Project Title: **Glove I/O**

Team: Ahmed Malik, Chase Muramoto, Luis Sanchez, Sachin DeYoung

EECS 149/249A Project, Fall 2018

Project Goal

The project is based off of project #8 on the list of suggested projects: Handheld Controller for AR Headsets. The projects uses a pre-existing AR or VR headset which we would attempt to acquire from a lab -- specifically through Bala Kumaravel/Bjoern Hartmann -- and a user-worn glove which we design and manufacture. We would use the AR/VR headset in conjunction with our glove to interact with a virtual environment and its objects; the user-worn glove would serve as a 6DOF tracking device, as well as an input/output device similar to that of a mouse/keyboard on the computer or fingers on a touchscreen, to the degree needed for the simulation application. We would also implement depth perception to allow us to move the virtualized hand/input-output device into the virtual environment to be able to interact with objects further away. We would showcase the features of our devices through some type of simulated, virtual environment -- likely through Unity. For example, the impetus for this project came from wanting to view 3D CAD models in an AR/VR setting, so we would enable the user to do so with the glove and AR/VR headset displaying Solidworks CAD.

Project Approach

The common ground between these projects is building a robust user-worn glove. We would determine orientation and movement of the glove in 3D space by using IMU sensors and/or accelerometers and gyroscopes. We would also allow the user to use any of their 5 fingers for some additional control using flex sensors. Then, we'd classify certain motions, gestures, or control to specific outputs on the desired application. Depth perception would be performed using some sort of distance measurement sensor -- such as ultrasonic or IR -- or by using a camera in conjunction with AR trackers stuck to the glove, or using computer vision techniques. We would model the the inputs from the glove and corresponding outputs in a state machine, mapped to the functions we want for the specific application(s).

Resources

We would need several sensors and some hardware to construct the user-worn glove. Orientation sensors can likely be used from the Buckler boards, though we may want to reduce the size of the circuit on the glove and so may need to purchase the specific sensors separately. Clicking-type functionality can be done through flex sensors. If performing most of the processing on the glove, we would need some slightly more powerful hardware than a simple Arduino/microcontroller, such as the Buckler & nRF52DK system, myRIO, or mbed FRMD

KL25Z. Otherwise, a simple microcontroller may be enough. For depth perception on the AR/VR headset, a camera or distance sensor would be needed, as well as access to simulation software, such as rviz and ROS or Unity. For the depth sensing, a Leap Motion or Kinect could be used.

Schedule

Oct 19 - 25: Determine which specific sensors and hardware we need to order, and prepare/submit purchase request for those. Get familiarized with Unity software. Research open source glove ideas, as well as depth perception techniques using leap motion, kinect, or cameras.
Oct 26 - Nov 1: Create architecture drawing showcasing main ideas of project and the overall system specific goals/outputs. Begin creating system to get basic 3 axis orientation for glove and interfacing this into Unity. Understand the basics requirements for depth perception.
Oct 30: Milestone 1
Nov 2 - 8: Finalize glove-to-Unity orientation, begin working on translational movement in conjunction with depth perception.
Nov 9 - 15: Showcase 6DOF movement in AR with a very basic system, where the glove is simply a rod. We don't care about individual fingers yet.
Nov 13: Milestone 2
Nov 16 - 22: Work on finer gesture control, as well as using individual fingers for control by employing the flex sensors. Begin to interface an actual hand within Unity. Work on sending data over wireless or wired connection to computer.
Nov 23 - Dec 6: Work on fine tuning, gesture control, and Unity integration. Work on sending data over wireless or wired connection to computer.
Dec 7 - 14: Work on poster and report.

Risks and Feasibility

There are many unknowns, starting with how accurate our baseline readings and classification for orientation and gestures is. We anticipate that making the gesture control robust and natural will require a lot of time and fine-tuning parameters to allow a user to use it comfortably. With poor sensors or noise, this process may become quite difficult. Inputting control quickly to the computer via a wifi or BLE connection may be difficult as well. Most definitely, performing the depth perception will be quite difficult, requiring lots of research on current techniques.

Github Link

The link may or may not be correctly setup to be viewed by the GSIs, so if that's the case please let us/me know (amalik@berkeley.edu).