VICTORIA UNIVERSITY OF WELLINGTON
Te Whare Wānanga o te Ūpoko o te Ika a Māui

School of Engineering and Computer Science
Te Kura Mātai Pūkaha, Pūrorohiko

PO Box 600
Wellington
New Zealand

Tel: +64 4 463 5341
Fax: +64 4 463 5045
Internet: office@ecs.vuw.ac.nz

Interactive 3D Visualisation of Exoplanets

Owen Bannister, 300172912

Supervisor: Stuart Marshall

Submitted in partial fulfilment of the requirements for
Bachelor of Software Engineering with Honors.

Abstract

We have lots of information about planets outside our solar system. This can be accessed from a database by anyone. However this information is complex and cannot be easily understood by laypeople. This is a problem as it means that the information gained about these planets is not being used effectively to convey knowledge to the masses. To resolve this a visualisation has been created that can convey this information in a way that interested lay people can understand. The result of this project is a visualisation system that can be used as an information source for laypeople wanting to increase their knowledge about planets outside of our solar system. This report outlines the project carried out, the visualisation created, as well as the evaluation to discover its effectiveness at fullfilling the goals driving its creation.

Contents

Figures

Chapter 1

Introduction

This project seeks to design, implement, and evaluate an interactive 3D visualisation software system for displaying the content in the Kepler Exoplanets dataset. The deliverable is intended as a standalone 3D visualisation system with two modes of interaction, keyboard and mouse or Microsoft Xbox Kinect sensor ((REF)). The resulting visualisation will convey the information in the dataset in a way that the target users, laypeople who have an interest in astronomy, can understand and interact with.

1.1 Problem statement

The complex nature of the data involved in this project causes a range of problems revolving around understandability to arise, which this project attempts to address. The following subsections outline these in detail.

1.1.1 Understanding the content in the dataset

Understanding and analysing large datasets whose size defies simplistic or trivial analysis is a known issue that many areas of research are attempting to address, these areas of research range from data mining to visualisations in order to discover or highlight important features in the data so that people can more efficiently use it.

Humans often rely on visualisation when we solve problems. We create an image in our mind of a situation in order to make sense of it http://nrich.maths.org/6447. This allows ..... The content in the dataset used for this project is made up of records of each exoplanet discovered, each of which contains 46 fields. This many textual fields, many of which are floating point numbers is next to impossible for a person to visualise internally, which means that an external way of visualising it is needed.

KOI	Dur	Depth	SNR	t0	t0_unc	Period	P_unc	a/R*	a/R*_unc
r/R*	r/R*_unc	b	b_unc	Rp	a	Teq	EB prob	V	FOP
N	P. Zone Class	P. Mass Class	P. Composition Class	P. Atmosphere Class	P. Habitable Class	P. Gravity (EU)	P. Esc Vel (EU)	P. Period (days)	S. Hab Zone Min (AU)
S. Hab Zone Max	P. HZD	P. HZC	P. HZA	P. HZI	P. Int ESI	P. Surf ESI	P. ESI	S. HabCat	P. Habitable
P. Hab Moon	P. Confirmed	P. Disc. Method	P. Disc. Year	S. Name	S. Constellation				

Figure 1.1: Data to be visualised

1.1.2 Comprehension of planetary information

Much of the information regarding planets is cryptic and unintuitive, this make its understandability difficult. Visualisations in general attempt to address this issue by displaying information in a way that conveys the information in simplistic ways which allows improved user comprehension.

1.1.3 Existing solutions lack functionality

Existing data visualisation techniques using this exoplanet dataset lack the ability to display sufficient detail on each exoplanet and do not provide answers to questions that can be answered by the Exoplanet attributes in the dataset. Existing solutions display only the size, temperature, and orbital information about the exoplanets. While this is useful information that informs users of important facts about the planets, it does leave a lot of potential information unseen and overlooked, for example, information about the type of planet, planets with similar traits, solar system information, similarity to earth and habitability. This project will therefore be focused on researching, implementing, and evaluating a new interactive visualisation system that will display additional information to users not included in previous visualisation systems.

1.1.4 Effective user interaction with visualisation

A visualisation that solely displays information without effective methods of interaction limits the immersive qualities that keeps users engaged. To address this interactive visualisations emerged, generally these visualisations allow users to modify the representation of information rather than the information itself. This means allowing user control some property of how the data is represented, be it something simple as the layout of elements or something more complex. Many mediums of interaction are possible from the mundane keyboards, mice, or touchpads to the more esoteric wired gloves, motion sensors, and omnidirectional treadmills or even a combination of a range of devices.

