Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

A survey of real locomotion techniques for immersive virtual reality applications on head-mounted displays

Jorge C. S. Cardoso[a,b,*], André Perrotta[b]

[a] *CISUC, Department of Informatics Engineering, University of Coimbra, Coimbra, Portugal*
[b] *CITAR, School of Arts, Portuguese Catholic University, Porto, Portugal*

ARTICLE INFO

Article history:
Received September 20, 2019

Keywords: Virtual Reality, Locomotion, Interaction Techniques, Survey

ABSTRACT

Locomotion is a fundamental activity in Virtual Reality (VR) and has been the focus of a large body of research since the implementation of the first systems. In real locomotion techniques, users physically move in the real world to affect movement in the Virtual Environment (VE). Real locomotion has been found to perform better than other forms of locomotion for many tasks. To overcome the challenges imposed by restricted physical space, researchers have devised ingenious interaction techniques for real locomotion. In this paper, we present a survey of interaction techniques for real locomotion in VR. Our analysis is focused on the interaction techniques – the combination of devices, user's actions, and system's responses. We categorize interaction techniques for real locomotion in VR into: unmediated, warnings, reorientation/resetting, scaling, redirection, and dynamic VE. These categories represent fundamentally different approaches to real locomotion and user action feedback. We further characterize techniques in each of these categories according to category-specific parameters. Finally, it is important to state that this paper was developed with the aim of helping newcomers to the field to understand and implement the techniques here presented.

1. Introduction

Locomotion is one of the most fundamental interaction tasks that users need to perform in VR systems. Consequently, user experience in a VE where the possibility of moving the perspective from one place to another is deprived, is likely to be considered incomplete, unintuitive and uninteresting. Given that locomotion is such a basic task, several interaction techniques have been developed and evaluated over the years. Some techniques allow users to naturally walk through the physical space as they move through the VE, others require users to mimic the movements of real walking while staying in the same physical place, others yet require users to use pointing devices to indicate the direction of movement, and other techniques even use brain-machine interfaces that allow users to think about movement in order to control the perspective of the VE.

Locomotion techniques have been developed for various kinds of VR systems, from those in which the user is fully immersed by putting on a Head-Mounted Display (HMD), to CAVE^{TM}-based VR systems that use projected wall screens. Furthermore, different interaction techniques for locomotion have different characteristics and constraints: they require the use of different types of input devices, require more or less training, have different requirements for the minimum physical interaction area, etc.

In real locomotion users have to physically move from one point to another (translational movement) in the real world to affect movement in the virtual environment. Real locomotion is generally accomplished through real walking, but vehicles

*Corresponding author.
e-mail: `jorgecardoso@dei.uc.pt` (Jorge C. S. Cardoso)

can also be used. Real locomotion in VR systems produces the same proprioceptive, vestibular, cues as walking in the real world. Consequently, in comparison with other forms of locomotion, real locomotion is associated with benefits for memory and cognition [1, 2], better feeling of presence and immersion [3], and superior performance on search tasks [4].

In this work, we present a survey of real locomotion interaction techniques for HMD-based VR systems. We focus on HMD systems due to the recent advancements on consumer HMD-based VR systems and consumer interaction devices that have fuelled the development of various interaction techniques. Most notably, the Oculus Rift HMD and associated development kit, have been associated with a new revival era for VR [5].

Although various surveys and taxonomies relating to locomotion and locomotion metaphors in VR have been published (e.g., Nilsson et al. [6], Boletsis [5], Anthes et al. [7], Jankowski and Hachet [8]), they generally provide only a high-level description of the interaction. Developers seeking to incorporate a given interaction technique into their system need specific details about the required devices, sensed user actions, and system's feedback to those actions in terms of audio, visual, or haptic responses.

In contrast to prior surveys, this survey is focused exclusively on real locomotion and provides more in depth coverage necessary for VR designers and developers. By focusing exclusively on real locomotion, we are able to provide more detailed descriptions and a characterization of the interaction techniques in terms of necessary input actions, devices, and system feedback. The current survey allows VR systems designers and developers to make more pertinent and conscious decisions when choosing the locomotion strategy that best suits a specific system.

More specifically, the contributions of this work are:

- A description of interaction techniques that cover the breadth of real locomotion techniques suitable for HMD-based VR systems.
- A categorization of these techniques, and a decomposition of the main characteristics of each technique.
- A mapping of the field of real locomotion techniques with examples of publications for the various implementations of locomotion techniques and associated variations.

1.1. Delimitation and definitions

This work is focused on locomotion techniques for HMD-based, egocentric, VR systems. Being HMD-based means that the VR world is experienced through the use of non-see-through headsets. Techniques that are only applicable, for example, on CAVE™-based VR systems, are not covered in this survey. We do, however, discuss techniques that have been created for CAVE™-based systems when these techniques have an obvious application to HMDs. As a consequence of our focus on HMD-based systems, we discuss only locomotion techniques for egocentric[1] view of the 3-dimensional environment.

Here, the term "locomotion" is used as a synonym for "travel" as defined by Bowman:

> "Travel, or Viewpoint Motion Control, is one of the most basic and universal interactions found in virtual environment applications. We define travel as the control of the users viewpoint motion in the three-dimensional environment." – Bowman [9, p. 26]

In accordance with Razzaque's [10] perspective, we prefer the term locomotion because the word "travel" is usually associated with long distances, which is not the case for VR in many situations.

Interaction technique refers to the combination of physical devices, the physical or mental actions one must perform while using those devices, and the system's response – be it physical or purely digital – including the feedback necessary for the user to understand the result of his actions. For example, one could devise a simple locomotion interaction technique to allow users to use portals in a VE. The technique would make use of a hand-held joystick controller that allowed users to select a portal by pushing the joystick to the left or right. This would trigger a Heads-Up Display (HUD) with a carousel of thumbnails depicting the various available portals. Once users chose their destination, they would press the joystick button and the selected portal would instantly appear in front of them, at which point they could walk through. This example technique makes use of joystick device which users can manipulate with a single hand and finger (e.g. thumb) to perform two functions: browsing and selecting. The system reacts in two different ways to the user's actions: by showing a HUD with a carousel of images and by placing a portal in the VE once users select one. A different interaction technique could be devised by slightly changing the required physical actions and feedback: instead of using a joystick, users could perform a pre-defined gesture with their arms and hands (detected, e.g., by a Kinect sensor) to instruct the system to place a portal in the VE, a different gesture would instruct the system to replace the portal with a another one in a pre-defined sequence. Once users were happy with the existing portal, they could simply walk through it. While both techniques allow users to use portals they imply very different experiences. To fully describe an interaction technique, we need to know not only what physical or virtual devices are necessary, but also how they are actuated and how the system responds visually, auditorily, tactilely, etc.

2. Related work

Making sense of the myriad of interaction possibilities that have been implemented for real locomotion in VR systems is a complex and substantial task. Other authors have made significant contributions to this topic by compiling surveys, describing interaction metaphors and creating taxonomies of interaction techniques and devices. Our work is guided by the contribution of several authors whose work is presented in the following subsections.

[1] Egocentric (or first-person) perspective refers to a VE where the perspective is a simulation of the user's vision of the virtual world. The virtual camera represents the user's eyes.

2.1. Metaphors

Metaphors are an intrinsic part of the realm of Human-Computer Interaction (HCI) and are unequivocally successful in their ability to introduce concepts and techniques that link the real and virtual worlds. In the context of VR systems, metaphors are important for providing users with an initial mental model of how the interaction works. Particularly for novice users, metaphors facilitate understanding what actions may be performed in the virtual world and how they can be triggered by actions in the real world.

However important, the use of metaphors might also present complications and difficulties when trying to introduce new interaction concepts and techniques that can not be compared to a real world phenomenon. Moreover, metaphors can also be culturally and even age biased, which in turn adds a whole new layer of problems and complexity [11, p. 170].

Ware and Osborne [12], described and compared three interaction metaphors for viewpoint manipulation in 3D interfaces: *eyeball in hand*, *scene in hand*, and *flying vehicle control*. The three metaphors assume a 6 Degrees of Freedom (DOF) input device: 3 for positional placement, and 3 for rotational placement. In the *eyeball in hand* metaphor, the user imagines the virtual world is an invisible model in the room, and that (s)he has a camera in her/his hand that (s)he can move and rotate in this invisible model. The view of the camera is mapped in the screen where the virtual world is visualized. In the *scene in hand* metaphor, the user imagines that the virtual world is in her/his hand. If the user twists the controller clockwise, the scene is rotated clockwise. In the *flying vehicle control* metaphor, the user imagines (s)he is in a vehicle from which (s)he can see the virtual world, controlling its translational and rotational speeds with the controller.

Arns [13] added *leaning* to the list of metaphors. In the *leaning* metaphor, users lean to the direction they wish to travel. In absolute leaning, the user returns to the original position once (s)he straightens back up; in relative leaning, (s)he stays at the current position when returning to the normal standing position. In relative leaning, the user acts as a kind of joystick for controlling the velocity of the movement. The *leaning* metaphor could arguably overlap with the *flying vehicle control* metaphor: one could imagine a vehicle in which the user stands and leans in the direction (s)he wishes the vehicle to move.

In the list of travel metaphors for 3D environments, others – e.g., De Boeck et al. [14] and Arns [13] – include *teleportations*, *world in miniature*, etc.

2.2. Taxonomies

Taxonomies are fundamental for our understanding of the interaction with VR worlds. By explicitly decomposing the interaction technique into various components and stating (even if not exhaustively) the range of values that each component may take, a taxonomy provides a kind of check-list that developers can follow to make sure that all the relevant aspects of the interaction design is well thought and resolved. They also allow us to easily compare different approaches and evaluate their effectiveness for specific applications. As an abstraction, taxonomies share the power of every abstraction: by ignoring unnecessary details they allow us to think in higher levels and manage more easily what would otherwise be too complex to handle. However, when implementing a specific interaction technique, particularly when we are focused in creating a good user experience, at some point we need to handle the details of how the interaction will be implemented (e.g., visual, audio feedback, timings, thresholds, etc.). This requires knowing more details about the interaction technique than the ones provided by a taxonomy.

Bowman et al. [15] proposed a taxonomy of immersive travel techniques which subdivides each technique into three components: *Direction/Target Selection* (e.g., gaze-directed steering, pointing/gesture steering, discrete selection, 2D pointing), *Velocity/Acceleration Selection* (e.g., constant velocity/acceleration, gesture-based, explicit selection, user/environment scaling, automatic/adaptive), and *Input Conditions* (e.g., constant travel/no input, continuous input, start and stop inputs, automatic start or stop). Arns [13] pointed out one limitation regarding the taxonomy by Bowman et al. [15]: the lack of explicit reference to interaction devices. This means that techniques using different devices and hence promoting very different user experiences, could be considered equal under Bowman's taxonomy. Arns [13] thus modified and expanded on Bowman's taxonomy, organizing it around three main components: *Rotation*, *Translation*, and *Interaction Device* (Figure 1). The *Rotation* component defines how one orients her/himself in the virtual world, in relation to the physical world. Usually, when moving through the real world, we rotate our body in the direction we wish to move. Even though it is possible to move without rotating (e.g. moving sideways) this is generally done for short distances. In a virtual environment, we can similarly physically rotate our body in relation to the virtual world. The rotation in the VE, however, does not necessarily need to have a 1-to-1 mapping: a scaling factor can be applied so that a 90° physical rotation corresponds e.g. to a 180° rotation in the VE. Additionally, although in the real world we usually walk in the direction our body is facing, in a virtual system we may choose to track another part of our body (the head is a typical candidate) and move in the direction faced by that part of the body. When immersed in a virtual world, we do not necessarily have to physically rotate our bodies: input devices may be used to rotate the virtual world. This is called *Virtual Rotation* and, in this case, the interaction technique may limit the system with respect to the degrees of freedom it provides. In *Virtual Rotation*, there are many ways to vary the speed/acceleration of rotation: we could have a constant speed set for all rotations, the user might be able to control the rotation speed by means of gestures, by pressing a gas pedal, or by means of other explicit selection mechanism, it could also be adjusted automatically by the system. The last factor in *Virtual Rotation* is the *Input Conditions* which specifies what input actions the user has to make to start, continue, and stop the locomotion rotation.

The *Translation* component is similar to the *Rotation* component, except for the *Physical* sub-category. One method for physical translation corresponds to simple walking (or otherwise moving) in the physical space. Provided the physical space is large enough to accommodate the exploration of the entire virtual world this would be the most effective solution in many cases. Another solution is to scale the user's translation so, e.g., that one meter in the real world corresponds to ten meters in the virtual world. This would allow the exploration of large virtual worlds even in rather small real spaces. The inverse would also be possible to allow precise control over small distances.

The final component is the *Interaction Device*. Different devices may be used to provide a locomotion technique that would otherwise be the same (when classified from just the *Rotation* and *Translation* components). However, different devices may result in very different user experiences (how they are held or worn, how much they weigh, etc., all influence the experience).

Our survey is also motivated by an understanding that devices have important implications on the resulting user experience, but we do not review particular interaction devices for real locomotion in VR. Instead, our focus in the interaction technique as a whole, which implicitly includes the required devices.

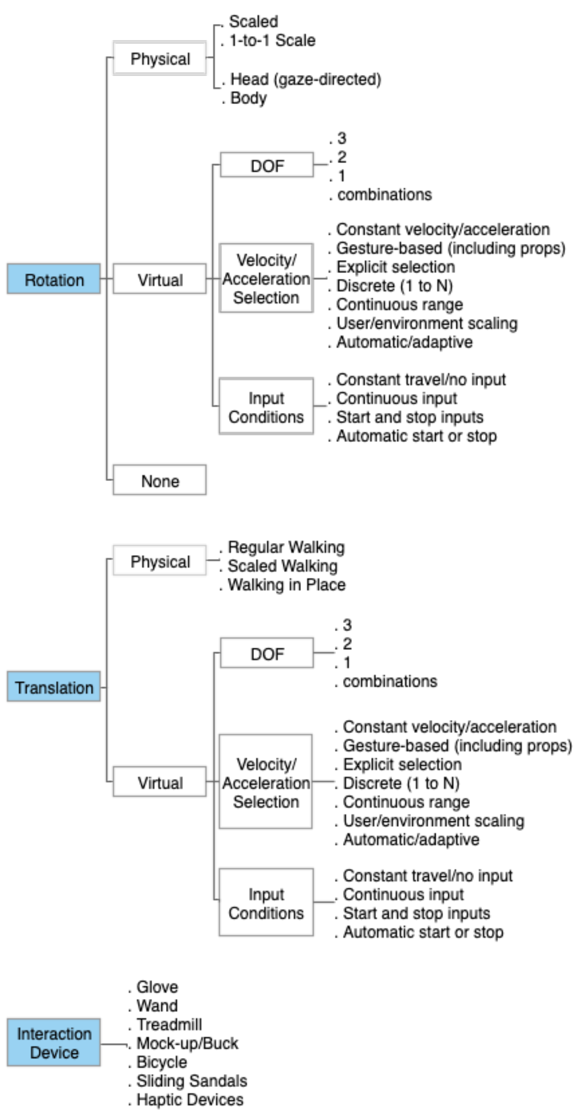


Fig. 1: Taxonomy of locomotion techniques proposed by Arns [13].

