COSC363 Assignment1 Report

Student Name: Rui Zhang Id: 53491204

Mark points are **bold**.

The simple graphic scene displays a model **train(locomotive with three wagons)** running around an **oval track**. The train **runs continuously** around the track. It will **stop 5 seconds** when it arrives at the **train station**. Then it will run through a **tunnel**. The whole scene is surrounded by mountains and grassland.

Image1 train stops at train station

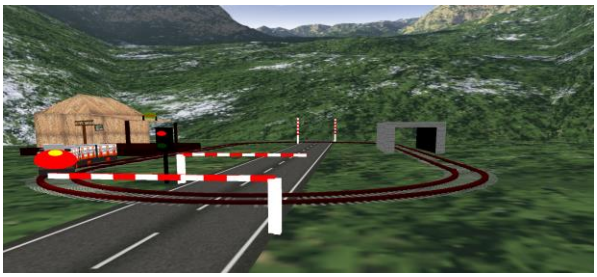

Image2 train is coming from train station

When the train is far away from the road, the **traffic light turns green**, and the **barriers' arms are automatically lifted**. It's possible for vehicles to drive across the railway.
When the train is coming towards the road, the **traffic light turns red**, and the **barriers' arms are shifted down**. Vehicles cannot pass the railway at the moment.
You can use **Up/Down/Left/Right/Page Up/Page Down** keys to control camera movements.

Features

1. An **oval track** was drawn using points that were provided in Oval.txt.

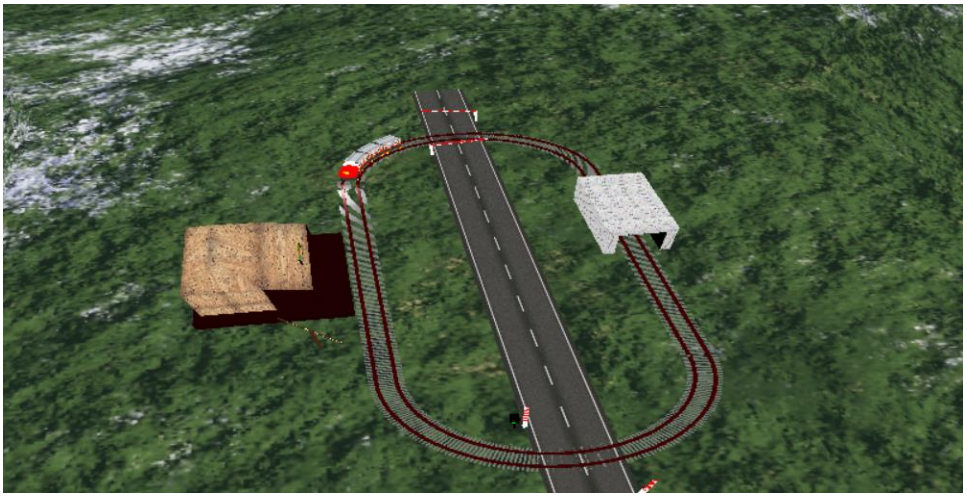

Image3 oval shape track

2. **Texture mapping is used in many objects** in this assignment. For example, train wagons are decorated with windows, train station is textured with wooden looking (*image4*). The gallery inside train station has four framed photos. They are textured with Christchurch scenery

photographs (*image5*). Other things include railway station sign that besides the station, road for vehicles, and tunnel concrete looking.

Image4 train station with wooden looking

Image5 train station gallery

3. From *image6*, a closer view of the train, we notice that there are black dots on red/yellow wheels. These patterns are designed to recognize the **self-rotation of train wheels**. The wheels rotate about z-axis certain angle. The angle changes over time, and when 360 degrees reached, it will be set back to 0 (*image8/9*).

