Virtual Navigation considering User Workspace: Automatic and Manual Positioning before Teleportation

Yiran Zhang, Nicolas Ladevèze, Thi Thuong Huyen Nguyen, Cédric Fleury, Patrick Bourdot

Virtual Navigation considering User Workspace: Automatic and Manual Positioning before Teleportation

Yiran Zhang [1,2], Nicolas Ladevèze [1], Huyen Nguyen [1], Cédric Fleury [2] and Patrick Bourdot [1]

[1]Université Paris-Saclay, CNRS, LIMSI, VENISE team, Orsay, France

[2]Université Paris-Saclay, CNRS, Inria, LRI, Orsay, France

{yiran.zhang,nicolas.ladeveze,huyen.nguyen,patrick.bourdot}@limsi.fr,cedric.fleury@lri.fr

ABSTRACT

Teleportation is a navigation technique widely used in virtual reality applications using head-mounted displays. Basic teleportation usually moves a user's viewpoint to a new destination of the virtual environment without taking into account the physical space surrounding them. However, considering the user's real workspace is crucial for preventing them from reaching its limits and thus managing direct access to multiple virtual objects. In this paper, we propose to display a virtual representation of the user's real workspace before the teleportation, and compare manual and automatic techniques for positioning such a virtual workspace. For manual positioning, the user adjusts the position and orientation of their future virtual workspace. A first controlled experiment compared exocentric and egocentric manipulation techniques with different virtual workspace representations, including or not an avatar at the user's future destination. Although exocentric and egocentric techniques result in a similar level of performance, representations with an avatar help the user to understand better how they will land after teleportation. For automatic positioning, the user selects their future virtual workspace among relevant options generated at runtime. A second controlled experiment shows that the manual technique selected from the first experiment and the automatic technique are more efficient than the basic teleportation. Besides, the manual technique seems to be more suitable for crowded scenes than the automatic one.

CCS CONCEPTS

• **Human-centered computing** → **Virtual reality**; Interaction techniques; User studies.

KEYWORDS

Locomotion; teleportation; real workspace; virtual workspace; virtual object access; spatial awareness.

1 INTRODUCTION

Teleportation is a popular locomotion technique that allows a user to move beyond the limits of their available physical space while minimizing simulator sickness [26, 48]. Using such technique, the user can select a destination point and instantaneously appear at this new location in the virtual environment (VE). However, this technique usually does not consider the physical space surrounding the user and their position inside this space. Due to a lack of awareness of the real workspace boundaries, the user may quickly reach the limits of their workspace while performing a virtual task.

Common solutions, such as alerting the user when they approach the boundaries, increase the user's mistrust in virtual reality (VR) systems, and often break their immersion and sense of presence. These solutions also induce the user to stay still and perform many small teleportations, instead of using their real movements to reach objects of the VE, especially when they get stuck in a corner or against a boundary of their real workspace. On the other hand, real walking could be highly beneficial for improving immersion. Solutions like redirected walking [35] provide a compelling approach for the user to explore large VEs while overcoming the constraints of the real workspace. However, the redirected walking algorithms usually require physical spaces larger than 6m × 6m [6]. It may not be possible for common users with head-mounted displays (HMD) because their real workspace is also limited by the room size.

This work aims to help the user to gain a prior knowledge of the accessible area of the VE, which allows them to access multiple virtual objects with real walking and avoid reaching the real workspace limits. To achieve this goal, we propose to display and manipulate a virtual representation of the real workspace when using the teleportation technique. This virtual representation of the real workspace, also called *virtual workspace* in this paper, is similar to the concepts of *vehicle* [11] or *stage* [20].

Optimizing future access of the user to multiple objects within their physically accessible area could be useful in many scenarios. For example, in VR escape room games [1, 2], if the user can position their virtual workspace close to some area of interest where the clues are possibly hidden, they can fully explore this area by walking and thus avoid a lot of unnecessary teleportations. With first-person shooter games [5], by choosing an appropriate virtual workspace position, they can physically move to hide and attack enemies. Another example is some complex VR training system [4] that consists of several assembly tasks at different locations in the VE. Positioning the virtual workspace around an assembly area of each sub-task might help the user to focus better on knowledge acquisition and assembly procedure learning as they do not need to manage virtual objects' accessibility while completing this sub-task.

In order to help the user to define the position and orientation of this virtual workspace, we explore two strategies, named manual positioning and automatic positioning. The former allows the user to manually adjust the position and orientation of their future virtual workspace. The latter, using clustering techniques, automatically generates a series of possible virtual workspaces considering the interactive object layout in the VE. The user can select their future virtual workspace among the relevant options proposed by the system. We investigated these different positioning techniques in two controlled experiments. In the first one (pre-study), we assessed three manual positioning techniques and selected the most appropriate one. In the second one, we compared the selected manual positioning and the automatic positioning techniques to the basic teleportation technique. We evaluated the user performance in terms of efficiency, number of teleportations, and cognitive load, considering various virtual object layouts. From the results, we derived some usability guidelines for such techniques.

The paper is organized as follows. Section 2 reviews related work about teleportation and navigation techniques that take the user's real workspace into account. Section 3 details the two positioning strategies. Section 4 describes the two experiments and analyze the results. Finally, section 5 concludes by proposing some guidelines and discussing open problems of this contribution.

2 RELATED WORK

The mapping between the real and virtual world is a fundamental issue of every VR applications, and various previous works explore solutions to manage this relationship. To avoid the user's collisions with the real world and grant direct access to virtual objects, some applications choose to have a fixed one-to-one mapping between the real and virtual environments. For example, Cheng et al. [17] and Sra et al. [41] propose to procedurally generate the virtual environment based on a 3D scan of the real world with a depth camera. Consequently, the size and shape of the virtual environment are constrained. Redirected walking [35] or other view distortion techniques [45] can be used to map a large virtual environment to a small real workspace while allowing the user to walk freely. Impossible Space [44] also uses a self-overlapping architectural layout to allow the user to walk through multiple virtual rooms while staying in the same real room. However, these solutions are not suitable for all applications since they require a reasonably large real workspace, and physical walking could also be tiresome when the user has to travel long distances.

Virtual navigation is a generic solution which allows the user to go beyond the real workspace limits [12]. However, it breaks the one-to-one mapping between the real and virtual environment. Such mismatch may result in safety issues as the user could collide with physical obstacles that are invisible in the virtual world. The user may feel afraid of encountering real-world obstacles [18] and thus alter their movement behaviors [18, 39]. In such cases, it is crucial to provide the user with a way to visualize and understand the limits of their real workspace. For example, 3DM [14] not only allows the user to walk naturally on a magic carpet representing the tracking space, but also to move this magic carpet over a long-distance using a steering technique. Magic Barrier Tape [19] employs a virtual barrier tape to indicate the available walking area

to the user. The user can go beyond the boundaries by "pushing" the tape. More recently, Chen et al. [15] present a human-joystick technique that takes the real workspace boundaries into account to prevent collisions during navigation.

Teleportation is another virtual-navigation technique widely used in VR applications. Basic teleportation allows the user to instantly appear at a remote target position using a pointing technique [12]. The instantaneous transition of viewpoint avoids the sensory conflict between the visual feedback and the user's vestibular systems, which reduces simulator sickness compared to other locomotion techniques [26, 48]. However, teleportation lacks optical flow, which limits the user's ability to perform path finding and leads to disorientation [7, 9]. Several existing approaches aim to improve teleportation. Point and Teleport technique [13, 23] allows the user to specify their orientation before the teleportation. Jumper [10] employs the user's eye gaze to specify the target destination and thus enables hand-free teleportation. Dash [9] quickly but continuously displaces the user's viewpoint to retain optical flow cues. The out-of-body locomotion technique [25] allows the user to seamlessly switch between a first-person and a third-person view to reduce the confusion caused by discontinuous avatar movements for multiple-user teleportation.