2.3. Surveys

Mine [16] described various interaction techniques for virtual environments structured around high-level interaction tasks: Movement, Selection, Manipulation, and Scaling. Movement is the task that allows locomotion within the VE and Mine [16] describes two key parameters that define the user's movement: direction of motion, and speed. Direction of motion is described as taking forms such as "Hand directed", "Gaze directed", "Physical controls", "Virtual controls", "Object driven", "Goal driven". Speed can take forms such as "Constant speed", "Constant acceleration", "Hand controlled", "Physical controls", "Virtual controls". For each parameter value, one or more interaction techniques were presented. The work by Mine [16] presents an interesting survey of interaction techniques, however, it does not address real locomotion and is limited in its account for the system's feedback for the user.

Jankowski and Hachet [8] wrote a comprehensive review of 3D interaction techniques, focused on non-immersive mouse- and touch-based interaction techniques. They specifically address navigation techniques and present several examples categorized in "General movement", "Targeted movement", "Specified coordinate movement", "Specified trajectory movement". Although some of those could be easily adapted, their review is not intended for practitioners of immersive VR, and they do not address real locomotion techniques.

In a recent study, Anthes et al. [7] analyze the state of the art of virtual reality technology, describing and categorizing many different input and output devices for VR. Many of the devices described can and are used for locomotion, among other types of interaction. However, their study focuses on the devices rather than the interaction techniques.

Another relevant work is Boletsis [5] systematic literature review and analysis of recent (2014-2017) empirical studies of HMD-based VR locomotion techniques. Thirty-six publications were analyzed and the interaction techniques were classified based on the following parameters: interaction type (the way in which the user triggers the interaction), motion type (continuous, non-continuous), interaction space (open, limited). As a systematic review, these interaction aspects are mainly used as categories to present an overview of the interaction techniques that have been developed in recent years. Although Boletsis [5] shares our concern with the interactivity aspects of the locomotion techniques, his work is necessarily less detailed in the description of the techniques, given that it covers all types of locomotion in VR. In our work, we aim at providing greater detail into the different characteristics of the various interaction techniques by focusing only on real locomotion. Additionally, we have not limited the analysis to recent years.

3. Real locomotion interaction techniques

Interaction technique refers to the combination of physical devices, the physical or mental actions one must perform while using those devices, and the system's response – be it physical or purely digital – including the feedback necessary for the user to understand the result of his actions.

Real locomotion interaction techniques require a positional tracking system with 6 DOF. In the examples discussed below, numerous tracking systems have been used, from generic optical or magnetic tracking systems to VR-specific systems such as Valve's Lighthouse or Oculus' Room Scale systems. Many of the tracking systems require instrumenting the tracking areas with sensors or laser emitters but more recent systems such as the Oculus Quest[2] are self contained. This kind of inside-out tracking can now even be used in smartphone-based VR [17]. The advancement of tracking systems is one more reason why real locomotion techniques are relevant – current tracking systems are now easier to deploy in various kinds of locations and can be set up rather quickly.

Although tracking systems may have an important impact in the user experience (given the possible additional devices to wear, cables, etc.), we mostly ignore them in the discussion below. We also ignore the system's update of the 3D perspective in response to head movements given that it is necessary for the VR experience itself and not particular to locomotion interaction. The following discussion focuses instead in the required user actions and system's response to the user's actions. This last property of the interaction technique is what distinguishes our categorization of real locomotion interaction techniques.

3.1. Unmediated

Unmediated locomotion is defined by the absence of any system intervention in the locomotion process. In unmediated locomotion, the physical position/orientation is directly mapped to a virtual position/orientation. The behavior of the system does not change in response to physical movement. This means that the system has no knowledge of the physical space and cannot enforce any restrictions on where users (try to) go. The only limitations that are imposed to the user are the physical layout of the space and the size and constraints of the tracking area. Unmediated systems can be dangerous to the user, thus, these are usually applied in controlled conditions, in which a (human) supervisor can warn and direct users to safety or immediately stop the experience altogether.

Most unmediated locomotion examples have very similar setups: they are mostly used in experiments concerned with locomotion itself or perception in immersive VR; the VE is built according to the available physical space; they use indoor medium/wide area trackers to track the user's HMD (usually with 6 DOF).

For example, in their classical study, Usoh et al. [3] used a $10\,x\,4$ m^2 physical area:

> "Although we are using wide-area tracking, the virtual scene still must fit into a finite area. We therefore divided the tracked space into a training area and an experimental area, each of 5 x 4 meters."

In any real locomotion technique, safety borders around the physical space are important, but even more so with unmediated real locomotion. The VR system has no knowledge of the physical area, and therefore, it is unable to alert users to possible collisions. In order to cope with this problem, Suma et al. [1] used a $4.3\,x\,4.9$ m^2 physical area and created a virtual environment that fits into the available physical area, leaving a safety border of about 15 cm (Figure 2).

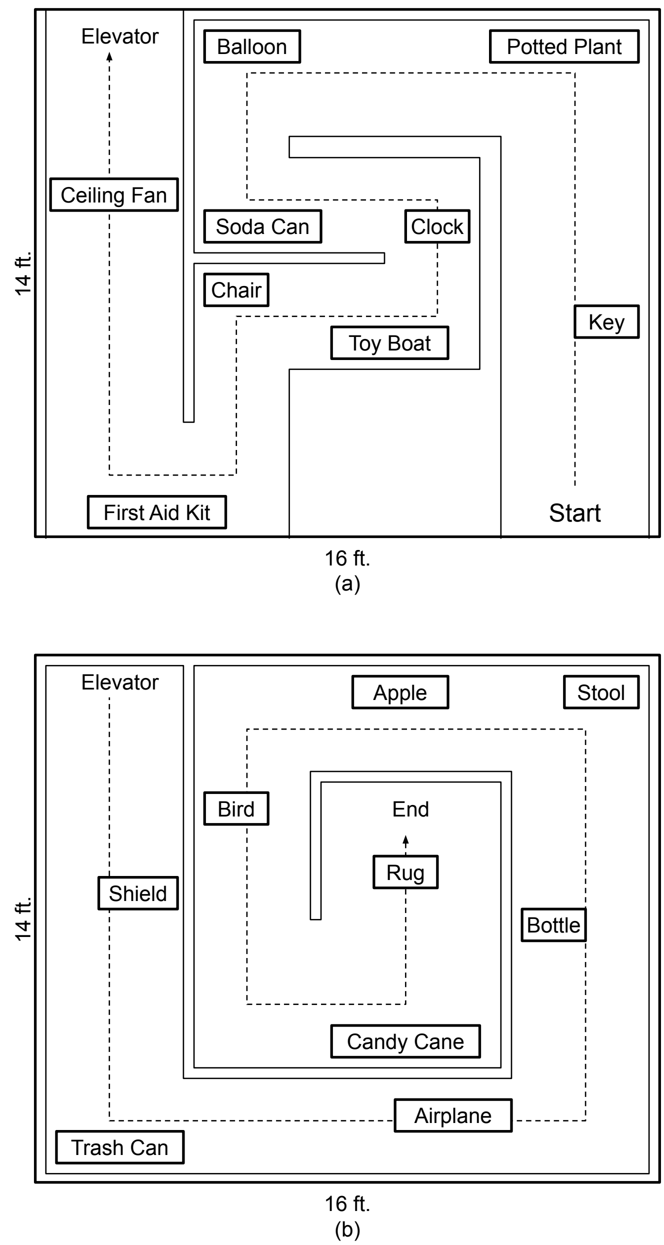


Fig. 2: To evaluate the cognitive effects of locomotion technique, participants were required to move through a maze in Suma et al.'s [1] study. The maze was designed to fit the available physical space, leaving a 15 cm safety border. (Redrawn from Suma et al. [1].)

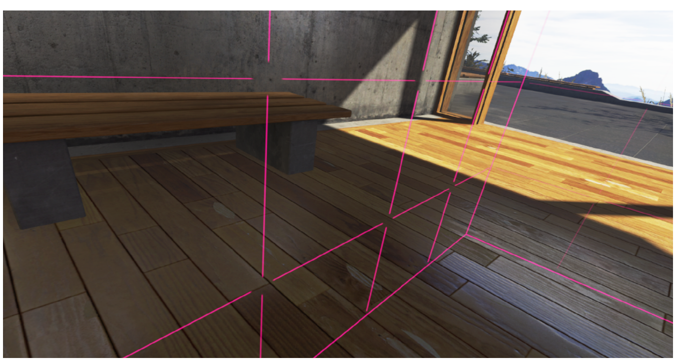
Although unmediated real locomotion is more often used for VR experiments, it can also be used for real-world applications. An excellent example is the work by Jung et al. [18], where a motion-capture studio of $10\,x\,10$ m^2 with a trackable region of $7\,x\,7$ m^2 is "transformed" into a virtual aquarium where users can freely explore the underwater environment.

[2] https://www.oculus.com/quest/

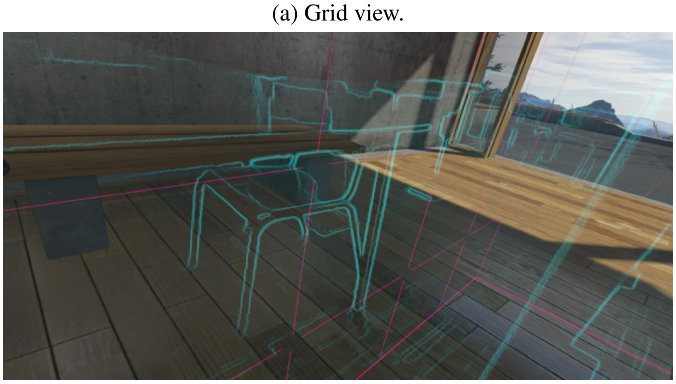
3.2. Warnings

To increase the safety of the VR experience, many real locomotion techniques implement various kinds of warnings to alert the user's proximity to the edges of the physical room or tracking area. Many of these are hybrid systems in that they allow the use of real locomotion within the tracked area, and also allow the use of other kind of virtual locomotion (e.g., using gestures, a wand, or other handheld controllers) at the edges of the tracked area (some allow both kinds of locomotion in any location). Warnings require the VR system to have knowledge about the limits of the physical space and to react to the position of the user (usually when the user approaches the limits of the physical space).

Commercial VR systems such as SteamVR platform, employ a type of warning system, called *Chaperone Bounds*, where a grid shows the limits of the tracked when users get too close to the limit (see Figure 3a). A grid can also be displayed on the floor if users choose so and the floor grid can be turned on permanently. There is currently also an experimental feature where the image (filtered) from the camera installed in the HMD is displayed when the *Chaperone Bounds* are activated (see Figure 3b).



(a) Grid view.



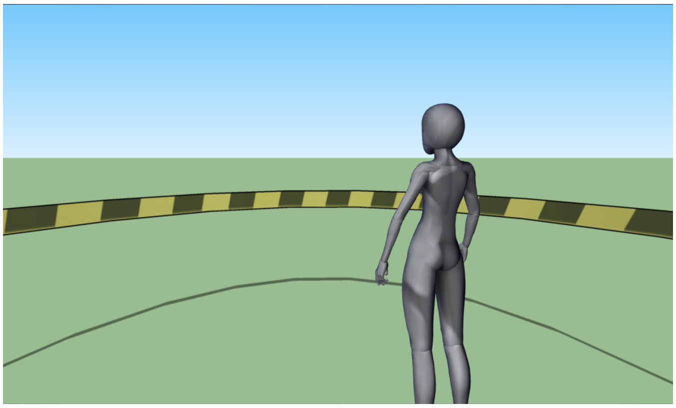
(b) Grid plus camera view.

Fig. 3: Chaperone Bounds on HTC Vive (SteamVR).

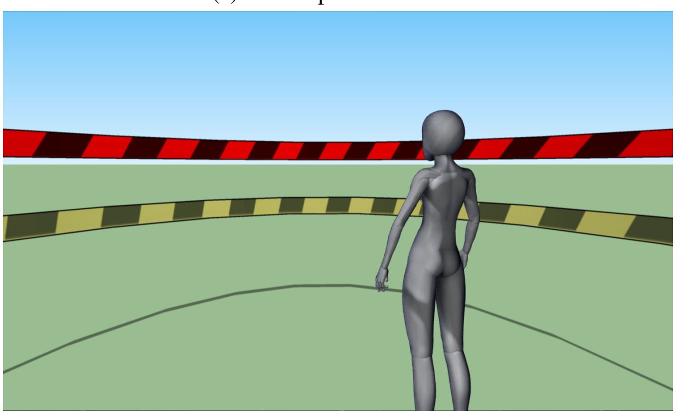
In *SpaceWalk* [19], a tracking station (a Kinect 2 device connected to a standalone computer) tracks the full body of the user in a cone-shaped area (of about 10 m^2). The implemented system defines a danger area on the limits of the tracking area and, as soon as the user enters this area, a visual alert with the message "Go Back! You are leaving the tracking area" is triggered and the whole display is blurred out.

In the *Magic Barrier Tape* [20], a tape is drawn at the contour of the physical area in order to display the limits of the physical area, and to provide a way to navigate to any location in the VE. The tape is composed of three components (Figure 4): a yellow striped tape, positioned at mid-height for constant visibility; a red striped tape, positioned at user's eyes height, that becomes progressively more visible as users approach the edges and acts as a danger signal; the tape's shadow projected onto the floor, providing an additional visual cue of the workspace limits in case users are looking straight down. Inside the workspace area, users' position is tracked so that they can naturally walk and have a fine control of their position in the VE. At the edges of the workspace, however, users can navigate to points beyond the barrier tape by pushing the tape with their hands (or any other tracked body part). When close to the edge of the workspace, the system adopts a rate control behavior: if users push the tape with their hands, the hand penetration distance is used to control the movement velocity. The *Magic Barrier Tape* and the previous examples make use of two zones: a safe zone where users are not near the limits of the physical area, and a danger zone where users are near the limits of the physical area causing the system to trigger additional warnings (Figure 5a). In addition, *Magic Barrier Tape* makes use of an unstated buffer area to allow user movements for interaction purposes: users can virtually locomote by pushing the tape so a buffer area is needed to allow users to extend their arms beyond the tape (i.e., beyond the danger zone).

