

Collaborative Interaction in Large Explorative Environments

Jason W. Woodworth

University of Louisiana at Lafayette

jwww7675@louisiana.edu

David Broussard

University of Louisiana at Lafayette

broussdm@gmail.com

Christoph W. Borst

University of Louisiana at Lafayette

cwborst@gmail.com

ABSTRACT

Building collaborative VR applications for exploring and interacting with large or abstract spaces presents several problems. Given a large space and a potentially large number of possible interactions, it is expected that users will need a tool selection menu that will be easily accessible at any point in the environment. Given the collaborative nature, users will also want to be able to maintain awareness of each other within the environment and communicate about what they are seeing or doing. We present a demo that shows solutions to these problems developed in the context of a collaborative geological dataset viewer.

CCS CONCEPTS

- Human-centered computing → Virtual reality; *User interface design*

KEYWORDS

collaborative virtual reality, menu interaction

ACM Reference Format:

Jason W. Woodworth, David Broussard, and Christoph W. Borst. 2019. Collaborative Interaction in Large Explorative Environments. In *Symposium on Spatial User Interaction (SUI '19), October 19–20, 2019, New Orleans, LA, USA*. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3357251.3360017>

1 INTRODUCTION

We present a demonstration of several techniques developed to mitigate problems that arose during development of a collaborative VR geological dataset viewer. In the application, pictured in Figure 1, we wanted remotely located geologists to be able to explore a large dataset (in this case an SRTM scan of the Chicxulub Impact Crater, described elsewhere [4]), and collaborate on interpreting and annotating the data. This presented two main design challenges: designing a menu interface and promoting collaborative awareness.

First, users of VR software that allow a large variety of tasks often require a menu system for selecting various tools to perform the tasks [2]. Common solutions often include integrating the menu into the environment or having a menu button summon an abstract in-world menu at a position near the user [3]. During early development stages, we inferred that static menus placed in a large virtual world could be lost or forgotten as the user moves around. We also consider that putting a large number of tools on a menu near the user's hands could easily become cluttered and hard to manage. Our

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SUI '19, October 19–20, 2019, New Orleans, LA, USA

© 2019 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-6975-6/19/10.

<https://doi.org/10.1145/3357251.3360017>

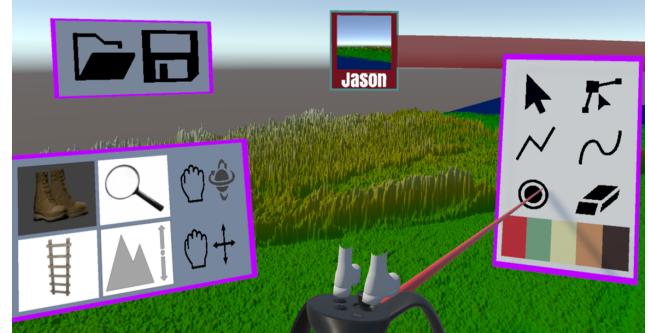


Figure 1: The spherical dashboard interface. Several widgets allow the user to change the current interaction context. The model of shoes over the controller's analog stick indicates that the standard walking movement tool is selected. A name tag for the remote user is placed on the dashboard, displaying the user's name and an image of what they currently see. A leading indicator connects it to the user's avatar to let users know where other users are at all times.

solution was to create a “spherical dashboard” containing multiple tool widgets floating in front of the user that would follow the user and rotate to keep it in view when the user needs it.

Second, users need tools for communicating with each other across the large environment and for maintaining awareness of other users when they are not in view. Basic tools like voice chat alone are insufficient to keep users aware of each others' locations and current tasks in the environment. Users quickly separate and are unable to find each other, compared to smaller environments where users can more easily find each other and see what others are doing. Several techniques have been developed to guide users to offscreen targets in a 2D environment, but this has not received as much attention in 3D virtual worlds [5]. Our solution involves a widget on the user's spherical dashboard with a tether connecting it to the remote users.

The demo will allow users to connect in VR with an experienced user to try out our proposed interaction methods.

2 INTERFACE DETAILS

2.1 Menu Interaction in a Large Environment

We consider that when users are expected to interact with the environment in many ways, we need to give them ready access to various tools. An initial solution relied on world-space menu panels and widgets to handle interaction and tool selection; a flat menu panel for selecting tools could be placed on the environment where the user was looking, scaled to be viewable from any distance, and a scale widget (similar to those commonly found in game engines or

3D modelers) could be placed on the environment and manipulated to scale the environment. Though these allowed for more tools to be used, initial tests showed that users would place the menu once, turn awkwardly to face it after they had moved from that spot, and eventually forget its location, even after being reminded that they could move the menu.

To minimize problems with world-space placement, we settled on a dashboard-style approach that positions tools as widgets on the surface of a large invisible sphere centered on the user's head. To keep it constantly in front of the user, an angular threshold from the center of the dashboard is noted by two bars on the horizontal edges; once a user rotates their head past this threshold, the entire dashboard rotates to catch up with them. Rotation is locked to the world's vertical axis to allow users to look down to survey the environment below them without obstruction, then look up to select tools. This effectively creates a stable usable area of the dashboard which is always in front of the user when they look up.

