# Games Technology Research Project Proposal

**Name:** Anthony Sturdy

**Supervisor Name:** Dr David White

**Course:** BSc Computer Games Programming

**Project Title:** Utilising raymarching and signed distance functions to render a scene of primitives

**Summary of project:**

This project will be a renderer which utilises the raymarching technique to render primitive objects (Sphere, cube, capsule, torus, etc) using signed distance functions (SDF). Raymarching steps along a ray until it collides with an object, it does this by stepping forward by the shortest distance to any object in the scene until the next distance is beneath a threshold, which is then considered to be a collision.

Due to the nature of using mathematical functions to define the shapes of objects, raymarching allows for perfectly smooth surfaces, object morphing, merging objects, and constructive solid geometry (CSG)/Boolean operations. Complex objects such as fractals are also able to be rendered. These effects are difficult to achieve in real-time using traditional rasterised rendering, but are either a by-product, or are achieved with little performance impact with raymarching. (Excluding fractals, the complexity of fractal SDF’s vary, and such does their performance).

(Michael Walczyk: <https://michaelwalczyk.com/assets/img/blog/ray-marching/complete-shader.gif>)

(Inigo Quilez: <https://www.iquilezles.org/www/articles/mandelbulb/mandelbulb.htm>)

(Jamie Wong: <https://www.shadertoy.com/media/shaders/4tcGDr.jpg>)

Other effects can be achieved through raymarching such as mirror reflections and refractions, which are difficult to achieve with rasterised rendering.

Games which use similar technology are:

* Claybook (<https://store.steampowered.com/app/661920/Claybook/>)
  + Uses SDFs
  + Integrated to Unreal Engine 4
* Marble Marcher (<https://github.com/WAUthethird/Marble-Marcher-Community-Edition>)
  + Raymarching fractal SDFs
  + Made using SFML
* Dreams (<https://www.playstation.com/en-gb/games/dreams/>)
  + Uses SDFs - “Point-splat” rendering technique
  + Hybrid rendering mixed with rasterising

Another common use of raymarching is for volumetric 3D objects, such as clouds. When using a rasterised renderer, clouds are usually made of many quads stacked upon one another to achieve depth, ray marching allows clouds/volumes with real depth.

(Felix/uhawkvr: <https://blog.uhawkvr.com/rendering/rendering-volumetric-clouds-using-signed-distance-fields/>)

**Outcomes:**

* A raymarching renderer/engine which is customisable by the user/developer.
* Multiple visual effects available for use to improve the visual fidelity of the scene.
* A modelling tool which allows the user to easily combine primitives and effects to create a cohesive 3D model, this model can then be saved and loaded into the engine for use.

**Ethics Form Required:** Disclaimer, Proportionate, Full

**Ethics Justification:**

No ethical considerations as I will not be working with, or collecting data, on other people/animals.

**Resources required:**

* Computer (specs used: Intel i9-9900k CPU, Nvidia GTX 970 GPU, Corsair 16GB DDR4 RAM)
* Mouse
* Keyboard
* Monitor
* Visual Studio 2019
* Git / GitHub