Cirio et al. [21] describes three locomotion techniques for CAVE environments. Although developed with CAVEs in mind, where there is usually a missing screen in the back of the user, two of these techniques could also be applied to HMD based VR experiences. Both techniques define 3 zones in the space: a safety zone, in which users are not near any screen; a reaction zone, in which the system starts presenting cues to guide the user; and a danger zone, in which there is danger of colliding with a screen (Figure 5b). In the *Constrained Wand and Signs* technique, users can walk naturally inside the safety zone, where the wand device is disabled. When users reach the reaction zone, a semi-transparent "no-way" sign appears and becomes fully opaque when users reach the danger zone. In the reaction zone, the wand becomes active and can be operated to move forward in the direction of the wand. In the *Virtual Companion* technique users have a virtual "companion" – a bird – that warns them about the area limits, and that also serves as a locomotion guide. In the safety zone, users can walk naturally and the bird stays near the closest wall calmly flapping its wings. In the reaction zone, the bird is at the position of the user's head, projected into the nearest wall. In the danger zone, the bird turns red, and flies in front of the user's face angrily flapping its wings. Locomotion beyond the physical area is achieved only in the safety zone by using a set of hand gestures. To initiate virtual locomotion, users bring their hands together for 1 second. This causes a pair of "reins" to appear connecting the users hands to the bird, which move to the front of the user. To move forward users makes up and down gestures. Although a bird was used by Cirio et al. [21], other virtual companions could be used for this locomotion technique.

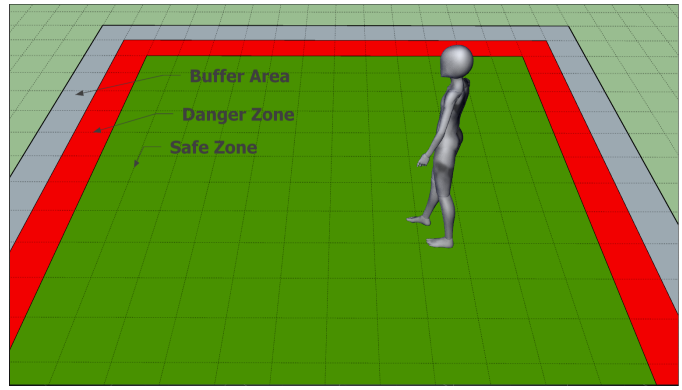


(a) Main tape and shadow.

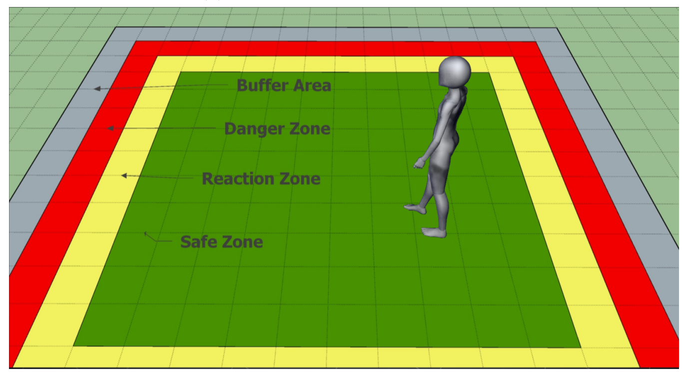


(b) Main tape, shadow, and warning tape.

Fig. 4: Components of the *Magic Barrier Tape*. Main (yellow) tape, slightly transparent; Warning (red) tape; Tape's shadow. (Redrawn from Cirio et al. [20])



(a) Two zones and buffer area.



(b) Three zones and buffer area.

Fig. 5: Zones in the Warnings locomotion technique.

Another hybrid locomotion technique is the *Cloud-Walker*[22]. In this technique users can fly around the VE in the direction pointed to by a wand, and at the same time walk within the limited tracking area. The metaphor is that of a cloud, where the cloud surface represents the area where users can walk, giving them small scale, precise locomotion. The cloud itself can be moved by pointing the wand in the intended direction. Users are warned about the proximity to the edges of the cloud via three feedback mechanisms: a vibrotactile belt with 8 tactors around the waist, auditory feedback for the footsteps with different sounds for the center and for the edges of the cloud, and different physical textures for the floor – the center of the cloud "feels" different than the edges. Additionally, there is a physical barrier around the locomotion area preventing users from exiting the "cloud". The *CloudWalker* technique also features a locomotion feedback: wind is generated by several fans around the physical barrier that are activated to produce a sensation of wind in the direction of the locomotion.

As described in the examples above, warnings can be applied in different ways, and in turn, each strategy can be analyzed by 4 principal parameters:

- Zones – the number of zones used to define safe (inside working area) and danger (edge/outside working area) limits.
- Modality – type of warning feedback: visual, audio, haptic.
- Visibility – the warning is constantly, progressively or instantly visible.
- Ecologicalness – the degree to which the warning fits within the concept of the VE in a natural and intuitive way [21].

Table 1 summarizes the properties of the presented Warnings techniques.

3.3. Reorientation / Resetting

Reorientation or resetting techniques reorient the user to a safe orientation or position when (s)he reaches the boundary of the tracked area. Reorientation can work either by making the user physically rotate so that (s)he faces away from the boundaries of the tracked area, or by making the user physically translate to a safe location (e.g. the center of the tracking area). A good reorientation strategy is implemented in such a way that the user never looses contact with the virtual experience.

Williams et al. [23] proposed and evaluated three reorientation techniques. In all techniques, the system warns the user through a text message in the HMD when (s)he reaches the boundary of the tracked space and starts the reorientation procedure. In the *Freeze-backup* technique, positional tracking stops and physical translational movements relative to the coordinates

Table 1: Summary of Warnings techniques.

Example	Zones	Modality	Visibility	Ecologicalness
Chaperon Bounds (HTC Vive)	2	Visual	- Inside danger zone - Floor can be always on	Low
SpaceWalk [19]	2	Visual	- Inside danger zone	Low
Magic Barrier Tape [20]	2	Visual	- Always (Main tape) - Inside danger zone (Warning tape)	Low/Medium
*Constrained Wand and Sign*s [21]	3	Visual	- Inside reaction and danger zones	Low/Medium
Virtual Companion [21]	3	Visual	- Always	Medium/High
CloudWalker [22]	3	Auditory, Haptic	- Always (haptic floor, footstep sound) - Inside danger zone (haptic belt)	High

of the ground plane of the working area (x and z axis)[3] are not reflected in the VE. Users are instructed to take steps back in the physical space. When enough steps have been taken, the system tells the user to stop and reactivates the positional tracking. While the user is stepping backward (s)he is able to look around the VE as orientation tracking is still in effect. This has the effect of repositioning the user in the middle of the physical area (provided (s)he approached the boundary from the center), or close to the farthest edge but facing away from it. In the *Freeze-turn* technique, the positional tracking stops (as in the *Freeze-backup*) and the yaw angle (rotation about the y angle) is also frozen. Users are instructed to rotate 180° degrees at which point tracking is resumed. In the *2:1-turn* technique, the system instructs users to make a 360° turn in the virtual environment but the rotational gain for the yaw angle is set at 2x so that users only rotate 180° in the physical space.

In the *2:1-turn* technique, the VE rotates while the user is also physically rotating. Other techniques rotate the VE so that afterwards users will naturally have to rotate to be facing the same initial direction. Peck et al. [24] compared a number of variations of reorientation techniques. In the *Turn without instruction* technique, the VE would automatically rotate 180° around the user at 120°/second when (s)he reaches the edge of the physical area. This will relocate the virtual path back to the tracking area and will require the user to rotate 180° to continue on that path. In the *Turn with audio instruction* the system instructs the user, via audio, to turn 360° when the edge of the tracked area is reached. While the user is turning, the system applies a rotational gain of 2 (this is equivalent to the 2:1-turn technique but instructions are provided in audio form). In the *Head turn with audio instruction* the user is asked to turn her/his head back and forth and then continue. While the user is turning her/his head, the system applies a 1.3 rotational gain (in one direction) until the VE is rotated 180° and the user will then have to physically rotate also 180° to continue on her/his path.

One of the challenges for reorientation techniques is to make them a part of the VE so that they don't break the sense of presence of the user. Giving direct instructions to users is something that obviously breaks the experience, hence, researchers have tried to develop solutions that can be seamlessly integrated into a VE. Peck et al. [24] proposed the use of distractors during reorientation. Instead of providing audio instructions, in the *Head turn with visual instruction distractor*, a visual distractor (a red sphere) appears in front of the user moving in an arc and following a sinusoidal displacement. To keep the sphere in view, users rotate their heads back and forth allowing the system to rotate the VE with a 1.5 rotational gain until the VE has rotated 180°(at which point the sphere disappears). Afterwards, users will have to reorient themselves by rotating 180°. Peck et al. [24] have also evaluated different distractors with increasing level of detail and different modalities: *improved distractor* – a butterfly that flies in and out (instead of appearing or disappearing suddenly); *distractor visual* – a hummingbird with realistic textures; *distractor, visual and audio* – a hummingbird with spatialized wings flapping sounds; *distractor audio* – an exclusively audible distractor made of the hummingbird's spatialized wings flapping sounds. They concluded that users found more natural and, thus, preferred the distractors that presented more realistic audiovisual cues.

This conclusion raises an interesting issue regarding distractors. They are meant to help maintain the user's illusion of being immersed in the virtual world, and as such, should be as ecological as possible. However, a distractor's main function is to distract users and make them follow the distractor's movement, which means that they must be noticed and will necessarily cause some break in the user's current task. Additionally, even though Peck et al.'s [24] study concluded that users preferred and found more natural a more realistic audiovisual distractor, it may be more important that the distractor's visual and auditory appearance is integrated to the underlying aesthetics of the VE (i.e., a realistic distractor may stand out in a non-realistic VE).

A very good example of a reorientation implemented seamlessly into the VE is that by Yu et al. [25]. In their narrative-driven system, users could walk through 3 straight segments of (virtual) tunnels, connected by 90° turns, used in World War I while physically moving in a very restricted physical space with about 4.5 x 1.5 m^2. To accomplish this, at the end of each segment, and before being able to proceed to the next, users where reoriented so that they would physically rotate 180°, while virtually they rotated the necessary 90° for the turn in the tunnel. To create a seamless experience, reorientations where done only in special zones (the end of the tunnel segment) and in a way that was embedded in the narrative of the VE: users were instructed to look for a light switch to turn on the lights of the new tunnel segment. While users were looking for the switch a rotation gain based on the rotation of the head was applied.

[3] In this work we use x and z to denote horizontal movement in the plane and y for vertical movement that changes elevation.

The reorientation zones were additionally filled with artifacts of historical value to encourage users to look around and an audio clip was played to direct attention to those artifacts.

From the presented examples, it is possible to point out a set of important parameters that distinguish different reorientation techniques:

- Tracking interruption – if and how the tracking system is suspended during the reorientation process.
- User action – explicit actions (such as walking backwards, turning around, moving the head back-and-forth, etc.) needed to be performed by the user in order to the proper reorientation process to take place.
- Instructions format – the method used to deliver explicit reorientation instructions during utilization. This can vary from simple and straightforward text based instructions to implicit symbolic ones.
- Distractor modality – if and how the system uses distractors.
- Additional system's response – automatic procedures performed by the system (such as VE rotation, transform gains, etc.).

Table 2 summarizes the main properties of the Reorientation/Resetting techniques.

3.4. Scaling

Another approach to dealing with the limited space for real locomotion is to change the mapping between the real movement and the virtual movement so that a small physical translation corresponds to a large virtual translation. This allows users to travel large distances in the virtual environment while confined to a relatively small physical area.

Williams et al. [26] evaluated the spatial orientation under different locomotion techniques (joystick and walking) and different translational gains (1:1 and 10:1). Spatial knowledge was tested from different locations in the room by asking participants to close their eyes and face a predefined object after they had locomoted to that location. Results showed that the turning error was larger when participants locomoted with a joystick and when they were subject to a larger translational gain. In another experiment they compared different translational gains (1:1, 2:1, and 10:1) applied to real walking. In this second experiment, no statistically significant difference in turning error was found between the different gains. The authors conjecture that the reason for the statistically significant differences in turning error due to gain in the first experiment might have been caused by error variance or differences across the experimental design (between subjects in the first, and within subjects in the second). In any case, the authors conclude that "manipulating gain is a viable method to assist people in fitting exploration of large virtual environments within the confines of small physical spaces".

Interrante et al. [27] applied the concept of translational scaling in the *Seven League Boots* technique. In this technique, scaling is applied only to the movement component that is aligned with the direction of travel. If scaling is applied to all axis, the small lateral head sways that people make while walking would be amplified, as well as the up and down movements of the head, causing discomfort and possibly simulator sickness. To determine the direction of movement, the authors used a combination of previous movement and gaze directions (integrated over a reasonable period of time). The weight assigned to the gaze is 1 when the magnitude of previous displacements is very small, but quickly falls to 0 when the user starts to move. This has the effect of considering movement direction as the gaze direction when users are standing still, but rapidly switching to considering the previous movement direction when users are walking. Although the authors considered several alternatives for activating the "boots", such as being constantly on, with ease-in and -out to ramp up the scaling when users start or stop moving, or using Artificial Intelligence (AI) to automatically determine when to turn on or off the boots, they considered manual control, activated by the push of a button on a wand, to be the best solution.

In *Arch-Explore*, Bruder et al. [28] use scaling (along with several other techniques) to allow users to explore architectural 3D models. In their implementation, the VE uses the actual physical walls as passive haptic feedback, and all virtual spaces (or rooms) fit to the actual physical space layout. Thus, in situations where the virtual space has a different size (bigger or smaller), translational gains are required in order to create a convincing correspondence between the virtual and real worlds. In this work, translational gains were applied exclusively to the xz-plane with values in the range of [0.78, 1.22] (although the authors considere that gain values up to 2.0 would not be overly disruptive).

Abtahi et al. [29] evaluated a different approach to scaling where the VE itself is scaled down, instead of scaling the user's movement. In the *Ground-level scaling* the scale center is placed at the midpoint between the users feet. This results in an illusion of becoming a giant (Figure 6b). In the *Eye-level scaling* the scale center is placed at the midpoint between the users eyes. This creates the illusion of walking through a miniature world placed at eye-level (Figure 6c). In both techniques the scaling is applied instantaneously and only when users are stationary. Abtahi et al. [29] compared the *Ground-level scaling* and *Eye-level scaling* techniques to a variation of *Seven League Boots* where the boots are only active when users are walking (they used foot tracking to detect when users are walking), and are turned off when users are stationary. Gains of $3x$, $10x$ and $30x$ were compared. They found that *Seven League Boots* was less preferred by users and positional accuracy diminishes at high speed gains. In the *Eye-level scaling* positional accuracy was maintained even at high speed gains.

