VR 3D Visualization Interface

Sergio Calabria, Charles Cullen, Bryan Duggan
Dublin Institute of Technology, Aungier Street, Dublin
+353 1 402 3000
sergio.calabria@dit.ie, charlie.cullen@dit.ie, bryan.duggan@dit.ie

Figure 1. The Hardware Setup.

ABSTRACT
Human activities produce large amounts of data, from online transactions to scientific experiments, this vast trove of information is being collected on a daily basis and great efforts are being made to produce tools to analyze it, in an attempt to gather insight and knowledge from this inexhaustible source. The use of 3D techniques to display information, and sorting techniques to arrange it are not new. Node-edge graphs predate computers, but their visualization in real-time, three-dimensional virtual reality is quite recent. Some efforts have been conducted by visualization researchers to verify the validity of these techniques, and some HCI experiments have been carried out in navigating 3D spaces. This paper describes a HCI framework to facilitate engagement in an immersive 3D virtual reality (VR) visualization, by means of a compound control system consisting of existing technologies, for navigation and highlighting, and a purpose built controller for selection and manipulation of data; within a software environment that was built using an existing real time 3D game engine.

CCS Concepts
•Human computer interaction (HCI)→Interaction paradigms→ Virtual Reality
• Interaction Design→Interaction design process and methods→ User interface design

Keywords
Visualization; Interaction; Controls; Interfaces

AVI '16, June 07-10, 2016, Bari, Italy
ACM978-1-4503-4131-8/16/06.
http://dx.doi.org/10.1145/2909132.2926089

1. INTRODUCTION
The goal of this demo is to assess the feasibility of a multimodal interface to access data nodes and interact with visualizations of large data sets in an immersive virtual reality environment. The project is part of ongoing research oriented towards building a framework for an interaction paradigm, to include a navigation control for movement in 3D, gaze interaction to highlight interactive elements, and a custom controller for ad-hoc selection and parameter manipulation. This *fused modality* interface is capable of affording interaction with multiple elements of the visualization, and of providing much sought after haptic feedback to VR interfaces, by means of a tangible navigation controller and a spring loaded mechanism on the manipulation controls, complemented by an intuitive gaze "highlighting" technique.

2. BACKGROUND
Prior research has established the utility of game engines for data analysis, visualization and research [1]. They are particularly convenient in proof of concept work, due to the ease of integrating animations, shaders, and an interface framework. Additionally, the provision of analytics tools in modern game engines can be used to validate new interaction paradigms.

The availability of consumer VR technologies means the possibility of an immersive tool to visualize data and generate insight is not prohibitively expensive. Research in HCI argues for the benefits of tangible interfaces in a VR environment [2], with haptic feedback constituting an unsolved problem in this field. Gesture and gaze interfaces are another interesting direction to explore [3], but current hardware and UI paradigms have been found to lack in precision for parameter manipulation. Research suggests that navigation in 3D is best served with a dedicated controller [4].

Table 1. Current interaction approach

INTERACTION LAYER	GAZE + MOD SELECTION	PITCH-BEND FUNCTIONS
At Graph	Light source	Light Azimuth / Intensity
	Graph Base	Rotate Graph
At Cluster	Boundary	Alpha / Color
	Edge	Go to Node / Bundling
At Node	Mesh	Rotate / Highlight
	Graph	Zoom
	Text	Scroll
Hud Actived	Cluster Icon	Alpha / Color
	Camera Icons	Move to preset position
	Edge Icon	Overall bundling strength
"No Gaze"	Empty space	Save preset position

3. IMPLEMENTATION

The project was developed using the Unity3D game engine, as an immersive 3D visualization of data rich nodes, connected by bundled edges. An interface module to navigate and interact with the data environment was also created. Details of the implementation are presented in this section, starting with the visualization, and continuing with interface implementation. This section concludes with some considerations and directions for future work.

3.1Visualization

For the purpose of this demonstration, a custom built 3D environment was designed and developed. It consists of nodes and connecting edges, positioned by a pre-existing force directed graph algorithm implementation, which gives an XML file as output. This is used at runtime to position the node objects. At each node location, a specific 3D object is also instantiated, alongside graph and text data. Some nodes are connected by edges, also specified in the originating XML file. These edges are drawn at runtime using meshes, procedurally generated to follow a B-Spline with midpoints used to create bundles. Nodes are then enveloped in a cluster, based on their location on the graph, using isosurfaces generated by an existing implementation of the marching cubes algorithm [5].

3.2 Interface

The current framework is based around a head mounted display (HMD), an off-the-shelf 6DOF 3D Mouse and a custom built ad-hoc controller driven by an Arduino Board, with access to a “Pitch Bend/Mod” wheel.

