Project4F17

From Immersive Visualization Lab Wiki

Contents

Project 4: Roller Coaster

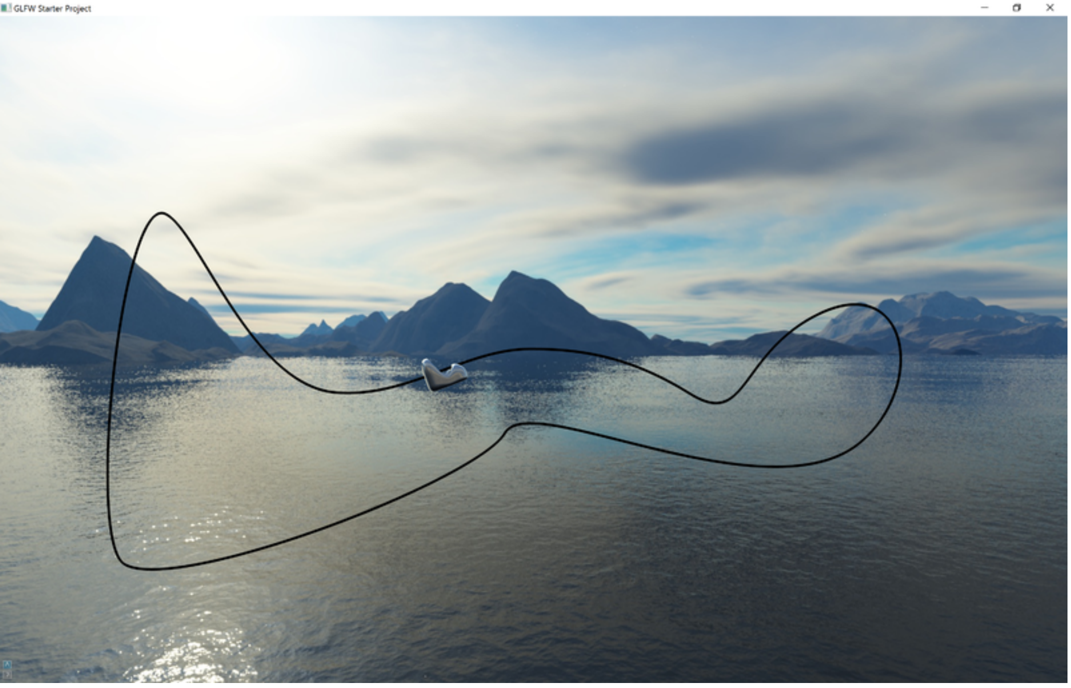
In this project, you need to create a track editor for a simple roller coaster. The roller coaster will consist of a 3D curve representing the track, and a sphere as a car. The track editor should work like the one in the video shown in class (https://www.youtube.com/watch?v=hlDYJNEiYvU) .

Start with your sky box from project 3 (feel free to use different textures), along with the code to manipulate the camera with keyboard and/or mouse. Also use your scene graph engine for this project if you can, it will make the project easier.

1. Track (30 Points)

Bezier Curves

- Create 8 connected cubic Bezier curves (i.e., 4 control points per curve). (20 points)
- Draw the Bezier curves by approximating them with at least 150 straight OpenGL segments each. (10 points)



Note that the screen shots are samples - your curves do not need to match the ones in these images exactly.

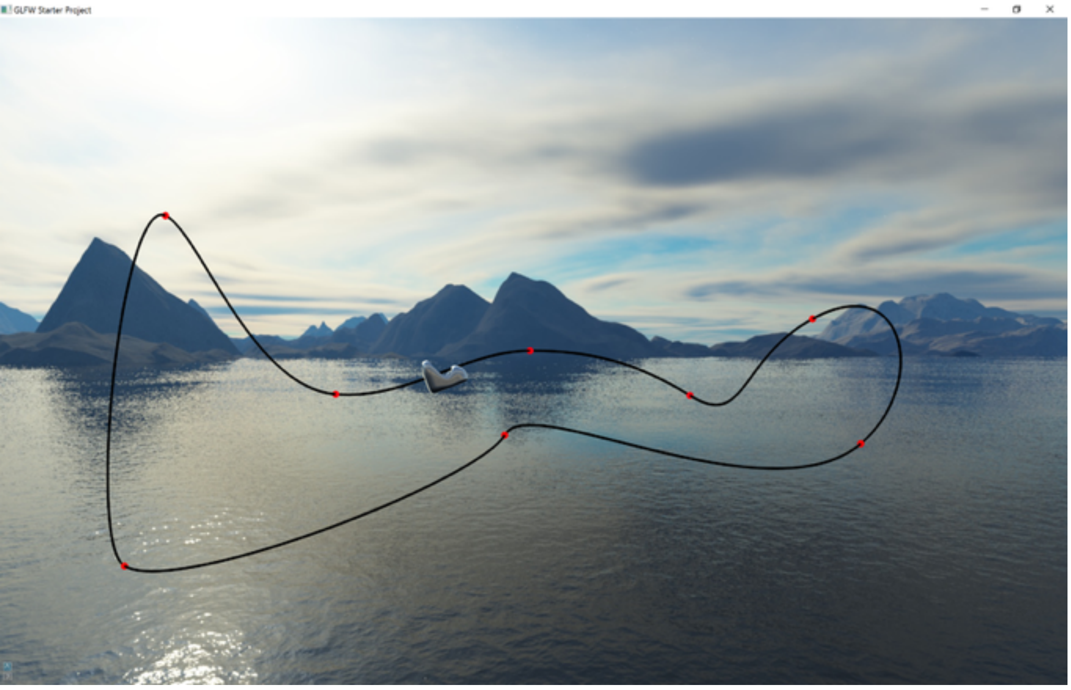
2. Control Handles (30 Points)