Some approaches combine teleportation with real walking to better use the available real workspace and facilitate real walking. For example, Redirected Teleportation [29] requires the user to step into a portal to activate teleportation, which unobtrusively reorients and re-positions the user away from the tracking space boundary. Interactive Portals [21] reorient the user to a safe position via portals sliding up from the ground in the center of the CAVE. Switch techniques [49] help the user to recover a one-to-one mapping between real and virtual workspace inside some areas in the VE. The user can access the virtual objects of the areas by walking. However, these areas need to be predefined according to the layout of VE and the shape of the real workspace, which makes it hard to apply in a generic context. Apart from that, the SteamVR plugin for Unity [3] can provide a virtual representation of the real workspace boundaries before the teleportation. However, the user cannot manipulate the orientation of this virtual representation.

In this paper, we overcome the limits of the existing approaches by designing two types of positioning techniques (manual vs. automatic) that allow the user to manage their virtual workspace before each teleportation, and to develop strategies to access multiple objects by real walking. These techniques do not rely on specific virtual object layouts or prior knowledge of the user's real workspace.

3 POSITIONING TECHNIQUES

In this section, we will present our considerations and design for the virtual workspace positioning techniques.

3.1 Manual Techniques

The manual positioning techniques use a 3D volume representing the user's real workspace, and the user can directly control its position and orientation in the VE to customize the teleportation.

Interacting with a predefined volume has been used for multiple-object selection (MOS) [31, 42] to select the enclosed objects at once. The user can use manipulation techniques, such as the go-go
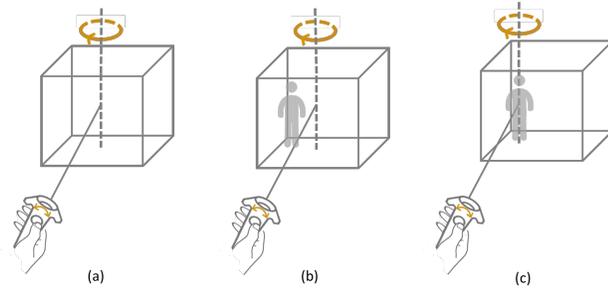
Figure 1: Three manual positioning techniques: the user (a) rotates the volume around its central vertical axis; (b) rotates the 3D volume and an avatar around the central axis; and (c) rotates the 3D volume around the avatar's vertical axis.

technique [31], to manipulate this volume remotely to select distant objects. In our approach, we extended the ray casting technique [33] to enable the virtual workspace manipulation.

In our first manual technique, the 3D volume of the virtual workspace appears when the user presses the touch pad of the controller. The intersection point between the virtual ray and the virtual ground determines the future position of this volume. The user can rotate the volume around its vertical axis (see Figure 1(a)) by sliding the finger in a circle on the touch pad with a one-to-one mapping. The objects fully or partially enclosed by the volume are selected and highlighted with a more intense colour. The user can release the touch pad to end the manipulation, and then they will be teleported into the newly specified virtual workspace with a correct matching to their real workspace.

To enhance the spatial awareness [12], we propose two other techniques, which add additional visual information to the virtual workspace representation: an avatar showing directly how the user will land after the teleportation. This avatar is a ghost representation of the user at their future location, which helps them to get self-related information.

These techniques differ in their rotation axis position. One technique uses an exocentric manipulation by rotating the 3D volume and the avatar around the volume's central vertical axis (see Figure 1(b)). The exocentric information can help people to see global trends [8] and enhances the size judgment [32]. The other technique uses an egocentric manipulation by rotating the 3D volume around the avatar's vertical axis (see Figure 1(c)). The egocentric cues can help people to gather the self-related information and result in more accurate distance estimation [30].

3.2 Automatic Technique

In many VR applications, the virtual objects that the user can directly interact with are usually predefined in the scenarios. Based on the layout of these objects and considering the user's actual real workspace, the system can compute possible virtual workspaces and propose them to the user. The user can then select their future virtual workspace among relevant options depending on the task requirements. Our approach to compute a set of suitable virtual workspaces is to: (i) organize objects into clusters by grouping or splitting them; (ii) compute bounding volumes for each cluster; and (iii) repeats the above steps until the size of the bounding volume and the size of user's real workspace become equivalent.

In the first step of the approach, bottom-up or top-down algorithms can be used to cluster virtual objects. The bottom-up algorithms treat each virtual object as a singleton cluster at the beginning, and then successively merge pairs of clusters until the user's real workspace can no longer enclose the bounding volume of a cluster. The top-down algorithms start with a cluster that includes all virtual objects and split the cluster recursively until each sub-cluster is smaller than the user's real workspace. Different criteria can be applied to organize objects into a cluster. For example, K-means is a fast and straightforward heuristic to group pairs of clusters based on their nearest mean [28]. Beyond those geometrical approaches, one can also use semantic knowledge about the scene to align objects of the same kind [43], or use collisions to define clusters when simulating the real-world behaviors [34].

Algorithm 1 Virtual workspaces positioning algorithm

Input: ObjList, UsrRWS
Output: PosList

1: **procedure** TopDown($ObjList, UsrRWS$)
2:     $N \leftarrow ObjList.length$
3:     **if** $N = 1$ **then**
4:         $PosList.Add(AABB(ObjList).position)$
5:         **return** $PosList$
6:     **else**
7:         $bbox \leftarrow AABB(ObjList)$
8:         **if** $!EncapTest(bbox, UserRealWS)$ **then**
9:             $lN, rN \leftarrow Kmeans(ObjList, 2)$
10:            $PosList.append(\text{TopDown}(lN, UsrRWS))$
11:            $PosList.append(\text{topDown}(rN, UsrRWS))$
12:         **else**
13:            **return** $PosList$

As a first prototype, we implemented a simple top-down algorithm (see Algorithm 1) to compute the possible virtual workspace positions in sublinear time. It uses K-means (with k=2) to split the inputting objects (ObjList) into disjoint subsets based on their 2D positions (x, z), and generates a bounding volume around each subset using axis-aligned bounding box (AABB) approach [24]. The recursive splitting creates a binary tree, and processes until a subset either contains only one virtual object, or its bounding volumes can be encapsulated in the user's real workspace (UsrRWS). By traversing the binary tree, the algorithm creates and returns a list of bounding volume positions from the leaf nodes. Based on this list, the system can subsequently instantiate 3D volumes at each bounding volume position to represent the possible virtual workspaces. In the example of Figure 2, our algorithm provides five virtual workspaces for a given configuration of the VE and a 3m × 3m user's real workspace. This algorithm is a first implementation to test the related interaction technique and user acceptability of an automatic technique. It can be improved later by considering arbitrary-oriented bounding boxes (OBB) or other clustering methods.

