Walking in Virtual Reality: Flexible Spaces and Other Techniques

Khrystyna Vasylevska, Iana Podkosova and Hannes Kaufmann[1]

Abstract

In many virtual reality applications the virtual world is larger than the available physical workspace. Multiple mechanical solutions have been developed to support the exploration of large virtual environments. However, real walking is still the most immersive way of supporting locomotion in a virtual environment. Redirected walking techniques enable natural locomotion through large scale virtual worlds.

In this chapter we briefly discuss some of the existing interfaces for walking and focus on existing approaches for redirected walking. We will concentrate specifically on spatial manipulation techniques and introduce a novel approach for their use – flexible spaces. This is a innovative redirection technique that enables infinite real walking in virtual environments that do not require the replication of real world layouts. This approach allows designers of virtual environments to focus on the content of the virtual world independently of the implementation details imposed by real walking, thereby making spatial manipulation techniques more practical for use in a variety of application domains.

1 Introduction

Virtual reality (VR) systems possess the capabilities to model and simulate an unlimited variety of environments and situations. Therefore the application domains of VR stretch from science to entertainment, covering almost every aspect of life. At any time the aim of such systems is to provide the most

[1] K. Vasylevska, I. Podkosova, H. Kaufmann
Institute of Software Technology and Interactive Systems,
Faculty of Informatics, Vienna University of Technology,
Favoritenstrasse 9-11 E188/2, A-1040 Vienna, Austria
e-mail: vasylevska@ims.tuwien.ac.at
ipodkosova@tuwien.ac.at
kaufmann@ims.tuwien.ac.at

compelling and useful experience. The purpose of the virtual environment (VE) defines the necessary properties, while the user's sensitivity and real world impose the limitations.

Navigation is one of the most universal tasks performed in real and virtual environments [1]. In the real world it can be performed in a number of ways: driving a car [2], riding a bicycle [3], flying [4], swimming [5], walking, or using unorthodox locomotion methods [6]–[8].

In this chapter we will focus on the most common navigation method – walking. It is a simple and intuitive technique for the interaction with an environment. Walking has been shown to have a positive impact on the sense of presence [4], spatial updating [9], search task performance [10], attention [11], and higher mental processes [12] in comparison to artificial locomotion techniques, such as using joysticks or other interactive devices.

The support of natural walking in VEs remains a significant challenge. The tracking technology and the size of the available workspace normally cannot accommodate exploration of large VEs in a straightforward manner.

To address the limitations of the natural locomotion in large-scale virtual environments, researchers have developed a class of techniques known as redirected walking [22].

Early redirected walking was based on manipulating the mapping between users' physical and virtual motions, resulting in a scaled rotation or translation in the virtual world. In an ideal setup it is possible to maintain the user within a considerably smaller working space while they explores a vast VE. Unfortunately, the ability of the human perceptual system to tolerate the mismatch between visual and vestibular cues is limited. Therefore manipulations become noticeable to users [27] and negatively impact the experience [4].

However, there are several radically different approaches that do not employ self-motion gains, but instead manipulate the architectural layout of the virtual environment through change blindness illusions [29] and self-overlapping architecture [32].

In this chapter, we will provide an overview of the existing redirection algorithms and discuss our approach for the procedural generation of architectural layouts that supports infinite walking through large, highly-occluded virtual environments, which we refer to as flexible spaces [37].

2 Existing Solutions for Walking in Virtual Environments

Walking as a dynamic ability to navigate in VEs is of great interest for many 3D application areas, such as rehabilitation, tourism, or entertainment. An obvious approach to bring real walking to VEs is a one-to-one mapping of the tracked user's head movements to the virtual camera in the VE. Unfortunately, the most natural solution is restricted by the tracking technology and size of the workspace.

Another option to give a user a walking-like experience is the walking-in-place approach that uses walk-like gestures to travel through VEs, while the physical location does not change [4], [13].

2.1 Locomotion Devices

Various prototypes of interface devices have been developed to provide a sense of walking in VEs while keeping the user within a relatively small space or even preventing him from changing the position in the real world.

Shoes based Virtual Perambulator uses roller skates while a user's body is fixed by a belt on the waist [14]. Therefore the body movements are restricted and the user has to adapt and perform the motion himself. Powered Shoes [15] uses motorized roller skates that bring the user back to the initial position. These shoes do not handle rotation, making it difficult to turn.

Most recent solutions, such as Omni [16] and Virtualizer [17], move the focus from shoes to the contacting surface. By providing a low-friction coating together with waist fixation they enable walk-like motions where the user's feet slip to the initial position. The Omni has a curved walking surface, rigidly fixes the waist and requires custom shoes. The Virtualizer uses a flat surface together with socks or special shoe stickers. Unlike previous solutions it offers some freedom in vertical motion. The Virtualizer supports jumping, crouching and sitting.

