An Integrated Design of World-in-Miniature Navigation in Virtual Reality

Samuel Truman and Sebastian von Mammen
samuel.truman@stud-mail.uni-wuerzburg.de,sebastian.von.mammen@uni-wuerzburg.de
Games Engineering, Julius-Maximilians-University
Würzburg, Germany

ABSTRACT

Navigation is considered one of the most fundamental challenges in Virtual Reality (VR) and has been extensively researched [11]. The world-in-miniature (WIM) navigation metaphor allows users to travel in large-scale virtual environments (VEs) regardless of available physical space while maintaining a high-level overview of the VE. It relies on a hand-held, scaled-down duplicate of the entire VE, where the user's current position is displayed, and an interface provided to introduce his/her next movements [16]. There are several extensions to deal with challenges of this navigation technique, e.g. scaling and scrolling [21]. In this work, a WIM is presented that integrates state-of-the-art research insights and incorporates additional features that became apparent during the integration process. These features are needed to improve user interactions and to provide both look-ahead and post-travel feedback. For instance, a novel occlusion handling feature hides the WIM geometry in a rounded space reaching from the user's hand to his/her forearm. This allows the user to interact with occluded areas of the WIM such as buildings. Further extensions include different visualizations for occlusion handling, an interactive preview screen, post-travel feedback, automatic WIM customization, a unified diegetic UI design concerning WIM and user representation, and an adaptation of widely established gestures to control scaling and scrolling of the WIM. Overall, the presented WIM design integrates and extends state-of-the-art interaction tasks and visualization concepts to overcome open conceptual gaps and to provide a comprehensive practical solution for traveling in VR.

ACM Reference Format:
Samuel Truman and Sebastian von Mammen. 2020. An Integrated Design of World-in-Miniature Navigation in Virtual Reality . In *Foundations of Digital Games (FDG20), September 15–18, 2020, Bugibba, Malta.* ACM, New York, NY, USA, 8 pages.

1 INTRODUCTION

VR refers to a VE, which is mediated to the user via a medium [15]. Usually, a head-mounted display (HMD) is used to present a stereoscopic image to the user. Contemporary HMDs track the user's head, to continuously update this image accordingly. VR is gaining popularity and is becoming increasingly affordable for consumers.

One of the most fundamental research topics regarding VR is navigation, i.e. the user's ability to travel in the VE. Numerous studies such as [18] indicate, that real-world walking or running is generally superior to other locomotion methods, e.g. gaze-directed steering. Real walking is very intuitive, facilitates presence [13, 14, 18], and induces less cybersickness since signals from visual, vestibular, and proprioception senses match [2, 18]. Disadvantages of real walking include slower locomotion compared to other methods and fatigue. Also, a direct mapping of real walking to VR is not always feasible. The VE is often larger than the real environment. The real environment might be restricted e.g. by room size, obstacles, or a limited tracking area. Since direct mapping is not applicable to all situations, there is a need for alternative means of navigation in VR. There are several methods that allow locomotion in large VEs either by direct mapping or indirectly using a metaphor, e.g. steering. However, most locomotion methods only allow relocating to a position in the user's field of view (FOV) or close vicinity.

The WIM navigation metaphor utilizes a scaled-down duplicate of the VE to display and manipulate the user's position and orientation. It, therefore, enables the user to quickly and precisely move to any position in the VE, without restrictions dictated by a line of sight or available physical space in the real environment. This work aims at implementing a miniature model-based navigation method that integrates state-of-the-art research insights and corresponding techniques such as pre-selection of orientation and look-ahead previews.

The result is a highly customizable, extensible WIM implementation which features and integrates the following options presented in preceding scientific works: Scaling and scrolling the WIM, designating a new location by pickup up and repositioning a user representation in the WIM, picking up and carrying the WIM itself, and the ability to render it semi-transparently or partly hidden based on user gaze. Due to our integration efforts, we had to master several design challenges. These include user interaction and user guidance. Therefore, the following features are added. Look-ahead previews can be used to provide additional information on the selected destination. A preview screen can be used to display the target location. Also, a travel preview animation visualizes the travel phase in the WIM. After the travel phase, a path trace effect helps to understand where the new location is in relation to the previous one. An additional occlusion handling strategy partly hides the WIM based on hand movement. A destination can be alternatively specified using a direct selection method based on pointing. Furthermore, the WIM can be grabbed from a distance, automatically generated, automatically adapted to user height and

FDG20, September 15–18, 2020, Bugibba, Malta

arm length, and instantiated anew. These new features are experimental, their effectiveness is yet to be confirmed by appropriate studies.

State-of-the-art research work that was built upon in our project is presented in the next section. We then continue, in Section 3 with our design decisions and the actual implementation. The results, including a reasonable standard configuration are discussed in Section 4. The work is summarized, and possible future work is outlined in Section 5.

2 RELATED WORK

The WIM navigation technique [16] is an object manipulation and locomotion metaphor. In its original form, the WIM is a handheld, scaled-down duplicate of the entire VE. The user can turn the WIM to spectate the VE from any direction. Therefore, the WIM provides a third-person point of view in addition to the first-person perspective which is common in most VR applications by mapping the HMD to the camera orientation and translation. The WIM technique has huge potential for a wide range of use-cases in VEs and was already successfully applied to Augmented Reality [1], architecture [3], astronomy [10], and medicine [4].

