CS 550 -- Fall Quarter 2015

Introduction to Computer Graphics Final Project Report

Need for Speed

Instructor: Prof. Mike Bailey

Xuhao Zhou

December 2015

1. Overview:

In this project, a player can control a vehicle to race against the computer.

2. Features:

(1). One player can choose one of vehicles that include supercar, police car and helicopter for racing.

(2). One player can choose difficulty. If a player chooses hard, the vehicles controlled by computer will be faster.

(3). The top light of police car is able to flash.

(4). One player can control the vehicles to move forward or backward, and to turn right or turn left.

(5). Vehicles may have insight view.

(6). The camera follows the player's vehicle

(7). A player can use rear view function.

(8). The program can show the result of competition.

(9). I draw some star objects, windmill objects and animated flower to adorn the scene.

3. What I did for this project:

(1). A player can choose one of three vehicles include one helicopter, one police car and one supercar.

Implement: I use a script program named obj2opengl.pl got from the Internet to convert the police car object file and the supercar object file into two .h files. These .h files include the object's vertexes array and normal vector array. I use these .h files to draw objects in the OpenGL program.

(2). A player can choose difficulty. If the player chooses hard, the rival vehicles move faster.

Implement: I use animation method to control the speed of rival vehicle. I set all vehicles to move along the Z-axis. -Z direction is forward and Z direction is backward. For example,

In Animate function, I set `rivalmove = rivalmove + 0.25`. In Display function, I set below code:

```
glTranslatef(-6, 0, -rivalmove);
DrawPoliceCar();
```

Therefore, the rival vehicle moves automatically and its speed can be controlled.

(3). The top light of police car is able to flash.

Implement: first, I use multiple cube objects to make a cuboid. Then, I use point light to make it flash blue or red light. Here, I use animation method again.

```
// Animation of flash light
int ms = glutGet( GLUT_ELAPSED_TIME );
ms %= MS_PER_CYCLE;
   Time = (float)ms / (float)( MS_PER_CYCLE - 1 );
// Below code used for flashing red light:
SetPointLight( GL_LIGHT1, 0.15, 2.3, -4., 1., 0., 0. );
if( Time<0.5 )
   glEnable( GL_LIGHT1 );
else
   glDisable( GL_LIGHT1 );
// Below code used for flashing blue light:
SetPointLight( GL_LIGHT2, -0.15, 2.3, -4., 0., 0., 1. );
if( Time>0.5 )
   glEnable( GL_LIGHT2 );
else
   glDisable( GL_LIGHT2 );
```

Then, I bind this flash light box to the top of the police car object.

(4). A player can control the vehicles to move forward or backward, and to turn right or turn left.

Implement: move forward or backward is stated in (2), I will explain how to turn right and turn left here. `MyRightWay` indicates the vehicle is on which side of the road. If `MyRightWay` is a true, it means player's vehicle is on the right side of the road, otherwise, it is on the left side of the road. For the player's vehicle, it is initially moving on the right side. I use `glTranslatef(-12.5, 0, 0)` to make it moving on left side. For the rival's vehicle, it is initially moving on the left side. I use `glTranslatef(12.5, 0, 0)` to make it moving on right side. When the player press 'a' key, it sets `MyRightWay` as false, and the player's vehicle will move on the left side. When the player press 'd' key, it sets `MyRightWay` as true, and the player's vehicle will move on the right side. In addition, the player can use 'w' key to make the vehicle moving forward and 's' key to move it backward.

```
// below code is a vehicle setting of player
    if (MyRightWay==true){
        glPushMatrix();
        glTranslatef(5, 0, -move);
        DrawCalifornia();
        glPopMatrix();
    }
    else{
        glPushMatrix();
        glTranslatef(-12.5, 0, 0);
        glPushMatrix();
        glTranslatef(5, 0, -move);
        DrawCalifornia();
        glPopMatrix();
        glPopMatrix();
    }
```

(5). A player can press 'i' key to use the insight view if the helicopter vehicle is chosen.