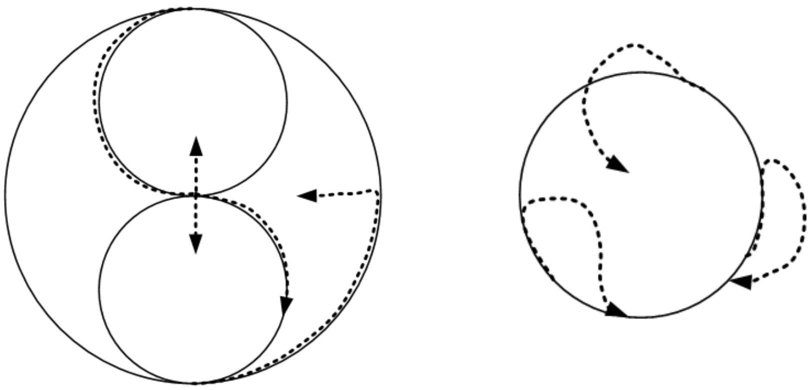
Treadmills Conventional treadmills allow natural walking in one direction. The torus treadmill consists of a closed chain of small treadmills arranged to move perpendicularly to the belts [18]. A flat surface formed by several treadmills allows walking in any possible direction. Belts provide movement along the X axis and treadmills move along the Y axis, bringing the user to the center of the surface.

Similarly, the CyberCarpet uses friction forces to keep the user within its area [19]. This device consists of an array of metal balls that are rolled by a single treadmill that is yawing under them to match the user's walking direction for omni-directional locomotion.

Robotic elements The CirculaFloor employs a number of robotic tiles that reorganize themselves according to the user's movements to simulate an infinite walking surface [20].

Spheres VirtuSphere represents another solution for locomotion, often referred to as a "hamster ball". A user is moving inside a hollow sphere with a diameter of 3 meters that stands on a special platform allowing free rotation. The user may walk or even run in any direction. The sphere's motion is tracked and used to change the user's position in the VE. The main concern of such a solution is inertia, which makes it difficult to stop.

For an excellent review of these and other interfaces for VR see [21].



a b

Fig. 1 Circular redirected walking algorithms. The user is following a circular trajectory with turns that are handled by the algorithms. a – Large circle algorithm. The user stays within the biggest circle. Smooth turns are following smaller circles, 90° turns are supported if directed to the center or started in the center. b – Small circle algorithm. The user walks inside and outside the small circle and is slowly redirected back to the circular trajectory.

2.2 Magical approaches

Among the means for virtual space compression are magical approaches. Typically they are outside of the real world experience and aim to accelerate locomotion in large-scale VEs.

Portals In the Arch-Explore project portals are used for the natural exploration [6]. The portal was incorporated into the VE as a door in the wall that vanished after the transfer to the new location was complete. The self-motion gains, described below, were used in order to fit the start and final locations within the tracked space during the transfer mode. For efficient use of portals additional information was represented by the picture of world in miniature.

Seven league boots and Jumper These metaphors accelerate the virtual motion. Seven league boots apply acceleration according to the user's speed [7]. Jumper uses real walking for short distances and virtual jumps to a predicted location for relatively large distances [8].

Flying This metaphor is typically implemented as a main navigation method or it is push-button activated [4].

2.3 Redirected Walking Algorithms

The domination of visual cues over other senses allows enlarging the virtual scene available for exploration with redirected walking techniques [22]. The most basic approach is to stop the user at the boundary of the tracked space and instruct him to perform head turns in order to unnoticeably rotate the scene [23]. Other

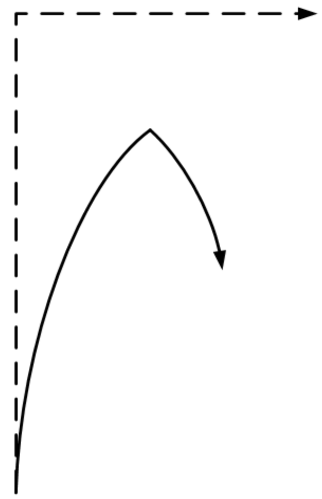


Fig. 2 Gain based redirected walking. *Dotted line* – path taken in the VE. *Solid line* – real world path: the straight virtual trajectory is curved and downscaled, 90° rotation is increased. As a result the space needed for real walking is a lot smaller.

wise, the user might be simply stopped and asked to turn off the HMD and to return to the center of the tracking space and continue from the same point in the VE [22]. Some methods introduce scene rotation with or without warning [24]. Rotation can also be introduced based on the user's position in the VE [25].

Circular algorithms keep the user on a circular trajectory. The user is either returned from the outside of a small circle or kept inside a large circle within the tracked area [25] (see Fig. 1).

Rotation can be applied constantly on an imperceptible level as in [26] or via system gains proportional to the change in user's position and orientation [27]. In [27] and [28] the human sensitivity was evaluated towards the changes of the gains from the initial one-to-one mapping. It was found that humans are unable to notice a difference between the virtual and the real-world movement below a certain threshold. Distances may be upscaled by 26% and downscaled by 14%, while rotation may be increased by 49% and decreased by 20% of the intended rotation (see Fig. 2). A user that is walking straight in a VE may be redirected in the real world to walk on a circular arc with a radius of 22 meters. Therefore an approach with self-motion gains still demands a large tracked space to remain unperceived.