The 3DMouse is used for navigation around the visualization, and to activate HUD controls via a button. The gaze interface, which is obtained from the HMD, is used to highlight interactable content while the custom controller is used for the selection of either the main function over highlighted interactables (by soft push on mod axis), or alternative control (by longer push). The chosen functionality is then available for the selected object using the pitch bend wheel, which allows a level of haptic feedback by means of the spring present.

Depending on the selected object, the functionality of the controller manipulates different parameters, but is always based on left/right approach adding or subtracting from the parameter at different rates.

The coordinated use of both hands and gaze is initially challenging, but grants precision in both navigation, selection, and parameter control, allowing a deep level of immersion with the visualization, with feedback provided for all the phases of interaction.

4. CONSIDERATIONS

The first prototype used a mouse controlled slider widget to access parameter manipulation, this did not translate well to the VR implementation, and an attempt was made at using hand tracking to control the same style of interface. After initial iterations, problems with scaling of virtual and real coordinates and lack of tangible feedback triggered the possibility of using an ad-hoc control for manipulation. The current data set consists of the names of a number of computer viruses, linked by similarities with the graph edges. Each node is paired to a virus name. A procedurally generated mesh is associated with each node; parameters for the mesh generation are related to characteristics of the virus. Statistics and a textual description of the virus are displayed upon reaching proximity with the related node. The setup was informally tested throughout prototype iterations, by the authors and a number of students under observation. Feedback and observations were recorded and taken into consideration for the current implementation, which is still a work in progress.

5. FUTURE WORK

An attempt to re-introduce the hand tracking implementation is envisaged, and tracking of the physical interface devices with retro-reflective markers. A static surface for touch control with haptic feedback will be added, by means of a rectangular piece of flexible material, attached to a tripod. Markers on the surface will be tracked in the VR space by the same imaging device used for hand tracking (likely the Leap Motion controller), and provide a touch surface for other forms of input. A formal study on the validity of the whole interface will then be undertaken.

6. REFERENCES

[1] Cubaud, P., Topol, A. and Dupire, J. (2005) ‘Using game engines for non 3D gaming applications’, In *Proceedings of CGAMES'05, Int. Conf. on Computer Games* , pp. 304–307.

[2] Billinghurst, M., Kato, H. and Poupyrev, I. (2008) ‘Tangible augmented reality’, ACM SIGGRAPH ASIA 2008 courses on - SIGGRAPH Asia ’08. doi: 10.1145/1508044.1508051.

[3] Chen, Y.-L. and Hung, P.-C. (2009) Intuitive augmented reality navigation system design- implementation by next-gene20 project. Available at: http://papers.cumincad.org/cgi-bin/works/Show?caadria2009_069 (Accessed: 12 April 2016).

[4] TANRIVERDI, V., AND JACOB, R. J. K. Interacting with Eye Movements in Virtual Environments. In CHI ’00 Proceedings (2000), ACM, pp. 265–272.

[5] Balzer, M. and Deussen, O. (2007) ‘Level-of-detail visualization of clustered graph layouts’, 2007 6th International Asia-Pacific Symposium on Visualization. doi: 10.1109/apvis.2007.329288.

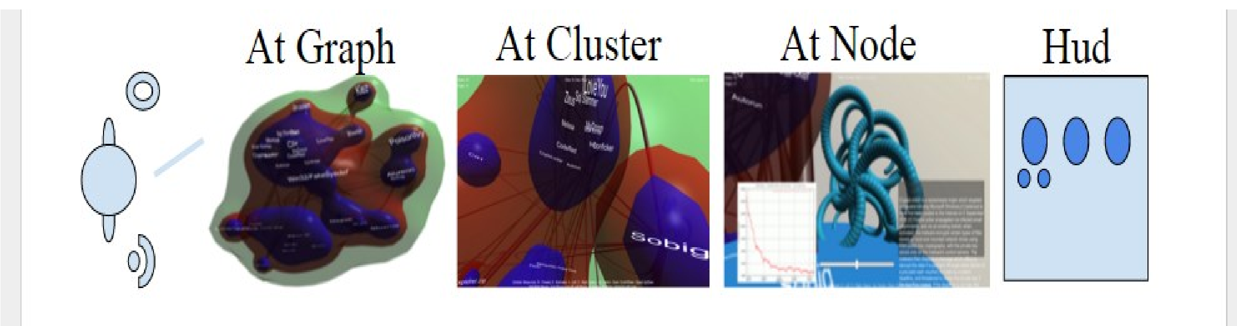

Figure 2. Visualization Diagram.