Implement: if the player presses 'i' key, the camera will be set in the cockpit.

```
    // ViewOn == 1 means it is inside view
    if( ViewOn == 1){
    glLoadIdentity( );
    gluLookAt( 6.0, 8.8, -4.9-move, 6.7, 8.8, -10.5-move, 0., 1., 0. );
    }
```

(6). The camera follows the player's vehicle

Implement: I use the 'move' variable to record the moving distance of the vehicle. Because the moving is along the Z-axes, we can use gluLookAt function and 'move' variable to implement this function. Such that:

```
 gluLookAt( 6.0, 8.8, -4.9-move, 6.7, 8.8, -10.5-move, 0., 1., 0. );
```

(7). A player can press 'r' key to use rear view function.

Implement: we can also use 'move' variable and gluLookAt function to implement this function. Eye position is set as in front of the vehicles. The position of the reference point is the intermediate position of the player vehicle and the rival vehicle.

```
 // rear view
 if (ViewOn==2){
    glLoadIdentity( );
    gluLookAt( -2., 12., -17.-move, -2., 7., -10.-move, 0., 1., 0. );
}
```

(8). The program can show the result of competition.

Implement: I use 'move' and 'rivalmove' variables to respectively record the moving distance of player vehicle and rival vehicle. Once one party of player vehicle and rival vehicle reaching 200, the program shows that the player is victorious or not.

```
 // Print result.
glDisable( GL_DEPTH_TEST );
glMatrixMode( GL_PROJECTION );
glLoadIdentity( );
gluOrtho2D( 0., 100.,    0., 100. );
glMatrixMode( GL_MODELVIEW );
glLoadIdentity( );
glColor3f( 1., 1., 1. );
if (move >= 200 && rivalmove <= move && Result == 0)
  Result=1;  // win
else if(rivalmove >= 200 && move < rivalmove && Result == 0)
  Result=2;  // the second
```

```
if (Result==1)
  DoRasterString( 5., 5., 0., "You Win!" );
else if (Result==2)
   DoStrokeString( 5., 5., 0., 5.0, "You are second! Try again? (Press t)");
```

(9). I draw some star objects, windmill objects and animated flower to adorn the scene.

4. How my project differs from what my proposed:

This project is match with my proposal, even added more features.

5. Any impressive cleverness:

(1). A player can control the vehicles to move forward or backward, and to turn right or turn left.

(2). The camera follows the player's vehicle.

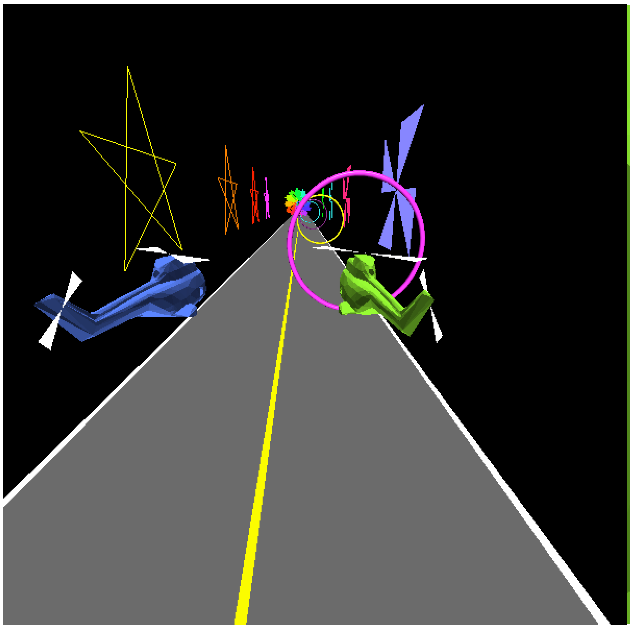
(3). A player can use rear view function.

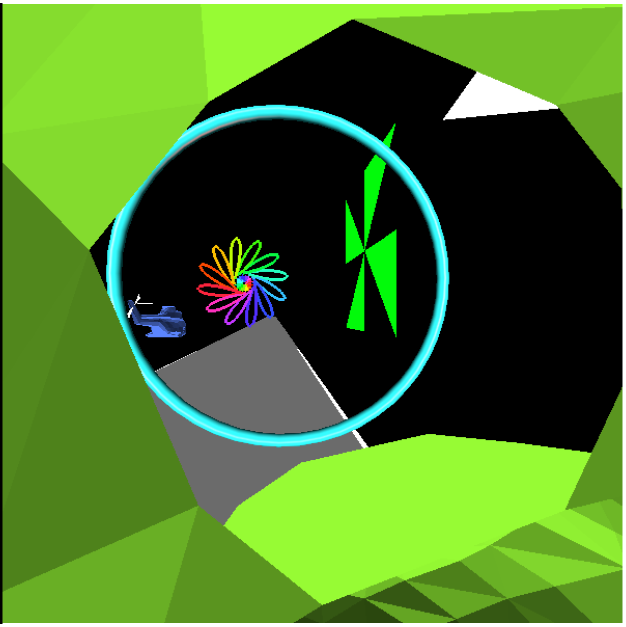
6. What I learned from doing this project:

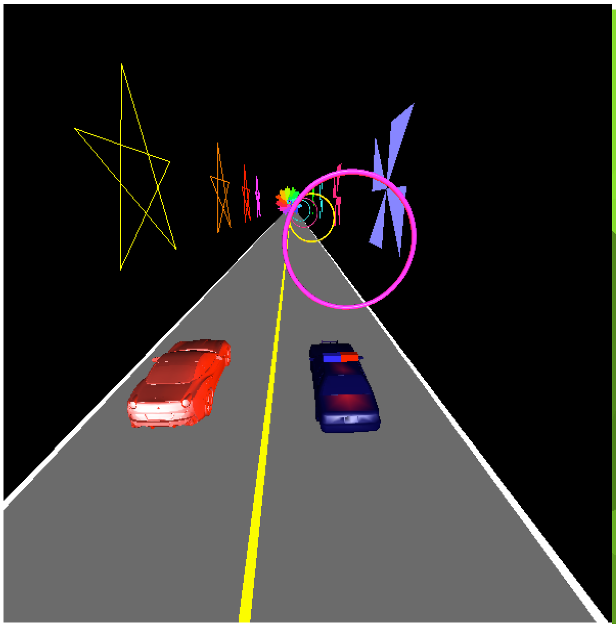
(1). I learned how to convert an .obj file into vertexes array and normal vector array that can be used by OpenGL program.

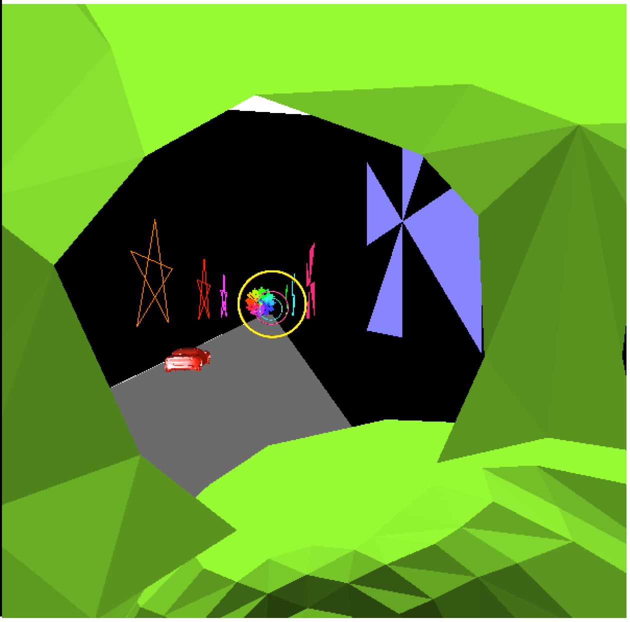
(2). I learned how to devise a simple game using OpenGL.

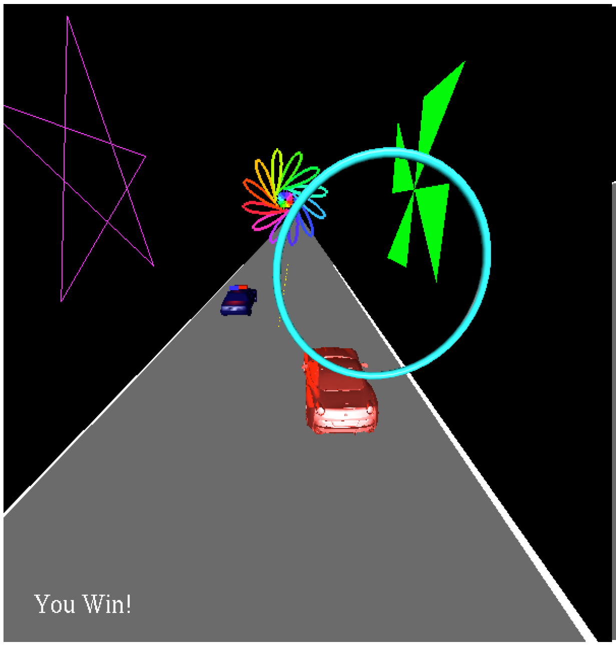
7. Any images that are especially representative of what I did:



