Design Document for Noctilucent VR

Taylor Fahlman, Joshua Bowen, Adam Puckette

December 1, 2016

Abstract

Contents

1 Overview

1.1 Scope

This document describes the software design and information regarding the Virtual Reality Lidar Point Cloud Viewing Software known as Noctilucent VR. This software is intended to be used by professionals seeking to improve their Lidar Cloud viewing experience. Uses can include anything from casual cloud data viewing to professional informatics.

Noctilucent and it's design can be used on any computer capable of handling the graphics rendering. This projects only applicability restriction is in the capabilities of the computer running it.

1.2 Purpose

Noctilucent VR exists to expand the current capabilities of humans to view Lidar Cloud data. Specifically the goal of this project is to take 3D information previously displayed in 2D and provide an easy to use, easily accessible, Open Source alternative to viewing this data which allows for it's viewing in 3D. This project exists as viewing 3D data in 3D is an important aspect of being able to garner easily all the information from that data as possible.

1.3 Intended Audience

This project is primarily intended to be used by scientists, engineers, and those who perform informatics, however, the project is not limited to those audiences. The Free and Open Source nature of this project allows it's use by anyone with access to cloud compare, a virtual reality headset, a wiimote, and a computer capable of running this software.

1.4 Conformance

Noctilucent VR conforms to this document if it satisfies the requirements as they are outlined in Sections 4 and 5 of this Document. Requirements are denoted by the verb shall.

2 Definitions

Free and Open Source Software (FOSS)

Any software that can be classified as both free software and open source software. This means that anyone is freely licensed to modify, copy, or stufy the code as they see fit. Users are encouraged to improve the software and share their work with others.

Virtual Reality (VR)

An Existing technology which allows the user to put ona headset and have an immersive experience in which turning your head turns the camera and mmoving your head moves the camera. Visuals are rendered seperately for both eyes allowing for different perspectives which come together to create a 3D stereoscopic environment.

CloudCompare (CC)

A FOSS program freely available to users who need to view and analyze point cloud data.

Open Source Virtual Reality (OSVR)

An existing Open Source VR Headset and VR Software platform. OSVR is designed so that it can be developed on and modified by the public at only the cost of the headwear.

Graphics Processing Unit (GPU)

The piece of hardware inside a computer responsible for performing microprosses often associated with graphics rendering or deep learning.

Frames Per Section (FPS)

The refresh rate at which the computer draws a scene over the course of a second.

3 Design

3.1 Stakeholders

Matt O'Banion - For use in geo-informatics

3.1.1 Design Concerns

- Shall have a user interface for interacting with the data
- Shall utilize an auxillery device for ineracting with the data
- Shall render at least million points of cloud data
- Shall be usable by any Virtual Reality headware and framework
- Shall interface with point-cloud viewing software

3.2 Context Viewpoint

The context of a software project is the environment and systems with which it interacts. These external elements are detailed below, along with a description of their interaction with our project and their responsibilities towards the user and each other. This is a broad overview of elements outside the scope of Noctilucent VR that are required in order for the project to be successful.

3.2.1 Design Concerns

1. The user, who is responsible for providing positional data, orientation data, and manipulation data. The user receives a graphical representation of the point-cloud data in return.
2. The point-cloud viewing software, which is responsible for providing the graphical representation of the point cloud data as well as processing the manipulation data.
3. The OSVR framework, which is responsible for processing the positional and orientation data.

3.2.2 Design Elements

There are three external factors that form the design context for Noctilucent VR. First there is the user, who will use our software to view and interact with point-cloud data in virtual reality. Second there is the point-cloud viewing software, which will handle the rendering and manipulation of the point cloud data as directed by the user. Finally, there is the OSVR virtual reality framework which will coordinate with the point-cloud viewing software to display the point-cloud data on the userâĂŹs virtual reality headset of choice. These three elements form a triangle of interaction. The user provides input in the form of positional tracking and orientation data to the OSVR framework, which in turn coordinates with the point-cloud viewing software to update the headset display. The user also provides interaction data directly to the point-cloud viewing software to measure and manipulate the point-cloud data. The point-cloud viewing software in turn updates the headset display of said measured and manipulated data via the OSVR framework.

3.3 Interface Viewpoint

The interface of Noctlilucent VR involves both hardware and a software componenets. The harware component will allow the user to manipulate the dataset with physical actions, as well as view the dataset. The software component will allow the user to perform pre-set complex operations and manipulations with one selection.

3.3.1 Design Concerns

1. The display device will enable the user to view and more importantly, move around the data set they are exploring.
2. The input device will be used for data manipulation. The input from the device allows the user to interact and analyze the data in a highly-controlled fashion.
3. The User interface overlay will be used to manipulate the display of data, as well as peform similar actions to the physical device, but pre-set so the user does not have to repeat these actions.

3.3.2   Design Elements

Name: Display device
Type: Hardware unit
Purpose: This device will enable the user to view the data they need to explore, specifically in steroscopic 3D. This device needs some sort of screen which enables the eyes to view the data in 3D. Most likely, a virtual reality headset will be used, as it has the required screens and technology to display the information in the desired way.

Name: Input device *Type*: Hardware unit
purpose: The input device is the primary way that the user will interact and manipulate the data viewed. The versatility of this device (or collection of devices if needed), is important, as there are many transformations that are possible in a 3D environment.
Name: User interface overlay *Type*: Component
Purpose: The user interface will be overlayed on top of the data display. This interface provides the user with information about the current dataset, as well as pre-set actions and transformations on the 3D data for the user. While not integral for the main goal of this project to work, this componenet provides useful tools that enable the user to more effectively perform research on their data.

3.4   Structure Viewpoint

3.4.1   Design Concerns

3.4.2   Design Elements

3.5   Interaction Viewpoint

Noctilucent VR will have six primary interactions. The first interaction is between the cloud viewing software and the GPU. The second interaction is between the GPU and the VR Software/Headset. The third interaction is between the user and the auxillery device. The fourth interaction is between the auxilery device and the cloud viewing software. The fifth interaction is between the user and the UI. The last ineraction is between the UI and the cloud viewing software

3.5.1   Design Concerns

1. Interacting with the auxilery device updates the display and edits the data
2. Interacting with the UI updates the display and edits the data

3.5.2   Design Elements

Name: User Interface
Type: Component
Purpose: Allows the user to perform actions commonly performed to manipulate the data in the point cloud.

Name: Cloud Viewing Software
Type: Component
Purpose: Houses the data, and performs all the backend workd necessary to edit and view the point cloud.

Name: Graphics Processing Unit
Type: Component
Purpose: Responsible for rendering the point cloud data so that it can be viewed in Virtual Reality.

Name: Virtual Reality Headset
Type: Component
Purpose: Used by the user to view the cloud data as it is created by the GPU and the Cloud Viewing Software.

Name: Auxillery Device *Type*: Component
Purpose: A device used by the user to edit, manipulate, and change views of the data without the need for a keyboard and mouse.

3.6   Design Rationale