WIM can be utilized as an object editing interface, i.e. the user can manipulate (e.g. scale) objects in the VE using the WIM. Changes to the WIM objects will be simultaneously applied to the full-scale objects in the VE. Likewise, changes to full-scale objects will also be applied to WIM objects. Utilizing the WIM, users can modify objects which aren't within their reach. Moreover, the WIM can also be used as navigation method.

This work focuses on the WIM as navigation method. The user is represented in the WIM by a doll as in [12]. Changes (e.g. translation) to the doll will be applied to the user's viewport. As opposed to some other methods, not only a target destination but also orientation can be specified using the WIM [9]. There are a couple of extensions and improvements to the original WIM technique. Valkov et al. proposed a VE locomotion technique utilizing multitouch input, a Nintendo Balance Board, and the WIM metaphor [19]. Other WIM implementations target specific limitations of the original WIM technique and are presented in the following sections.

2.1 Disorientation

Navigation is generally considered to be the aggregate task of wayfinding and travel [5]. Wayfinding refers to the cognitive part of navigation, while travel refers to the actual motion. The WIM technique provides mechanics to support both aspects of navigation. According to [16], updating the viewport at the same time the user's representation in the WIM is modified is highly disorientating. One consequence of this insight is the realization of the travel phase after a target destination and orientation are confirmed. There are several different strategies to visualize the travel phase. The simplest one instantly places the user at the destination with the specified orientation. However, this will also foster disorientation, because continuous spatial updates are missing [20]. To provide continuous spatial updates, the user could be moved to the destination by either a continuous or slow in/slow out animation. Stoakley et al. choose the latter [16]. However, using an animation is likely to induce cybersickness because the user has no direct control over the movement. Sensory information from vision, vestibular sense, and proprioception will be in conflict [7, 9]. To reduce the likelihood of causing cybersickness while still providing spatial updates, the travel phase is often visualized by moving the viewport very quickly towards the destination [9]. Pausch et al. additionally identified the issue that users had to shift their focus between the full-scale VE and the WIM [12]. Therefore, they proposed a visualization method that does not require the user to refocus. Once the destination is confirmed, the WIM starts to grow until it replaces the full-scale VE. The user takes the place of his or her representation in the WIM and thus is now located at the specified destination. The old WIM replaces the full-scale VE and is thus becoming the new full-scale VE. A new WIM is then instantiated.

2.2 Input Devices

The original WIM was interacted with by means of a tracked clipboard and two buttons attached to a tracked tennis ball [16]. According to [16], these props were used to provide haptic feedback to the user which was especially important while rotating the WIM. These props, however, introduced two problems. First, holding the props for a prolonged period of time introduced fatigue. Stoakley et al. therefore provided a mechanic to detach the WIM from the user's arm [16]. Once detached, the WIM floats in front of the user until it's attached to the user's arm again. To prevent fatigue, we designed the WIM such that it is not attached to the user's arm, but floats in space. It can be grabbed or instantiated in front of the user by pressing a button. Second, the physical objects often presented an obstruction, e.g. preventing the user from selecting the desired destination. This issue was solved by temporarily detaching the WIM from the clipboard [16]. In our test setup, we used the Oculus Quest touch controllers as input devices.

2.3 Occlusion

Another challenge every WIM implementation must deal with is occlusion. Since the WIM is a scaled-down version of the VE, there will almost certainly be obstacles preventing the user from seeing all parts of the WIM. For example, walls, furniture, and multiple floors might obstruct the user's vision in the case of indoor areas. Stoakley et al. mitigated occlusion issues using backface-culling, thus drawing walls and the ceiling only from one side so that the user can look inside the building [16]. The original WIM use-case was a single-story building. Diepstraten et al. proposed a set of approaches to create cut-away illustrations so that the user can look inside opaque objects [6]. Trueba et al. implemented a WIM utilizing a cut-away view and compared it to a semi-transparent WIM [17]. Users performed better with the cut-away approach. The most likely reason is that although the semi-transparent WIM allowed the user to look inside, occluding walls were still visible, thus contributing to visual clutter. Chittaro et al. created a desktop application in which a WIM can be used to navigate a multistory virtual building utilizing different perspectives [3]. They used transparency to only show the floor important to the user. Floors below the floor of interest are occluded since the floor of interest is displayed fully opaque, while floors above the floor of interest are fully transparent. In cases where multiple floors might be important to the user, those floors are displayed semi-transparent.