Make your control points editable by the user with the mouse. You need to support both the interpolating and the approximating control points.

Interpolating control points

These are the control points which the curve goes through, also called anchor points (points p0 and p3 for a given curve).

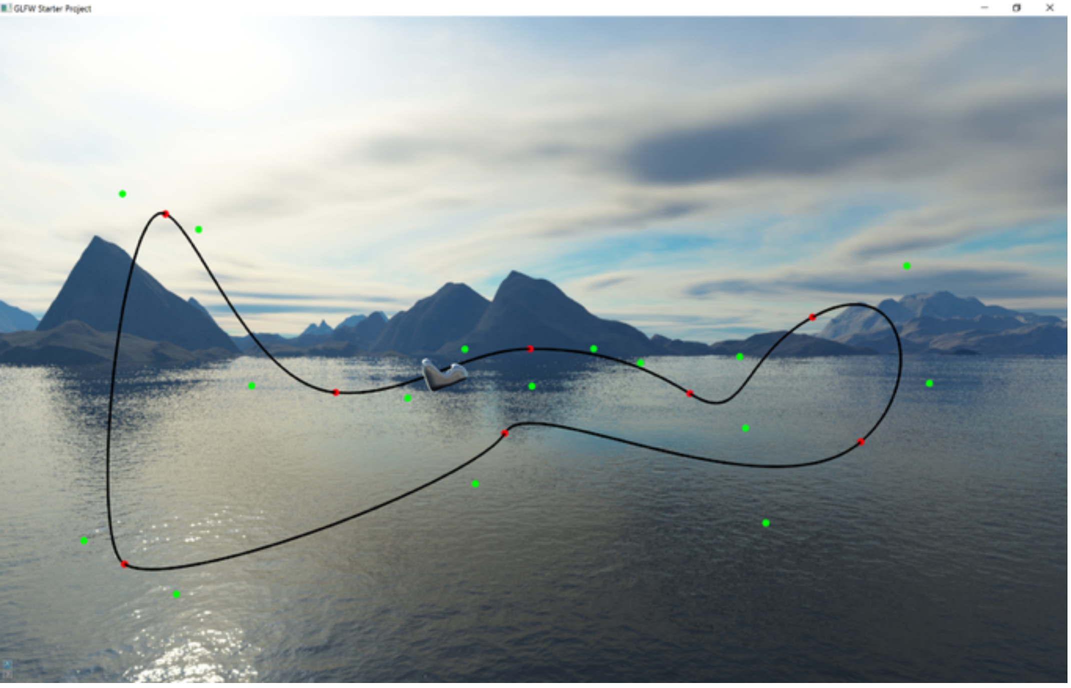
- Highlight the points with spheres or otherwise to make it easy to see where they are.



Approximating control points

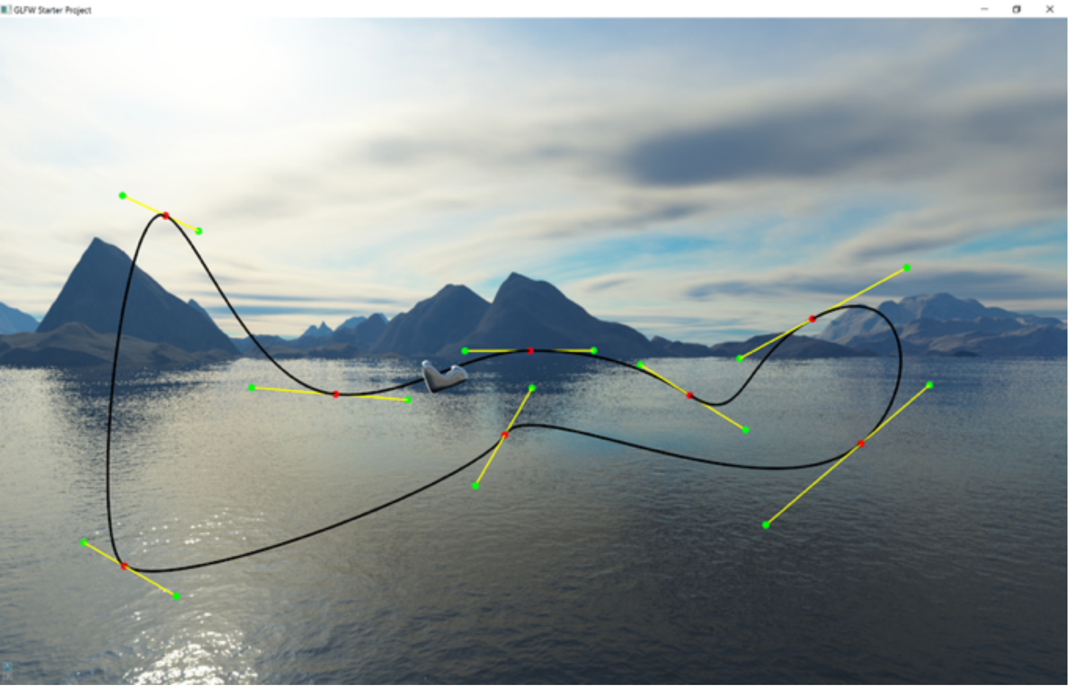
These control points "pull" on the curve but the curve does not go through them (points p1 and p2 for a given curve).

