

# 3D Graph Visualization with the Oculus Rift

## Virtual Graph Reality

Farshad Barahimi, Stephen Wismath

University of Lethbridge, Canada

University of  
Lethbridge



Faculty of Arts & Science

### Why?

There are many theoretical results in the graph drawing literature regarding three-dimensional (3D) representations of graphs. However, the actual usefulness of such 3D drawings has always been questioned; in particular, the effectiveness of visualization has been a main concern, as ultimately the graph must be presented to the user, commonly:

- projected onto a (2D) monitor, possibly with a stereographic effect such as shutter glasses, or anaglyphically,
- displayed in a virtual reality environment such as a CAVE, or
- printed as a physical model with a 3D printer.

Early studies on the effectiveness of head-mounted VR systems vs. fish tank VR noted that the hardware suffered from latency and resolution problems [4]. However, more recent HCI experiments indicate that users can improve performance on graph tasks (such as determining paths) for large graphs in a 3D VR environment [3].

### What?

The GLuskap [1] software package allows for the creation and editing of graphs in 3D and has been extended over the past decade to include each of the above three output techniques; interactive visualization of 3D graphs has always been a priority.

The Oculus Rift [2] headset provides an affordable, individual, 3D stereo experience with head tracking and display of appropriate left and right eye information. The low latency and accurate orientational tracking allow an acceptable immersive experience.

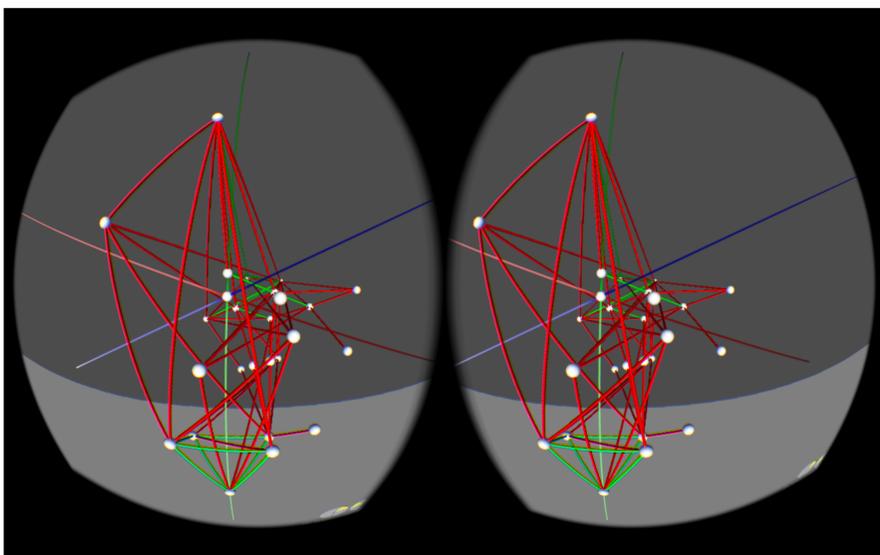


Figure 1: Left and right eye information provided by the rift.

### How?

The Oculus Rift Software Development Kit provides programmable support in a C environment. Although GLuskap is written in python, suitable wrappers and libraries permit its use as the main engine for creating the virtual (graph) world. Head tracking on the first generation (DK1) rift is only rotational and does not detect translational movements. For more effective and convenient navigation, a simple game controller was added, allowing the user to efficiently and intuitively move through the displayed graph. For visualization purposes, when a user discovers an appropriate viewpoint and viewing angle, a *snapshot* or picture of the visible portion of the graph can be made. Note that the left/right eye information provided by the rift

does not provide an acceptable 2D picture and thus the scene is totally recreated and supplied to separate rendering software (POVRay) for processing and generation.

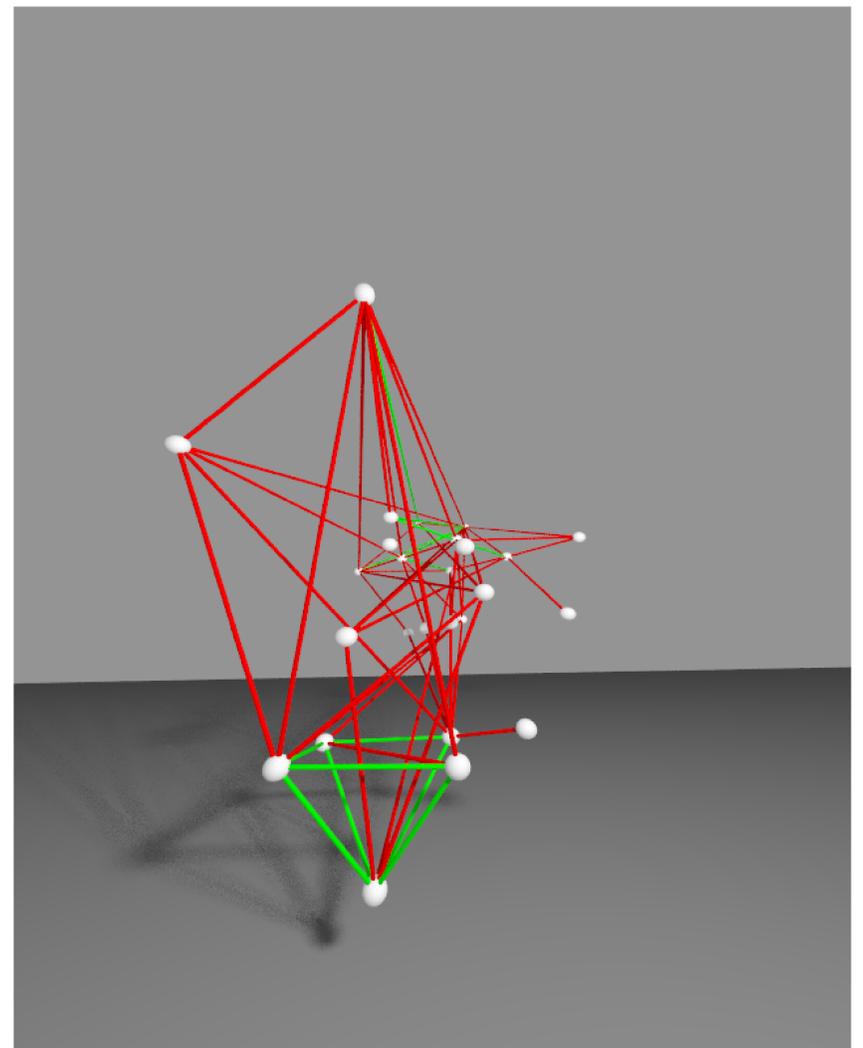


Figure 2: Snapshot generated.

### Why Not?

The rift headset prevents the wearer from easily accessing the keyboard, mouse, and computer screen and thus the graph editing features of GLuskap (such as adding, deleting, and moving vertices and edges) are unavailable – only navigation, scrolling through stored graphs, and snapshots are provided.

### References

- [1] GLuskap information, manual, source code and plugins: [github.com/u leth HCI/GLuskap/](https://github.com/u leth HCI/GLuskap/)
- [2] Oculus Rift: [www.oculusvr.com/rift/](http://www.oculusvr.com/rift/)
- [3] C. Ware, P. Mitchell, Visualizing graphs in three dimensions, TAP, Vol. 5, 1, 2008.
- [4] K. Arthur, K. Booth, C. Ware, Evaluating 3D Task Performance for Fish Tank Virtual Worlds, ACM Trans. Inf. Syst., Vol. 11, No. 3, 1993, pp. 239–265.

### Acknowledgments

N.S.E.R.C. support is gratefully acknowledged.