base movement, a first step toward upcoming social features. <https://www.roadtovr.com/quest-2-horizon-home-social-node-locomotion/>
- [17] 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.
- [18] Changyang Li, Haikun Huang, Jyh-Ming Lien, and Lap-Fai Yu. 2021. Synthesizing Scene-Aware Virtual Reality Teleport Graphs. *ACM Trans. Graph.* 40, 6, Article 229 (dec 2021), 15 pages. <https://doi.org/10.1145/3478513.3480478>
- [19] Esteban Segarra Martinez, Annie S Wu, and Ryan P McMahan. 2022. Research trends in virtual reality locomotion techniques. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 270–280.
- [20] Kimihiro Noguchi, Riley S Abel, Fernando Marmolejo-Ramos, and Frank Konietzschke. 2020. Nonparametric multiple comparisons. *Behavior Research Methods* 52, 2 (2020), 489–502.
- [21] Playstation. 2017. The Elder Scrolls V: Skyrim VR - Tutorial | PS VR. <https://www.youtube.com/watch?v=TznIBmvKvXs>.
- [22] Rocksteady Studios. 2016. Batman: Arkham VR.
- [23] Unity Technologies. 2019. Fps microgame. <https://learn.unity.com/project/fps-template>.
- [24] Unity Technologies. 2023. Unity. <https://unity.com/>.
- [25] Valve. 2020. Half-Life: Alyx - Locomotion Deep Dive. <https://www.youtube.com/watch?v=TX58AbJq-xo>.
- [26] Tim Weißker, André Kunert, Bernd Fröhlich, and Alexander Kulik. 2018. Spatial updating and simulator sickness during steering and jumping in immersive virtual environments. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 97–104.