From the strategies previously mentioned, it is possible to recognize important parameters that distinguishes different scaling techniques:

- Scale axis and origin – the axis and origin of the applied scaling. The decision on the implementation of this parameter directly affects the complexity of the system, hence, applying the scaling factor only to the movement compo-

Table 2: Summary of Reorientation/Resetting techniques.
†It is not clear from [24] if/how tracking was interrupted but they state that their technique is similar to a method described in [23].
‡We assume that auditory instructions were also present even though the techniques are named with "visual instruction".

Example	**Tracking interrupted**	**User action**	**Instructions**	**Distractor/ Modality**	**Additional system's response**
Freeze-backup ([23])	Yes (Positional - x and z axes)	Walk backwards	Text	-	-
Freeze-turn ([23])	Yes (Positional - x and z axes, Rotational - yaw)	Turn around	Text	-	-
2:1-turn ([23])	Yes (Positional - x and z axes)	Turn around	Text	-	Rotational gain (2x)
Turn without instruction ([24])	No	Turn around (Reorient 180°)	Auditory	-	Automatically rotates VE 180°at 120°/second
Turn with audio instruction ([24])	Yes (Positional - x and z axes)†	Turn around	Auditory	-	Rotational gain (2x)
Head turn with audio instruction ([24])	No	Turn head back/forth, then turn around (Reorient 180°)	Auditory	-	Rotational gain (1.3x)
Head turn with visual instruction, distractor ([24])	No	Turn head back/forth (watch distractor), then turn around (Reorient 180°)	Auditory‡	Visual	Rotational gain (1.5x)
Improved distractor ([24])	No	Turn head back/forth (watch distractor), then turn around (Reorients 180°)	Auditory‡	Visual (butterfly)	Rotational gain (1.5x)
Distractor visual ([24])	No	Turn head back/forth (watch distractor), then turn around (Reorient 180°)	Auditory‡	Visual (hummingbird)	Rotational gain (1.5x)
Distractor, visual and audio ([24])	No	Turn head back/forth (watch distractor), then turn around (Reorient 180°)	Auditory‡	Visual (hummingbird) , Auditory (wings flapping sound)	Rotational gain (1.5x)
Distractor, audio ([24])	No	Turn head back/forth (follow sound), then turn around (Reorient 180°)	Auditory‡	Auditory (wings flapping sound)	Rotational gain (1.5x)
Yu et al. [25]	No	Turn head (look for light switch, look at artifacts)	No explicit instruc-tions	-	Rotational gain (1.1x)

nent aligned with the direction of locomotion has the difficulty of requiring the system to predict the direction of locomotion that is intended by the user. The scale origin is also important as it will have implications in the resulting users perspective (*Ground-level scaling* for example, causes a significant vertical shift in the users perspective).

- Scale factor – the range of the used scale factor and if the used values are constant or dynamic. Even though very high scale factors might still result in a controllable experience, lower factors such as the ones proposed in Arch-Explore [28] are probably more sensible to use in most situations. Nevertheless, lower factors may render the technique useless if the environment is much larger than the available space. A large factor such as 10:1 requires environments where objects are also highly spaced.

- Activation method – scaling may be activated manually – by user's explicit control, or automatically – as a response to specific users' action. This decision is specifically important in VEs that require large space exploration and also detailed inspection of objects. Being always on means less control for the user. For example, in a situation where the user is standing still and wants to closely inspect an object, scaling may prevent or hinder this task. However, it means that the system can guarantee that the available physical space is enough for users to explore the full extension of the VE. Activation through an explicit user command, means that users have greater control (for example to turn off scaling when inspecting an object), but it also means that they may choose not to activate scaling and reach the limits of the physical space.

Table 3 summarizes the main properties of the Scaling techniques presented in the previously described examples.

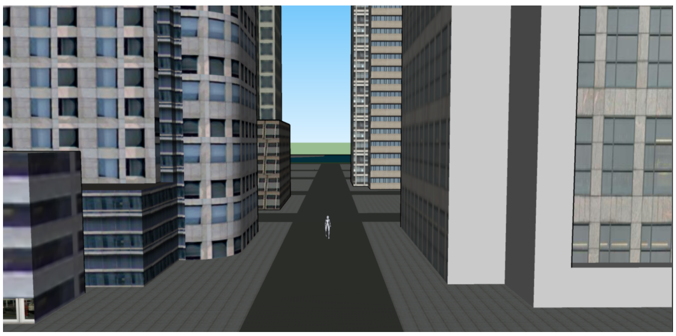
3.5. Redirection

Another common strategy to deal with the limited space for real locomotion is to steer the user away from the boundaries of the tracked area while (s)he is moving through the VE. Razzaque et al. [30] introduced the Redirected Walking (RDW) technique:

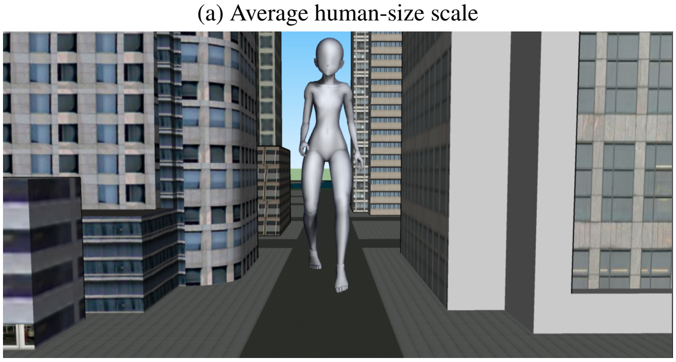
> "Redirected Walking works by interactively rotating the virtual scene about the user, such that the user is made to continually walk towards the farthest 'wall' of the tracker area. The user does not notice this rotation because the algorithm exploits the limitations of

Table 3: Summary of Scaling techniques.

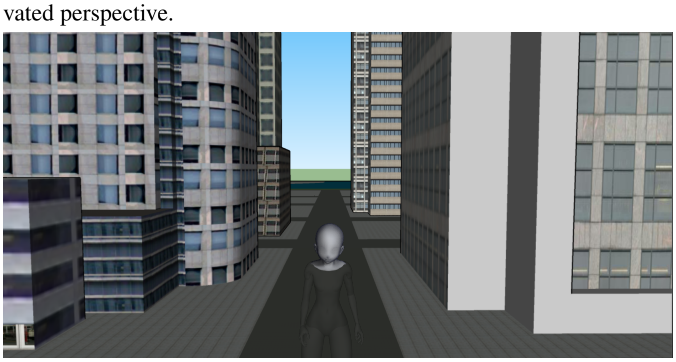
Example	Scale Factor	Scale Axis/Origin	Activation
Williams et al. [26]	Fixed (10:1, 2:1)	(x, y, z)	Always on
Seven League Boots [27]	Fixed (10:1) / Ramp up/down	Direction of movement	Button press
Arch-Explore [28]	Automatic (0.78:1, 1.22:1)	(x, z)	Automatic, as needed
Ground-level scaling [29]	Fixed (3:1, 10:1, 30:1)	Mid-point between user's feet	Manually by experimenter
Eye-level scaling [29]	Fixed (3:1, 10:1, 30:1)	Mid-point between user's eyes	Manually by experimenter
Seven League boots variation [29]	Fixed (3:1, 10:1, 30:1)	Direction of movement	Automatic, when walking



(a) Average human-size scale



(b) Ground-level scaling to achieve a 10x speed gain, resulting in elevated perspective.



(c) Eye-level scaling to achieve a 10x speed gain while maintaining regular eye-level.

Fig. 6: Different scaling centers. (Redrawn from Abtahi et al. [29])

> human perceptual mechanisms for sensing position, orientation and movement. The amount of rotational distortion injected is a function of the user's real orientation and position in the lab, linear velocity, and angular velocity."

Redirection applies small rotations to the scene while the user translates so that, in physical space, users are describing arcs instead of straight lines. This allows the system to continually steer the user to a predetermined physical location (usually the center of the tracked area). When walking in a virtual straight line, users do not notice small artificial rotations introduced in the VE and compensate by turning slightly to keep the virtual path a straight line. The result is that users physically describe an arc, but perceive their virtual path as a straight line. Even if users are standing still, it is also possible to introduce imperceptible rotations. These rotations can be increased while users are turning themselves in the VE. This allows the system to steer the user more quickly than what would be possible in normal circumstances.

The general RDW algorithm is outlined in Figure 7. The algorithm applies three rotational parameters: a baseline rotation applied regardless of movement velocity or users' rotational rate, a rotational gain (scaling) applied to users' rotation rate, and a curvature gain proportional to users' linear velocity. The maximum of these three parameters is taken and scaled by the sine of the angle between the virtual and physical targets. Finally, the result is clamped to make sure the rotational gain is not above the detection threshold of a user. RDW is sometimes referred to as Motion Compression (MC) [31, 32]. MC originated in the field of telepresence for exploring a large remote physical environment through remote control of a mobile robot. It is usually described through a different algorithm, but the end effect and strategies are similar to RDW.

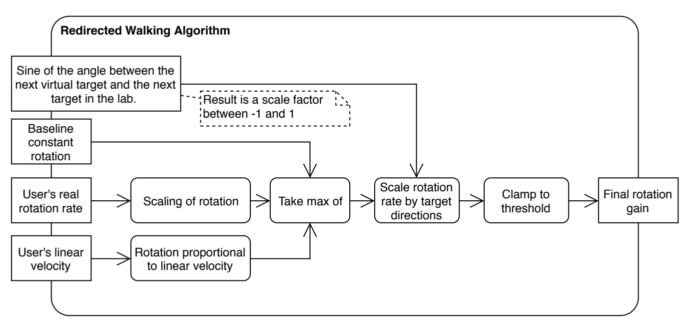


Fig. 7: Redirected walking algorithm. Redrawn from [30].

Applying this algorithm requires the determination of users' detection thresholds (for rotational and curvature gains), the physical location to steer users to, and determining users' intended direction or next virtual target.

Regarding the usage of scaling factors, the RDW algorithm applies three different types of rotation gains: a constant baseline, or time-dependant gain (gT), a rotational gain (gR) applied when users rotate their heads, and curvature gains (gC) applied as users translate through the VE. Time-dependent gains from 0.5 deg/s [33] to 5 deg/s [34] can be found in the literature.

(Translational gains, i.e., Scaling, is also often referred to as a possibility for RDW, however, we have not found any concrete implementation of RDW that also employed translational gains.)

Steinicke et al. [35] have quantified, through a psychophysical experiment with a two-alternative forced-choice task, Detection Thresholds (DTs) for how much humans can be redirected. In an experiment to determine the ability to discriminate whether a physical rotation was smaller or greater than the simulated virtual rotation they found that rotational gains (gR) between 0.67 and 1.24 cannot be reliably estimated by humans. In another experiment to determine the ability to discriminate the direction of bending of the physical path (gC), they found that for curvature gains in the interval [-0.045, +0.045] m^{-1} (which translates to a circular arc with a minimum radius of about 22 m) humans cannot reliably estimate if they are walking straight or in a curve. However, it is common to find examples where both larger and smaller rotational and curvature gains have been applied. In some situations, values are chosen so as not to be overly distracting rather than being imperceptible; in other situations, values are chosen conservatively to guarantee they are not perceptible by users.

Kruse et al. [36] have studied the effect of type of VE (high fidelity, or low fidelity), and visual self-representation (visible virtual feet, or invisible virtual feet) on the sensitivity to translation gains. Their results generally suggest that the type of VE is more important that self-representation and it impacts the detection thresholds. Although they studied translational gains, their results may also be applicable to rotational gains.

An interesting variation on DT is the Threshold of Limited Immersion (TLI), defined as the level above which a manipulation will become disturbing for the VE [37]. This is different from the detection threshold because, for the TLI users are not asked whether they can detect the manipulation, but rather they are asked to tell when a manipulation affects the quality of their experience. Schmitz et al. [37] measured TLI for rotation gains and found that, for decreasing gains the TLI of 0.58 was not significantly different from the DT of 0.67 but 43% of participants reported breaks in immersion before the DT was reached. For increasing gains, however, the TLI of 1.85 was significantly higher than the DT of 1.24.

A complementary approach to manipulate the VE for RDW is to take advantage of blink-induced suppression and saccadic suppression. In these approaches the VE is rotated when users blink [38] or when they perform a rapid eye movement (saccade) [39] taking advantage of the fact that humans are blind during these events.

Neth et al. [40] have measured the effect of walking speed on the detection threshold for curvature gains and found that users are less sensitive to curvature gains when walking at slower speeds. They found that, while walking at 0.75 m/s, the detection threshold was about 0.095 m^{-1} (circular radius of 10.5 m); while walking at 1 m/s, the detection threshold was $0.04m^{-1}$ (radius of $23.8m$); and while walking at 1.25 m/s, the detection threshold was about 0.38 m^{-1} (radius of 27.0 m). This means that an RDW algorithm that takes walking speed into account to calculate curvature gains can be more efficient. They proposed a dynamic gain algorithm where gC is calculated as a function of movement velocity v (Equation 1[4]).

$$gC = \begin{cases} 0.2, & v < 0.75\, m/s \\ -0.2v + 0.35, & 0.75 < v < 1\, m/s \\ -0.04v + 0.09, & 1 < v < 1.25\, m/s \\ 0.13, & v > 1.25\, m/s \end{cases} \tag{1}$$

Although RDW is mostly applied to walking situations, Bruder et al. [41] applied it to locomotion in a wheelchair. Their study was about comparing the perceptual thresholds for rotational, translational and curvature gains. The results suggest that users driving the wheelchair could be more redirected than walking users, but the thresholds for rotation and translation are similar.

Another important aspect to take in consideration when implementing a RDW strategy is the steering target. In one of the first RDW implementations, Razzaque et al. [30] steered users into pre-defined physical locations, so that the physical distance between those locations matched the virtual distance between the virtual buttons that users had to push. This approach worked because the VE was created for a very specific and structured task and users were instructed to follow a sequence of steps.

For more generic situations, Razzaque [10, p. 86] proposed three redirection algorithms that differed in the physical target of the redirection:

steer-to-center tries to steer the user to the center of the tracked area;

steer-to-orbit tries to keep the user in a circular path around the center of the tracked area;

steer-to-multiple targets tries to steer the user to one of several targets distributed in the tracked area.