With interactive visualisations, response time of the system to user actions is important and so changes made by the user must be incorporated into the visualization in a timely manner. Experiments have shown that a delay of more than 20 ms between when input is provided and a visualisation is updated is noticeable by most people ((REFERENCE)). Thus it is important for an interactive visualization to provide a rendering based on human input within this time frame or else risk breaking user immersion.

When the information being presented is altered, the visualization is usually part of a feedback loop. For example, consider an aircraft avionics system where the pilot inputs roll, pitch, and yaw and the visualization system provides a rendering of the aircraft's new attitude. Another example would be a scientist who changes a simulation while it is running in response to a visualization (see Visualization) of its current progress. This is called computational steering.

1.2 Key issues project addresses

To summarize the above sections, this project addresses the following key issues:

I1. Content in database form is difficult to view and understand.

I2. Planetary information if complex and difficult to comprehend without a visual reference.

I3. Existing visualisations for this dataset lack functionality.

I4. User interaction is needed in a visualisation to make the most of data displayed.

1.3 Contributions of this project

This project will provide an extension of the Kepler Visualisation Tool [**?**] that conveys more information and is easier for users to interact with than the original. This extension will be evaluated by a user experiment to ensure that it is successful in conveying the information contained in the dataset.

The work and research completed for this project will allow for further improvement by other developers and researchers to extend and improve the visualisation created. This will provide further exposure of the Kepler dataset which will encourage learning about Exoplanets.

Chapter 2

Project Methodologies

2.1 Project management approach

Following a structured project management approach is important as it avoids the problem caused by following a code-and-fix approach as described as 1) write some code. 2) Fix the problems in the code [?]. By following a process model it encourages thinking about requirements, design, and testing before coding is commenced.

The project methodology chosen for this project was a customized Spiral Model made up of requirements analysis, design, implementation, and evaluation phases as shown in the below figure. The reason for limiting the model to these 4 phases was because.... Using a

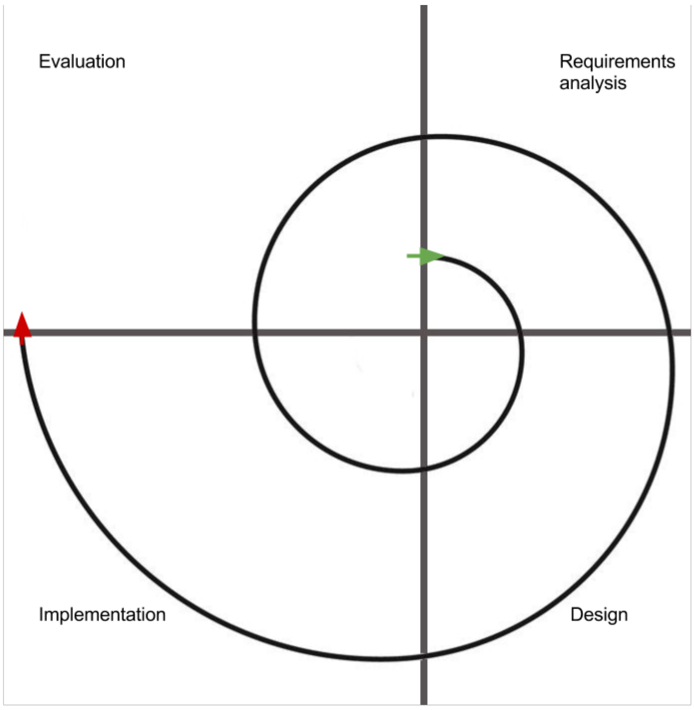

Figure 2.1: Spiral process model followed

spiral model allowed me to produce a deliverable feature at the end of each iteration of the model, this ensured that I did not become delayed or stuck in my development. By using this methodology it also allowed me to prioritize the features that were the most important to the visualisation which reduced the risk that there would be missing or incomplete components at the end of the project.

The advantages of this methodology over other choices such as the waterfall model or an agile approach such as Scrum was that it provided me with the benefits of a structured work flow that is a feature of the waterfall model as well as a flexible iterative process that is a feature of Agile methodologies. Following a pure waterfall methodology would not have allowed me to iteratively design, develop, and evaluate each feature which would have forced more upfront design which limits flexibility and support for changing requirements and design as was needed for this project. Following a pure agile approach would not have been optimal as most Agile methodologies are more beneficial to projects that have a team working on them. As it was I was agile in my approach to the project as I produced a deliverable at the end of each of the cycles of the spiral model and responded to change in the form of feedback and ideas from the supervisor of the project.

This project management technique supported the creation of a visualisation as it allowed the flexibility to add and remove components into the visualisation as they were discovered to be beneficial or not. It also supported the expansion of the project brief to include using a kinect sensor in order to control the visualisation. The choice of this project management approach meant that whilst I had the freedom to explore visualisation options I also had a structured software development life cycle to guide me and provide the project through the necessary steps to end in completion of each component.