- Highlight these points in a similar way as the interpolating points.
- Use a different color or shape to distinguish them from the anchor points.



C1 Continuity

To ensure C1 continuity, you will need to support control handles like in the video. They connect the two last control points from one curve with the first two control points of the neighboring curve, and ensure that they lie on a straight line. Draw the control handles as lines.



Control Point Manipulation

- Use the control handles with the mouse to modify the control points.
  - To detect which control point is being modified, we recommend using ray casting from the mouse cursor, or a selection buffer (http://www.lighthouse3d.com/tutorials/opengl-selection-tutorial/) .
- Update the shape of the Bezier curve in real-time, just like the video demonstrates.
- The control point interaction has to be intuitive: when you pull a point with the mouse it should move in the same general direction as the mouse cursor, regardless of the camera position.

Grading:

- Displaying control points (5 points)
- Selecting correct point for mouse position (5 points)
- Correct grouping for control handles (5 points)
- Correct movement based on camera view (5 points)
- Updating curve (5 points)

- Ensuring C1 continuity (5 points)

3. Environment mapping (20 Points)

Use the sphere code you created for project 3 to render a solid sphere. It will represent the roller coaster car. Its surface should look like polished metal, so you need to use environment mapping on it.

The Cube map tutorial (http://learnopengl.com/#!Advanced-OpenGL/Cubemaps) explains how environment mapping is done with a sky box as the environment. Feel free to follow it and use the code from the site to make it work. The goal is to have the sky box reflect on the spere, to give the impression of the sphere's surface being polished metal (i.e., act like a perfect mirror).

Grading:

- Something resembling environment mapping (10 points)
- Correct environment mapping (10 points)

4. Physics (20 Points)

Sphere Movement (15 points)

The approach to determine the velocity of the sphere is to use a friction-less energy preservation model: the sum of the potential energy (proportional to the height the sphere is at) and the kinetic energy (determined by its velocity) remains constant. In other words: the lower the sphere is the faster it goes. This way, when the sphere starts at the highest point of your roller coaster, it should go through the entire track and end up at the highest point again without getting stuck.

Make the sphere slide along the roller coaster track like the ring in the video. Make sure your sphere moves at the correct speed even when different segments of your Bezier curve have different spacing between sample points. Use physics: measure the wall clock time since the last update (delta_time), then move the sphere according to this equation: distance_traveled = current_speed * delta_time.

Sphere Pause and Reset (5 points)

- Support two buttons - they can be keyboard keys, buttons on the input device, or even a GUI widget, as you choose:
    - One to pause/unpause the movement of the sphere.
    - One to automatically place the sphere at the highest point of your track, with a small amount of forward momentum.

5. Extra Credit (10 Points max.)

There are multiple options for extra credit. Each will get you a maximum of 10 points, but you can only get a combined maximum of 10 points of extra credit.

- **Virtual Reality:** Render your application to a virtual reality head set, and control the track's control points with the Touch controllers. You can use the Oculus Rift HMDs in the CSE VR lab (room B210). Email the instructor to get access to a Rift and controllers. (10 Points)

- **Improved Reflections:** Make the environment map more realistic by reflecting the track in it, in addition to the sky box. Do this by placing a virtual camera in the middle of the sphere, with a square image and a 90 degree field of view. Then take six pictures with the camera, facing in perpendicular directions to cover all six faces of a cube. These pictures now make up your sky box from this perspective. Then use environment mapping on the pod using this temporary sky box. This process has to be repeated for every frame rendered. (10 Points)

- **Improved Physics:** Add friction to your physics equation, gradually slowing down the sphere as it moves along the track. To keep it from coming to a standstill, make one of your Bezier curves an "accelerator segment", which gradually adds speed to the sphere as it moves along it. Draw this Bezier segment in a different color than the rest of them. (5 Points)
- **Rider's View:** Create a mode in which the camera is located on top of the sphere and always points forward along the track, simulating the view from the first row of a rollercoaster car. Support a button on the keyboard to switch between the regular camera and this special action camera. (5 Points)

You can also suggest a custom extra credit option and if it's granted we'll add it to the list.

- This page was last modified on 20 November 2017, at 14:28.