Hodgson and Bachmann [42] compared and evaluated the three algorithms proposed by Razzaque (in their steer-to-multiple, they used three targets around the center separated by 120°) plus a combination of the steer-to-multiple and steer-to-center (Figure 8):

steer-to-multiple+center tries to steer the user to several targets around the center plus the center point itself.

They evaluated the algorithms using simulated virtual paths (synthetic paths such as straight lines, zig-zag lines and figure-8 paths) and also with live user paths in an open VE – a lightly wooded forest with snow and fog that obscured distance objects. Although the different algorithms behave differently for different paths, they all performed reasonably well in constraining users to a small physical space (but none succeeded completely in preventing users from reaching the limits of the physical area). In general however, the steer-to-center gave the best results in keeping the user close to the center of the tracked area. Also, all algorithms had difficulty with frequent and rapid

[4] We assume there was a typo in [40] and that the authors meant $-0.04v+0.19$ instead of $-0.04v + 0.09$.

changes in direction. In a subsequent study, Hodgson et al. [33] compared the steer-to-center and steer-to-orbit algorithms in a constrained virtual world (a grocery store with various narrow parallel corridors) and found that the steer-to-orbit algorithm outperformed the steer-to-center in keeping the user in a smaller physical area and in reducing the number of potential wall contacts.

When there is some knowledge available about the VE, the steering target can be dependant on the VE. For example, Steinicke et al. [43] steered users to a physical prop in order to provide users with haptic feedback. Bachmann et al. [44] proposes an algorithm that searches for possible future locations and allows steering targets that are close to the limits of the physical area if it is known (from the structure of the VE) that users will not turn towards the edges.

RDW algorithms require that the system predicts the user's movement direction so that an appropriate rotation, that steers users efficiently without them noticing it, can be applied. Generalized RDW algorithms assume no knowledge of the VE, thus, predictions are usually based on the previous user movements and/or current gaze direction. For example, in one of their direction prediction versions, Peck et al. [45] averaged the previous 29 consecutive differences in virtual location and used the result as the intended future direction.

When applied to constrained VEs RDW algorithms may take advantage of the knowledge about the VE for better prediction of the user's future direction.

In their fire drill exercise VE, Razzaque et al. [30] took advantage of the highly structured task to predict that the following virtual target would be the following waypoint in the task.

Field et al. [34] proposed the *large circle* and *small circle* algorithms conceived essentially for VEs based on corridors where the system knows the distance between the users' current position and the next position where the user must make a turn. The *large circle* algorithm, for example, attempts to keep users on an orbit but when approaching a turn it will "prepare" for the turn by steering users to the center and possibly inverting the orbit's direction (if the turn direction is known). Peck et al. [45] used an algorithm that models the VE as a bi-directed graph such that nodes are points where users may change directions and edges are straight paths between the nodes. In their maze environment, hallways were modelled as edges and intersections and dead-ends were modelled as nodes. The algorithm for predicting the future directions first calculates the closest node to the user and then the nodes connected to that closest node. The angle between the current user heading and the vectors from the user position to the connected nodes is then calculated. The node that has the smallest angle is used as the prediction of the future user direction. Bachmann et al. [44] proposes a probabilistic model not only for predicting users' paths but also for determining the best physical location to steer them to:

> "By preprocessing the map of the virtual environment, the major corridors of travel can be identified and these corridors can be used to produce real-time probabilistic predictions concerning the users future actions. This, in turn, allows RDW to utilize otherwise undesirable space near the physical limits of the

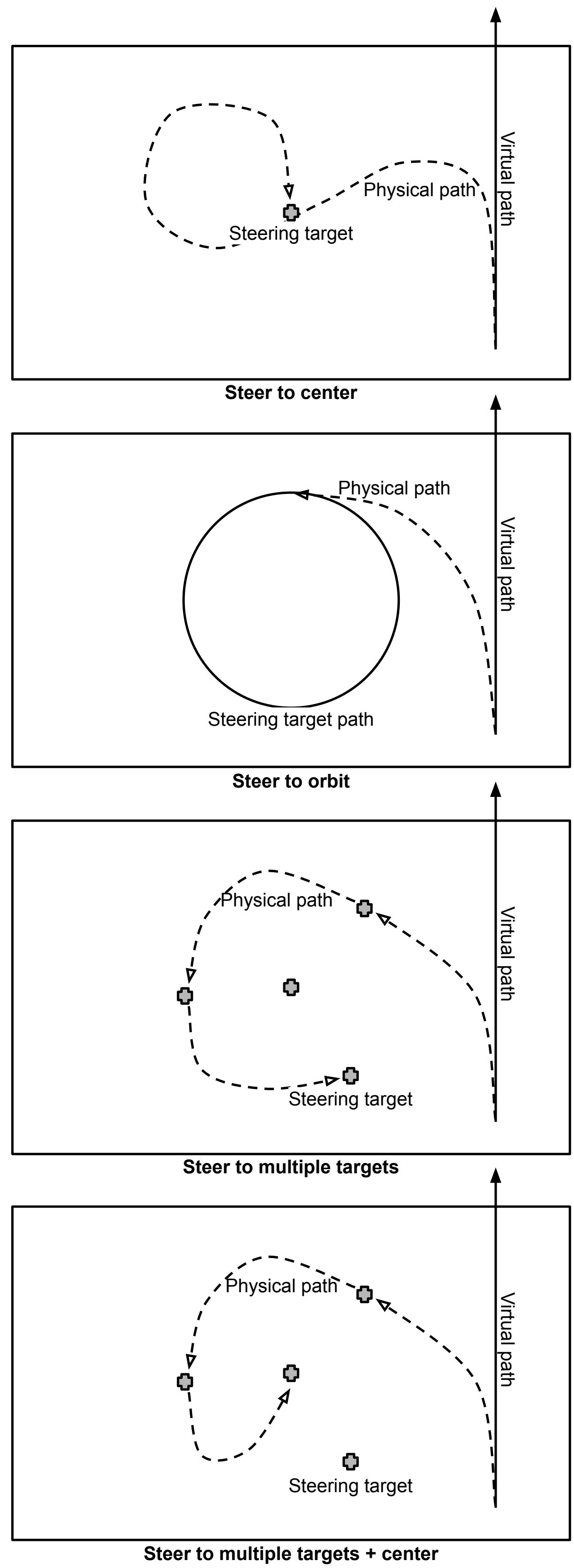


Fig. 8: Redirected walking steering algorithms. Redrawn from Hodgson and Bachmann [42].

> tracking space during periods in which it is known that users will not spontaneously turn in that direction" [44].

Other strategies to predict users' paths have been proposed. For example, Zank and Kunz [46] proposes the use of eye tracking and shows that it allows earlier predictions. Azmandian et al. [47] propose the use of navigation meshes – a method used for path planning in gaming applications – generated automatically to predict possible user's future locations. Cho et al. [48] propose using deep networks to predict users' path based previously collected data samples.

The fact that RDW alters the mapping between the physical space and the virtual space makes it harder to provide users with physical haptic feedback for virtual objects, i.e., designers cannot simply place a physical table next to a wall and a virtual table next to the equivalent virtual wall because there is no 1:1 mapping. In some situations however, being able to provide physical proxy objects that produce haptic sensation to users is important. Steinicke et al. [43] considered the use of passive haptics in their VR experiments with RDW – the ability to steer users to physical objects that proxy virtual objects so that users can touch and feel them. The system has a description of both the objects in the VE and the objects in the physical world so that it can first predict the target virtual object and then redirect the user to the correct physical proxy at the correct intersection angle (see Figure 9). After the target object has been found, the system applies a path transformation that redirects users to the proxy object, including avoiding possible physical objects.

Matsumoto et al. [49] explored the additional haptic cue provided by a physical wall to create an unlimited virtual corridor. A physical circular wall (with an opening in the middle) provides a haptic cue to users walking down a virtual unlimited corridor. The added haptic cue causes users to perceive the corridor as straight, even though the physical wall is a circular arc with 5 m of diameter. Assuming users will keep on walking down the corridor and touching the wall (or go through the opening to another parallel corridor), this has the benefit of requiring much less physical area.

Similarly to the previously described reorientation strategies, RDW may also use distractors in order to get users' attention when a more accentuated redirection is required. In that regard, Peck et al. [45] used a hummingbird that appears in the user's visual field to distract him and elicit head turns. If the user is distracted, the system can apply a greater rotation to the scene and perform a faster redirection. In one version, Peck et al. [45], used a simple algorithm to determine when the distractor should appear or disappear, based on the inversely related values of t, the time since the previous distractor appeared, and d, the distance of the user to the center of the physical area.

Following their study on walking speed influence in curvature gain thresholds, Neth et al. [40] proposed the use of avatars to influence the walking speed of the user and thus allowing for greater curvature gains. Their avatar technique employs two types of avatar. One that walks in front of the user, at a distance that is dependent on the user's walking speed. The faster the user walks, the closer the avatar gets, thus, forcing the user to slow down. The second avatar type is used when the user approaches the boundaries of the tracked area and follows an intercept path that makes the user rotate to avoid colliding with the avatar. The rotation during the collision avoidance can be used by the system to apply a greater rotational gain without the user noticing it.

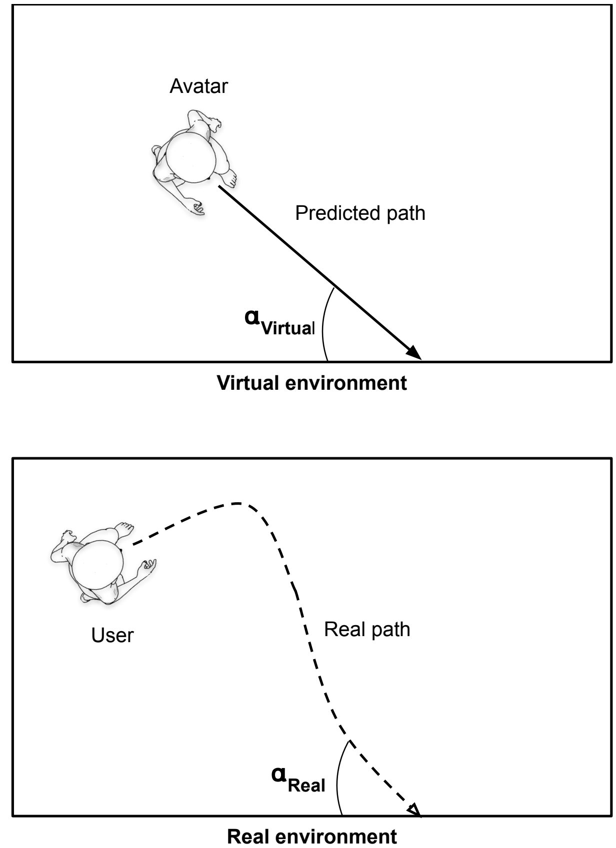


Fig. 9: Redirected walking technique for passive haptics. Redrawn from Steinicke et al. [43].

Most RDW examples are applied to situations where only one user is experiencing the VE at a time. Redirecting multiple users simultaneously is a challenge due to the potential for collisions between users. Bachmann et al. [50] discussed an algorithm for redirection of two simultaneous users using a variation of the steer-to-center algorithm. The algorithm predicts the future positions of the two users and checks for collisions (if the two paths come closer than a threshold at nearly the same time in the future). If a collision is detected, the algorithm decides on: a) steering each user away to a temporary point 90° off their current direction b) stopping one user to allow the other to pass c) stopping both users and asking them to manually adjust their routes. (It is not clear how the decision is made). The evaluation based on simulation studies showed that the algorithm is capable of eliminating all collisions (without collision avoidance the number of unsafe situations was 44 at 31.5 per hour). Most collisions were avoided by steering the users. Stopping users occurred only about 3 times per hour.

Table 4 presents a classified overview of the redirection walking techniques, evidencing their most relevant parameters such as:

- Knowledge about the VE – whether the system has any knowledge about the structure or contents of the VE and can use this information to predict users' paths or steer them to specific locations.
- Direction prediction – how the system predicts the future user direction of movement.
- Steering target – where users are steered to in the physical space.
- Gains – what scaling gains are used.
- Distractors – whether distractors are used and what kind of distractors.
- Multi-user – whether the system supports more than one user simultaneously.

3.6. Dynamic VE

The last technique for real locomotion can be thought of as the inverse of redirection. While redirection attempts to manipulate the user's physical path to fit the virtual environment, dynamic VEs manipulate the environment itself to fit a predefined physical path. Unlike redirection, dynamic manipulation of the VE does not introduce any visual-vestibular conflict. It does, however, introduce physical incongruences in the structure of the VE making it impossible to be physically recreated. The challenge is to make the changes as "invisible" or natural as possible to the user.

One of the simplest forms of manipulating the VE to allow users to explore large spaces is to dynamically create portals into other locations. Portals are passages to other virtual locations that are (temporarily) inserted into the current location. When traversed, users are instantly and seamlessly transported to the remote location. This means that, while portals are open, two virtual locations occupy the same physical space. Portals are transient so they need to be dynamically created and destroyed. Bruder et al. [28] made use of portals in their *Arch-Explore* system. While users were in the virtual replica room (a room modelled after the physical room where users could walk around a virtual architectural model in a very similar way as one does in the real world) they could select a virtual room using a Nintendo Wiimote controller and open a portal to that room. The portal would open instantly in the virtual room as shown in Figure 10. Once users went through the portal they would be inside the chosen room and be able to explore it in real size. The system opens the portal in the direction of the farthest real wall so that, by combining redirection and scaling techniques, the user can safely go through the portal without hitting a real wall. Portals disappear automatically in a few seconds, after users cross them.

Freitag et al. [52] also made use of portals, although in a CAVE-like physical environment, with the purpose of re-orienting users when they approached the workspace's boundaries. Upon reaching the boundaries, users could create a portal to a different location by using a wand and raycasting to select the target location. The portal would then be automatically placed in a location that forced users to turn away from the wall. The portal itself is rendered as a stone arch that smoothly rises from the ground.