By supporting this project methodology with other project management tools such as Gantt charts [APPENDIX] and work breakdown structures(WBS) [APPENDIX], it encouraged efficient documentation of planning and work completed in the project as well as displaying the upcoming stages required to complete the project.

Weekly meetings with the supervisor of the project, Dr Stuart Marshall, were used to provide guidance and ideas for innovation of the visualisation throughout the project. These meeting ensured that vital components and deliverables were implemented in the required timeframe and also provided a sounding board for ideas for elements to be included in the visualization. Another important aspect of having an involved supervisor was that he provided me the guidance of an experienced academic which was indispensable when navigating the administrative side of organizing delicate matters such as ethics approval for human evaluation of the visualisation.

2.2 Key difficulties in project

As this project builds upon a previous system much of the existing code and execution flow needs to be modified. This requires understanding of how the system was originally built and designed. Because this system does not have any unit or integration tests, going ahead without a comprehensive knowledge of the core functionality would be foolish.

Encountered errors in Processing framework due to number of elements needing to be displayed on screen.

Having a time constraint of 300 hours for this project over the course of a year meant that prioritization of visualization features needed to be made to ensure that

Libraries used for gesture detection in kinect are opensource in order to work with processing did not have decent detection

Chapter 3

Requirements Analysis

To guide the creation of the visualisation a user oriented design approach was used, in particular making use of user models (personas) which were created to give a sense of empathy and understanding for the foreseen users of the visualisation in order to better understand the requirements and design decisions to be made.

The design of the visualisation was based heavily on User Centered Design as it provided a method of user interface design as well as visualisation design. User Centered Design is a process in which the needs, wants, and limitations of the end users of a system are given extensive attention. To achieve this, personas were created (also known as archetypal users), which are a personification the needs of a larger group of related users. These personas act as stand-ins for real users, describing them in terms of their goals and personal characteristics, and although they are fictitious, they are based on knowledge of real users. This design methodology supported my understanding of how users were likely to use the visualisation.

An additional tool used during requirements analysis was User Scenarios which describe the foreseeable interactions of the user personas with the visualisation. A scenario is made up of a functional goal for the visualisation and describes how it is carried out by a persona. Both of these tools force you to think about the tasks needed for the visualisation and their context in the system as a whole. Once the personas and scenarios have been completed you can then start to design specific elements of the user interface and visualisation based on the requirements and interactions described in the scenarios.

3.1 User models

Below are the two personas that were used in the design of the visualisation for this project. They depict users that would use the visualisation in the context of a terminal or display in an observatory environment. These personas can be validated during evaluation of the visualization by finding real users that match the core values of the personas.

3.1.1 John Truman (Primary Persona - The interested layperson)

24 year old John is interested in planets and space and has a basic knowledge about both. He frequently visits attractions catering to this interest at locations such as planetariums and observatories. Some of his favourite things to do when visiting these attractions is to go to the computer terminals that allow users to choose what information they see.

John is used to playing computer games and using visualisations and is not overwhelmed understanding and using new systems. He finds that he learns better when provided with visual examples than when reading or listening to information. John is most comfortable using keyboard and mouse when interacting with a computer.

3.1.2 Cara Thompson (Secondary Persona - Likes gesture based systems)

23 year old Cara likes using interactive visualisations when visiting attractions, she finds that they are more entertaining and provide a better level of interaction and more of a novelty experience with a visualisation than simply a keyboard and mouse.

Both of these users are similar in their need for information from the visualisation but differ in the methods that they wish to access the information and interact with the visualisation. John wants to interact with keyboard and mouse as it is more straight forward and accurate. Cara wants to interact with gestures as she finds it more of a novelty and more immersive.

3.2 Scenarios

A good use scenario does a number of things:

- Describes the user's goals and motivations.
- Describes a specific task or tasks that need to be accomplished.
- Describes some of the interaction, with enough detail to make it compelling, but not so much detail as to be overwhelming.
- Provides a shared understanding for everyone on your team about what a user might want to do and how they might do it.
- Helps you construct the sequence of events that are necessary to address in your user interface.
- Can be sketchy, as long as it provokes ideas and discussion.

3.2.1 Scenario 1: View planets ordered by their similarity to Earth

Primary Persona:
When John first sees the system the first thing he notices is that their are many planets orbiting what looks like a star. He doesn't have any point of reference for these planets so their sizes, colours, and movement speeds are meaningless. By providing a way of comparing the planets to Earth it gives a point of reference which is well documented and known by most.