2.4 Motion Manipulation Based on Perception Flaws

A completely different approach was proposed in [29], which used a perceptual phenomenon known as change blindness to redirect the user. According to [30], change blindness is a striking failure to see large changes that should be noticed easily. The technique was applied while the user was distracted by a task. By changing the positions of doors in a series of virtual rooms it was possible to modify the direction of the user's movement within multiple scenes in a systematic way without it becoming noticeable (see Fig. 3).

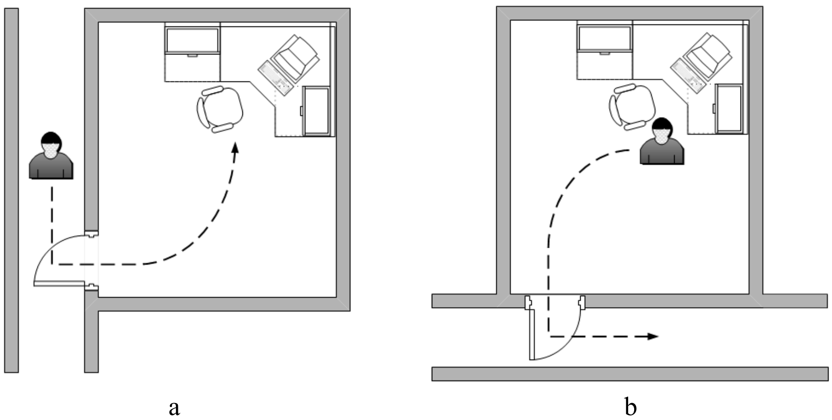


Fig. 3 Change blindness based redirected walking. a – The user enters a room to perform a task. b – During the task performance the door is relocated in order to keep the user within the tracking space.

Bruder and Steinicke used change blindness to create a self-motion illusion [31]. Approximately four times per second there is an opportunity to change the scene due to blinking and saccades. Displaying a grey screen for 60-100 ms masked the change in position and orientation of the virtual camera, after the next grey screen the changes were reversed. As a result the same physical distances walked in a VE with and without this illusion were estimated differently.

Another approach to expand the size of the environment uses impossible spaces with overlapping geometry [32]. An example is shown in Fig. 4. The user is walking from one room to the adjacent one. The shared wall between them is moved in such a way that the visited room is enlarged at the expense of the other one. This approach allows more efficient use of the physical space. The percentage of the unperceived overlap between two rooms depends on their sizes. It is higher for the small rooms and decreases as the rooms' sizes grow.

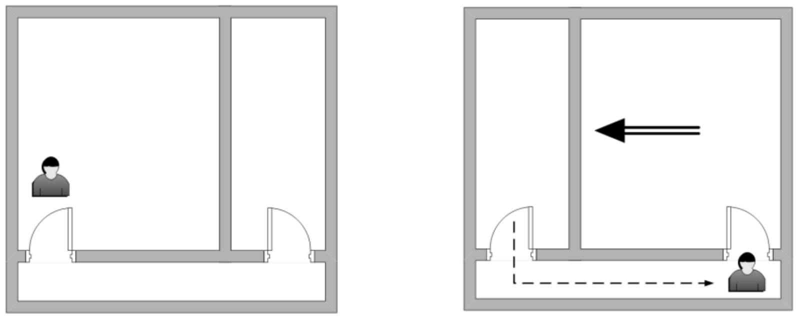


Fig. 4 Impossible spaces use overlapping spaces. While the user is walking from one room to the other, the wall between them is moved in order to increase the size of the targeted room. The available tracked space is used more efficiently.

3 Spatial Cognition

Virtual training is one of the application areas for immersive VEs where real walking is potentially advantageous. For many scenarios it is important to create a representation of the environment called a cognitive map for successful orientation in it (e.g. soldiers in a tactical training system) [33]. With this in mind, prior VEs were trying to recreate the real world in detail while limiting the possible applications of redirection techniques.

However, the word “map” is rather misleading. It suggests a graphical representation of the memories, which is not always true. In reality, our cognitive maps often contain not graphical, but categorical and hierarchical representations of the world [33], [34]. According to [33] detailed spatial knowledge might be useful for navigation, but is not necessary. In addition, cognitive maps are often distorted. Some of them even cannot be represented by images [35]. For instance, distance asymmetry: the distance between two points might differ depending on direction. These distortions originate from the hierarchical structure of the cognitive maps and mental heuristics that help us remember information about the environment [33].

Like in change blindness, the consistency of the environment (the existence of a connection between two points in both ways) is more important than the respective details (the actual distance between them). In [34] it is shown that objects are often used for more precise orientation than landmarks. Without them the error in orientation and navigation significantly increases even in a very basic environment. Scholl in [36], among other aspects, describes a connection between the object-based attention and `packaging of the world into units'.

Therefore the human perception allows us to create a new class of environments. If the purpose of the VE does not depend specifically on the spatial layout of the environment, the requirements for architectural conformity may be relaxed and allow more extensive manipulations.