2.4 Scalability

The WIM lacks a scaling mechanism [9]. Such a mechanism might be useful to quickly change the level of scale, e.g. to switch from navigating a house to navigating a city or state [21]. Similarly, a major challenge of the WIM technique is that it doesn't scale well due to discrepancy in dimensionality of the actual VE and the WIM model [9]. Using the WIM technique for a very large VE would require an appropriately small scale factor. This, however, would dramatically reduce destination selection accuracy. There are some extensions to the WIM technique mitigating this issue either by providing the ability to scale the WIM or by representing only an excerpt of the VE in the WIM. LaViola et al. introduced some drastic changes to the original WIM technique by projecting the WIM on the floor of a CAVE system [8]. The user can interact with the WIM by means of foot gestures. To show or hide the WIM, the user can click his toes together. Similarly, the user can click his heels together to enter or exit scale mode. While in scale mode, the user can walk forward or backward to scale up or down, respectively. Alternatively, the user can stand up on tiptoes or crouch to scale the WIM. The user can walk to the desired destination in the WIM while it is visible. Afterwards, he will be moved to the specified destination in the full-scale VE. Li et al. used a scalable WIM technique to navigate simulations of the astrophysical universe in a desktop application [10]. The WIM is also used to give an overview of the entire observable universe by means of logarithmic scale mapping. Chittaro et al. provided sliders to adjust their WIM's size, position, and x-axis rotation, respectively [3]. As opposed to the previous three approaches, [17] created an algorithm to automatically select the section from the WIM which is most important to the user. Therefore, the part of the WIM model relevant to the user can be displayed at a reasonable scale independent of the VE's total size. Utilizing a WIM in combination with a multi-touch table, [4] created a method to navigate volumetric data e.g. for medical use-cases. The WIM provides, thereby, an overview of the entire dataset. Using gestures on the table, a slice of the dataset can be selected and is then displayed on the table. Finally, [21] proposed a WIM which is both scalable and scrollable and thereby enabling the user to operate on multiple levels of scale. Using the approaches presented in this section the scale and scalability issues imposed by the original WIM technique can be dealt with. In the next section, we discuss our WIM design that combines state-of-the-art approaches and introduces novel aspects that became apparent when pursuing their integration.

3 METHODOLOGY

This section describes the individual WIM features and extensions that are implemented.

3.1 Using the WIM Technique

The WIM is mainly interacted with by means of hand gestures (the Oculus Quest along with the corresponding Oculus touch controllers were used). Table ?? provides an overview of all interaction tasks. The individual interactions are further explained in their respective sections. Figure 1 provides an overview of all techniques that are merged in our implementation (we used Unity engine) [1].

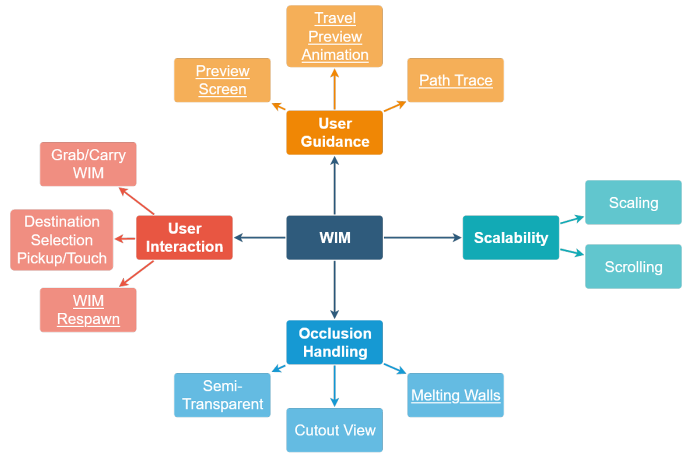

Figure 1: WIM navigation challenges and respective technical solutions. We did not find the underlined features mentioned in preceding works.

The user's current position in the VE is indicated by a red capsule in the WIM, at all times. This corresponds to the red dot which commonly marks the spectator's perspective on maps. Additionally, the red capsule indicates the user's orientation in the VE. The user can interact with the WIM object to change e.g. the view angle or visible excerpt of the WIM. This can be useful for example to assist with orientation and path-planning tasks. More on how the WIM can be interacted with is described in sections 3.2 and 3.3. To use the WIM for locomotion, the following four steps are required. First, the desired target position and orientation need to be selected utilizing the WIM. At this point, visual aids can be useful to provide look-ahead previews. This step may be repeated until the user is satisfied with the selected destination. Second, the selected destination must be confirmed. Third, actual travel takes place. Several alternative travel visualization strategies were introduced in section 2.1. Finally, the user might be provided with visual aids to help with reorientation at the new location. Although the least obvious, this step might have a significant impact on user experience. In the next sections approaches to mitigate the WIM limitations introduced in section 2, i.e. occlusion, are presented.

3.2 User Interaction

Intuitive user interaction is vital to any VR navigation method. Therefore, this WIM implementation provides a set of techniques to enhance usability. The WIM size can be automatically adapted to the user's height. It will be increased in size for taller users and vice versa. As explained in section 2.2, some usability issues can be mitigated by temporarily detaching the WIM from the user's hand. To avoid these sorts of issues, the WIM floats in space independent from the user by default. The user can grab the WIM to translate and rotate it (table ??). There is also an option to allow grabbing the

[1] A video of all the integrated WIM-design's features in action can be watched here: https://youtu.be/6dS7TTdePsw

Interaction task	Gesture/button
Pickup/carry WIM	Hold Grab button (squeeze hand); release to let go
Distance grab	Point at WIM using index finger and hold Grab button
Translate/rotate WIM	Move/turn hand while carrying WIM
Scale WIM	Grab with both hands and spread/narrow hands
Scroll WIM (manually)	move thumbstick; optionally use second thumbstick to scroll vertically
Respawn WIM	Press Respawn button
Select destination	Pickup destination selection: pickup player representation using index and thumb fingers, and drop at destination Direct selection: touch destination with index finger and press Confirm button
Move/rotate destination	Pickup destination selection: pickup and reposition destination indicator Direct selection: thumbstick to rotate, re-select new destination to move
Confirm destination	Pickup destination selection: double-tap destination indicator with index finger Direct selection: press Travel button
Open preview screen (manually)	Pickup destination indicator's view frustum using index and thumb fingers; release to let go
Pickup/carry preview screen	Pickup using index and thumb fingers; release to let go
Close preview screen	Touch Close button in the upper right corner
Detect arm length	Fully extend arm and press Confirm button

Table 1. All interaction tasks and their corresponding gesture or button.