Procedure:

1. John clicks a prominent button stating view planet similarity to earth.
2. The planets on screen move so that the earth is located at the center and top of the screen with all others orbiting it. The planets with a high internal similarity to earth are higher on the Y axis whilst the planets with a high surface similarity to earth are closer to the center of the orbiting planets on the X axis.
3. From here John can select any of the planets for further analysis.

3.2.2 Scenario 2: Select ranges for attributes of each planet displayed

Primary Persona:
John has become comfortable with selecting the planets and has some idea of the scale and basic attributes of the planets. Now he wants to select more planets to find out more information. However due to the large number of planets he finds it difficult to accurately select them due to overlapping and fast moving small planets.

Procedure:

1. John uses a range of filters to remove planets from his view that don't match the criteria he chooses (temp ,size ,KOI , ESI).
2. As planets disappear the graph of planets expands into the space that frees up, this causes more space to appear between planets making them more selectable.

3.2.3 Scenario 3: Select planets to display more information

Primary Persona:
John wants to see more information about each of the planets he can see orbiting in the visualisation. To do this he wants to be able to select the planets and have textual information appear on screen.

Procedure:

1. John has the option to pause the rotation of planets in order to make more accurate selections.
2. John clicks on a planet orbiting a planet.
3. The planet selected becomes larger and its outline grows, making it more visible.
4. The text window has all of the information about the planet selected added to it.

Secondary Persona:
Cara wants to see more information about each of the planets she can see orbiting in the visualisation. To do this she wants to be able to hover her hand over a planet to get the information to display on screen.

Procedure:

1. Cara hovers her hand over a planet to make a selection
2. The planet selected becomes larger and its outline grows, making it more visible.
3. The text window has all of the information about the planet selected added to it.

3.2.4 Scenario 4: View planets in the same solar system

Primary and Secondary Personas:
When a planet it selected John and Cara want all ofter planets that are in the same Solar System as the selected planet to become highlighted.

Procedure:

1. When a planet is selected, all planets in the same Solar System become larger and its outline grows, making it more visible.
2. A label appears on these planets indicating that they are related planets.

3.2.5 Scenario 5: View the Goldilocks zones of each planet

Primary Persona:
John wants to see which planets are in the habitable zones of their stars.
Procedure:

1. John clicks a button saying ”Show habitable zones”
2. Coloured rings appear showing the cold (blue), habitable (green), and hot (red) zones of the selected planets star.
3. When a planet from a different star system is clicked the coloured rings will change to that stars zones.

3.2.6 Scenario 6: Select two planets to compare against one another

Primary Persona:
John wants to be able to compare two planets against one another. **Procedure:**

1. When John selects a planet a button becomes ungreyed called ”Compare” with a note next to it saying ”Please select another planet to compare to”.
2. When the second planet is selected a second text box fills up with the information about the second planet. This information can be compared with that in the first text box.

3.2.7 Scenario 7: Navigate the visualisation with gestures

Secondary Persona:
Cara wants to be able to navigate around the visualisation by using hand gestures **Procedure:**

1. By moving her hand to the edges of the screen the visualisation with pan in the correstponding direction, ie if the hand goes to the top of the screen the visualisation pans up.
2. By moving her hand backwards and forwards the visualisation will zoom in and out.

3.3 Requirements summary

3.3.1 Functional Requirements

3.3.2 Nonfunctional Requirements

3.4 Existing systems

3.4.1 Worlds: The Kepler Planet Candidates - Non Interactive

This animation [**?**] shows planet candidates found by NASA’s Kepler mission. These candidates are animated in orbit around a single star. They are drawn to scale with accurate radii, orbital periods, and orbital distances. They range in size from 1/3 to 84 times the radius of Earth. Colors represent an estimate of temperature with red indicating warmest, and blue indicating coldest candidates. The layout of this animation is very similarly to the Kepler Visualisation Tool that I am extending. This means that it provides insights into how

