Bi-Weekly Report 5

Date: *11/12/2015*

Project: *MSF Google Project 1*

Project Title: *Video and sensor display on the Google Cardboard*

(Previously: Alzheimer's Experience using the Google Cardboard)

Team Members:

Garrett May (Team Leader)

Hekla Helgadóttir

Overview

We are finalising our Proof of Concept in order to present graph visualisation. We have been able to draw graphs quickly without overworking the main thread.

Video is also present on one of the boxes. This is currently done using a list of images, however we feel this will easily be applicable to an image stream from a video stream.

Meetings

Date: *10/12/2015*

Attendees: *Garrett May*

Hekla Helgadóttir

Carla Hyenne

Daniel Eldar

Aron Monszpart

In this meeting the team met to discuss the progress of the project so far, and to discuss the final implementation of the website and the video. We made changes to the storyline for the video, which we had made a week earlier, and split the task of writing the remaining sections for the website between us.

Completed Tasks and Project Projection

During the last two weeks, we were successful in our aim to draw graphs in our application. We had intended to draw graphs in a JavaScript library, such as Three.js or D3.js, render the images onto a texture in OpenGL, and then update these images accordingly. However, it proved quite difficult to get this system working, as it would involve running JavaScript code within the Android application itself.

Instead, a simpler solution presented itself. We were currently rendering boxes, each actually simply being a quad, and drawing triangles each time. We decided that, if we were

to replace this quad with a graph, it would simplify our program greatly. Although we would have to create the graphing library ourselves in OpenGL in order to create the graphs we needed, this method allowed us to make our graph drawing more efficient and more natural. It also meant that we had the freedom to create different types of visualisations for data in the future, such as bar graphs, pie charts, or even just displaying a number.

As today is a major milestone for a project, it seems suitable to analyse our project projection to date. We are currently following our timeline, and intend to work on the drone, networking, and video in the second half of term. We also have a basic prototype of how we will display data. Everything seems to be on schedule.

Problems to be resolved

As of now, our prototype of the project is completed. We now need to make improvements upon it, and the progress further through our project, in order to have the whole system working.

For the Google Cardboard, we should aim to abstract and clean up any code. This is to make the application more efficient and to give it a better structure, which will be useful for adding or changing features, and less memory intensive.

We also need to focus on adding the ability to display video in our application. This may prove to be difficult, as we must find a way to render the video onto a texture for a box. Normal image uploading for textures is mainly for a single static image, so even if we could transform a video stream into a stream of images, substituting each texture each frame may end up being too costly.

The drone system will need to be built soon. The circuit board that will be used will either be a variant of an Arduino, or a variant of a Raspberry Pi. Which type we use will depend on whether there will be enough processing power to send video and sensor data. We should begin designing the system and the code needed for it to function, which will most likely be programmed in C. We also need to think about the gimbel subsystem.

The networking side is progressing, but we will need to discuss how it is going to work. The Google Cardboard application is written in Android, which is similar and based upon Java. However, the drone system will probably be written in C. In order for our networking to run smoothly, we will have to create a program, or perhaps two programs linked with each other, that will be able to send data between it and the drone, in C, and between it and the application, in Java. It will have to be able to go both ways, in case we wish to send a command request to the drone, an example being in gimbel movement.

Workload - Past Two Weeks

Garrett:

I had been focusing on creating Frame Buffer Objects, by wrapping them into my own custom object, allowing us to abstract their functionality. This would have proved useful for image rendering, as we had hoped to use it for displaying graphs.

I also worked on replacing the model for the GUI box for the model for a graph. This was done by creating a Graph object, which contained the model, and then providing getter methods in order for the GUI to have a reference to the model.

I was also able to provide a rudimental LineGraph, which was able to draw lines between points on a set of axes.

Hekla:

At the start of the two weeks I began looking into rendering a video as a texture to apply to the menu boxes in the main menu on the application, using PixelBufferObjects. However because of

I wrote content for our website, drew explanatory diagrams and developed the business context around our application, including designing the logo.

I also worked on the video script, recorded drone footage to be used in the video, edited and recorded the video.

Workload - Next Two Weeks

Garrett:

I will work on improving the structure of the application by refactoring some of the program and removing any duplicate code. I'll also attempt to add more types of Graphs, such as a bar graph and a pie chart.

Another thing I should look at is to create a texture atlas for glyphs, in order to display text, such as alphanumerical and punctuation. This will be important if we wish to display any data in a non-graph form.

It may also be worthwhile in looking into turning a video stream into an image stream and to look at some of the networking, as part of this will need to be done in the Google Cardboard application.

I should also help Hekla in the drone system. We will need to discuss whether we should use an Arduino or a Raspberry Pi, and how to send sensor and video data.

Hekla:

I will work towards connecting the telemetry, Arduino and the sensors. When the system has been connected I will look into transforming the sensor output to metrics that will be used by the application.

I will also start researching how to use the built-in accelerometer of the phone to send movement instructions to the gimbal, and whether we can use the Jeelink/Jeenode telemetry for this as well as sensor data transfers.

Lastly I will do research on which infrared camera, gimbal and possible low resolution camera to order – and place the order once I have found suitable parts.