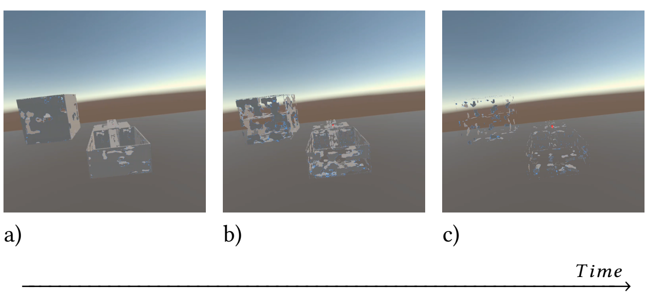

Figure 2: The dissolve effect. Part of the respawn visualization. The WIM is dissolved at the old position and simultaneously resolved at the new position.

WIM from the distance. This requires the WIM to be in the line of sight of the user and within range. Furthermore, there is a function to respawn the WIM. Respawning the WIM is not restricted by range or line of sight. This is especially useful if the user left the WIM somewhere in the VE. The WIM will be spawned at the user's position using an offset specified beforehand. Alternatively, the user's arm length can be detected and used as an offset. The WIM will be respawned to maintain its position and orientation relative to the user after the travel phase and therefore assist with orientation. Respawning the WIM is visualized to facilitate understanding. The WIM is dissolved (fig. 2) at the old position and resolved (fig. 3) at the new one simultaneously.

3.3 Ensuring Scalability

As discussed in section 2.4, scalability is a serious limitation of the original WIM technique. WIM implementations, therefore, must provide mechanisms to ensure scalability.

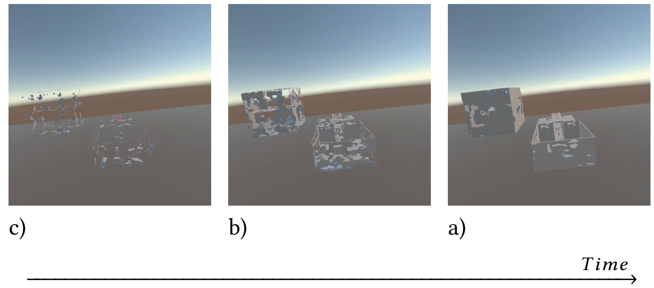

Figure 3: The resolve effect. Part of the respawn visualization.

The WIM in this work deploys the scale and scroll concept by [21]. However, the implementation deviates from the one by [21] in terms of user interaction and visualization. Instead of utilizing the mouse wheel of a wireless mouse to scale and moving towards one edge of the WIM to scroll, a widely known interaction pattern, which is used on touch input devices like smartphones, was adopted. To resize the WIM, the user can grab it using both hands. The WIM can be scaled up by moving the hands apart from each other and scaled-down by bringing the hands closer together. If scrolling is enabled, only an excerpt of the WIM is visible at a time. The user can scroll the WIM to change the visible excerpt. There are two possible scrolling modes. The automatic scrolling mode is useful in combination with the direct selection method described in Section 3.5.1. If a destination is selected, the visual excerpt will be centered on the destination. The visual excerpt will be centered on the user representation otherwise. Using the automatic scrolling mode requires no additional user interaction. It is therefore very easy to use. The manual scrolling mode is useful in combination with the pickup destination selection method described in Section 3.5.2. Using the manual scrolling mode, the user can use a thumbstick on the Oculus touch controllers to move the visual excerpt of the WIM. This mode is advantageous because the user can observe an area of the WIM without the need to walk there or select a destination in the proximity of the area. Since both manual scrolling and specifying the destination rotation using the direct selection method are controlled by use of a thumbstick, we decided not to use them at the same time. We present in the next section the WIM visualization strategies to deal with occlusion.

3.4 Occlusion Handling Strategies

To travel, the user must first select a destination. Therefore, the user sometimes needs to be able to see inside opaque structures like buildings. A few occlusion handling strategies were introduced in Section 2.3. This WIM implementation includes three occlusion handling strategies as well as the option not to use any.

3.4.1 Transparency. The first occlusion handling strategy is to draw the WIM semi-transparently. This strategy is very easy to implement and yet preliminary tests indicate that it is quite beneficial to deal with occlusion in small VEs. However, a more complex VE might render this strategy impractical due to visual clutter. Transparency can be used together with other occlusion handling strategies.

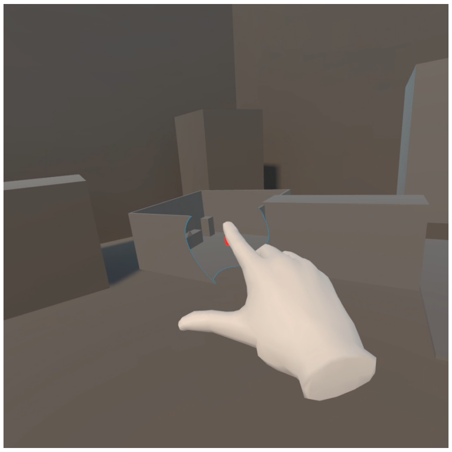

Figure 4: The "melting walls" occlusion handling strategy hides those parts of the WIM that are currently in close proximity to the user's hand. The same visualization is used for the cutout view strategy.