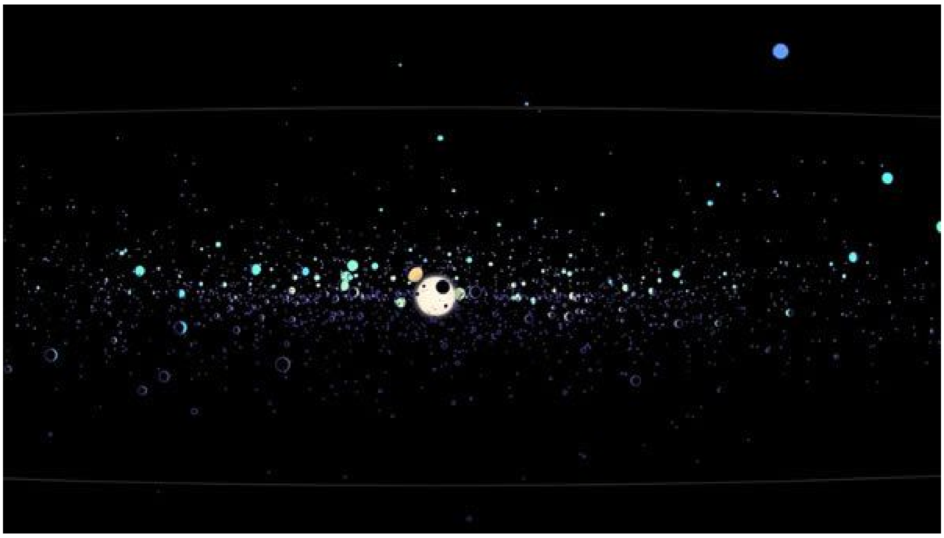

Figure 3.1: Image of Worlds Visualisation

my visualisation can be improved as Worlds is a much more visually appealing system. By researching how it displays its Exoplanets I can further improve my own visualisation.

3.4.2 The Kepler Orrery and The Kepler Orrery 2 - Non interactive

The Kepler Orrery [?] illustrates the exoplanet candidates in their own solar systems. The orbit radii are to scale with respect to each other and planet sizes are to scale with respect to each other, but orbits and planet sizes are different scales. The colors are in order of semi-major axis: two-planet systems (242 in all) have a yellow outer planet; 3-planet (85) green, 4-planet (25) light blue, 5-planet (8) dark blue, 6-planet (1, Kepler-11) purple. This system

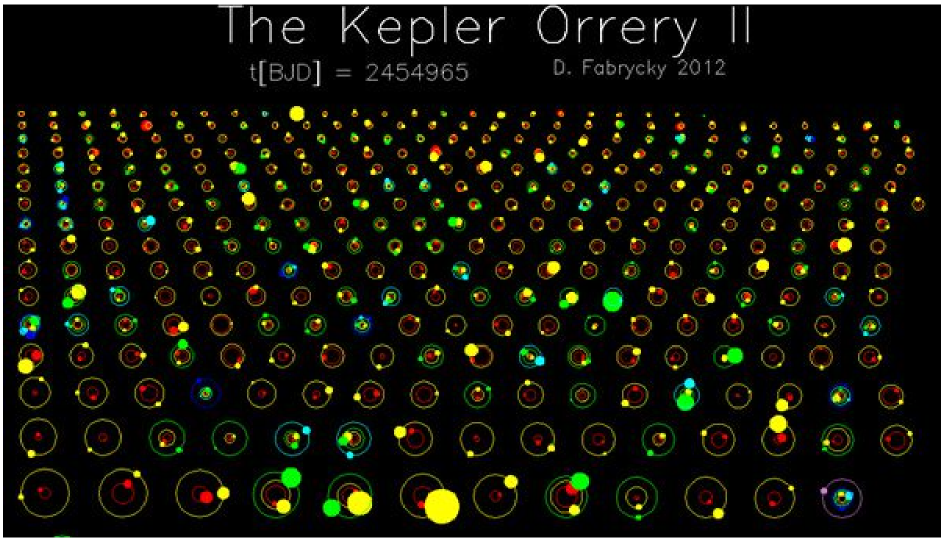

Figure 3.2: Image of The Kepler Orrery Visualisation

exhibits small multiples, a grid of small similar graphics or charts, allowing them to be easily compared. This provides insights into how I can use small multiples to display information about groups of planets. This will be important for displaying which planets share a solar system.

3.4.3 Celestia - Interactive

Celestia [?] is a free real-time space simulation that lets you visually experience the universe in three dimensions. It is an open source system written in C++. This visualisation is much

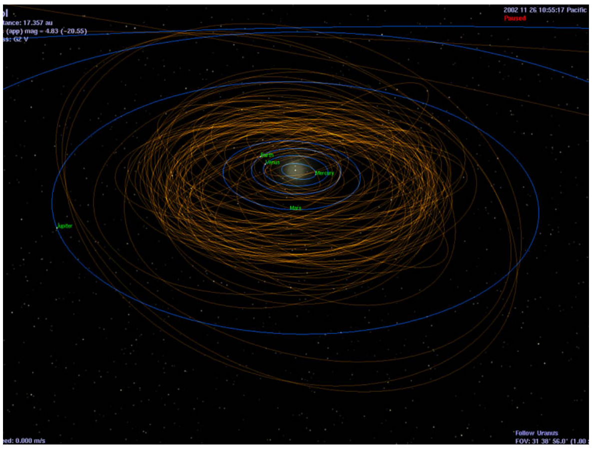

Figure 3.3: Image of Celestia Visualisation

larger and more encompassing system than is needed for this project, as it is a full 3D space simulation. However is does offer insights into how to effectively portray planets and their orbits (See Figure 2.3). It also provides textures that can be used in my visualisation to depict what planets actually look like to increase user immersion.