The proposed virtual workspaces are normally invisible to the user. As soon as the user's virtual ray collides with a virtual workspace, it is displayed along with an avatar indicating the user's future destination (see Figure 3(a)). When the user walks into an overlapping
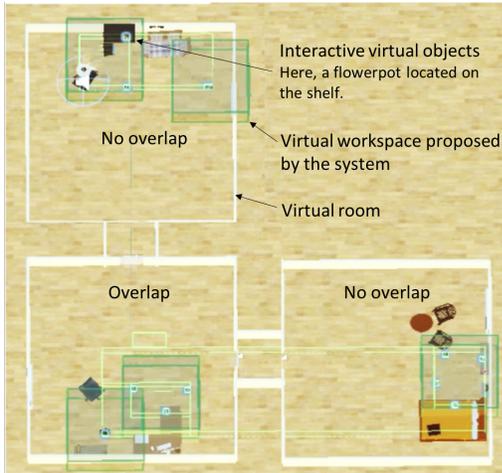
Figure 2: Results of the positioning algorithm: five virtual workspaces (green) are proposed to the user for 12 interactive objects (light blue) located in three virtual rooms (white). Three workspaces are disjointed (no overlap rooms), and two overlap each other (overlap room).
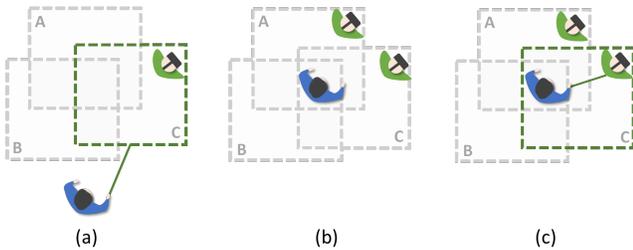

Figure 3: Top view of the user (a) selects a virtual workspace among relevant options; (b) enters an overlapping area and the system displays avatars to represent the connected virtual workspaces; and (c) selects a subsequent workspace by pointing the virtual ray to its corresponding avatar.

area between multiple virtual workspaces, the system allows the user to switch directly from their current virtual workspace to one of the connected workspaces by showing their related avatars (see Figure 3 (b)). The user can then select a new workspace by pointing the virtual ray to its associated avatar (see Figure 3 (c)). The virtual objects enclosed inside the selected workspace are highlighted with a more intense color.

4 EXPERIMENT 1

As a first step, we conducted a controlled experiment to evaluate the three manual techniques proposed in section 3.1. We aimed to assess the benefits of including an avatar at the user's future destination in the virtual workspace. We also wanted to compare egocentric and exocentric manipulation techniques in terms of user spatial awareness and performance. The experiment thus compared the following TECHNIQUES (see Figure 4):

- *Exo-without-avatar* is an exocentric technique allowing the user to move the virtual workspace representation and to rotate it around its central axis. No preview of the user's future position is offered.
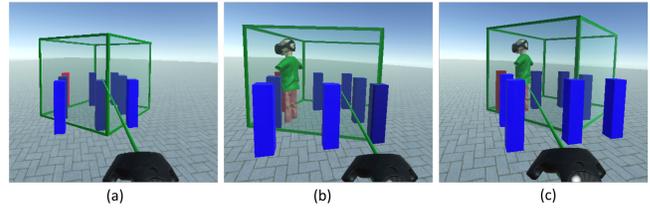

Figure 4: Conditions of the first experiment: (a) *Exo-without-avatar*, (b) *Exo-with-avatar* and (c) *Ego-with-avatar*.

- *Exo-with-avatar* is an exocentric technique for which the rotation axis is still at the center of the virtual workspace representation. An avatar is included to show the user's future position. This avatar is a simplified human body wearing a head-mounted display.
- *Ego-with-avatar* is an egocentric technique that uses the future position of the user (i.e. the avatar position) as the rotation axis of the virtual workspace representation. The same simplified avatar is used in this condition.

We did not include an egocentric technique without the avatar because the rotation axis is invisible, making it difficult for the user to understand the manipulation. The experiment was a within-subject design with TECHNIQUE as a factor. The order of the TECHNIQUES was counterbalanced across participants using a balanced Latin square.

4.1 Hypothesis

We expected the conditions with the avatar would help the user to anticipate their next position in the virtual scene. We also assumed that *Exo-with-avatar* would highlight the entire virtual workspace and would make it easier to enclose the targeted virtual objects, while *Ego-with-avatar* focuses on the user's future destination and causes less disorientation. Therefore we formulated the following hypotheses:

H1 *Exo-with-avatar* and *Ego-with-avatar* will reduce disorientation and the time required to find a target after the teleportation, compared to *Exo-without-avatar*.

H2 Less time will be required for positioning the virtual workspace representation with *Exo-with-avatar* than with *Ego-with-avatar*.

H3 Less time will be required to find a target after the teleportation with *Ego-with-avatar* than with *Exo-with-avatar*.

4.2 Participants

We recruited 12 participants, aged between 25 and 31 (6 men and 6 women). Only one person was left-handed. Three participants had VR experience. 11 out of 12 rated their everyday usage of head-mounted displays as very low.

4.3 Experiment setup

The VR setup consisted of an HTC Vive Pro Eye with both position and orientation tracking, as well as integrated eye-tracking technology. The virtual environment was rendered using Unity with a resolution of 1440 × 1600 pixels per eye at 90 Hz. The experiment room supported a 3m × 3m tracking area. User input was detected using a Vive handheld controller.
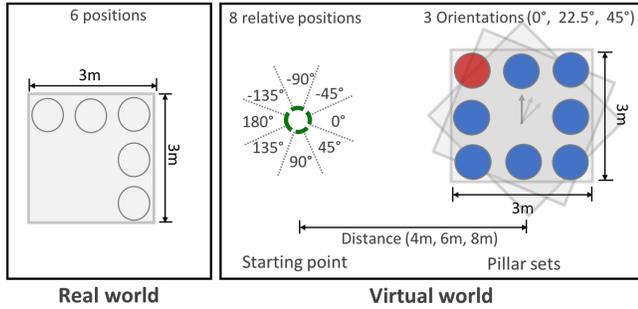
Figure 5: Each starting position in the virtual scene positioned the participant at one of the 6 positions in the real space (left). The pillar set was located with 3 different orientations, in one of the 8 directions around the participant, and at one of the 3 distances from the participant (right).

4.4 Experimental task

To assess spatial awareness, existing studies [12, 47] measure the time needed for participants to reorient themselves and find objects previously seen in the virtual scene. We used a similar task to evaluate the three techniques in terms of spatial awareness and manipulation efficiency. Before each trial, the participant was asked to walk to a starting point presented by a green dotted circle on the floor. Then, a set of pillars were displayed, and the trial started. The pillar set consists of one red and seven blue pillars located on four sides of a 3m × 3m square. The participant had to adjust the virtual workspace position to enclose all the pillars as if they wanted to be able to access all of them without having to perform additional teleportations. Once all the pillars were enclosed, the participant could release the Vive controller touch pad to travel to the selected destination. Subsequently, the participant needed to touch the red pillar with the Vive controller to end the trial.

4.5 Procedure

Each participant was welcomed, received instructions on the task, and signed an informed consent form. After setting up the head-mounted display, we calibrated the eye tracker. For each technique, the participant first experienced training trials. Next, the participant completed 24 trials in randomized order resulting from a particular subset of the full combination of 6 starting points in the real space, 8 relative directions between the starting point and the pillar set (-135°, -90°, -45°, 0°, 45°, 90°, 135°, 180°), 3 distances between the starting point and the pillar set (4m, 6m, 8m) and 3 orientations of the pillar set (0°, 22.5°, 45°), as illustrated in Figure 5. After each technique, the participant filled out a questionnaire. At the end of the experiment, the participant also ranked the three techniques according to their preference. The whole experiment lasted approximately 45 minutes.