3.4.2 Melting Walls. The second strategy is called "melting walls". As soon as the user's hand approaches the WIM, those parts of the WIM in close proximity to the hand start to fade (fig. 4). When the hand is moved elsewhere, the entire WIM is visible again. In the first implementation, the proximity was defined using a sphere located at the index finger. This approach, however, leads to issues when the user tried to select a destination deep inside the WIM because the walls closest to the user started reappearing as they were no longer near the user's index finger. In the current implementation, the proximity is defined using a cylinder to prevent those kinds of issues. The cylinder extends from the user's hand to the user's upper arm (fig. 5).

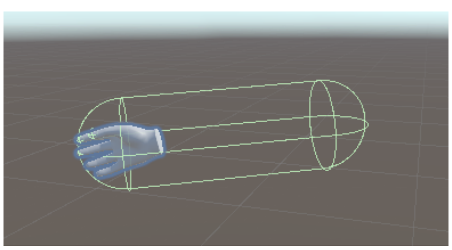

Figure 5: The collider which defines the proximity used by the "melting walls" occlusion handling strategy.

3.4.3 Cutout View. The third occlusion management strategy is a cutout view similar to those described in Section 2.3. However, contrary to some existing implementations, this strategy does not hide entire faces of geometries, for example walls, which are occluding e.g. a room. Instead, the same technique used for "melting walls" is in use, i.e. geometries which are within proximity of an area defined by a cylinder are clipped (fig. 4). As opposed to the "melting walls" technique, the cylinder is attached to the user's virtual head in the VE. This implies, that the cutout view technique is limited in range. A limited range allows the user to select what geometries should be hidden by adapting the distance and angle to the WIM accordingly. Thereby, also destinations deep inside the WIM can also be selected, as with the "melting walls" strategy.

With an occlusion handling strategy in place, the user is able to select destinations even in normally occluded areas. The process of selecting a destination and orientation is described in the next section.

3.5 Destination Selection

There are two alternative methods to select a desired position and orientation for travel.

3.5.1 Direct Selection. The first destination selection method is direct selection. Using direct selection, the user can point at any location in the WIM using his or her index finger and mark the location as a destination using a button on the Oculus touch controller. A blue destination indicator will be instantiated at the selected location in both the WIM and full-scale VE (fig. 6). The destination indicator's orientation thereby represents the selected orientation at the destination, i.e. the user's view orientation will match the destination indicator's orientation after the travel phase. This visualization is similar to the red user representation indicating the user's current position and orientation in the VE. To increase orientation visibility, the destination indicator in the WIM is augmented with a semi-transparent view frustum. The destination indicator's initial orientation will match the index finger's forward direction. The orientation can then be changed using a thumbstick on the Oculus touch controller. This two-step position and orientation selection process can be seen in some implementations of the teleport navigation method, e.g. in the Oculus sample framework for Unity. The destination can be confirmed by pressing another button. A disadvantage of the direct selection method might be the inconsistency in interaction. The WIM is grabbed like a physical object, but the destination is selected by pointing with the index finger and the orientation is changed by means of a thumbstick. A more consistent alternative to direct selection is the pickup method, which is explained next.

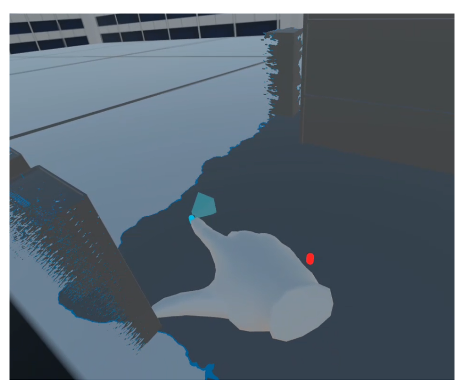

Figure 6: A destination was selected in the WIM using direct selection. The blue destination indicator provides a preview of the user's position and rotation after travel.

3.5.2 Pickup. The pickup method is conceptually similar to the original WIM locomotion by [16]. However, there are some significant differences, especially in user interaction. It was our goal to provide an input modality which is as intuitive as possible. We therefore refrained from using abstract input metaphors such as buttons or axis input. Instead, we wanted to utilize natural hand gestures. The user can instantiate a destination indicator, identical to the one used with direct selection, by grabbing the red user representation in the WIM. Therefore, the user must pinch the index finger and thumb. The destination indicator will be picked up, while the user's representation stays in place. The user can then place the destination indicator anywhere in the WIM. To change the orientation, the user must turn his or her hand accordingly. The destination indicator can be dropped and picked up again to change the selected destination or orientation. Alternatively, the user can pull another destination indicator out of the red user representation in the WIM. In this case, the old destination indicator will disappear. Figure 7 illustrates the pickup destination selection process. To confirm the destination, the destination indicator in the WIM must be double-tapped.

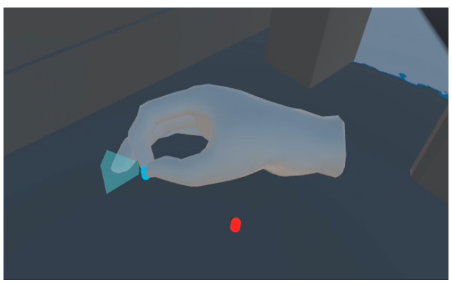

Figure 7: The user can pick up the red user representation in the WIM and place a destination indicator to select a target for the travel phase.