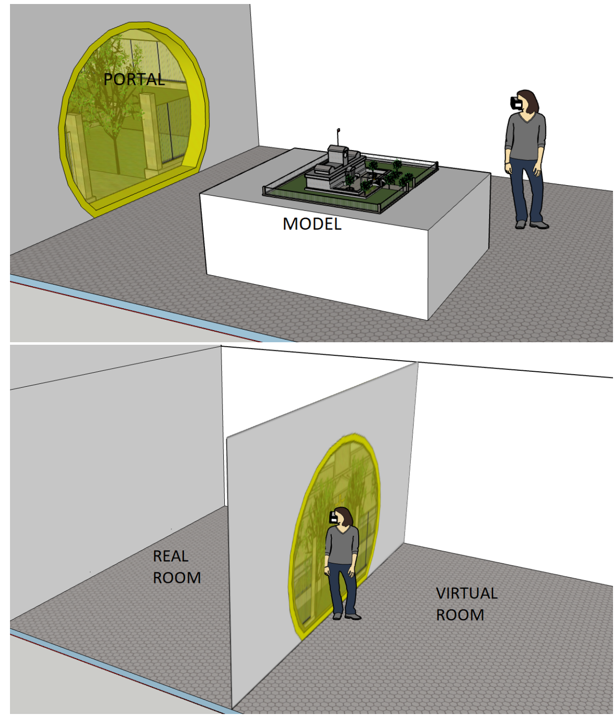


Fig. 10: Portals in *Arch-Explore*. Redrawn from Bruder et al. [28].

Suma et al. [53] explored the concept of overlapping areas: impossible spaces composed of virtual rooms that overlap each other so that the combined virtual area is greater than the available physical area. Their study was limited to two rooms connected by a corridor and sharing a single wall. While the user was in the corridor, the rooms were switched out so that the user would only see the correct sized room. Their study included fixed-size rooms where the overlap was controlled by "sliding" the rooms partially on top of each other (meaning that the available physical area might not be completely used) and expanding rooms where the overlap was controlled by expanding the size of the rooms (the rooms expand to fill in the available physical area and to make sure that a certain level of overlap exists). They tested different levels of overlap and found that for fixed-sized rooms the overlap could be as high as 56% before users noticed that they were in an impossible space and as high as 31% for expanding rooms.

Another way to manipulate the VE is to explore the phenomenon of change blindness – "a perceptual phenomenon that occurs when a change in a visual stimulus is introduced and the observer does not notice it" [54]. Suma et al. [55] performed an experiment where they would ask people to visit 12 rooms in sequence. In each room, users had to go to the corner where a computer was sitting on a desk and turn the computer on. Then they would go back to the corridor and enter another room. While users were facing away from the door, the virtual environment was dynamically manipulated so that the door through which the user entered was switched to the adjoining wall. The net result was that although the VE was very large, the physical

Table 4: Summary of Redirection techniques.

Example	**Knowledge about VE**	**Direction prediction**	**Steering target**	**Gains**	**Distractors**	**Multi-user**
Razzaque et al. [30]	No	Next waypoint in task	Predefined physical targets	n.a.	No	No
Field et al. [34]	No	Pre-defined (simulation)	Steer-to-center	gT: 5 °/sec gR: [0.60, 1.66] gC: n.a.	No	No
Field et al. [34]	Yes	Pre-defined (simulation)	Large circle	gT: 5 °/sec gR: [0.60, 1.66] gC: n.a.	No	No
Field et al. [34]	Yes	Pre-defined (simulation)	Small circle	gT: 5 °/sec gR: [0.60, 1.66] gC: n.a.	No	No
Steinicke et al. [43]	Yes	Walking direction[1], or virtual object[2]	Proxy haptic object	n.a.	No	No
[1]if angle between walking direction and viewing direction less than 45° [2]intersected by raycast in the walking direction						
Bruder et al. [28]	Yes	Virtual doors	n.a.	gT: n.a. gR: [0.59, 1.1] gC: [-0.045, 0.045][1]	No	No
[1]range of [-0.64, 0.64] considered “notice-able, but still not overly distracting”						
Steinicke et al. [51]	n.a.	n.a.	n.a.	gT: n.a. gR: [0.67, 1.24] gC: [-0.045, 0.045]	No	No
Neth et al. [40][1]	No	n.a.	n.a.	gT: n.a. gR: n. a. gC: [-0.095, 0.095]: 0.75 m/s gC: [-0.04, 0.04]: 1 m/s gC: [-0.038, 0.038]: 1.25 m/s	No	No
[1]psychophysical experiment to determine sensitivity to curved path.						
Neth et al. [40] (static condition)	No	n.a.	Steer-to-orbit	gT: 1 deg/s gR: n. a. gC: [-0.13, 0.13]	Avatars (slow down, intersect)	No
Neth et al. [40] (dynamic condition)	No	n.a.	Steer-to-orbit	gT: 1 deg/s gR: n. a. gC: $0.2, v < 0.75m/s$ gC: $0.13, v > 1.25m/s$ gC: $-0.2v + 0.35, 0.75 < v < 1m/s$ gC: $-0.04v + 0.09, 1 < v < 1.25m/s$[1]	Avatars (slow down, intersect)	No
[1]we assume authors meant $-0.04v + 0.19$.						
Bachmann et al. [44], Bachmann et al. [50]	No	n.a.	Steer-to-center; When potential collision: targets placed ad 90 from direction of travel	gT: n.a gR: [0.71, 1.25] gC: n.a.	No	Yes
Bachmann et al. [44]	Yes	Based on structure of VE	Dynamically defined by search process	gT: n.a gR: [0.71, 1.25] gC: n.a.	No	No
Bruder et al. [41]	No	n.a.	n.a.	gT: n.a gR: [0.6810, 1.2594] gC: [-0.0670, 0.0670]	No	No
Bruder et al. [41] wheelchair	No	n.a.	n.a.	gT: n.a gR: [0.7719, 1.2620] gC: [-0.1115, 0.1115]	No	No
Peck et al. [45]	No	Walking direction[1]	Steer-to-center	n.a.	Yes (Hummingbird)	No
[1]average of previous 29 consecutive differences in virtual location.						
Peck et al. [45]	Yes	Bi-directional graph model of VE.[1]	Steer-to-center	n.a.	Yes (Hummingbird)	No
[1]nodes are locations where people may change location. Determine nearest node. User will predicably walk to node connected to nearest node.						
Hodgson and Bachmann [42], Hodgson et al. [33]	No	Walking direction	Steer-to-center (modified); Steer-to-Orbit; Steer-to-Multiple Targets; Steer-to-Multiple+Center	gT: 0.5 deg/s gR: [0.85, 1.30] (clips at 30 deg/s) gC: [-0.13, 0.13] (if linear vel > 0.2 m/s, clips rot speed at 15 deg/s)[1]	No	No
[1]also applies various dampening functions (see paper for details).						

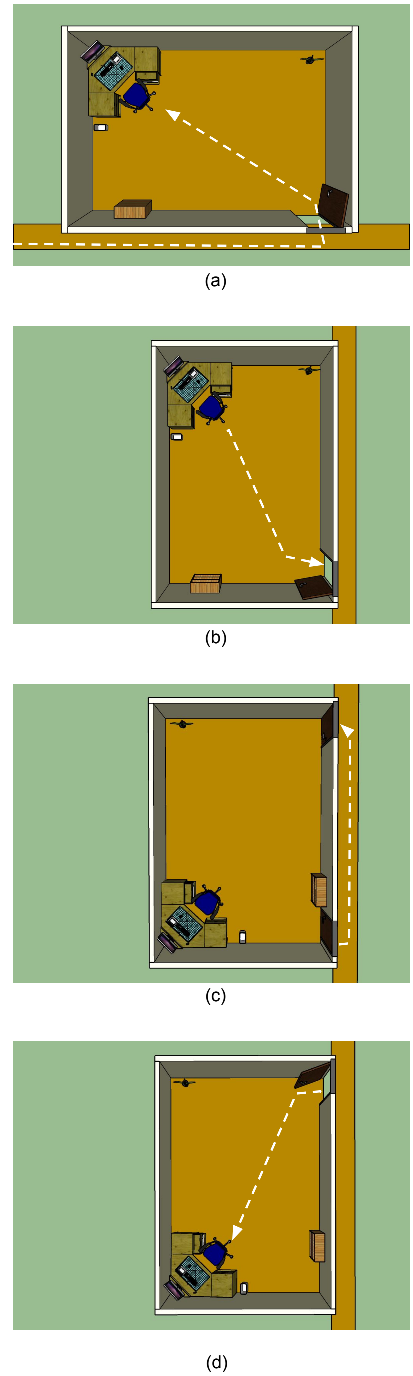


Fig. 11: Layout of rooms in Suma et al.'s [55] experiment. (a) Users enter the office and walk towards the computer on the desk. (b) When the user approaches the computer, the system instantaneously rotates the door and corridor by 90°. (c) When the user exits the office to the corridor, a second door is added and the contents of the office are swapped with the next office. (d) The first door is removed as the user enters the second office. The process can then be repeated. Redrawn from Suma et al. [55].

space necessary to explore the entire VE was only the size of one room plus corridor. As users visited each room, the corridor and doors were rotated 90° so that they would go around in circles in the physical area. Few users noticed the change in the VE although most noticed they were going round in circles. Also, most were able to sketch out the correct layout of the VE. This technique however, only works when users visit the rooms in an orderly sequence. If the user skips a room or continues walking the entire length of the corridor the technique fails.

Another way to dynamically modify a VEs is to restructure the environment for situations where its spatial layout is not important. Vasylevska et al. [56, 57] proposes an approach where both rooms' positions and corridors are dynamically generated every time the user exits a door in the room. For each room that can be accessed from a given room, there must be a specific door (in the original study, rooms are identified by colors, so doors have the same color as the room they lead to). When a user exits through a door, the algorithm regenerates a corridor that leads to the destination room by placing and connecting random points. Their algorithm reserves an outside border in the physical area for placing the corridors so the area available for the rooms is a bit smaller than the total physical space.

In a dynamic VE, such as the ones previously presented, the VR system must know when to apply the necessary modifications. In some cases, it is desirable to hinder the user from seeing the actual modification process, in which case the system is responsible for triggering the necessary changes. Other strategies, such as the use of portals, usually require that the user assert a specific command in order to generate the actual portal. This means that some form of interaction must be available (usually through some kind of handheld device) so that users can select the destination and position for creating the portal. It is conceivable to think about automatic portals that open without an explicit user command, but we have not found an example of this.

When the modification is not supposed to be noticed by users, the system must be programmed to automatically apply it. This means detecting that the user is not looking (either by providing an explicit task that must be performed as in [55], or by applying the change while the user is passing through a location necessary to reach the modification but where the modified region is not yet visible [53, 56])

Table 5 summarizes the possibilities for dynamic VEs:

- Type of modification – whether the system creates a portal, slightly modifies the VE taking advantage of change blindness, overlaps areas or completely restructures the VE.
- When the modification is applied – whether the modification is applied automatically and in what circumstances.
- How the modification appears – whether the modification appears instantly or progressively and whether it is seen appearing by users.

4. Discussion

It is generally agreed that real locomotion is better than other means of locomotion because it results in higher feeling of presence [3] and immersiveness, better orientation [59], faster navigation [1], and less cognitive load [60]. Real walking in VR systems produces the same proprioceptive, vestibular, and ocular cues as walking in the real world, which explains in part, these results.

When implementing VR systems that support real locomo-

Table 5: Summary of Dynamic VE techniques.

Example	Type of modification	When the modification is applied	How the modification appears
Steinicke et al. [58], Bruder et al. [28]	Portal	Requested by user (hand controller)	Instantly
Freitag et al. [52]	Portal	Requested by user (hand controller)	Slowly
Suma et al. [55]	Change blindness	When user not looking (turning on computer)	Not seen by user
Suma et al. [53]	Overlapping	When user not looking (walking in corridor)	Not seen by user
Vasylevska et al. [56, 57]	Restructuring	When leaving/opening door	Not seen by user

tion various practical issues have to be addressed:

Usable area. Clearly, the greatest limitation to real locomotion is the available area. The available area is limited by the size of the room but also by the tracking system. Wide area tracking systems are usually expensive and their cost depends on the total area that one wishes to track. Larger areas require more sensors, equipment, calibration time and most importantly: more real state. Thus, even though commercial systems such Oculus Quest[5] or AntiLatency[6] are pushing the boundaries of tracking capabilities and are bringing down the cost of the technology, the size of the usable area is always dependent on the cost of the actual physical space (real state costs). As a consequence, VR prepared rooms usually represent a considerable investment and are kept dedicated to tracking activities to prevent damage or decalibration of the tracking system. Still, there are examples of commercial systems that use large tracking areas to provide gaming VR experiences such as Zero Latency[7].

If vehicles are to be used, the size of the tracking area is an even more important issue. In their psychophysical experiments with redirected driving, Bruder et al. [41] used a $11\,x\,9.5$ m^2 room and asked participants to move (walk or move the wheelchair) in a straight line or to rotate in place while different gains were applied to the virtual translations and rotations. A room with 104.5 m^2 and associated tracking system is clearly a resource that is not available to everyone.

A related issue is the existence of obstacles in the physical area. When building VEs for un-mediated locomotion, care has to be taken to ensure that users don't bump into the walls or obstacles. For example, Suma et al. [1] explicitly designed the VE so that a small border would exist around the perimeter:

> The dimensions of the environment were precisely designed to fit our 14 x 16 tracking area, leaving 6-inch borders around the perimeter of the area to avoid collisions with the physical environment.

This implies a great care in adapting and calibrating the VE to existing space to make sure that the limits of the VE are adjusted to the limits of the tracked space. These techniques are usually only applied to experimental settings where researchers know exactly where the VE will be experienced.

Even if the virtual area fits in the physical area, designers must take steps to ensure that users do not hurt themselves by bumping into walls when wandering off the VE, or other obstacles while inside the VE. Many HMDs are still tethered, which means that real locomotion techniques must somehow handle cables so that users don't trip on them while exploring the VE. Grechkin et al. [61] point out the encumbrance of the HMD machinery and cables as a factor that affects the resulting interaction:

> The lower walking speed in the HMD condition relative to the real world walking is consistent with earlier studies [35] which show that participants wearing an HMD tend to walk slower compared to real-world walking, which is likely explained by the encumbrance of the HMD machinery.

As VR systems mature, we may assume that these devices will progressively become untethered, alleviating this issue.

Irregular terrains. Even with real locomotion it is very difficult to provide a realistic experience of walking that takes into account the irregularities and textures of different kinds of surfaces. VR tracking rooms are mostly agnostic of the concrete VE that will be experienced there, and floors are also usually flat and hard. The haptic experience of e.g., running on a field of grass, will not be fully provided in such a room. Perhaps even more difficult is to create tracking rooms that allow users to experience going up or down a flight of stairs or simply climb a small hill. Active floor tiles such as the ones described by Visell et al. [62] can provide 3 DOF sensing and vibrotactile feedback and thus be used to create fuller walking experiences. Although it is hard to provide the physical feedback for this kind of irregularities in the VE terrain, Marchal et al. [63] considered providing at least a psychological hint by adjusting the visual feedback. They tested different visual feedback techniques to simulate walking through a bump or a hole in the ground by varying the height of the camera, varying the orientation of the camera, and varying the speed of the camera as users passed through the bump or hole, and found that the combination of all factors works best.