4.6 Data collection

We registered 864 trials: 3 techniques × 24 repetitions × 12 participants. For each trial, we logged the following measures:

- *Task Completion Time (TCT)*: the total duration of a trial. The measurement started when the participant arrived at the starting point and ended when the red pillar was touched.
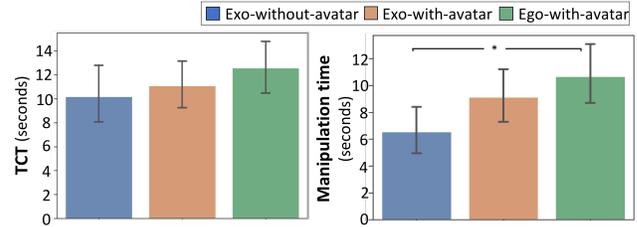

Figure 6: Mean *TCT* (left) and *manipulation time* (right) by technique. Error bars show 95% confidence intervals (CI).

- *Manipulation time*: the time used to enclose all the pillars. The measurement started when the virtual workspace representation collided with one of the pillars and ended when the participant triggered the teleportation.
- *Target identification time*: the time that the participant needed to reorient themself to find the red pillar. The measurement started just after the teleportation, and ended when the eye gaze of the participant collided with the red pillar (measured by the eye-tracking system of the HTC Vive).

We used the NASA-TLX questionnaire [27] and also added two more questions about *anticipation* (Were you able to anticipate where you would be after teleportation?) and *disorientation* (Did you feel disoriented after teleportation?). Criteria were graded on a 21-point scale and later converted to a 100-point score.

4.7 Statistical results

For each measure, we used normal QQ-plots and Shapiro-Wilk Tests to analyze data normality. *TCT*, *manipulation time* and *target identification time* were not normally distributed, so we applied a log-transformation to analyze them, as recommended by Robertson & Kaptein [37] (p. 316). To minimize the noise in our data, we averaged the 24 repetitions of each technique. We then ran a one-way ANOVA test and conducted post-hoc analysis with paired sample T-tests with Bonferroni corrections[1]. Means (M) are reported with standard deviations.

For *TCT* (see Figure 6, left), we did not find a significant effect of technique ($F_{2,22} = 1.291$, $p = 0.295$), and all conditions had close mean values: *Exo-without-avatar* ($M = 10.18\pm4.21$s), *Exo-with-avatar* ($M = 11.09\pm3.43$s) and *Ego-with-avatar* ($M = 12.56\pm3.85$s).

For *manipulation time* (see Figure 6, right), we observed a significant effect of technique ($F_{2,22} = 4.683$, $p = 0.0202$). Pairwise comparisons showed that participants spent less time with *Exo-without-avatar* ($M = 6.54\pm3.32$s) than with *Ego-with-avatar* ($M = 10.66\pm4.07$s, $p = 0.0042$). No significant differences were found between *Exo-without-avatar* and *Exo-with-avatar* ($M = 10.66\pm4.07$s, $p = 0.132$), and between *Exo-with-avatar* and *Ego-with-avatar* ($p = 0.64$).

For *target identification time* (see Figure 7, left), we detected a significant effect of technique ($F_{2,22} = 17.40$, $p < 0.0001$). Pairwise comparisons showed that target identification time was significantly shorter with *Exo-with-avatar* ($M = 0.81\pm0.20$s, $p = 0.0021$) and *Ego-with-avatar* ($M = 0.77\pm0.38$s, $p = 0.0015$) than with *Exo-without-avatar* ($M = 1.79\pm1.06$s). No significant differences were found between *Exo-with-avatar* and *Ego-with-avatar* ($p = 0.57$).

---

[1]All statistical analyses were performed with R and we used a significance level of $\alpha = 0.05$ for all tests.
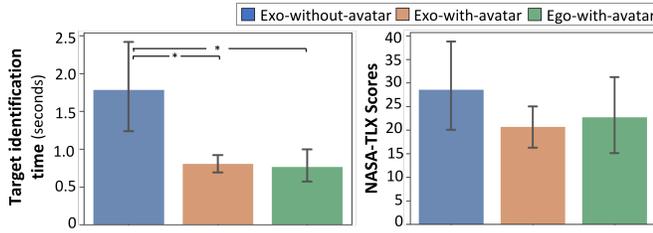
Figure 7: Mean *target identification time* (left) and NASA-TLX score (right) by TECHNIQUE. Error bars show 95% CI.
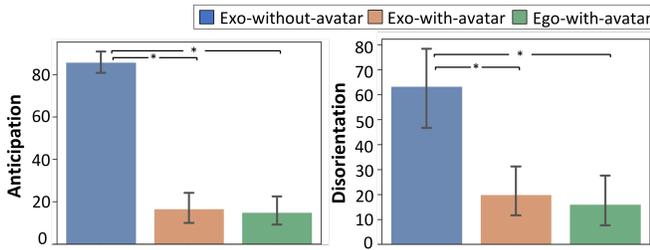

Figure 8: Mean *anticipation* (left) and *disorientation* (right) score by TECHNIQUE. Error bars show 95% CI.

For the subjective questionnaire, we used Friedman's tests and Wilcoxon signed-rank tests for post-hoc analysis in conformity with such non-parametric data. For cognitive load, we did not find a significant effect of TECHNIQUE ($\chi^2(2) = 1.721$, $p = 0.423$) on the NASA-TLX score (see Figure 7, right). However, we detected a significant effect of TECHNIQUE on *anticipation* ($\chi^2(2) = 19.19$, $p < 0.0001$) and *disorientation* ($\chi^2(2) = 21.80$, $p < 0.0001$). Post-hoc analysis shows that *Exo-with-avatar* ($M = 16.67 \pm 12.67$, $p = 0.0057$) and *Ego-with-avatar* ($M = 15.00 \pm 12.06$, $p = 0.0061$) resulted in a significantly better anticipation (see Figure 8, left) compared to *Exo-without-avatar* ($M = 85.83 \pm 9.00$). It shows that significantly less disorientation (see Figure 8, right) was perceived by the participants with *Exo-with-avatar* ($M = 20.00 \pm 19.19$, $p = 0.015$) and *Ego-with-avatar* ($M = 16.08 \pm 19.08$, $p = 0.0099$) than with *Exo-without-avatar* ($M = 63.33 \pm 30.70$). In addition, 11 out of 12 participants preferred *Exo-with-avatar* and *Ego-with-avatar* over *Exo-without-avatar*, and 8 out of 12 participants ranked *Ego-with-avatar* as their favorite condition.

4.8 Discussion

Despite the non-significant difference in task completion time, we found that the use of an avatar has a significant impact on manipulation time and target identification time. On the one hand, participants performed faster the manipulation task of the virtual workspace with *Exo-without-avatar* than with *Ego-with-avatar*. Manipulation with *Exo-without-avatar* seems also slightly faster than with *Exo-with-avatar*, but the difference is not significant. On the other hand, the two conditions with avatar resulted in significantly less disorientation and thus a shorter target identification time, which supports **H1**. Even if the results could be predictable since the visual feedback directly shows how the user will "land" in the VE, it is interesting to measure its actual impact. While the user spends slightly more time positioning the avatar, they can plan and better

understand the upcoming teleportation, decreasing disorientation and the time needed to complete the task after the teleportation. In the conditions with avatar, positioning the avatar can increase the cognitive load, but finding the target requires less cognitive effort. This can explain why the overall difference in cognitive load is not significant.