While perception manipulation techniques have received a lot of attention, the full potential of scene manipulation is still to be discovered. Although exploitation of change blindness and impossible spaces improved flexibility of the virtual environments, most of the abovementioned techniques are trying to maintain structures inherent in the real world with no to minimal changes.

4 Flexible Spaces: an extreme non-architectural approach to redirection

A key observation that may be derived from the fuzziness of the cognitive maps is that maintaining an architectural structure consistent within a single virtual environment is not always necessary. It largely depends on the specific application domain of the VE and desired knowledge or experience that should be obtained.

We distinguish two different types of virtual environments:

Structural VEs where the spatial layout of the environment is the key information for the user. They might be used for tactical simulation, architectural design or virtual excursions, etc.

Informational VEs where the particular layout is not critical and the focus is on the content (objects and information) within the VE. This type of VEs may be applied for virtual therapy, educational applications or entertainment.

For informational VEs there is no need to copy the spatial layout of real, existing environments. Instead, they can be treated as virtual, non-physical environments that resemble the real ones visually, but do not obey the same laws.

The concept of flexible spaces [37] may be described as an impossible environment that violates the real world constancy in favor of providing the experience of seamless, unrestricted natural walking over a large-scale virtual space. It is a dynamic self-overlapping layout generation algorithm that executes automatic relocation and restructuring of parts of the environment to fit into the tracked space.

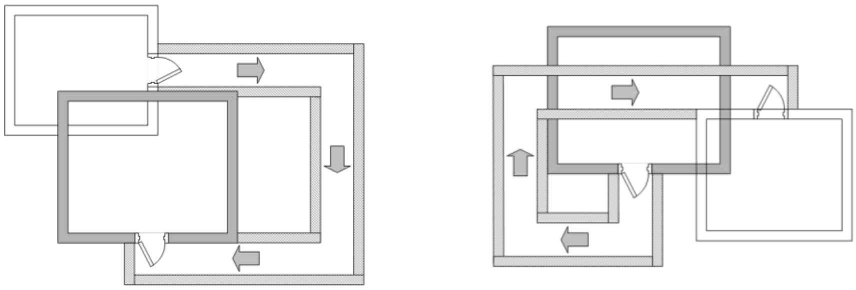
Figure 5 shows examples of flexible spaces for two rooms. The user navigates from one room to another following one route and eventually comes back using a different route. Despite a self-overlapping layout his view is kept consistent.

One of the possible use cases of our approach is a virtual museum - a large-sized environment, where visitors are more interested in exhibits than specific paths. Corridors connect to rooms with related topics. Another application domain of flexible spaces is military training, in particular for training in search for key landmarks or other cues and for the orientation in an unfamiliar environment.

Consequently, this approach opens new possibilities for exploring the environment with less physical restrictions. To the best of our knowledge, our algorithm is the first attempt to generalize architectural manipulation illusions for practical use in immersive virtual environments.

4.1 Design

The test environment is designed as a complicated building that users are able to explore via real walking without invoking movement gains manipulation, distractions [26] or explicit instructions to the user. To this end, we distinguish between two subtypes of the environment architecture: (1) informational, which consists of a room with its features and content, and (2) transitional, represented by corridors. The informational part of the VE undergoes minimal changes necessary to maintain consistent orientation cues. All changes are applied before the user enters the room. This eliminates the requirement for a specific route inside the room and allows the changes to be left unnoticed. The transitional part (corridor) and targeted room position in the tracked space are procedurally generated and vary according to the algorithm described below.



a b

Fig. 5 Flexible spaces approach. a – The corridor from one room to another is generated by the algorithm. b – The corridor is regenerated with a new shape every time the user walks from one room to another.

We perform dynamic restructuring of the layout with random factors to avoid a buildup of knowledge of fixed layout patterns. It would be possible to pre-generate the whole VE using our algorithm. However, depending on the size of the environment, the user might learn the layout over time. This might expose the impossibility of the VE. From our perspective, flexible spaces alleviate the need for a detailed cognitive map, shifting the responsibility for orientation cues to the VE, while a user decides where to go. The induced inconsistency encourages the user to succumb to such an approach. Moreover, flexible spaces do not limit the user's route and support infinite walking.

In our algorithm, rooms' positions are changed randomly and the positions of the doors are changed to ensure that it is possible to access rooms from all directions. We reserve a corridor wide space (1m from each side) along the perimeter of the tracked area, placing rooms in the *inner* space. Procedural generation automatically adjusts the VE to the tracked space.