Using any of the two destination selection methods, the user could in theory travel to any location in the VE, including a location in mid-air. There is, however, an option to automatically snap the destination indicator to the floor and set its rotation to be perpendicular to the floor. Refraining from using this option might cause cybersickness or disorientation. Another possible destination selection method might be utilizing a laser pointer to select a destination. The process of selecting a destination and orientation can be repeated indefinitely until the destination is confirmed, and the travel phase begins. Once a destination is selected, the user can be presented with visual cues to provide information on the selected destination in order to mitigate disorientation after the travel took place.

3.6 Look-Ahead Previews

As identified in section 2.1, disorientation is a major issue of the WIM technique. Luckily, there are several design approaches to mitigate it. Visual aids can be employed at three stages of the navigation process described in section 3.1: before travel, during travel, and after travel. Some good mechanics to provide spatial updates without inducing too much cybersickness already exist and were presented in section 2.1. Therefore, this work focuses on the pre- and post-travel stages. Both the red user representation and destination indicator in the WIM can already be considered to be visual cues to mitigate disorientation. Two additional pre-travel aids are presented in this section. A method to help maintain orientation is presented in the next section.

3.6.1 Preview Screen. The preview screen is a 2D panel that floats in the VE and always faces the user (fig. 8). On the preview screen, the user can see a live preview of what the currently specified destination looks like, i.e. a continuously updated image showing exactly what the user will see once translated and rotated to the specified location and orientation. There are two different preview screen modes.

When the automatic mode is active, the WIM will be automatically displayed while a destination is selected. The preview screen will float above the WIM using a consistent offset. There are some disadvantages to using the automatic mode. To look at the preview screen, the user has to look up from the WIM. This means the user has to shift his or her focus from the WIM to the preview screen back and forth when selecting a destination, looking at the preview screen, modifying the destination, etc. Also, the user might not notice the preview screen appearing when a destination is selected. The pickup mode provides a more natural interaction while mitigating these problems.

The pickup mode works very similarly to the pickup destination selection method described in section 3.5.2. As described in section 3.5.1, the destination indicator's view frustum is displayed in the WIM. The user can grab the view frustum by pinching the index finger and thumb. The preview screen will become visible and attached to the user's hand. Afterwards, the user can position the preview screen anywhere in the VE. As soon as the preview screen is dropped, it will retain its position relative to the WIM. The preview screen can be closed by pressing a Close button which is displayed in the upper right corner of the preview screen. In a previous version it was closed by double-tapping it. This interaction was unintuitive and prone to error. The preview screen is rather narrow, making it easy to accidentally trigger a second tap when pulling one's finger back.

3.6.2 Travel Preview Animation. Another visual pre-travel aid is the travel preview animation. The user's current position is marked by the red user representation in the WIM. A blue user representation is used to indicate the destination in both WIM and full-scale VE. The travel preview animation adds a visual path between the user's current position and selected destination in the WIM. Displaying the path to a selected destination in either a minimap or the full-scale VE is a field-tested visual aid commonly used in car navigation systems. A semi-transparent user representation is animated to move along the path, thus providing a preview of the travel phase (fig. 9). The travel preview animation was extended to not only show a preview of the translation but the rotation as well. A Similar visualization can be used to provide post-travel visual cues.

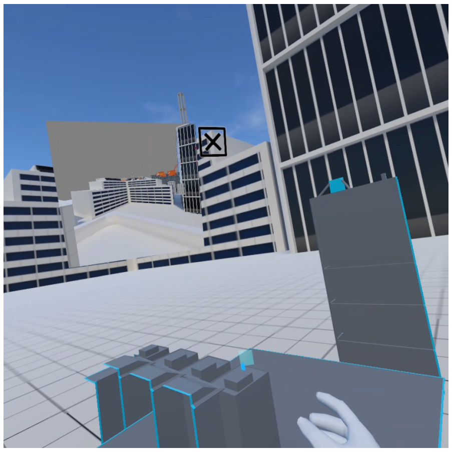

Figure 8: The preview screen floats in the VE providing a look-ahead preview of the specified destination.

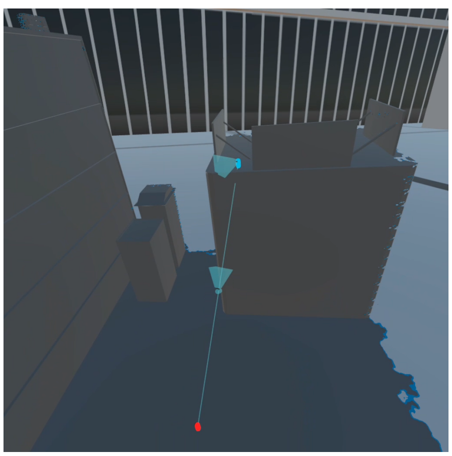

Figure 9: The travel preview animations provide a preview of the travel phase.

3.7 Post-Travel Feedback

After the travel phase, users might have trouble to orient themselves at the new position and orientation. The path trace visualization aid can be used to mitigate this issue.

3.7.1 Path Trace. If the path trace option is enabled, a line from the previous position to the new position is displayed in the WIM, visualizing the locomotion that just took place. The path trace is faded-out over time, from the previous position towards the new one. In the next section, the individual features presented in this section are discussed, and a standard configuration is proposed.

4 DISCUSSION

The WIM implementation developed for this work provides several mechanics to mitigate the limitations imposed by the WIM metaphor. Some mechanics exclude each other, some are complementary to each other. The set of mechanics enabled in the default configuration and the corresponding reasons are discussed in this section. The goal of the default configuration is to be applicable to most use-cases. Therefore, the default mechanics and settings are chosen to comply with the following two principles: First, ease of use is favored over immersion. Second, intuitive interactions are favored over efficiency in terms of speed.