Contrary to our expectations, we were not able to find significant differences between exocentric and egocentric techniques in term of user performance. For manipulation time, the difference is not significant, which does not support **H2**. For target identification time, no significant difference was found between *Exo-with-avatar* and *Ego-with-avatar*, which rejects **H3**. We also did not detect significant differences between *Exo-with-avatar* and *Ego-with-avatar* for cognitive load, anticipation and disorientation. However, participants preferred *Ego-with-avatar* to *Exo-with-avatar* according to the questionnaires. In particular, participants reported that *Ego-with-avatar* allowed them to "focus more on themselves" (P6) during the manipulation step, was "easier for positioning themselves" (P9), and was "easier for finding" (P3) the target objects after teleportation.

The two conditions with avatar seem to reach close performance levels. However, we wanted to select one of them to compare it with the automatic positioning technique in the second experiment. Consequently, we decided to choose the *Ego-with-avatar* based on the user preference.

5 EXPERIMENT 2

The goal of this experiment is to compare the manual technique selected from the first experiment (i.e., *Ego-with-avatar*) and the automatic technique to a basic teleportation. In this experiment, we set up a more realistic task similar to an escape room game. The VE consisted of a series of virtual rooms. Participants needed to select multiple objects to escape each room and continue the exploration. The experiment followed a [3×2] within-subject design with the following factors:

- TECHNIQUE: *Basic*, *Manual* and *Automatic*,
- LAYOUT: *Overlap* and *No-overlap*.

For TECHNIQUE (see Figure 9), the three variations are:

- *Basic* is the basic teleportation technique used as a baseline. A virtual ray appeared when participants pressed the controller's touchpad. The teleportation position was determined by the collision point between the ray and the virtual floor. It was represented by a green dotted circle. Participants activated teleportation by releasing the touch pad.
- *Manual* is the manual *Ego-with-avatar* technique described in section 4. Participants used the virtual ray to manipulate the virtual workspace representation instead of the dotted circle used in the *Basic* technique.
- *Automatic* is the automatic technique described in Section 3. Participants used the virtual ray to select a virtual workspace among the relevant options proposed by Algorithm 1.

For LAYOUT, two different object layouts were used:

- *No-overlap*: objects were laid out in two separate areas, which could be included in the participant's real workspace. The *Automatic* technique thus proposed one virtual workspace for each area.
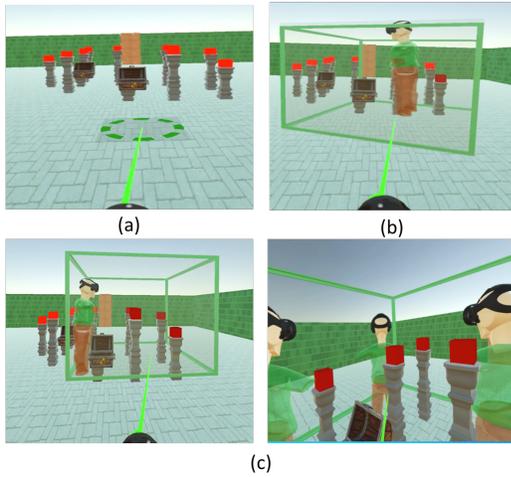
Figure 9: Three conditions for TECHNIQUE: (a) *Basic* teleportation; (b) *Manual* technique selected from EXPERIMENT 1; (c) *Automatic* technique: the user selects their future virtual workspace among relevant options (left), or selecting avatars in overlapping conditions (right).

- *Overlap*: objects were spread in an area larger than the participant's real workspace. The *Automatic* technique proposed a set of virtual workspaces which enclosed only a subset of the objects.

The order of the TECHNIQUEs was counter-balanced across participants using a balanced Latin square, and the order of the LAYOUT was also counter-balanced for each TECHNIQUE.

5.1 Hypothesis

In comparison to the basic teleportation, we expected that the manual and the automatic positioning techniques would allow the user to access more easily multiple objects using physical walking. With the automatic technique, the user could select their future virtual workspace among the proposed ones and thus would be able to avoid the manipulation step required for positioning it. However, in a crowded virtual environment, the large number of proposed virtual workspaces could be confusing for the user. We, therefore, formulated the following hypotheses:

- **H1** *Automatic* and *Manual* will result in better user performance, compared to the *Basic* teleportation.
- **H2** *Automatic* and *Manual* will result in better sense of presence, compared to the *Basic* teleportation.
- **H3** In *No-overlap*, *Automatic* performs better than *Manual*.
- **H4** In *Overlap*, *Manual* performs better than *Automatic*.

5.2 Participants

We recruited 12 participants, aged between 25 and 32 (7 men, 5 women). 6 participants had VR experience. 8 out of 12 rated their everyday usage of head-mounted displays as very low.

5.3 Experiment setup
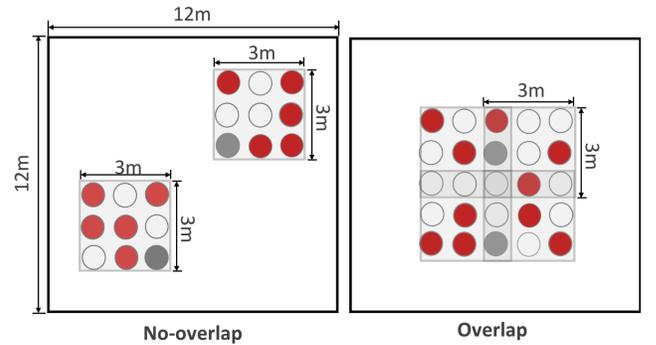
The VR setup was the same as in Experiment 1.

Figure 10: Two object layouts were used: (left) *No-overlap* in which objects (red) and treasure boxes (gray) were randomly located in two disjointed 3m × 3m areas; and (right) *Overlap* in which objects (red) and treasure boxes (gray) were randomly placed in a single 5m × 5m area. The light gray rectangles represent the virtual workspaces computed in the *Automatic* condition.

5.4 Experimental task

Participants traveled in a large virtual environment composed of nine rooms. In each room, they had to select multiple objects. When participants reached their real workspace limits, a warning sign appeared in their field of view with an alarm sound to ensure participants' safety. The first room was used for the training task, and the other eight were set up for the evaluation: half with the *No-overlap* layout and half with the *Overlap* layout. Both types of layouts required participants to access ten target objects, grab them with the controller, and bring them back to one of two treasure boxes one by one (see Figure 10). With the *No-overlap* layout, the objects and the treasure boxes were located within two disjointed 3m × 3m areas. Each area contained five targets and one treasure box located randomly on 6 of 9 positions. With the *Overlap* layout, the objects and the treasure boxes were placed randomly on 12 of 25 positions located in a single 5m × 5m area. In *Automatic* condition of this layout, the algorithm computed four overlapping virtual workspace positions to cover the full area.

5.5 Procedure

Each participant was welcomed, received instructions on the task, and signed an informed consent form. For each TECHNIQUE, the participant first completed the training task with eight targets and two treasure boxes located inside four 3m × 3m areas (three overlapped and one not-overlapped). During this step, the experimenter was allowed to answer their questions, if any. Next, the participant completed eight trials (4 with *No-overlap* and 4 with *Overlap*, or vice versa). After each TECHNIQUE, they filled out an Igroup Presence Questionnaire (IPQ) [36, 38] to measure the sense of presence and two NASA-TLX questionnaires [27] to assess the cognitive load of each LAYOUT. We used the color of the rooms to help the participant to differentiate the two LAYOUTs. At the end of the experiment, the participant also ranked the three techniques according to their preference. The whole experiment lasted around 60 min.