Image6 close look at train wheels

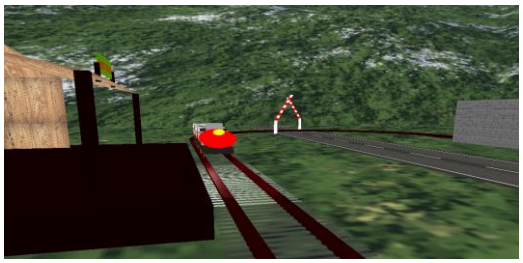

Image7 train lightens tracks

```
508
509     GLUquadric* q = gluNewQuadric();   //Disc
510
511     glPushMatrix();
512     for (int i = 0; i < 4; i++){
513         glPushMatrix();
514         glTranslatef(wx[i], 2.2, wz[i]);
515         glRotatef(wheelRotateAngle, 0., 0., 1.);
516
517         glPushMatrix();
518         glColor4f(0.5, 0., 0., 1.0);    //Wheel color
519         gluDisk(q, 0.0, 1.0, 20, 2); // outer r = 1.0 unit
520         glPopMatrix();
521
522         glPushMatrix();
523         glColor4f(1., 0.5, 0., 1.0); // yellow
524         glTranslatef(0., 0., wz1[i]);
525         gluDisk(q, 0.0, 0.7, 20, 2); // inner r = 0.7 unit
526         glPopMatrix();
527
528         glPushMatrix();
529         glColor4f(0., 0., 0., 1.0); // black
530         glTranslatef(0., 0.3, wz2[i]);
531         gluDisk(q, 0.0, 0.2, 20, 2);
532         glPopMatrix();
```

Image8

```
167 void myTimer(int value)
168 {
169     glutPostRedisplay();
170     if (timeClock < trainStayPeriod) {
171         timeClock++;
172     }
173     else {
174         if (icurr < NPTS) {
175             icurr += 1;
176             if (icurr == NPTS) {
177                 icurr = 1;
178             }
179         }
180
181         wheelRotateAngle+=60;
182         if (wheelRotateAngle > 360) wheelRotateAngle = 0;
```

Image9

4. There is a **spotlight on the top front of engine**. The yellow disk on train's head represents the light. It is toward the track and follows the train's movements. In *Image7* above, the track in front of train is shiner than other track segments.
5. **Four barriers and one signal light** were drawn. The movement of barrier arms and the change of signal light followed train's positions. A position variable and 'if' conditions are used to control these objects (*Image11*).

Image10 barrier arm are lifting up

```
184     // angles control left barrier arm
185     if (icurr >= 0 && icurr < 75) {
186         armAngleLeft += 1.2;
187     }
188     else if (icurr >= 76 && icurr < 200) {
189         armAngleLeft = 90.;
190     }
191     else if (icurr >= 200 and icurr <= 275) {
192         armAngleLeft -= 1.2;
193     }
194     // angles control right barrier arm
195     if (icurr >= 265 && icurr < 340) {
196         armAngleRight += 1.2;
197     }
198     else if (icurr >= 340 && icurr <= 415) {
199         armAngleRight = 90.;
200     }
201     else if (icurr > 415 or icurr <= 0) {
202         armAngleRight -= 1.2;
203     }
```

Image11 code for barrier arm movements

6. There are **three view modes** that you could shift by using **'c' key**: **driver's view, view from signal light and view from train station**. *Image1* shows the view set from signal light position. *Image12/13* are screenshots from the driver's view. One is when train is entering tunnel, the other is when train is entering train station.

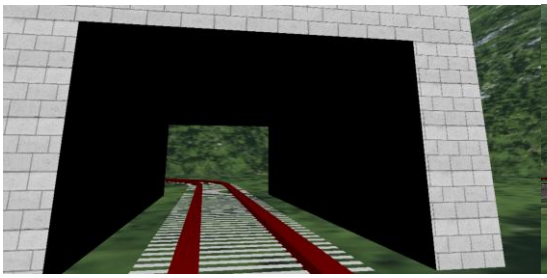

Image12

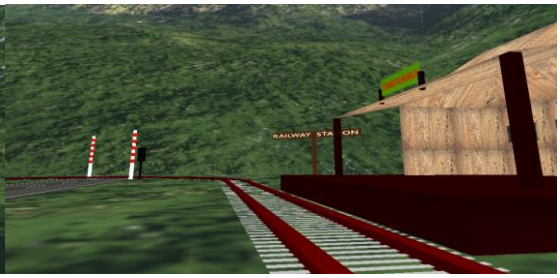

Image13

7. **GLU/GLUT objects and polygonal shapes** are created and assembled to form objects in the scene. **Surface generation** is used for drawing tunnel.
8. **Sky box** and **collision detection**, controller cannot move view out of box.

Control functions

Except minimum requirements:

1. **Mouse click** left side, height goes up. Mouse click right side, height goes down.
2. Key 'C' for shifting three view modes.

Build and Run Command

g++ -Wall -o "assignment1" "assignment1.cpp" -lm -lGL -lGLU -lglut

./assignment1

Reference

https://www.cgtrader.com/3d-models/exterior/exterior-public/train-station-778144de-9cfb-4dc0-bfc4-6c06a291d405
https://www.afar.com/places/christchurch-botanic-gardens-christchurch
https://en.wikipedia.org/wiki/Christchurch
https://www.google.co.nz/search?q=road+texture
https://www.khronos.org/opengl/