Content production. Producing the content for the VEs may also be a challenge, specifically for the techniques that require the system to have knowledge of the structure or layout of the VE. RDW techniques that take into account the layout of the VE for path prediction for example, need meta-data about the VE. This means either manually annotating the VE – adding to the complexity of producing the content, or preprocessing the VE for automatic annotation – usually less accurate and often

[5]`https://www.oculus.com/quest/`
[6]`https://antilatency.com/)`
[7]`https://zerolatencyvr.com/`

still requiring manual intervention. For dynamic VEs, producing the VEs is also challenging as they may be harder to model statically in traditional 3D modelling software.

Integration with other interaction techniques. In this work we have not specifically addressed the combination of locomotion techniques with other necessary interaction techniques for VR. However, most often this issue will have to be addressed in order to provide a fully interactive VR experience. It might also be important to consider the combination of real locomotion techniques with virtual locomotion for situations where it is not possible or desirable to provide real locomotion only. A good example is the Magic Carpet [64] interaction technique for flying where in one of the evaluated alternatives for speed control – the speed circle – the human body is used as an analog stick controlled by walking within a pre-defined circle.

4.1. Research opportunities

Although various studies about perception in VE have already been performed, there is still various phenomena that are not well understood. For example, although Marchal et al. [63] studied how camera manipulations could be used to provide a sense of walking through bumps and holes and Matsumoto et al. [65] explored the idea of creating a feeling of walking up and downhill, how to emulate sensations like these in VR is still under-explored. While RDW is perhaps the locomotion technique that has attracted most research including psychophysical studies, there is still work to be done regarding for example, how scale changes affect sensitivity to rotational gains [29], how steering interfaces affect the detectability of rotation or translation manipulations [41], how detection thresholds vary as users get accustomed to the manipulations [35], or what is the effect of different virtual self-representations on the detection thresholds [36]. More generally, and perhaps more practically, it would also be important to have more studies that aimed to understand the TLI – the level above which a manipulation will become disturbing for the VE experience [37] – on various types of tasks and manipulations. The effect of age and gender on detection threshold also seems to not have been fully studied. In particular, it might be interesting to study if children and elderly are equally sensitive to redirection manipulations and whether those manipulations induce motion sickness for those groups.

Regarding RDW algorithms, there is some work on how the structure of the VE affects the performance of different algorithms (e.g., Hodgson and Bachmann [42], Hodgson et al. [33]), but it might also be interesting to study how VEs may be purposefully constructed to improve the performance of specific RDW algorithms.

For Dynamic VEs, more research is needed on the perception of overlapping areas and on the effects of these manipulations in the performance of spatial orientation [56]. The emergence of mobile 3D scanning tools with inside-out tracking poses various interesting challenges for Dynamic VEs and real locomotion. For example, how can VEs be dynamically laid out in an unknown physical space so that a single VR experience can be deployed in multiple and completely different physical spaces? (How) can VEs be adapted to physical spaces with various irregular connected spaces such as those found in people's homes? Perhaps more interesting even, can it be done dynamically as the user moves about the physical space, i.e., lay out the VE and map the physical space simultaneously? How would RDW algorithms behave in such a situation and how could they be jointly optimized with dynamic mapping of physical spaces? What tools are needed for designers of VEs to express layout constraints that guarantee the soundness of the VE regardless of the physical space it is experienced in?

There is also a clear opportunity in research for the integration of additional modalities in real locomotion techniques. The sense of smell, for example, is largely un-explored in this context. Even though there well known challenges such as the persistency of smell, there are also clear opportunities not only for the direct augmentation of the virtual experience but also for understanding the interaction effects that smell may have in spatial orientation, detection thresholds, sense of presence, etc. The use of positional sound also seems to be under-explored. Although sound was explicitly used by Peck et al. [24] in their evaluation of distractors, and Razzaque's [10] seminal thesis on RDW called attention for the importance of sound in the VR experience, the direct influence of sound on the locomotion experience does not seem to have been studied in depth. Haptics have received more attention than smell, or sound in relation to real locomotion. Matsumoto et al. [66] for example, has shown how haptic cues can be used to increase redirection gains, Steinicke et al. [43] has adapted the redirection algorithm to redirect users to physical proxy objects for haptic feedback and Wang et al. [22] has used both passive and active haptic feedback in their CloudWalker locomotion interface. However, the effects of haptic fidelity on redirection gains, or the performance of vibrotactiles for warnings or distractors, or even the interaction effects of vibrotactile feedback on redirection are largely unknown.

Another area with scarce research is the use of real locomotion in the outside world. Apple Inc. has submitted a patent for an automotive VR system [67] where various car sensors could be used, including localization, to generate virtual content. The idea of using vehicles while immersed in a VE is not new, but being in a vehicle that the user does not control directly (e.g., in an autonomous car, train, bus) presents challenges that have not been addressed yet. In the perspective of real locomotion techniques, perhaps the most relevant question is to what extent this context could be used as locomotion – does it make sense to create a VE which users move in without direct control? What applications could take advantage of this?

Finally, an important field of research related to the development of VR technologies is the use of immersive interactive virtual environments in medical and clinical treatments. The advances in neurosciences of the last decades have resulted in the achievement of several milestones in the development of Brain-Machine Interfaces (BMIs) capable of interpreting brain activity to control physical actuators and prosthetic limbs [68, 69]. In this context, VR, and more specifically VR locomotion strategies, plays a fundamental role. It is used to create virtual simulations of distinct realistic scenarios capable of inducing specific brain activity and consequent neuroplasticity [70]. Given that the realism of the VE is deeply significant to the success and efficiency of clinical and medical treatments that take advantage of VR technologies [71], understanding how real locomotion techniques can affect and interact with brain-machine interfaces is definitely an open door for multidisciplinary research.

5. Conclusion

This work has presented a survey of real locomotion interaction techniques for HMD-based VR.

We categorize interaction techniques for real locomotion according to the types of system's responses to users' actions which reflect different approaches to locomotion: into Unmediated, Warnings, Reorientation/Resetting, Scaling, Redirection, and Dynamic VE. We further decompose each category and describe the main characteristics of each type of real locomotion technique.

By focusing on the interaction technique – the devices, the user's actions, and the system's responses – we aimed at providing a map of real locomotion techniques and their main characteristics that helps HCI and interaction design practitioners making sense and implementing their own real locomotion techniques. This survey can be used not only by newcomers to the field, to help them make sense of the variety of approaches for real locomotion, but also by current practitioners to better understand and relate different locomotion techniques.

Acknowledgments

This work was financed by national funds through the FCT - Foundation for Science and Technology, I.P., within the scope of the project CISUC - UID/CEC/00326/2019

References

[1] Suma, E, Finkelstein, S, Reid, M, Babu, S, Ulinski, A, Hodges, LF. Evaluation of the cognitive effects of travel technique in complex real and virtual environments. IEEE Transactions on Visualization and Computer Graphics 2010;16(4):690–702. doi:10.1109/TVCG.2009.93.

[2] Zanbaka, C, Lok, B, Babu, S, Ulinski, A, Hodges, L. Comparison of Path Visualizations and Cognitive Measures Relative to Travel Technique in a Virtual Environment. IEEE Transactions on Visualization and Computer Graphics 2005;11(6):694–705. URL: http://ieeexplore.ieee.org/document/1512020/. doi:10.1109/TVCG.2005.92.

[3] Usoh, M, Arthur, K, Whitton, MC, Bastos, R, Steed, A, Slater, M, et al. Walking > walking-in-place > flying, in virtual environments. In: Proceedings of the 26th annual conference on Computer graphics and interactive techniques - SIGGRAPH '99. New York, New York, USA: ACM Press. ISBN 0201485605; 1999, p. 359–364. URL: http://portal.acm.org/citation.cfm?doid=311535.311589. doi:10.1145/311535.311589.

[4] Ruddle, RA, Lessels, S. The benefits of using a walking interface to navigate virtual environments. ACM Transactions on Computer-Human Interaction 2009;16(1):1–18. URL: http://portal.acm.org/citation.cfm?doid=1502800.1502805. doi:10.1145/1502800.1502805.

[5] Boletsis, C. The New Era of Virtual Reality Locomotion: A Systematic Literature Review of Techniques and a Proposed Typology. Multimodal Technologies and Interaction 2017;1(4):24. URL: http://www.mdpi.com/2414-4088/1/4/24. doi:10.3390/mti1040024.

[6] Nilsson, NC, Peck, T, Bruder, G, Hodgson, E, Serafin, S, Whitton, M, et al. 15 Years of Research on Redirected Walking in Immersive Virtual Environments. IEEE Computer Graphics and Applications 2018;38(2):44–56. URL: https://ieeexplore.ieee.org/document/8255772/. doi:10.1109/MCG.2018.111125628.

[7] Anthes, C, Garcia-Hernandez, RJ, Wiedemann, M, Kranzlmuller, D. State of the art of virtual reality technology. In: 2016 IEEE Aerospace Conference. April; IEEE. ISBN 978-1-4673-7676-1; 2016, p. 1–19. URL: http://ieeexplore.ieee.org/document/7500674/. doi:10.1109/AERO.2016.7500674.

[8] Jankowski, J, Hachet, M. Advances in Interaction with 3D Environments. Computer Graphics Forum 2015;34(1):152–190. URL: http://doi.wiley.com/10.1111/cgf.12466. doi:10.1111/cgf.12466.

[9] Bowman, DA. Interaction Techniques for Common Tasks in Immersive Virtual Environments. Ph.D. thesis; Georgia Institute of Technology; 1999.

[10] Razzaque, S. Redirected walking. Phd thesis; University of North Carolina at Chapel Hill; 2005. URL: http://wwwx.cs.unc.edu/{~}eve/dissertations/2005-Razzaque.

[11] Dix, A, Finlay, JE, Abowd, GD, Beale, R. Human-Computer Interaction. 3 edition ed.; Pearson Education Limited; 2004. ISBN 9780130461094. URL: https://amzn.to/2EwAduT.

[12] Ware, C, Osborne, S. Exploration and virtual camera control in virtual three dimensional environments. ACM SIGGRAPH Computer Graphics 1990;24(2):175–183. URL: http://dl.acm.org/citation.cfm?id=91394.91442.

[13] Arns, L. A new taxonomy for locomotion in virtual environments. Ph.D. thesis; Iowa State University; 2002.

[14] De Boeck, J, Raymaekers, C, Coninx, K. Are existing metaphors in virtual environments suitable for haptic interaction. In: Proceedings of the 7th International Conference on Virtual Reality (VRIC2005). s.l. 2005; 2005, p. 261–268. URL: https://uhdspace.uhasselt.be/dspace/handle/1942/5130.

[15] Bowman, DA, Koller, D, Hodges, LF. A methodology for the evaluation of travel techniques for immersive virtual environments. Virtual Reality 1998;3(2):120–131. URL: http://link.springer.com/10.1007/BF01417673.

[16] Mine, MR. Virtual Environment Interaction Techniques. Tech. Rep.; 1995. URL: http://www.cs.unc.edu/techreports/95-018.pdf.

[17] Mendez, RL, Lopez, R. Mobile inside-out VR tracking, now available on your phone. In: ACM SIGGRAPH 2018 Appy Hour on - SIGGRAPH '18. New York, New York, USA: ACM Press. ISBN 9781450358071; 2018, p. 1–2. URL: http://dl.acm.org/citation.cfm?doid=3213779.3213781. doi:10.1145/3213779.3213781.

[18] Jung, S, Choi, YS, Choi, JS, Koo, BK, Lee, WH. Immersive virtual aquarium with real-walking navigation. Proceedings of the 12th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry - VRCAI '13 2013;:291–294URL: http://dl.acm.org/citation.cfm?id=2534329.2534367. doi:10.1145/2534329.2534367.

[19] Greuter, S, Roberts, DJ. SpaceWalk : Movement and Interaction in Virtual Space with Commodity Hardware. Proceedings of the 2014 Conference on Interactive Entertainment 2014;:1–7.

[20] Cirio, G, Marchal, M, Regia-Corte, T, Lécuyer, A. 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 - VRST '09; vol. 1. New York, New York, USA: ACM Press. ISBN 9781605588698; 2009, p. 155. URL: http://portal.acm.org/citation.cfm?doid=1643928.1643965. doi:10.1145/1643928.1643965.

[21] Cirio, G, Vangorp, P, Chapoulie, E, Marchal, M, Lecuyer, A, Drettakis, G. Walking in a cube: Novel metaphors for safely navigating large virtual environments in restricted real workspaces. IEEE Transactions on Visualization and Computer Graphics 2012;18(4):546–554. doi:10.1109/TVCG.2012.60.

[22] Wang, J, Leach, O, Lindeman, RW. DIY World Builder: An immersive level-editing system. IEEE Symposium on 3D User Interface 2013, 3DUI 2013 - Proceedings 2013;:195–196doi:10.1109/3DUI.2013.6550245.

[23] Williams, B, Narasimham, G, Rump, B, McNamara, TP, Carr, TH, Rieser, J, et al. Exploring large virtual environments with an HMD when physical space is limited. Proceedings of the 4th symposium on Applied perception in graphics and visualization - APGV '07

2007;1(212):41. URL: http://portal.acm.org/citation.cfm?doid=1272582.1272590. doi:10.1145/1272582.1272590.

[24] Peck, T, Fuchs, H, Whitton, MC. Evaluation of Reorientation Techniques and Distractors for Walking in Large Virtual Environments. IEEE Transactions on Visualization and Computer Graphics 2009;15(3):383–394. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4663065. doi:10.1109/TVCG.2008.191.

[25] Yu, R, Duer, Z, Ogle, T, Bowman, DA, Tucker, T, Hicks, D, et al. Experiencing an Invisible World War I Battlefield Through Narrative-Driven Redirected Walking in Virtual Reality. In: 2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR). IEEE. ISBN 978-1-5386-3365-6; 2018, p. 313–319. URL: https://ieeexplore.ieee.org/document/8448288/. doi:10.1109/VR.2018.8448288.

[26] Williams, B, Narasimham, G, McNamara, TP, Carr, TH, Rieser, JJ, Bodenheimer, B. Updating orientation in large virtual environments using scaled translational gain. In: Proceedings of the 3rd symposium on Applied perception in graphics and visualization - APGV '06. New York, New York, USA: ACM Press. ISBN 1595934294; 2006, p. 21. URL: http://portal.acm.org/citation.cfm?doid=1140491.1140495. doi:10.1145/1140491.1140495.