5.6 Data collection

We registered 288 trials: 3 TECHNIQUEs × 2 LAYOUTs × 4 repetitions × 12 participants. For each trial, we collected the following measures:
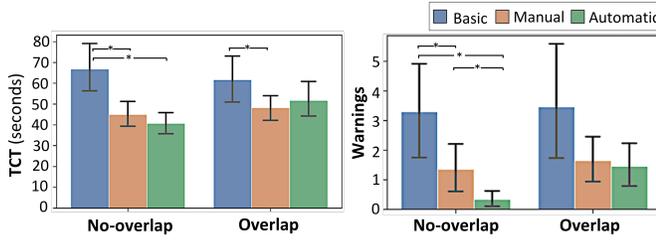
Figure 11: Mean *TCT* (left) and *warnings* (right) by TECH-NIQUE × LAYOUT. Error bars show 95% CI.

- *Task Completion Time (TCT)*: the total duration of a trial. The measurement started when the participant entered a room and ended when all objects were put in the treasure boxes.
- *Warnings*: the number of times the participant triggered the warning sign.
- *Teleportations*: the number of teleportations performed.

Each question of the NASA-TLX was graded on a 21-point scale and converted to a 100-point score. Each question of the IPQ was graded on a 7-point Likert scale (from 0 to 6).

5.7 Statistical results

To minimize the noise in our data, we averaged the 4 repetitions of each TECHNIQUE × LAYOUT. Means (M) are reported with standard deviations.

For *TCT*, we used normal QQ-plots and Shapiro-Wilk Tests to analyze data normality. The data was not normally distributed, so we applied a log-transformation to analyze it following statistical recommendations [37]. A two-way repeated measures ANOVA with the model TECHNIQUE × LAYOUT revealed a significant effect of TECHNIQUE ($F_{2,22} = 11.08$, $p = 0.0005$) and interaction effect ($F_{2,22} = 4.79$, $p = 0.019$), but no significant effect of LAYOUT ($F_{1,11} = 2.13$, $p = 0.17$) was found. For TECHNIQUE, post-hoc Tukey HSD tests indicated that performing the task with *Manual* ($M = 45.70 \pm 9.65s$, $p = 0.0019$) and *Automatic* ($M = 45.40 \pm 9.46s$, $p = 0.0011$) was significant faster than with *Basic* ($M = 61.70 \pm 20.19s$). For TECHNIQUE × LAYOUT (see Figure 11, left), post-hoc Tukey HSD tests shown the task with the *No-overlap* layout was significant faster to achieve with *Manual* ($M = 43.75 \pm 10.63s$, $p = 0.0005$) and *Automatic* ($M = 39.80 \pm 8.75s$, $p < 0.0001$) than with *Basic* ($M = 64.02 \pm 20.60s$). For the task with the *Overlap* layout, *Basic* ($M = 58.99 \pm 19.23s$) was significantly different from *Manual* ($M = 46.97 \pm 10.86s$, $p = 0.044$), but not from *Automatic* ($M = 49.87 \pm 13.75s$, $p = 0.17$). In all cases, no significant differences were found between *Manual* and *Automatic*.

For *Warnings*, we used non-parametric tests in conformity with the nature of count data. We first aggregated the data by TECHNIQUE and a Friedman test revealed a significant effect of TECHNIQUE ($\chi^2(2) = 11.51$, $p = 0.0032$). Wilcoxon Signed Rank tests[2] shown that participants triggered significantly more *Warnings* with *Basic* ($M = 3.38 \pm 3.32$) than with *Manual* ($M = 1.50 \pm 1.11$, $p = 0.041$) and *Automatic* ($M = 0.90 \pm 0.67$, $p = 0.034$). No significant differences were found between *Manual* and *Automatic*. We then split the data by LAYOUT and ran a Friedman test for each LAYOUT (see Figure 11, right). For *No-overlap*, it indicated a significant effect of TECHNIQUE

---

[2]In this experiment, all Wilcoxon Signed Rank tests were performed with Holm-Bonferroni corrections.
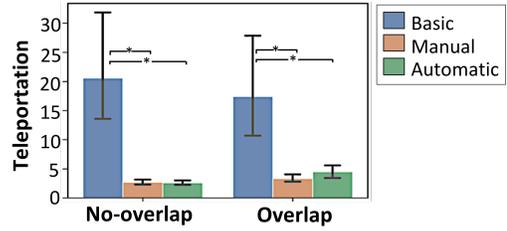

Figure 12: Mean *Teleportations* by TECHNIQUE × LAYOUT. Error bars show 95% CI.

($\chi^2(2) = 14.28$, $p = 0.0008$). Wilcoxon Signed Rank tests shown that participants triggered significant more *Warnings* for the *No-overlap* layout with *Basic* ($M = 3.29 \pm 2.91$) than with *Manual* ($M = 1.35 \pm 1.41$, $p = 0.029$) and *Automatic* ($M = 0.33 \pm 0.44$, $p = 0.011$). *Manual* was also significantly different than *Automatic* ($p = 0.032$) for *No-overlap*. For *Overlap*, no significant effect of TECHNIQUE was found ($\chi^2(2) = 4.95$, $p = 0.084$).

For *Teleportations*, we also used non-parametric tests. We first aggregated the data by TECHNIQUE and a Friedman test revealed a significant effect of TECHNIQUE ($\chi^2(2) = 19.50$, $p < 0.0001$). Wilcoxon Signed Rank tests shown that participants teleported significantly more with *Basic* ($M = 19.00 \pm 17.49$) than with *Manual* ($M = 3.00 \pm 0.85$, $p = 0.0050$) and *Automatic* ($M = 3.53 \pm 0.90$, $p = 0.0015$). No significant difference was found between *Manual* and *Automatic*. We then split the data by LAYOUT and ran a Friedman test for each LAYOUT (see Figure 12). For both LAYOUTs, we had similar results to that of the aggregated data, pointing out that there was probably no interaction effect of TECHNIQUE×LAYOUT.

For the NASA-TLX questionnaires, we aggregated the data by TECHNIQUE and did not detect a significant effect of TECHNIQUE ($\chi^2(2) = 2.426$, $p = 0.2974$). We also aggregated the data by LAYOUT and observed that the *Overlap* layout ($M = 42.22 \pm 21.72$) induced a significantly higher cognitive load than the *No-overlap* layout ($M = 38.75 \pm 19.76$, $p = 0.0050$). Further analysis on the data split by LAYOUT revealed that a significant higher cognitive load was required with *Automatic* ($M = 32.98 \pm 18.76$) than with *Manual* ($M = 28.96 \pm 17.95$, $p = 0.031$) for the *Overlap* layout.

For the IPQ questionnaire (see Figure 13), a Friedman test revealed a significant effect of TECHNIQUE ($\chi^2(2) = 19.63$, $p < 0.0001$). Wilcoxon Signed Rank tests reported a significantly better presence with *Manual* ($M = 3.89 \pm 0.47$, $p = 0.0025$) and *Automatic* ($M = 4.80 \pm 0.53$, $p = 0.0025$) than with *Basic* ($M = 4.75 \pm 0.69$).

Finally, 11 out of 12 participants preferred *Manual* and *Automatic* for both tasks over the *Basic* condition. For the *No-overlap* layout, 6 out of 12 participants ranked *Automatic* as their favorite, and 5 participants preferred *Manual*. For the *Overlap* layout, 6 out of 12 participants preferred *Manual*, and 5 participants preferred *Automatic*.