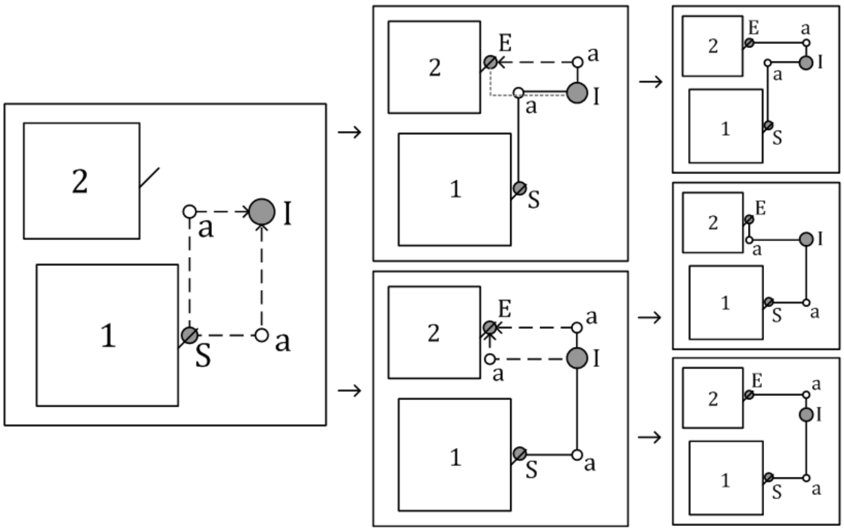
To ensure that users can navigate successfully we decided that the connections assigned to each room should not change and should be bidirectional. This way the user is able to return to the previously explored rooms at any moment.

While in a custom built environment the information for navigation is provided by a designer, for content oriented flexible VEs a simpler solution is to extend the room specific information to the doors that lead to these rooms. In our test environment we used rooms' wall colors for this purpose (partially shown in Fig. 5).

4.2 Algorithm

Our algorithm for procedural layout generation computes room positions and generates connecting corridors in the following way (see Fig. 6):

1) The user inside initial room 1 selects and opens a door to a target room 2.



a b c

Fig. 6 The algorithm of flexible spaces. a - An intermediate point *I* is randomly selected and the end point *E* is defined by the nearest door of the target room 2. b - Random selection of point *a* creates a variation of twists of the corridor. c - Several possible corridor routes can be generated from these combinations of points.

2) The target room 2 is relocated randomly in the inner space of the tracked area so that it fits in. There are no restrictions on the room position. Initial and target rooms may fully overlap.

3) The position of the opened door is taken as a starting point *S* of the corridor.

4) Then an intermediary point *I* is selected randomly (see Fig. 6 a). A first point *I* is selected with a condition that its position is not inside or behind the initial room. This ensures that the room space is not broken immediately after the user leaves the room. Similarly, the last *I* point should not be inside the target room.

With multiple *I* points it is important to ensure that the coordinates of the consecutive points are not too close to each other, defined by a minimum segment size.

5) A door of the target room 2 is selected on the basis of minimal distance to the last intermediary point *I*. The position of the door is taken as the end-point of the corridor – point *E* as shown at Fig. 6 b.

6) To connect the main points of the corridor additional points *a* are calculated. The decision which additional point is chosen for corridor construction is also made randomly, except the cases when a chosen *a* conflicts with a previous choice and creates a deadlock or if it is next to the *E* or *S* points and breaks into

the room. Fig. 6 c shows the possible corridors that could be built depending on the choice of additional points. One *a* is discarded because of a deadlock.

7) When the corridor route calculation is finished the doors in the target room are relocated so that the door at the point *E* corresponds to the initial room and the rest of the doors are relocated relative to it.

8) After that the corridor is built.

The length of the corridor depends on the number of intermediate points. Absence of I points is equivalent to a short straight corridor or a corridor with a maximum of three corners. During initial testing, the corridors of this type exposed the technique and, therefore, were declared inefficient and excluded from the test VE.

The shape of the corridor depends on the size and location of the rooms it connects. With each intermediate point the maximum amount of the corners is increased by two and the length of the corridor depends on the specific location of the point. We suggest using 1 or 2 *I* points per corridor. Examples of layouts for suggested settings are shown in Fig. 6 c. There is a tradeoff between available modifications of a corridor and the sizes of the rooms it connects. The variability of the corridor is automatically reduced to avoid overlap detection, so that the corridor does not intersect with the room in the proximity of the door.

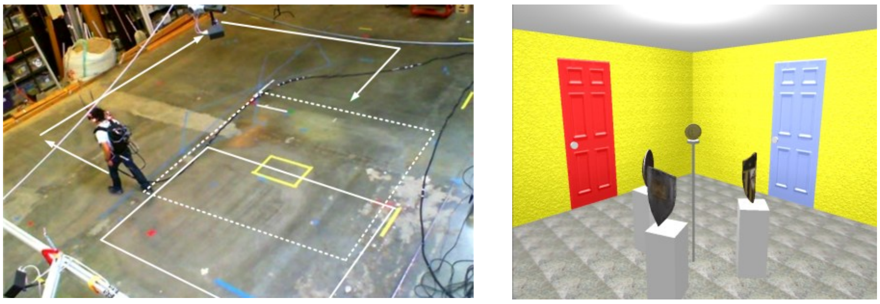
This approach does not limit the number of rooms in a VE. To provide users a consistent view of the environment despite dense overlaps we render only part of the VE at a time based on the user's position in the VE and his movement direction. We determine which objects can be seen by the user and do not render occluded objects.

