3D Visualization of Data Problem Statement

CS461 - Senior Software Engr Project

Instructor: D. Kevin McGrath
Instructor: Kristen Winters

Fall 2018

Eli Laudi

October 11th, 2018

Abstract

This paper goes over the broad strokes of the data visualization problems that our capstone project will attempt to solve and why it needs to be solved. It will then go over our proposed solutions and what tools and open source libraries we may use. Then, it will show the metrics which we will hold ourselves accountable to.

1 PROBLEM

In many fields, there is often an issue of data presentation. Data is not an easily consumable thing in its raw form. Data scientists often have to come up with innovative ways to show data to customers or even other researchers via blogs, videos, presentations, or other media forms. There are current solutions to the problem, but they are not universally accessible and require a lot of overhead to set up and transfer data. This leaves room for a one size fits all, three-dimensional data display tool. Our goal is to provide a solution that fits in that niche.

2 PROPOSED SOLUTION

There are several open source solutions to display data in a 3D manner. However, these solutions have their own input type that a lot of data might not come prepackaged as. Our solution is to come up with a program which will consume CSV spreadsheet files, extract the data from them, and display that data in a friendly 3D manner.

The initial starting point of this project is to handle data both geographically and temporally. For example, the tool will consume a CSV file that McAfee will provide us. It will contain data such as IP-Addresses, hashes of malware, geo-locations, tags, and time stamps. The tool will have to parse the data and translate it into 3D coordinates as well as mapping all associated data to that point.

The first part of this process is to consume the CSV file and parse the data in an efficient manner. There are solutions to do this in nearly every programming language, so the choice in languages comes down to a performance and preference basis, the top contenders being Java or Python. We would most likely translate the data into JSON format, allowing for hierarchical sorting with least-pertinent data nested the deepest. The principle of transforming the data is simple. The complexity of this part of the application would come from the ability for the application to consume CSV files that contain differing types of information. One CSV file might contain geographical data, while another may contain spatially related data; the application would be able to handle both. This is a feature that may not be part of version one, but could very well be implemented.

The second part of the process is the visualization itself. The data obtained from the first part of the process will be used and mapped onto a 3D space where the application will represent the data sets with points in space or on a globe. This data will be organized spatially and use colors to denote user-desired specifications. These specifications will be supplied in the CSV file from part one. The user will be able to explore the 3D data set using click and drag

3 PERFORMANCE METRICS

There are two main steps in this project. It can be effectively divided into the first step of consuming the CSV and transforming it into easily navigable and mappable data and the second step of displaying the data in a friendly 3D environment. The two sections combine into our minimal viable product. That is the overarching performance metric, though there are other sections of the project that are easily measurable.

Performance of an application that utilizes computer graphics is quite important. A slow 3D application is often useless as the entire purpose of it is to be navigable enough to provide value to the user. There is a metric that can be obtained from this, be it a minimal frame rate or a responsiveness rating. Until there development process moves along and the tools and resources we use are selected, the metric for display will have to remain vague.

There is one more metric that the application can be held to. While not at the topmost priority, the speed with which the application goes through its conversion of a CSV file to its display of the data on the 3D space can be measured and held accountable. Ideally, the application should be able to complete its "step one" process within an allotted amount of time. Once again, the project itself needs more definition before a definitive time goal can be set.