Rendering Invisibility

|  |  |
| --- | --- |
| Report Name | Design Specification |
| Author (User Id) | Katherine Rose Farmer (krf) |
| Supervisor | Bernie Tiddeman (bpt) |
|  |  |
| Module | CS39440 |
| Degree Scheme | G451 (Computer Graphics, Vision and Games inc. Industrial Placement) |
|  |  |
| Date | February 17, 2015 |
| Revision | 0.1 |
| Status | Draft |

# Introduction

The purpose of this document is to lay out a design for the system as a whole and for the invisibility cloak model. This specification is intended to inform me and my supervisor of the design I have envisioned for the project and as a reference for my future work. Another aim of this specification is to work out the feature list which I will be building from, according to the feature driven development I have chosen to work through. There will also be a section working through considerations I have had to make through research and limitations of what I have to work with. There will also be two overall designs, detailing the system design, including what I have already implemented, and the model design, which will be a design of different models I will aim to work through, applying my research to creating them.

# Design Considerations

## Assumptions and Dependencies

The language I am using for this project is Javascript, mainly working with WebGL and Three.js as this is a graphics project. Therefore I am making the assumption that it will work on all operating systems but may not work on all web browsers. I am expecting it to run on Chrome, Safari and Firefox due to these having good support for WebGL. I know that Opera has it disabled by default but should support WebGL, and Internet Explorer has still got some issues with WebGL. I will make sure to include support for the three browsers as required, but I will make less of an effort to make it work for Opera and IE.

I will need to decide on what rendering technique I am using, as the research I have done has concluded for me that I need to be able to do complex refractions and the basic refraction support in WebGL or Three.js has some trouble modelling complex refractions. This means I am looking into real-time rendering techniques, as these will make for a better demo. I have decided that my final rendering technique it likely to be either ray tracing or photon mapping. I will intend to make the decision soon, after some more research into the dependencies of the two techniques.

## Goals and Constraints

The goal of the system is to

# Feature List

# System Design

# Model Design

# Bibliography