Contains exerpts from FINAL REPORT.pdf

The source is located at a group of repositories found here: https://github.com/NCSALOVR. The repository labeled Unreal contains all the source for the view: https://github.com/NCSALOVR/Unreal. Unfortunately, it is not simple to make use of the source straight from the repository. There are two core things that need to be done to make use of the source. First, the Unreal Engine must be the correct version. Unreal projects have little portability, and so version 4.4.3 must be used. Second, once launching the correct version of the Unreal Engine, create a new project of type "Code First Person." This will launch Visual Studios C++ 2013. from there, we can adjust the source of each file, and all source in the Public and Private directories need to be replaced with the source from the repositories.

For future version control, the Unreal UI has options for version control that can be looked into. To start expanding on the project, observe ViveViewCharacter.cpp, and notice the Tick function. This is what can be manipulated to make adjustments to what happens every frame. You will also need to eliminate all keybindings within the Unreal Engine UI. To bring the project from source to UI, right click the project and select the debug option, to run the project. If everything compiles, the UI will launch and you will be able to edit the environment from the editor. It is also recommended that you delete all visible objects in the room, so that the provided source is doing all of the rendering.

From here, the code is ready to be expanded on. If editing from the Unreal Engine is desired, it should not interfere with the source usually, since the final design was very modular. However it is risky, and changing some settings or properties with the UI or Blueprints has broken the project on certain occasions. Only proceed to do so if there is a good understanding of the Unreal Engine. It is also smart to understand how the VIVE and the server source works. Please take a look at their repositories to make sure there is a clear understanding of the role of each piece of source.

Once everything is set up, launch the server code, launch the appropriate software from VIVE, and then launch the ViveView from the Unreal UI by clicking Launch. If there are issues of getting the data between all of the parts, make sure the file paths in the source code match between all parts of the project. The Unreal engine currently writes and reads at a very specific location, and it is crucial that VIVE writes and reads the same location. Feel free to adjust these locations to your needs. For the Unreal Engine part of the project, all file paths are in the UpdateFromVive function. For more information on what needs to be adjusted for the other parts of the project, read the descriptions for other individual group parts. Because of the strong division of labor, the team is fairly modular.

Step-by-step:

1) Install Unreal Engine 4.4.3

2) Install Visual Studios 2013 or later

3) Create a "First Person Code" project.

4) Copy the source from the files in the repositories to the files in the source code editor

5) Launch with Debug option (either click Local Windows Debugger, or right-click project name->debug->start new instance). This leads to the Unreal UI

6) Remove key bindings in the project settings

7) Delete the cubes, walls, other solid visible objects

8) Select Launch from the UI to launch the program.

9) After confirming a successful launch, all source in Tick() and UpdateFromVive() is up for edit. Good luck!

FINAL NOTES: Make sure you understand how and when to use VIVE and the networking service to properly run the full project. Unreal and VIVE currently communicate with a very specific filepath. Feel free to edit this in the Unreal source in UpdateFromVive(), just make sure VIVE is reading and writing to the correct new place.