3.4.4 Kepler Visualisation Tool

An existing system built with Processing is the Kepler Visualisation Tool[?, ?]. It is a simple visualisation focusing on displaying the candidate Exoplanets temperatures and their locations in relation to their distance from their nearest star, so that a sense of scale can be perceived. Each candidates estimated size, orbital speed, and orbital separation is accurately depicted, and each planet is color-coded according to its estimated effective temperature.

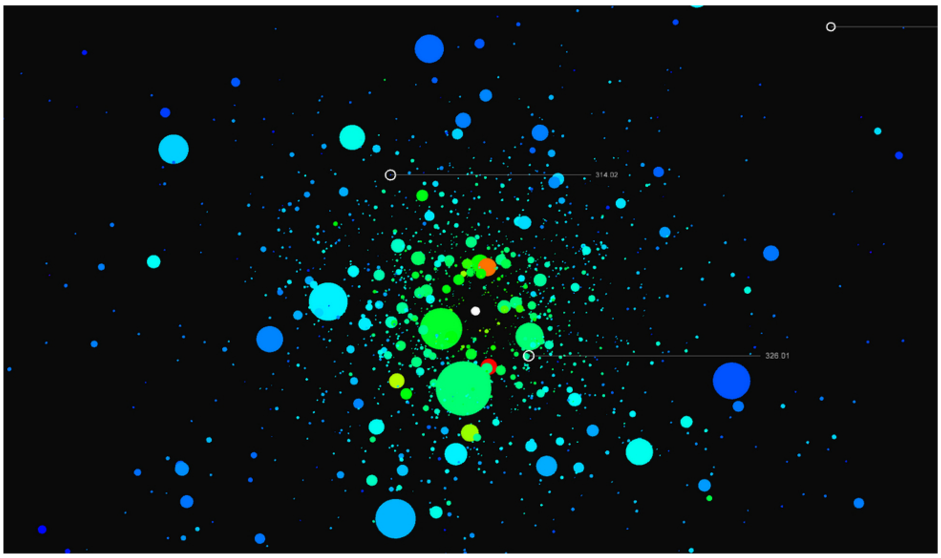

Figure 3.4: Kepler Visualisation Tool Orbital View

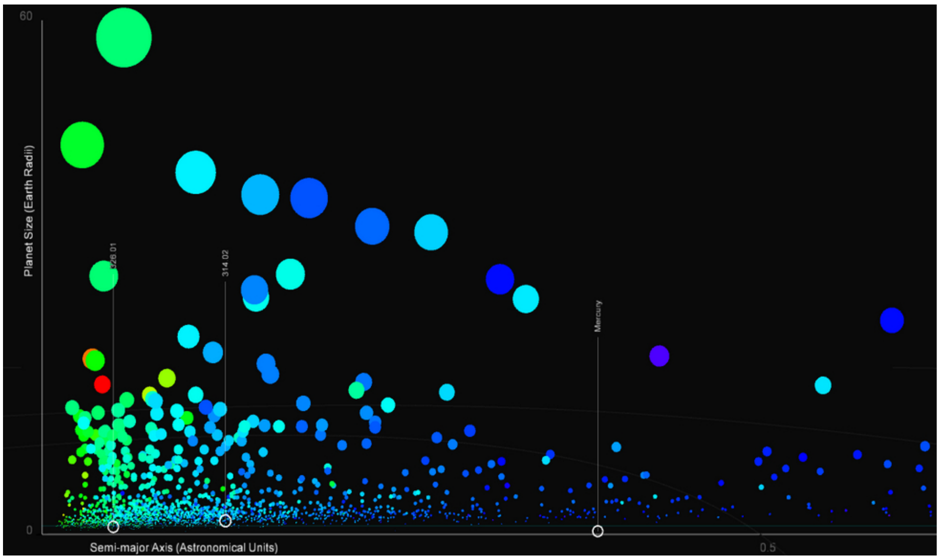

Figure 3.5: Kepler Visualisation Tool Graph View

The existing work in this system would serve as foundation for this project. Because much of the visual aspects, and initial data manipulation of the existing system are already complete. It means that implementing the features needed for this projects completion could be focused on more heavily and larger improvements to the existing system can be undertaken, such as better labeling and information displays and user interaction methods.