[27] Interrante, V, Ries, B, Anderson, L. Seven league boots: A new metaphor for augmented locomotion through moderately large scale immersive virtual environments. IEEE Symposium on 3D User Interfaces 2007 - Proceedings, 3DUI 2007 2007;:167–170doi:10.1109/3DUI.2007.340791.

[28] Bruder, G, Steinicke, F, Hinrichs, KH. Arch-explore: A natural user interface for immersive architectural walkthroughs. In: 3DUI - IEEE Symposium on 3D User Interfaces 2009 - Proceedings. ISBN 9781424439652; 2009, p. 75–82. doi:10.1109/3DUI.2009.4811208.

[29] Abtahi, P, Gonzalez-Franco, M, Ofek, E, Steed, A. I'm a Giant: Walking in Large Virtual Environments at High Speed Gains. In: Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems - CHI '19. New York, New York, USA: ACM Press. ISBN 9781450359702; 2019, p. 1–13. URL: http://dl.acm.org/citation.cfm?doid=3290605.3300752. doi:10.1145/3290605.3300752.

[30] Razzaque, S, Kohn, Z, Whitton, MC. Redirected Walking. In: Proceedings of EUROGRAPHICS. 2001, p. 289–294.

[31] Nitzsche, N, Hanebeck, UD, Schmidt, G. Motion compression for telepresent walking in large-scale remote environments. In: Rash, CE, Reese, CE, editors. Proceedings of SPIE - The International Society for Optical Engineering; vol. 13. 2003, p. 265. URL: http://proceedings.spiedigitallibrary.org/proceeding.aspx?doi=10.1117/12.488379. doi:10.1117/12.488379.

[32] Su, J. Motion Compression for Telepresence Locomotion. Presence: Teleoperators and Virtual Environments 2007;16(4):385–398. URL: http://www.mitpressjournals.org/doi/10.1162/pres.16.4.385. doi:10.1162/pres.16.4.385.

[33] Hodgson, E, Bachmann, E, Thrash, T. Performance of redirected walking algorithms in a constrained virtual world. IEEE Transactions on Visualization and Computer Graphics 2014;20(4):579–587. doi:10.1109/TVCG.2014.34.

[34] Field, T, Bay, S, Vamplew, P. Generalised Algorithms for Redirected Walking in Virtual Environments. International Conference on Artificial Intelligence in Science and Technology 2004;.

[35] Steinicke, F, Bruder, G, Jerald, J, Frenz, H, Lappe, M. Estimation of Detection Thresholds for Redirected Walking Thechniques. IEEE Transactions on Visualization and Computer Graphics 2010;16(1):17–27. URL: papers2://publication/uuid/827DC4DE-CE85-40E0-B997-181D7EDBA5B1.

[36] Kruse, L, Langbehn, E, Stelnlcke, F. I Can See on My Feet While Walking: Sensitivity to Translation Gains with Visible Feet. In: 2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR). IEEE. ISBN 978-1-5386-3365-6; 2018, p. 305–312. URL: https://ieeexplore.ieee.org/document/8446216/. doi:10.1109/VR.2018.8446216.

[37] Schmitz, P, Hildebrandt, J, Valdez, AC, Kobbelt, L, Ziefle, M. You Spin my Head Right Round: Threshold of Limited Immersion for Rotation Gains in Redirected Walking. IEEE Transactions on Visualization and Computer Graphics 2018;24(4):1623–1632. URL: http://ieeexplore.ieee.org/document/8260943/. doi:10.1109/TVCG.2018.2793671.

[38] Langbehn, E, Steinicke, F, Lappe, M, Welch, GF, Bruder, G. In the Blink of an Eye Leveraging Blink-Induced Suppression for Imperceptible Position and Orientation Redirection in Virtual Reality. ACM Transactions on Graphics 2018;37(4):1–11. URL: http://dl.acm.org/citation.cfm?doid=3197517.3201335. doi:10.1145/3197517.3201335.

[39] Sun, Q, Kaufman, A, Patney, A, Wei, LY, Shapira, O, Lu, J, et al. Towards virtual reality infinite walking: dynamic saccadic redirection. ACM Transactions on Graphics 2018;37(4):1–13. URL: http://dl.acm.org/citation.cfm?doid=3197517.3201294. doi:10.1145/3197517.3201294.

[40] Neth, CT, Souman, JL, Engel, D, Kloos, U, Bülthoff, HH, Mohler, BJ. Velocity-dependent dynamic curvature gain for redirected walking. IEEE Transactions on Visualization and Computer Graphics 2012;18(7):1041–1052. doi:10.1109/TVCG.2011.275.

[41] Bruder, G, Interrante, V, Phillips, L, Steinicke, F. Redirecting walking and driving for natural navigation in immersive virtual environments. IEEE Transactions on Visualization and Computer Graphics 2012;18(4):538–545. doi:10.1109/TVCG.2012.55.

[42] Hodgson, E, Bachmann, E. Comparing four approaches to generalized redirected walking: simulation and live user data. IEEE transactions on visualization and computer graphics 2013;19(4):634–43. URL: http://www.ncbi.nlm.nih.gov/pubmed/23428448. doi:10.1109/TVCG.2013.28.

[43] Steinicke, F, Bruder, G, Kohli, L, Jerald, J, Hinrichs, K. Taxonomy and implementation of redirection techniques for ubiquitous passive haptic feedback. Proceedings of the 2008 International Conference on Cyberworlds, CW 2008 2008;:217–223doi:10.1109/CW.2008.53.

[44] Bachmann, ER, Zmuda, M, Calusdian, J, Yun, X, Hodgson, E, Waller, D. Going anywhere anywhere: Creating a low cost portable immersive VE system. Proceedings of CGAMES'2012 USA - 17th International Conference on Computer Games: AI, Animation, Mobile, Interactive Multimedia, Educational and Serious Games 2012;:108–115doi:10.1109/CGames.2012.6314560.

[45] Peck, TC, Fuchs, H, Whitton, MC. The design and evaluation of a large-scale real-walking locomotion interface. IEEE Transactions on Visualization and Computer Graphics 2012;18(7):1053–1067. doi:10.1109/TVCG.2011.289.

[46] Zank, M, Kunz, A. Eye tracking for locomotion prediction in redirected walking. In: 2016 IEEE Symposium on 3D User Interfaces (3DUI). IEEE. ISBN 978-1-5090-0842-1; 2016, p. 49–58. URL: http://ieeexplore.ieee.org/document/7460030/. doi:10.1109/3DUI.2016.7460030.

[47] Azmandian, M, Grechkin, T, Bolas, M, Suma, E. Automated path prediction for redirected walking using navigation meshes. In: 2016 IEEE Symposium on 3D User Interfaces (3DUI). IEEE. ISBN 978-1-5090-0842-1; 2016, p. 63–66. URL: http://ieeexplore.ieee.org/document/7460032/. doi:10.1109/3DUI.2016.7460032.

[48] Cho, YH, Lee, DY, Lee, IK. Path Prediction Using LSTM Network for Redirected Walking. In: 2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR). IEEE. ISBN 978-1-5386-3365-6; 2018, p. 527–528. URL: https://ieeexplore.ieee.org/document/8446442/. doi:10.1109/VR.2018.8446442.

[49] Matsumoto, K, Ban, Y, Narumi, T, Yanase, Y, Tanikawa, T, Hirose, M. Unlimited Corridor: Redirected Walking Techniques using Visuo Haptic Interaction. In: ACM SIGGRAPH 2016 Emerging Technologies on - SIGGRAPH '16. New York, New York, USA: ACM Press. ISBN 9781450343725; 2016, p. 1–2. URL: http://dl.acm.org/citation.cfm?doid=2929464.2929482. doi:10.1145/2929464.2929482.

[50] Bachmann, ER, Holm, J, Zmuda, MA, Hodgson, E. Collision prediction and prevention in a simultaneous two-user immersive virtual environment. In: 2013 IEEE Virtual Reality (VR). IEEE. ISBN 978-1-4673-4796-9; 2013, p. 89–90. URL: http://ieeexplore.ieee.org/document/6549377/. doi:10.1109/VR.2013.6549377.

[51] Steinicke, F, Bruder, G, Hinrichs, KH, Jerald, J, Frenz, H, Lappe, M. Real Walking through Virtual Environments by Redirection Techniques. Journal of Virtual Reality and Broadcasting 2009;.

[52] Freitag, S, Rausch, D, Kuhlen, T. Reorientation in virtual environments using interactive portals. In: 2014 IEEE Symposium on 3D User Interfaces (3DUI). IEEE. ISBN 978-1-4799-3624-3; 2014, p. 119–122. URL: http://ieeexplore.ieee.org/document/6798852/. doi:10.1109/3DUI.2014.6798852.

[53] Suma, EA, Lipps, Z, Finkelstein, S, Krum, DM, Bolas, M. Impossible Spaces: Maximizing Natural Walking in Virtual Environments with Self-Overlapping Architecture. IEEE Transactions on Visualization and Computer Graphics 2012;18(4):555–564. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6165136. doi:10.1109/TVCG.2012.47.

[54] contributors, W. Change blindness. 2019. URL: https://en.wikipedia.org/w/index.php?title=Change{_}blindness{&}oldid=892272478.

[55] Suma, EA, Clark, S, Krum, D, Finkelstein, S, Bolas, M, Warte, Z. Leveraging change blindness for redirection in virtual environments. In: 2011 IEEE Virtual Reality Conference; vol. 19. IEEE. ISBN 978-1-4577-0039-2; 2011, p. 159–166. URL: http://ieeexplore.ieee.org/xpls/abs{_}all.jsp?arnumber=7412174http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5759455. doi:10.1109/VR.2011.5759455.

[56] Vasylevska, K, Kaufmann, H, Bolas, M, Suma, Ea. Flexible spaces: Dynamic layout generation for infinite walking in virtual environments. IEEE Symposium on 3D User Interface 2013, 3DUI 2013 - Proceedings 2013;:39–42doi:10.1109/3DUI.2013.6550194.

[57] Vasylevska, K, Podkosova, I, Kaufmann, H. Walking in Virtual Reality: Flexible Spaces and Other Techniques. The Visual Language of Technique 2015;:81–97doi:10.1007/978-3-319-05341-7.

[58] Steinicke, F, Bruder, G, Hinrichs, K, Lappe, M, Ries, B, Interrante, V. Transitional environments enhance distance perception in immersive virtual reality systems. In: Proceedings of the 6th Symposium on Applied Perception in Graphics and Visualization - APGV '09. New York, New York, USA: ACM Press. ISBN 9781605587431; 2009, p. 19. URL: http://portal.acm.org/citation.cfm?doid=1620993.1620998. doi:10.1145/1620993.1620998.

[59] Chance, SS, Gaunet, F, Beall, AC, Loomis, JM. Locomotion Mode Affects the Updating of Objects Encountered During Travel: The Contribution of Vestibular and Proprioceptive Inputs to Path Integration. Presence: Teleoperators and Virtual Environments 1998;7(2):168–178. URL: http://www.mitpressjournals.org/doi/abs/10.1162/105474698565659. doi:10.1162/105474698565659.

[60] Suma, Ea, Clark, S, Finkelstein, SL, Wartell, Z. Exploiting change blindness to expand walkable space in a virtual environment. Proceedings - IEEE Virtual Reality 2010;:305–306doi:10.1109/VR.2010.5444752.

[61] Grechkin, TY, Plumert, JM, Kearney, JK. Dynamic affordances in embodied interactive systems: The role of display and mode of locomotion. IEEE Transactions on Visualization and Computer Graphics 2014;20(4):596–605. doi:10.1109/TVCG.2014.18.

[62] Visell, Y, Smith, S, Law, A, Rajalingham, R, Cooperstock, JR. Contact sensing and interaction techniques for a distributed, multimodal floor display. In: 2010 IEEE Symposium on 3D User Interfaces (3DUI). IEEE. ISBN 978-1-4244-6846-1; 2010, p. 75–78. URL: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5444718. doi:10.1109/3DUI.2010.5444718.

[63] Marchal, M, Lecuyer, A, Cirio, G, Bonnet, L, Emily, M. Walking up and down in immersive virtual worlds: Novel interactive techniques based on visual feedback. 3D User Interfaces (3DUI), 2010 IEEE Symposium on 2010;:19–26doi:10.1109/3DUI.2010.5446238.

[64] Medeiros, D, Sousa, A, Raposo, A, Jorge, J. Magic Carpet: Interaction Fidelity for Flying in VR. IEEE Transactions on Visualization and Computer Graphics 2019;:1–1URL: https://ieeexplore.ieee.org/document/8667739/. doi:10.1109/TVCG.2019.2905200.

[65] Matsumoto, K, Narumi, T, Tanikawa, T, Hirose, M. Walking uphill and downhill. In: ACM SIGGRAPH 2017 Posters on - SIGGRAPH '17. New York, New York, USA: ACM Press. ISBN 9781450350150; 2017, p. 1–2. URL: http://dl.acm.org/citation.cfm?doid=3102163.3102227. doi:10.1145/3102163.3102227.

[66] Matsumoto, K, Ban, Y, Narumi, T, Tanikawa, T, Hirose, M. Curvature manipulation techniques in redirection using haptic cues. In: 2016 IEEE Symposium on 3D User Interfaces (3DUI). IEEE. ISBN 978-1-5090-0842-1; 2016, p. 105–108. URL: http://ieeexplore.ieee.org/document/7460038/. doi:10.1109/3DUI.2016.7460038.

[67] Rober, MB, Cohen, SI, Kurz, D, Holl, T, Lyon, BB, Meier, PG, et al. Immersive Virtual Display. 2018. URL: http://pdfaiw.uspto.gov/.aiw?docid=20180089901{%}7B.

[68] Lebedev, MA, Nicolelis, MAL. Brain{\textendash}machine interfaces: past, present and future. Trends in Neurosciences 2006;29(9):536–546. doi:10.1016/j.tins.2006.07.004.

[69] Bohil, CJ, Alicea, B, Biocca, FA. Virtual reality in neuroscience research and therapy. Nature reviews neuroscience 2011;12(12):752.

[70] Donati, ARC, Shokur, S, Morya, E, Campos, DSF, Moioli, RC, Gitti, CM, et al. Long-term training with a brain-machine interface-based gait protocol induces partial neurological recovery in paraplegic patients. Scientific reports 2016;6:30383.

[71] Slater, M, Khanna, P, Mortensen, J, Yu, I. Visual Realism Enhances Realistic Response in an Immersive Virtual Environment. {IEEE} Computer Graphics and Applications 2009;29(3):76–84. doi:10.1109/mcg.2009.55.