To avoid fatigue, the WIM is detached from the user by default so that the user is not required to lift his or her arm to see the WIM. However, the WIM can be picked up, moved and rotated at any time. Default values for arm length and user height are provided to fit most adult users. The properties can be easily modified to adapt the application to a specific target group. Automatic adaption to the user can be enabled. This is disabled by default so that developers are in full control of the user experience. Distance grabbing is disabled by default because informal testing did not indicate, that the benefit outweighs the added complexity.

Scaling is enabled to allow users to quickly adapt the WIM to favor precise destination selection or getting a better overview of the VE depending on the situation. Scrolling, however, is disabled by default. Many current VR applications do stick to a very limited VE size, partly due to performance issues of contemporary VR hardware. Therefore, providing better orientation by depicting the entire VE in the WIM can currently be considered more useful to most applications. The default scrolling mode is manual scrolling. Manual scrolling allows users to observe an area of the WIM without the need to walk there or select a destination in the proximity of that area. Also, manual scrolling works better in combination with the pickup destination selection method, which is the standard method.

Although it introduces visual clutter, the default occlusion handling strategy is the semi-transparent WIM. No user interaction is required, which renders transparency the easiest-usable occlusion handling strategy. In situations in which transparency cannot be used, melting walls should be favored over the cutout view strategy. Informal tests indicate that melting walls is more comfortable to use compared to cutout view, because it allows occlusion handling independent from gaze direction and head movement. The pickup destination selection is the default method instead of direct selection because it favors intuitive and consistent user interactions as opposed to speed.

Both look-ahead previews, preview screen, and travel preview animation are enabled by default because they provide powerful visual aids. The default preview screen mode is pickup, as it's consistent with other user interactions and mitigates issues introduced by the automatic mode. Finally, the path trace mechanic is enabled, because its minimalistic visualization barely adds complexity but yields a valuable visual cue. The contributions of this work are summarized, and a brief overview of possible future work is provided in the next section.

5 CONCLUSION

5.1 Summary

In this work, we give an in-depth overview of the World-in-Miniature (WIM) navigation metaphor along with its limitations and state-of-the-art implementations to mitigate these limitations. Furthermore, we present an integrated WIM metaphor implementation to improve on the current-state-of-the-art by adopting field-tested best practices from both miniature-based and unrelated navigation techniques.

The limitations of early WIM implementations can be summarized as occlusion, disorientation, scalability, and usability. The process of creating the WIM is automated to enhance usability for developers and make utilization of the WIM navigation method more accessible. A set of default settings useful for most use-cases is provided. The implementation is customizable to fit a specific user's needs. Up to a certain point, this customization can be automated, e.g. by probing the user's height and arm length to adapt the generated WIM's size and position the user. A respawn mechanic is introduced to the WIM metaphor to allow quick access in every situation. The WIM is interacted with using a set of intuitive gestures known from everyday life domains such as smartphones. The WIM can be grabbed like physical objects. Equally, the user sees himself represented in the WIM and can move and reorient his representation naturally. As a shortcut, a target can be selected by touching a location with the index finger.

Scale and scroll options are provided to ensure usability even in very large VEs. Therefore, the technique by [21] is adopted. The input modalities based on a wireless mouse are replaced by hand gestures. The user can grab the WIM and stretch and condense it using both hands. Scrolling is either initiated automatically depending on the user's position and the selected destination or manually controlled using a thumbstick. Manually scrolling works particularly well in combination with the pickup destination selection, due to consistent interaction modalities.

Three occlusion handling strategies are provided, which are transparency, melting walls, and cutout view. Melting walls is a new strategy similar to the cutout view presented by [17]. However, instead of hiding objects based on gaze-direction, a shader is used to clip geometry within range of the user's arm. To this end, a capsule-shaped area aligned with the user's arm is used to define what should be clipped. The cutout view is limited in range to enable users to only hide walls within a chosen range. To compensate for disorientation which might be introduced by the travel phase, both pre-travel and post-travel visual aids are provided. The WIM metaphor is therefore extended by a travel preview animation as well as a preview screen which can be grabbed and positioned by the user. The preview screen displays a livestream of the currently selected target position. Also, a path trace visualization is used as visual aid after travel.

5.2 Future Work

Some possible improvements to the WIM implementation presented in this work were already mentioned in the respective sections. However, further promising future work is outlined in this section.

An interesting extension to the WIM metaphor would be the utilization of machine learning algorithms to identify and categorize geometry in the full-scale VE, to provide automatic abstractions or stylized representations of the full-scale VE in the WIM. Multiple levels of abstractions, and therefore multiple levels of details, could be toggled automatically or manually by the user situation-dependently. We would very much like to see this WIM implementation put to use. Studies evaluating the usefulness of the added features would be particularly interesting.