5.8 Discussion

The results provide evidence that the *Manual* and *Automatic* techniques outperformed the *Basic* teleportation. In particular, participants completed the task significantly faster when they were able to choose the position of the future virtual workspace, compared to the *Basic* teleportation. This supports **H1**. It can be explained by the fact that the participants could reach multiple virtual objects easily with physical walking, avoiding unnecessary teleportations. It is
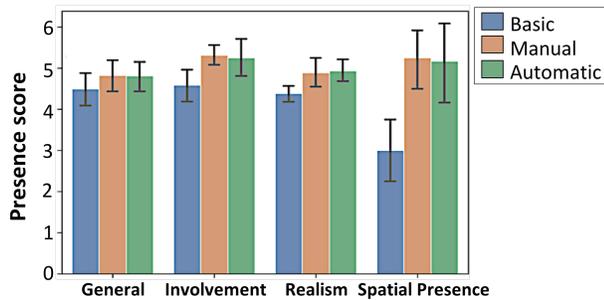
Figure 13: Results from the IPQ questionnaire by TECH-NIQUE × LAYOUT. Error bars show 95% CI.

also confirmed by the significantly smaller number of teleportations executed with *Manual* and *Automatic* compared to *Basic*.

In addition to the better performance, the *Manual* and *Automatic* techniques also resulted in higher sense of presence compared to the *Basic* teleportation, according to the IPQ questionnaire. This supports **H2**. With *Basic* teleportation, since the user "cannot imagine the accessible area" (P3) and "cannot determine if an object is accessible" (P7), a larger number of warnings were triggered while performing the task, compared to *Manual* and *Automatic*. Excessive warnings often lead the user to distrust the VR system and break the immersion. For example, participants "feel fear" (P4) and were "afraid to move" (P5) with *Basic* teleportation. The fact that the user performs a more significant number of teleportations and walks less with the *Basic* teleportation is also detrimental to their immersion.

For the task with the *No-overlap* layout, no significant differences in *TCT* were found between *Manual* and *Automatic*, which does not support **H3**. However, significantly more warnings were detected with *Manual* condition than with *Automatic*. The *Manual* technique requires the user to position the virtual workspace manually, and they may sometimes make mistakes, e.g., not including all the target objects or putting the objects too close to the real workspace's limits. As a result, they trigger warnings when they try to access these objects. But the user seems to be able to quickly reposition the virtual workspace when facing issues with the *Manual* technique, which explains the non-significant differences for *TCT*. Consequently, both techniques are suitable in a virtual environment with no-overlapping interaction areas. However, the *Automatic* technique seems more appropriate in such a context since fewer warnings are triggered, thus avoiding immersion breaks.

For the task with the *Overlap* layout, no significant differences in *TCT* were found between *Manual* and *Automatic*, which does not support **H4**. However, while the task was significantly faster to achieve with *Manual* compared to *Basic*, similar results were not reported for *Automatic*, suggesting that a small difference could exist between *Manual* and *Automatic*. In addition, the score from the NASA-TLX shown that *Manual* significantly reduced the cognitive load, compared to *Automatic*. As the virtual workspaces proposed by the system can be numerous and overlapped each other, the user sometimes has to pass through an intermediate virtual workspace to reach the one behind, which can be time consuming and increase the cognitive load. Users also felt "constrained" (P2, P7) as they needed "to adapt to a previously defined position" (P5). Consequently, the

Manual technique seems more appropriate than the *Automatic* one when the VE is crowded with many objects in the same area.

6 CONCLUSION

In this paper, we proposed and evaluated several techniques helping a user to be aware of their future virtual workspace and manage its position and orientation before the teleportation. Such techniques are interesting to facilitate access to multiple virtual objects through the user's physical movements in their real workspace without reaching its limits. To this aim, we investigated manual and automatic techniques for positioning this virtual workspace.

A first experiment focused on manual positioning techniques. It demonstrates that using an avatar to represent the user's future position in the virtual workspace reduces disorientation and thus the time needed to locate targeted objects after teleportation. It also shows that exocentric and egocentric techniques with an avatar result in a close performance levels, but the egocentric technique seems to be preferred by users. A second experiment compared the egocentric manual technique with an avatar and an automatic technique to a basic teleportation. The manual and automatic positioning techniques outperform the basic teleportation in terms of efficiency and immersion. Although these two positioning techniques reach equivalent performances, each one seems to have its advantages depending on the layout of the virtual environment. Compared to the manual technique, the automatic one causes fewer collisions with the real workspace limits in sparse virtual object layouts, but it induces a higher cognitive load for crowded scenes. In conclusion, this study demonstrates the benefits of the virtual workspace positioning approaches over the basic teleportation.

The future work consists of enhancing the automatic technique by adapting the clustering algorithm to obtain a good balance between the number of proposed virtual workspaces according to the virtual scenario and the real workspace configuration. Moreover, further investigations are required to evaluate these techniques considering different shapes and sizes of real workspaces, the density of the interactive objects in the virtual environment, and the needs of the VR applications. In addition, it remains unclear how the techniques would perform in other VR scenarios or more cognitively challenging situations, for example, in an exploration task where the main goal is not purely to interact with objects. Depending on user expectations, skills and experiences, the virtual workspace's visual feedback could be automatically adjusted based on the amount of the information contained in the scene [22] to avoid overloading the user's field of view and to fit virtual scenario needs. Finally, the suitability of automatic techniques could be studied for specific scenarios that require a perfect match between the real and virtual environments, for example, to provide tangibility to virtual objects [16, 40, 46] or in collaborative co-located applications.

REFERENCES

[1] 2017. Tales of escape. https://store.steampowered.com/app/587860/Tales_of_Escape/. [Online; accessed].

[2] 2017. We were here. https://store.steampowered.com/app/582500/We_Were_Here/. [Online; accessed].

[3] 2018. Essential introduction to Steam VR and Unity part 2: Teleporting and navigating around your scene. https://www.youtube.com/watch?v=GozB7zlD1wQ. [Online; accessed].

[4] 2018. Self Guided VR Training. https://pixovr.com/vr-training-center/. [Online; accessed].

[5] 2020. HalfLife Alyx. https://store.steampowered.com/app/546560/HalfLife_Alyx/. [Online; accessed].

[6] Mahdi Azmandian, Timofey Grechkin, Mark T Bolas, and Evan A Suma. 2015. Physical Space Requirements for Redirected Walking: How Size and Shape Affect Performance.. In *ICAT-EGVE*. 93–100.

[7] Niels H Bakker, Peter O Passenier, and Peter J Werkhoven. 2003. Effects of head-slaved navigation and the use of teleports on spatial orientation in virtual environments. *Human factors* 45, 1 (2003), 160–169.

[8] Woodrow Barfield, Craig Rosenberg, and Thomass A. Furness III. 1995. Situation awareness as a function of frame of reference, computer-graphics eyepoint elevation, and geometric field of view. *The International Journal of Aviation Psychology* 5, 3, 233–256.

[9] Jiwan Bhandari, Paul MacNeilage, and Eelke Folmer. 2018. Teleportation without spatial disorientation using optical flow cues. In *Proceedings of Graphics Interface*, Vol. 2018.

[10] Benjamin Bolte, Frank Steinicke, and Gerd Bruder. 2011. The jumper metaphor: an effective navigation technique for immersive display setups. In *Proceedings of Virtual Reality International Conference*.

[11] Patrick Bourdot and Damien Touraine. 2002. Polyvalent Display Framework to Control Virtual Navigations by 6DOF Tracking. In *Proceedings of IEEE Virtual Reality Conference, VR '02*. IEEE Computer Society, 277–278.

[12] Doug A Bowman, David Koller, and Larry 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*. IEEE, 45–52.