Chapter 4

Solution Design: Improved Kepler Visualisation Tool

This section discusses the deliverable visualisation that will be created as part of this project. The visualisation for this project will be created using Processing, a Java framework for visualisations. As the time is short for this project, extending a previous visualisation that uses the same dataset, the Kepler Visualisation Tool [**?**, **?**], will increase the amount of progress that can be made in the time afforded.

The visualisation was designed to emphasis small multiples and filtering of the Exoplanets to display the information more clearly to users.

Instead of the visualisation only answering the 5 key questions as proposed in the proposal, the aim will now be displaying as much of the information in the dataset as possible without detracting from the effectiveness of the visualisation whilst still answering the questions.

This is because the 5 questions did not fully utilize the information in the dataset.

However I will need to ensure the effectiveness of the visualisation does not become diminished by trying to convey to much information which would lead to cluttering and overlapping in the visualisation, as well as information overload for users. There will also be larger emphasis placed on making the existing system more usable by improving the interaction methods for users. The following list outlines the new requirements for the visualisation being developed. This will be done by providing GUI elements for each form of interaction with the system, as well as ensuring all interaction methods are intuitive for users.

4.0.5 Visualisation Layout

Component Layout

As the majority of the interaction and movement of visualisation elements occurs in the center of the window it caused a aspect ratio that was not suitable . It was BETTER to use 2 vertical columns to view and control the visualisation as it had a higher aspect ratio which allowed more of the content to be seen on the screen at once thanks to the fact that the majority of computer screens have a wide ratio.

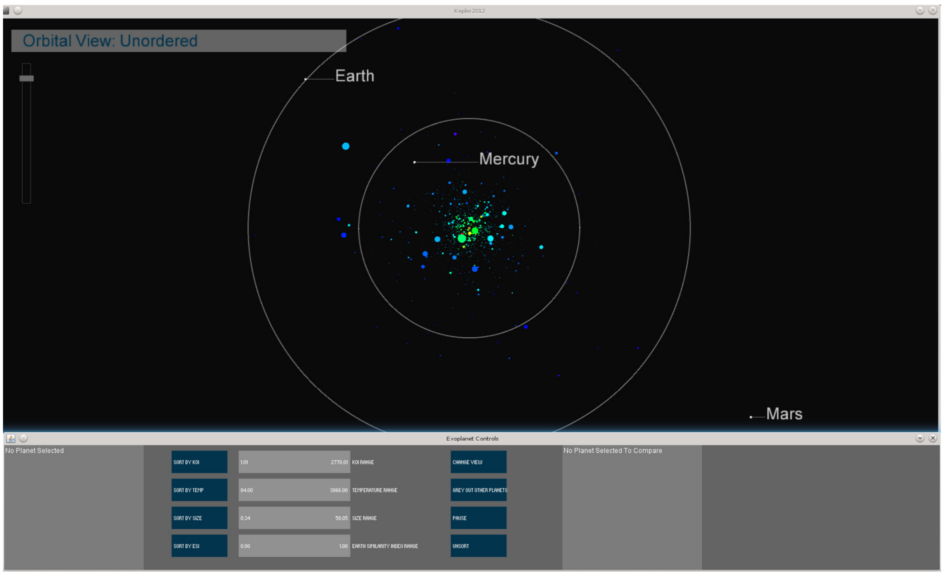

Figure 4.1: Original Horizontal Layout

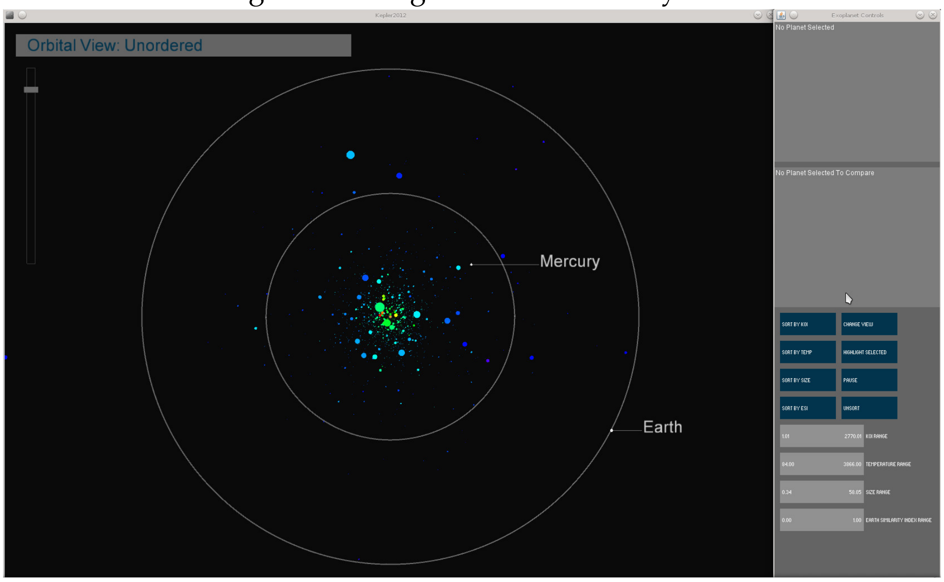

Figure 4.2: Improved Vertical Layout

Chapter 5

Visualisation implementation

This section details the implementation of the Improved Kepler Visualisation Tool. It details decisions that were made during the project such as choosing a viable framework, the choice of extending a previous system, and platform choice.