REFERENCES

[1] Blaine Bell, Tobias Höllerer, and Steven Feiner. 2002. An annotated situation-awareness aid for augmented reality. https://doi.org/10.1145/571985.572017

[2] Sarah S. Chance, Florence Gaunet, Andrew C. Beall, and Jack M. Loomis. 1998. 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* 7, 2 (1998), 168–178. https://doi.org/10.1162/105474698565659

[3] L. Chittaro, V. K. Gatla, and S. Venkataraman. 2005. The Interactive 3D BreakAway Map: a navigation and examination aid for multi-floor 3D worlds. In *International Conference on Cyberworlds, 2005*, Tosiyasu L. Kunii (Ed.). IEEE Computer Society, Los Alamitos, Calif., 8 pp–66. https://doi.org/10.1109/CW.2005.88

[4] Dane Coffey, Nicholas Malbraaten, Trung Le, Iman Borazjani, Fotis Sotiropoulos, and Daniel F. Keefe. 2011. Slice WIM: a multi-surface, multi-touch interface for overview+detail exploration of volume datasets in virtual reality. https://doi.org/10.1145/1944745.1944777

[5] Rudolph Darken and Barry Peterson. 2015. Spatial Orientation, Wayfinding, and Representation. In *Handbook of virtual environments*, Kelly S. Hale and Kay M. Stanney (Eds.). Human factors and ergonomics series, Vol. 20143245. CRC Press, Boca Raton, Fla., 467–491. https://doi.org/10.1201/b17360-24

[6] J. Diepstraten, D. Weiskopf, and T. Ertl. 2003. Interactive Cutaway Illustrations. *Computer Graphics Forum* 22, 3 (2003), 523–532. https://doi.org/10.1111/1467-8659.t01-3-00700

[7] Joseph J. LaViola. 2000. A discussion of cybersickness in virtual environments. *ACM SIGCHI Bulletin* 32, 1 (2000), 47–56. https://doi.org/10.1145/333329.333344

[8] Joseph J. LaViola, Daniel Acevedo Feliz, Daniel F. Keefe, Robert C. Zeleznik, et al. 2001. Hands-free multi-scale navigation in virtual environments. *SI3D* 1 (2001), 9–15.

[9] Joseph J. LaViola, Ernst Kruijff, Ryan P. McMahan, Doug A. Bowman, and Ivan Poupyrev. 2017. *3D user interfaces: Theory and practice* (second edition ed.). Addison-Wesley, Boston.

[10] Yinggang Li, Chi-Wing Fu, and Andrew J. Hanson. 2006. Scalable WIM: effective exploration in large-scale astrophysical environments. *IEEE Transactions on Visualization and Computer Graphics* 12, 5 (2006), 1005–1011. https://doi.org/10.1109/TVCG.2006.176

[11] Niels Christian Nilsson, Tabitha Peck, Gerd Bruder, Eri Hodgson, Stefania Serafin, Mary Whitton, Frank Steinicke, and Evan Suma Rosenberg. 2018. 15 Years of Research on Redirected Walking in Immersive Virtual Environments. *IEEE Computer Graphics and Applications* 38, 2 (2018), 44–56. https://doi.org/10.1109/MCG.2018.111125628

[12] Randy Pausch, Tommy Burnette, Dan Brockway, and Michael E. Weiblen. 1995. Navigation and locomotion in virtual worlds via flight into hand-held miniatures. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. 399–400.

[13] Roy A. Ruddle and Simon Lessels. 2009. The benefits of using a walking interface to navigate virtual environments. *ACM Transactions on Computer-Human Interaction (TOCHI)* 16, 1 (2009), 5. https://doi.org/10.1145/1502800.1502805

[14] Mel Slater, Martin Usoh, and Anthony Steed. 1995. Taking steps: the influence of a walking technique on presence in virtual reality. *ACM Transactions on Computer-Human Interaction (TOCHI)* 2, 3 (1995), 201–219. https://doi.org/10.1145/210079.210084

[15] Jonathan Steuer. 1992. Defining Virtual Reality: Dimensions Determining Telepresence. *Journal of Communication* 42, 4 (1992), 73–93. https://doi.org/10.1111/j.1460-2466.1992.tb00812.x

[16] Richard Stoakley, Matthew J. Conway, and Randy Pausch. 1995. Virtual reality on a WIM: interactive worlds in miniature. In *CHI*, Vol. 95. 265–272.

[17] Ramón Trueba, Carlos Andujar, and Ferran Argelaguet. 2009. Complexity and Occlusion Management for the World-in-Miniature Metaphor. In *Smart graphics*, Andreas Butz (Ed.). Lecture Notes in Computer Science, Vol. 5531. Springer, Berlin, 155–166. https://doi.org/10.1007/978-3-642-02115-2{_}13

[18] Martin Usoh, Kevin Arthur, Mary C. Whitton, Rui Bastos, Anthony Steed, Mel Slater, and Frederick P. Brooks Jr. 1999. Walking> walking-in-place> flying, in virtual environments. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. 359–364.

[19] Dimitar Valkov, Frank Steinicke, Gerd Bruder, and Klaus H. Hinrichs. 2010. Traveling in 3d virtual environments with foot gestures and a multi-touch enabled wim. In *Proceedings of virtual reality international conference (VRIC 2010)*. 171–180.

[20] D. A. Bowman, D. Koller, and L. F. Hodges. 1997. Travel in immersive virtual environments: an evaluation of viewpoint motion control techniques. In *IEEE 1997 Virtual Reality Annual International Symposium*. IEEE Computer Society Press, Los Alamitos, Calif, 45–52. https://doi.org/10.1109/VRAIS.1997.583043

[21] C. A. Wingrave, Y. Haciahmetoglu, and D. A. Bowman. 2006. Overcoming World in Miniature Limitations by a Scaled and Scrolling WIM. In *3DUI 2006*. IEEE, Piscataway, N.J., 11–16. https://doi.org/10.1109/VR.2006.106