[13] 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*. ACM, 205–216.

[14] Jeff Butterworth. 1992. 3DM: a three-dimensional modeler using a head-mounted display. (1992).

[15] Weiya Chen, Nicolas Ladeveze, Céline Clavel, and Patrick Bourdot. 2016. Refined experiment of the altered human joystick for user cohabitation in multi-stereocopic immersive CVEs. In *IEEE International Workshop on Collaborative Virtual Environments, 3DCVE@IEEEVR*. IEEE Computer Society, 1–8.

[16] Lung-Pan Cheng, Li Chang, Sebastian Marwecki, and Patrick Baudisch. 2018. iTurk: Turning Passive Haptics into Active Haptics by Making Users Reconfigure Props in Virtual Reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 89.

[17] Lung-Pan Cheng, Eyal Ofek, Christian Holz, and Andrew D Wilson. 2019. VRoamer: Generating On-The-Fly VR Experiences While Walking inside Large, Unknown Real-World Building Environments. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, 359–366.

[18] Alexandros Koilias Christos Mousas, Dominic Kao and Banafsheh Rekabdar. 2020. Real and Virtual Environment Mismatching Induces Arousal and Alters. In *Virtual Reality Conference, 2020. VR 2020. IEEE*. IEEE.

[19] Gabriel Cirio, Maud Marchal, Tony Regia-Corte, and Anatole Lécuyer. 2009. The magic barrier tape: a novel metaphor for infinite navigation in virtual worlds with a restricted walking workspace. In *Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology*. ACM, 155–162.

[20] Cédric Fleury, Alain Chauffaut, Thierry Duval, Valérie Gouranton, and Bruno Arnaldi. 2010. A generic model for embedding users' physical workspaces into multi-scale collaborative virtual environments. In *ICAT 2010 (20th International Conference on Artificial Reality and Telexistence)*.

[21] Sebastian Freitag, Dominik Rausch, and Torsten Kuhlen. 2014. Reorientation in virtual environments using interactive portals. In *2014 IEEE Symposium on 3D User Interfaces (3DUI)*. IEEE, 119–122.

[22] Sebastian Freitag, Benjamin Weyers, Andrea Bönsch, and Torsten W Kuhlen. 2015. Comparison and Evaluation of Viewpoint Quality Estimation Algorithms for Immersive Virtual Environments. *ICAT-EGVE* 15 (2015), 53–60.

[23] Markus Funk, Florian Müller, Marco Fendrich, Megan Shene, Moritz Kolvenbach, Niclas Dobbertin, Sebastian Günther, and Max Mühlhäuser. 2019. Assessing the Accuracy of Point & Teleport Locomotion with Orientation Indication for Virtual Reality using Curved Trajectories. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–12.

[24] Jeffrey Goldsmith and John Salmon. 1987. Automatic creation of object hierarchies for ray tracing. *IEEE Computer Graphics and Applications* 7, 5 (1987), 14–20.

[25] Nathan Navarro Griffin and Eelke Folmer. 2019. Out-of-body locomotion: Vectionless navigation with a continuous avatar representation. In *25th ACM Symposium on Virtual Reality Software and Technology*. 1–8.

[26] MP 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)*. IEEE, 371–378.

[27] Sandra G. Hart and Lowell E. Staveland. 1988. Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In *Human Mental Workload*, Peter A. Hancock and Najmedin Meshkati (Eds.). Advances in Psychology, Vol. 52. North-Holland, 139 – 183.

[28] K Krishna and M Narasimha Murty. 1999. Genetic K-means algorithm. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 29, 3 (1999), 433–439.

[29] James Liu, Hirav Parekh, Majed Al-Zayer, and Eelke Folmer. 2018. Increasing walking in VR using redirected teleportation. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*. 521–529.

[30] Jack M Loomis, José A Da Silva, Naofumi Fujita, and Sergio S Fukusima. 1992. Visual space perception and visually directed action. *Journal of Experimental Psychology: Human Perception and Performance* 18, 4 (1992), 906.

[31] John Finley Lucas. 2005. *Design and evaluation of 3D multiple object selection techniques*. Ph.D. Dissertation. Virginia Tech.

[32] Suzanne P McKee and Harvey S Smallman. 1998. 14 Size and speed constancy. *Perceptual constancy: Why things look as they do* (1998), 373.

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

[34] Ji-Young Oh, Wolfgang Stuerzlinger, and Darius Dadgari. 2006. Group selection techniques for efficient 3D modeling. In *3D User Interfaces (3DUI'06)*. IEEE, 95–102.

[35] Sharif Razzaque, Zachariah Kohn, and Mary C. Whitton. 2001. Redirected Walking. In *Eurographics 2001 - Short Presentations*. Eurographics Association.

[36] Holger Regenbrecht and Thomas Schubert. 2002. Real and illusory interactions enhance presence in virtual environments. *Presence: Teleoperators & Virtual Environments* 11, 4 (2002), 425–434.

[37] Judy Robertson and Maurits Kaptein. 2016. *Modern Statistical Methods for HCI*. Springer.

[38] Thomas W Schubert. 2003. The sense of presence in virtual environments: A three-component scale measuring spatial presence, involvement, and realness. *Zeitschrift für Medienpsychologie* 15, 2 (2003), 69–71.

[39] Adalberto L Simeone, Ifigeneia Mavridou, and Wendy Powell. 2017. Altering user movement behaviour in virtual environments. *IEEE transactions on visualization and computer graphics* 23, 4 (2017), 1312–1321.

[40] Adalberto L Simeone, Eduardo Velloso, and Hans Gellersen. 2015. Substitutional reality: Using the physical environment to design virtual reality experiences. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 3307–3316.

[41] Misha Sra, Sergio Garrido-Jurado, Chris Schmandt, and Pattie Maes. 2016. Procedurally generated virtual reality from 3D reconstructed physical space. In *Proceedings of the 22nd ACM Conference on Virtual Reality Software and Technology*. 191–200.

[42] Rasmus Stenholt. 2012. Efficient selection of multiple objects on a large scale. In *Proceedings of the 18th ACM symposium on Virtual reality software and technology*. 105–112.

[43] Wolfgang Stuerzlinger and Graham Smith. 2002. Efficient manipulation of object groups in virtual environments. In *Proceedings IEEE Virtual Reality 2002*. IEEE, 251–258.

[44] Evan A Suma, Zachary Lipps, Samantha Finkelstein, David M Krum, and Mark Bolas. 2012. Impossible spaces: Maximizing natural walking in virtual environments with self-overlapping architecture. *IEEE Transactions on Visualization and Computer Graphics* 18, 4 (2012), 555–564.

[45] Qi Sun, Li-Yi Wei, and Arie Kaufman. 2016. Mapping virtual and physical reality. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–12.

[46] Keisuke Suzuki, Sohei Wakisaka, and Naotaka Fujii. 2012. Substitutional reality system: a novel experimental platform for experiencing alternative reality. *Scientific reports* 2 (2012), 459.

[47] Bernd Froehlich Tim Weissker, Alexander Kulik. 2019. Multi-Ray Jumping: Comprehensible Group Navigation for Collocated Users in Immersive Virtual Reality. IEEE.

[48] Tim Weißker, André Kunert, Bernd Frohlich, 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.

[49] Yiran Zhang, Nicolas Ladeveze, Cédric Fleury, and Patrick Bourdot. 2019. Switch Techniques to Recover Spatial Consistency Between Virtual and Real World for Navigation with Teleportation. In *International Conference on Virtual Reality and Augmented Reality*. Springer, 3–23.