To handle having many tools available to the user while maintaining ease of use, we enable only a single controller and have each tool widget change the context of a part of the controller when selected. For example, all movement options are delegated to the controller's analog stick, but selecting a movement tool widget changes which type of movement the stick controls. Widgets are grouped into logical tool sets; in our demo application we have six for movement, two for saving and loading, six for annotation tools, and six for changing the dataset. Users can move these sets around the sphere to adjust for different scenarios (e.g. moving them higher up to see another user directly in front of them).

Because we consider users may want to change their movement type more often than other tools, we provide a secondary interaction method. To quickly change movement tools, the user can look at the analog stick on their controller model; once their view direction is within a threshold of the stick, it will display 3D models associated with the movement type (e.g. boots for simple walking, a ladder for vertical movement, etc...) arranged in a line with the current active tool closest to the analog stick. The user can use the stick to scroll through the options, and look away to select the chosen movement type.

2.2 Collaborative Tools

Given the collaborative nature of the application, users will want to stay aware of each other as they explore. Remote users are each represented by a model of a human head, as seen in Figure (2), but they are small compared to the large environment and initial tests showed users can lose each other easily with only voice chat available. As an initial solution, we placed a large "name tag" above their head displaying that user's name and a viewport showing their current view. The name tag scaled as the local user changed distance to keep a static size in view. This became distractingly large when user's were on other ends of the map, and didn't give any indication to what direction they were in when the remote user was not already in view.

To keep the local user more aware of remote users' positions, we placed a name tag with a leading indicator to the user on the local user's dashboard. This name tag similarly shows the remote user's name, along with the viewport showing their view. A thick line

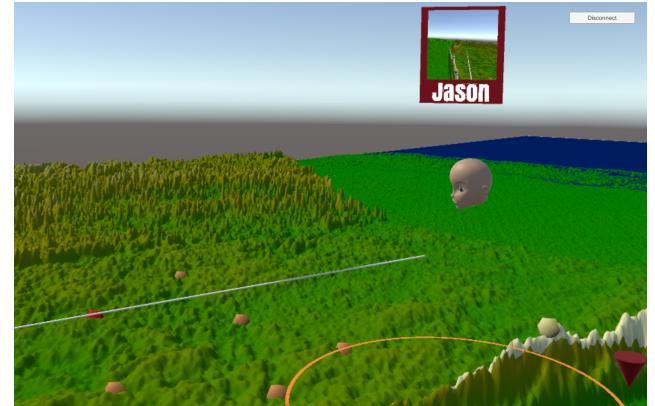


Figure 2: User representation in the application. Remote users are represented by a human head; a name tag floats above them indicating their name and current view.

follows an cubic Hermite curve (as described by Biocca et al. [1]) from this name tag that leads the local user's gaze to the remote user's representation. The line fades when the local user is looking near the correct direction to become less obtrusive when it is less necessary as a guide. To alleviate the need for one user to manually travel to another, the name tag can be selected to immediately warp to that remote user, being placed nearby and rotated to face the point on the environment where the user was looking.

3 CONCLUSION

We are presenting tools and techniques for collaborative interaction in large environments meant to address the difficulty of providing a persistent menu and keeping remote users aware of each other. Our demo will present these in the context of a collaborative geological dataset viewer in which remote geologists can annotate and interpret data. Users of our demo will use these tools to interact with the data with a knowledgeable collaborative user on-site.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. 1815976 and by the Louisiana Board of Regents Support Fund under contract LEQSF(2019-20)-ENH-DE-22.

REFERENCES

- [1] Frank Biocca, Arthur Tang, Charles Owen, and Fan Xiao. 2006. Attention Funnel: Omnidirectional 3D Cursor for Mobile Augmented Reality Platforms. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. ACM, New York, NY, USA, 1115–1122. <https://doi.org/10.1145/1124772.1124939>
- [2] Raimund Dachselt and Anett Hübner. 2007. Three-dimensional menus: A survey and taxonomy. *Computers & Graphics* 31, 1 (2007), 53 – 65. <https://doi.org/10.1016/j.cag.2006.09.006>
- [3] M. M. Davis, J. L. Gabbard, D. A. Bowman, and D. Gracanin. 2016. Depth-based 3D gesture multi-level radial menu for virtual object manipulation. In *2016 IEEE Virtual Reality (VR)*. 169–170. <https://doi.org/10.1109/VR.2016.7504707>
- [4] Gary L Kinsland and Christoph W Borst. 2015. Visualization and interpretation of geologic data in 3D virtual reality. *Interpretation* 3, 3 (2015), SX13–SX20.
- [5] Sylvia Rothe, Daniel Buschek, and Heinrich Hußmann. 2019. Guidance in Cinematic Virtual Reality-Taxonomy, Research Status and Challenges. *Multimodal Technologies and Interaction* 3, 1 (2019). <https://www.mdpi.com/2414-4088/3/1/19>