4.3 Benefits and Limitations

The benefits of our solution are the support of an unlimited number of rooms in the VE, multiple entrances/exits in a virtual room and absolute freedom of the user's choice where to go. Infinite walking is supported without any additional limitations. These features are unusual for the highly constrained indoors spaces that typically require a tight control over users' actions. The combination of the three of them makes flexible spaces a unique approach.

Flexible spaces are limited to individual spaces interconnected by walkways bound by natural constraints. The necessity of the inner space puts constraints on the size of the tracked area that should fit at least one room and provide a corridor reserved space (at least 1m wide) around them. The flexible spaces algorithm might be used without this limitation at the cost of high probability of creating an over-constrained space where redirection fails.

The corridor length for rectangular tracked spaces has an upper limit described by $n \bullet (P-4c)$, where n is the number of I points, P is the perimeter of the tracked space and c is the corridor width. In practice the average lengths of the corridors tend to 44% and 38.5% of the said limit for one and two I points respectively. In our 9x9m test environment the corresponding lengths are 14.25 and 24.95 m.



a b

Fig. 7 Flexible spaces test environment. a – The user walking within a tracked space from one virtual room to another through a simple corridor. b – A virtual room with a 3D content, a numbered token and multiple doors.

The algorithm supports rectangular rooms and may be extended to other shapes. In case of a room approaching the size of the inner space our solution will be equivalent to impossible spaces with a noticeable overlap. A potential way to solve the issue of interconnecting multiple large rooms will be the nonlinear application of translational gain and/or to increase the number of *I* points in the corridor.

4.4 Usability Testing

Virtual environments that were generated by our algorithm were heavily tested during the development stage to identify the factors that have an impact on the user's experience. The development was concluded with a pilot study.

Pilot study To test the flexible spaces approach we limited the environment to 5 rooms of different sizes, so that the users would take some routes several times during test sessions. We defined the connections between the rooms by corresponding colors and provided rooms' content: sets of 3D objects and numbered tokens. The environment was tested by five people. Two of them were naïve users. We explained to users the meaning of the door colors and instructed them to think aloud while in the VE. Each person spent approximately 30-35 minutes in the VE and performed two sessions: first - with the task to remember the correspondence between rooms and numbered tokens, second - with the task to obtain the tokens in ascending order. Fig. 7 shows the user in the tracked space and the virtual room.

We observed that the search task was performed successfully and faster than exploration. This suggests that users are able to successfully navigate in flexible spaces. One participant commented "So many turns. It's like a maze." and we got similar comments from the rest of the users. During the interview naïve users indicated they felt that it might be possible to build the VE in the real world. The users seemed to be comfortable with following the corridors to reach the targeted room. When asked to compare the sizes of the VE and tracked area some users testified that flexible spaces were perceived to be larger.

Once when the user entered the room and then immediately decided to return to the previous room the change of the corridor was suspected, as the corridors were changing every time the user opens the door. We suggest countering such situations with preserving the structure of the last visited corridor unless the user opens another door. We also got some comments for small inner parts of the corridor, approx. 0.5 m wide, which was equivalent to the defined minimal corridor segment, formed by two corners placed close to each other. They were described as “a bit weird”. That might be amended by increasing this parameter to 1 or 2 m.

Observational data and feedback given in interviews suggest that with the modifications mentioned above, our technique tends to be unnoticed by users.

Robustness testing As the VE is procedurally generated based on multiple random factors together with user defined parameters, it is hard to identify all possible combinations that might occur. The change of the size or shape of the tracked space, generation and limitation parameters of the VE might lead to an over-constrained solution. In this case there is a high probability of a redirection failure or a severe reduction of the corridor variety. Therefore extensive testing is often needed, while user tests are not cost- or time-efficient.

In such a situation, a computer simulation is preferred to user testing. For this purpose we have developed an autonomous walking agent. This agent moves through the VE just as a real user would do. Once in a room, it chooses a door to go to randomly and then continues moving forward along the corridor until the door to the next room is reached. In the current solution, we do not account for rare situations where a user would make a 180° turn in the middle of the corridor and continue moving backwards. However, it is possible to extend the set of possible movement choices in the future.

Our simulation helps to speed up testing, provides automatic identification of problematic situations and corridor patterns without the need for an actual tracking space.

Observations Based on our informal observations, we suggest that the probability that the spatial manipulation will remain unnoticed depends on the length and number of corners in the corridors. Short corridors with no intermediate points were remembered and exposed the technique, while corridors with more than two intermediary points were too long to be practical. The effectiveness of our approach, therefore, seems to depend on both the number of corners and distances between them. Ultimately, we suggest that there may be a tradeoff between obscuring the user’s sense of direction and invoking a sense of feeling lost. We assume that this effect might also be related to inherent in the VEs lack of cues for orientation. We plan to overcome this issue with wayfinding aids, such as displaying a connection graph of the environment with the user’s relative position.

