Luke Sanderson

Maths & Graphics 2

Bezier Racer

12/02/2017

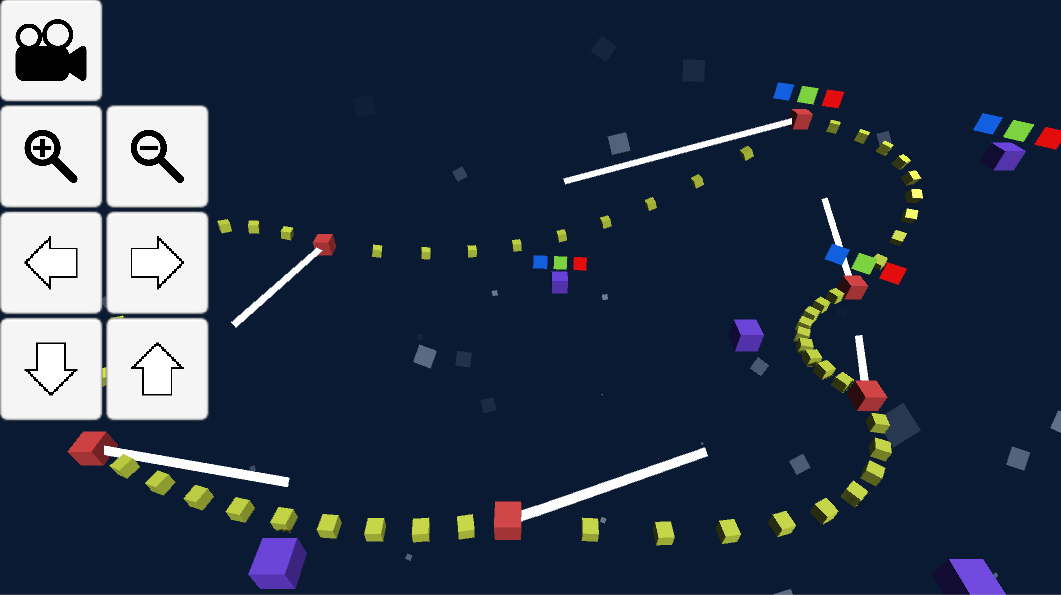
Introduction

*Bezier Racer* consists of a tool to create a tube-like course using various points of a Bezier curve, as well as a hover car-like racing game in order for the player to test out their new creation. The construction of the tube is done in real-time can be moved around on all 3 axes, as well as made more complex by adding additional nodes. Once the course is completed, a track is generated the player has the ability to drive on it.

I coded and designed the track editor and its ability to create ‘nodes’ along a Bezier path, as well as the hover car, including its modelling, texturing and functionality.

*Bezier Racer* could potentially be used as a track generation system – the result could easily be saved as a mesh and raced on later, for example, and the use of Bezier curves allows a lot of freedom, especially when the player is creating the track in real-time.

Track Creation



*Figure 1*

Figure 1 shows the general layout of the track generation scene.

The **RED** nodes are the crux of the track generation. Each node has the ability to find the next node in the loop, and the last to the first. The next node in the loop is shown by the white line emitted from the node. Each node creates 10 **CHILDNODES**, which are distributed evenly distributed from the node’s origin to the next. The Bezier points along a given line are found using this function:

public Vector3 GetBezier(float t)

{

bezierPoint.x = (1 - t) \* (1 - t) \* startPoint.x + 2 \* (1 - t) \* t \* controlPoint.x + t \* t \* endPoint.x;

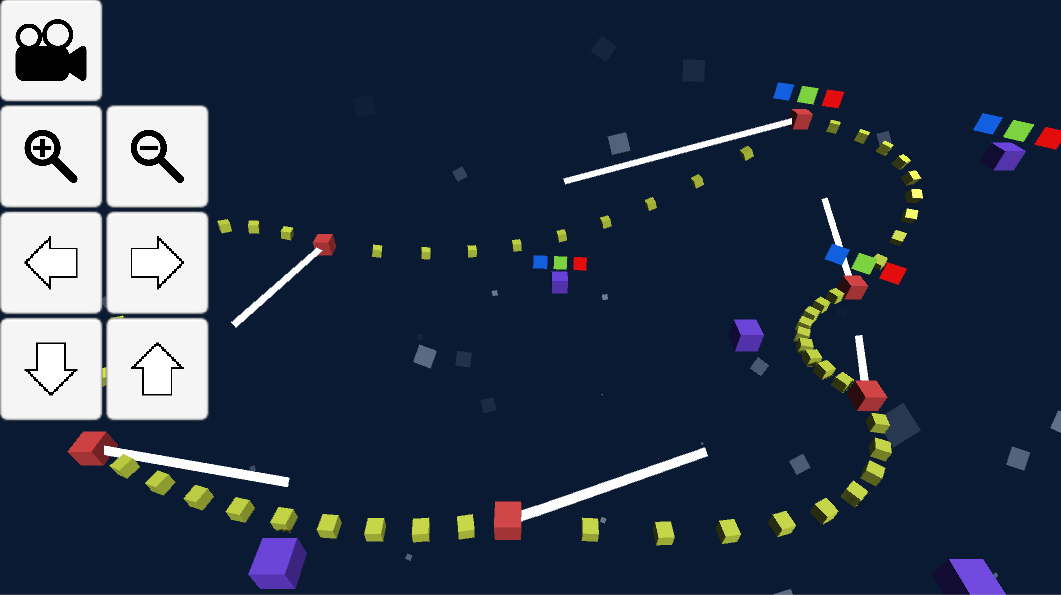
bezierPoint.y = (1 - t) \* (1 - t) \* startPoint.y + 2 \* (1 - t) \* t \* controlPoint.y + t \* t \* endPoint.y;

bezierPoint.z = (1 - t) \* (1 - t) \* startPoint.z + 2 \* (1 - t) \* t \* controlPoint.z + t \* t \* endPoint.z;

return bezierPoint;

}

Each of the child nodes are placed on this line depending on their position in the list of children.

The main method of moving and curving the track between nodes is by moving the **CONTROL NODES**. These (as well as the red nodes) can be moved by clicking on them to ensure they’re ‘selected’, and then by clicking and dragging the corresponding box to move the node in that direction – for example, by dragging the RED box, the player can move the node along the X axis. This is to ensure that the box can be moved properly around a 3D space without unwanted movement along a third axis.

On the left hand side of *Figure 1* we can see the Camera Controls, which can be opened and closed by using the top-left button. These keys allow the player to move the camera freely around the track.