5.1 Technology choice

Many technologies were looked into,experimented with, and the positives and negatives of each weighed up before a decision was made about which would be the choice for the visualisation.

D3 (Data Driven Documents)

D3 is a JavaScript library that allows the displaying of data in dynamic graphics. Embedded within an HTML web page, the JavaScript D3.js library uses pre-built JavaScript functions to select elements, create Scalable Vector Graphic (SVG)[17] objects, style them, and add transitions, dynamic effects and tooltips. Large datasets can be easily bound to SVG objects using simple D3 functions to generate rich charts and diagrams. D3 was created because of the need for a balance of expressiveness, efficiency, and accessibility that previous visualization toolkits did not allow [4].

D3 allows the binding of input data to arbitrary input elements. This means that the exoplanet dataset can easily be bound to SVG elements for creating visualizations. D3 adopts the W3C Selectors API to identify document elements queried. This results in a rich but concise selection method of elements in a visualisation.

D3 allows debugging thanks to Google chrome and other modern browsers development tools. A downside to D3 is that it does not allow 3D diagrams, although it does allow pseudo 3D by using the painters algorithm and textures.

Prefuse

Prefuse is a set of software tools for creating rich interactive data visualizations [13]. The Prefuse toolkit provides a visualization framework for Java. It supports a set of features for visualizing and interacting with data. It provides optimized data structures for tables, graphs, and trees. It can be used to build standalone applications, visual components embedded in larger applications, and web applets. Prefuse to greatly simplifies the process of representing and efficiently handling data, mapping data to visual representations (e.g., through spatial position, size, shape, color, etc), and interacting with the data. To use Prefuse a basic familiarity with the Java is required, including setting up and building Java projects.

A knowledge of Swing or another similar user interface toolkit is also useful for understanding some of the concepts behind Prefuse and for integrating Prefuse visualizations into larger applications. Experience with database systems is also helpful. However the complexity of Prefuse means that the learning curve will be out of scope for this project.

Processing

Processing is an open source programming language and development environment that was initially created to serve as a software sketchbook and to teach the fundamentals of computer programming with a visual context. Using processing would mean that the visualization could be built with Java while still using a successful visualisation framework. The most complete existing visualization using the same exoplanet dataset (Kepler Visualization Tool) is built using Processing . Using this solution would involve learning the Processing language, however Processing is a library built in Java so the syntax is the same. This means the learning curve in in regards to the program itself should be shallow.
Using processing and would mean that 3D elements could be included, this wouldnt be possible with D3. However it does require a strong knowledge in 3D transformations which I do not possess. This may be a limiting factor in the speed at which I could understand the existing code and may push using this solution out of scope in favor of D3.

	D3 (Appendix A.1.2)	Processing (Appendix A.1.1)	Prefuse (Appendix A.1.3)
Potential for 3D	***No***	***Yes***	***No***
Has low learning curve	***Yes***	***Yes***	***No***
Prior evidence of successful visualisations	***Yes***	***Yes***	***Yes***
Interactive	***Yes***	***Yes***	***Yes***
Dynamic transitions	***Yes***	***Yes***	***Yes***
Has existing solution related to planets	***No***	***Yes***	***No***

Figure 5.1: Table of technology choices

5.1.1 System design and structure

5.1.2 Tools and artifacts used

5.2 Extension to initial design

Chapter 6

Visualisation Evaluation

Following the completion of the implementation stage of this project a final user evaluation was carried out on the visualisation to discover whether the visualisation designed and implemented

Chapter 7

Conclusions

7.1 Future Work

The work from this project can be taken further in many different ways depending on how it is intended to be used. There is the option of using the system as a terminal that users would use at an observatory or attraction where prior knowledge of the system is limmited and amount of time users would spend on the system would be small. In this case further expanding the user experience and improved Kinect interaction would be benificial as immersion would be the decider on its success. Another option for the system would be for a standalone desktop system that users would use multiple times and so prior knowledge of how to use the system could be expected. This would mean that more complex functionality could be introduced witht he expectation that it could be used by users. The systems current state could me modified to fit into either of these two options.

Occulus rift, look at paper boy

Bibliography