5 Conclusion

While most natural locomotion techniques support movement at close range, there is also the need of larger distance locomotion. This, however, often conflicts with real world constraints. Locomotion interfaces, walking-in-place and magical approaches are able to provide long distance locomotion at the cost of naturalness of the experience. The lack of actual movement in space reduces the effectiveness of proprioceptive and vestibular feedback as it does not match the real world experience anymore. Keeping the user inside a small area influences the sense of spatial presence, attention, and cognition.

Redirected walking algorithms enable real walking but require precise planning and control for the successful redirection and for avoidance of collision with real world obstacles. Our body's imperfect connection between proprioceptive feedback and spatial perception allow spatial manipulation techniques and perceptual tricks, which are very effective. Nevertheless, each of the abovementioned approaches is a partial non-generic solution of the problem and introduces its own limitations in order to be unnoticeable and not to cause cybersickness.

Walking in VR is getting closer to natural locomotion in the real world. That increases demands to the techniques used, which are sometimes incompatible with real world constraints. When developing a new VR locomotion technique, it is already important to take the user's conscious and subconscious mental functions, his wishes, and expectations into consideration. They certainly impose the need for new inconspicuous space compression methods. The aim is to develop a universal redirection methodology that allows full freedom of navigation within a VE, with minimum limitations. We see flexible spaces as another step in this direction, where the final goal is true virtual reality.

Acknowledgements This work was partially funded by the Vienna PhD School of Informatics and USC Institute for Creative Technologies under W91 1NF-04-D-0005. The authors thank Evan Suma, Mark Bolas, Adam Jones, and Thai Phan from the USC ICT Mixed Reality Lab for their support during the initial development of flexible spaces.

References

[1] D. Bowman, E. Kruijff, J. LaViola Jr., and I. Poupyrev, *3D User Interfaces: Theory and Practice*. Addison-Wesley, 2005.

[2] H. S. Kang, M. K. A. Jalil, and M. Mailah, "A PC-based driving simulator using virtual reality technology," in *SIGGRAPH VRCAI*, 2004, pp. 273–277.

[3] R. Ranky, M. Sivak, J. Lewis, V. Gade, J. E. Deutsch, and C. Mavroidis, “VRACK - Virtual Reality Augmented Cycling Kit : Design and Validation,” in *IEEE Virtual Reality*, 2010, pp. 135–138.

[4] M. Usoh, K. Arthur, M. C. Whitton, R. Bastos, A. Steed, M. Slater, and F. P. Brooks, “Walking > Walking-in-Place > Flying, in Virtual Environments,” in *Computer graphics and interactive techniques SIGGRAPH 99*, 1999, pp. 359–364.

[5] S. Fels, S. Yohanan, S. Takahashi, Y. Kinoshita, K. Funahashi, Y. Takama, and G. T. Chen, “User Experiences with a Virtual Swimming Interface Exhibit,” in *International Conference on Entertainment Computing*, 2005, pp. 433–444.

[6] G. Bruder, F. Steinicke, and K. H. Hinrichs, “Arch-Explore: A natural user interface for immersive architectural walkthroughs,” *IEEE 3DUI*, pp. 75–82, 2009.

[7] V. Interrante, B. Ries, and L. Anderson, “Seven League Boots: A New Metaphor for Augmented Locomotion through Moderately Large Scale Immersive Virtual Environments,” in *IEEE 3DUI*, 2007.

[8] B. Bolte, F. Steinicke, and G. Bruder, “The Jumper Metaphor: An Effective Navigation Technique for Immersive Display Setups,” in *Virtual Reality International Conference*, 2011, no. April.

[9] S. S. Chance, F. Gaunet, A. C. Beall, and J. M. Loomis, “Locomotion Mode Affects the Updating of Objects Encountered During Travel: The Contribution of Vestibular and Proprioceptive Inputs to Path Integration,” *Presence Teleoperators Virtual Environ.*, vol. 7, no. 2, pp. 168–178, Apr. 1998.

[10] R. A. Ruddle and S. Lessels, “The benefits of using a walking interface to navigate virtual environments,” *ACM Trans. Comput. Interact.*, vol. 16(1), pp. 1–18, 2009.

[11] E. A. Suma, S. L. Finkelstein, S. Clark, P. Goolkasian, and L. F. Hodges, “Effects of travel technique and gender on a divided attention task in a virtual environment,” in *IEEE 3DUI*, 2010, pp. 27–34.

[12] C. A. Zanbaka, B. C. Lok, A. C. Babu, A. C. Ulinski, and L. F. Hodges, “Comparison of path visualizations and cognitive measures relative to travel technique in a virtual environment,” *IEEE Trans. Visu- alization Comput. Graph.*, vol. 11(6), pp. 694–705, 2005.

[13] J. Feasel, M. C. Whitton, and J. D. Wendt, “LLCM-WIP: Low-Latency, Continuous-Motion Walking-in-Place,” in *IEEE Symposium on 3D User Interfaces*, 2008, pp. 97–104.

[14] H. Iwata and T. Fujii, “Virtual perambulator: a novel interface device for locomotion in virtual environment,” in *IEEE Virtual Reality Annual International Symposium*, 1996, pp. 60 – 65.

[15] H. Iwata, Y. Hiroaki, and H. Tomioka, “Powered Shoes,” in *SIGGRAPH Emerging Technologies*, 2006.

[16] “Virtuix Omni.” [Online]. Available: http://www.virtuix.com/.

[17] “Cyberith Virtualizer.” [Online]. Available: http://www.cyberith.com/.

[18] L. Bouguila and M. Sato, *Virtual locomotion system for large-scale virtual environment*. ACM Press, 2002, p. 227.

[19] A. De Luca, R. Mattone, P. Robuffo Giordano, H. Ulbrich, M. Schwaiger, M. Van Den Bergh, E. Koller-Meier, and L. Van Gool, “Motion Control of the CyberCarpet Platform,” *IEEE Trans. Control Syst. Technol.*, vol. PP, no. 99, pp. 1–17, 2012.

[20] H. Iwata, H. Yano, H. Fukushima, and H. Noma, “CirculaFloor,” in *IEEE Computer Graphics and Applications*, 2005, pp. 64–67.

[21] H. Iwata, “Locomotion Interfaces,” in *Human Walking in Virtual Environments*, F. Steinicke, Y. Visell, J. Campos, and A. Lécuyer, Eds. Springer New York, 2013, pp. 199–219.

[22] S. Razzaque, “Redirected Walking,” PhD Thesis. UNC at Chapel Hill, 2005.

[23] B. Williams, G. Narasimham, B. Rump, T. McNamara, T. H. Carr, J. Rieser, and B. Bodenheimer, “Exploring Large Virtual Environments with an HMD when Physical Space is Limited,” in *APGV*, 2007, vol. 1, no. 212, pp. 41–48.

[24] T. C. Peck, M. C. Whitton, and H. Fuchs, *Evaluation of Reorientation Techniques for Walking in Large Virtual Environments*, vol. 15, no. 3. IEEE, 2008, pp. 121–127.

[25] T. Field, S. Bay, and P. Vamplew, “Generalised Algorithms for Redirected Walking in Virtual Environments,” in *AISAT2004 International Conference on Artificial Intelligence in Science and Technology*, 2004, vol. 65, no. 11, pp. 1357–1366.

[26] T. C. Peck, H. Fuchs, and M. C. Whitton, “Improved Redirection with Distractors: A large-scale-real-walking locomotion interface and its effect on navigation in virtual environments,” *2010 IEEE Virtual Real. Conf.*, pp. 35–38, Mar. 2010.

[27] F. Steinicke, G. Bruder, J. Jerald, M. Lappe, and H. Frenz, “Analyses of Human Sensitivity to Redirected Walking,” in *VRST*, 2008, pp. 149–156.

[28] F. Steinicke, G. Bruder, J. Jerald, H. Frenz, and M. Lappe, “Estimation of Detection Thresholds for Redirected Walking Techniques,” *IEEE Trans. Vis. Comput. Graph.*, vol. 16, no. 1, pp. 17–27, 2010.

[29] E. A. Suma, S. Clark, D. Krum, S. Finkelstein, M. Bolas, and Z. Warte, “Leveraging change blindness for redirection in virtual environments,” *2011 IEEE Virtual Real. Conf.*, pp. 159–166, Mar. 2011.

[30] D. J. Simons and R. a Rensink, “Change blindness: past, present, and future.,” *Trends Cogn. Sci.*, vol. 9, no. 1, pp. 16–20, Jan. 2005.

[31] G. Bruder, F. Steinicke, and P. Wieland, “Self-motion illusions in immersive virtual reality environments,” in *IEEE Virtual Reality*, 2011, pp. 39–46.

[32] E. A. Suma, Z. Lipps, S. Finkelstein, D. M. Krum, and M. Bolas, “Impossible spaces: maximizing natural walking in virtual environments with self-overlapping architecture.,” *IEEE Trans. Vis. Comput. Graph.*, vol. 18(4), pp. 555–564, Apr. 2012.

[33] N. G. Vinson, “Design Guidelines for Landmarks to Support Navigation in Virtual Environments,” in *Conference on Human Factors in Computing Systems*, 1999, vol. 99, no. May, p. 8.

[34] P. Foo, W. H. Warren, A. Duchon, and M. J. Tarr, “Do humans integrate routes into a cognitive map? Map- versus landmark-based navigation of novel shortcuts.,” *J. Exp. Psychol. Learn. Mem. Cogn.*, vol. 31, no. 2, pp. 195–215, Mar. 2005.

[35] I. Moar and G. H. Bower, “Inconsistency in spatial knowledge,” *Mem. Cognit.*, vol. 11, no. 2, pp. 107–13, Mar. 1983.

[36] B. J. Scholl, “Objects and attention: the state of the art.,” *Cognition*, vol. 80, no. 1–2, pp. 1–46, Jun. 2001.

[37] K.,Vasylevska, H., Kaufmann, M., Bolas, E. A., Suma “Flexible spaces: dynamic layout generation for infinite walking in virtual environments”. in *IEEE 3D User Interfaces*, pp. 39-42 (2013) doi:10.1109/3DUI.2013.6550194