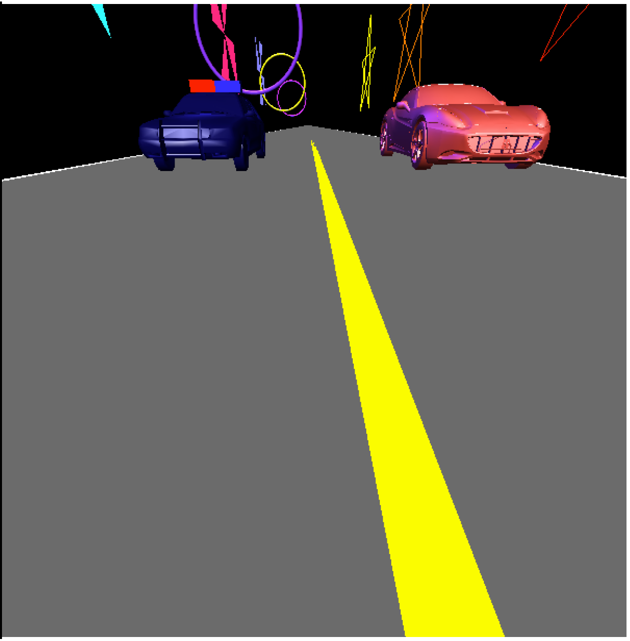


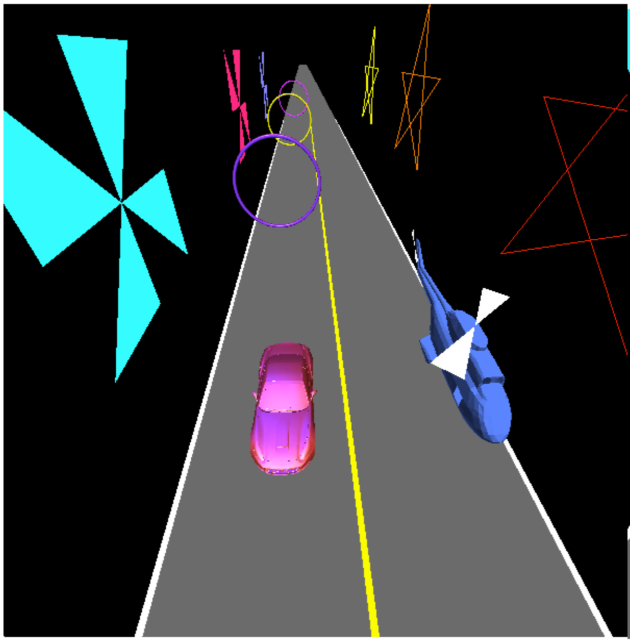


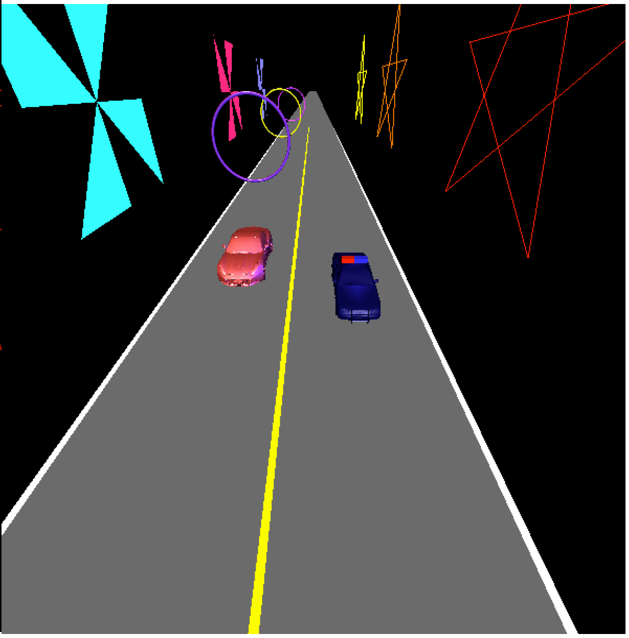


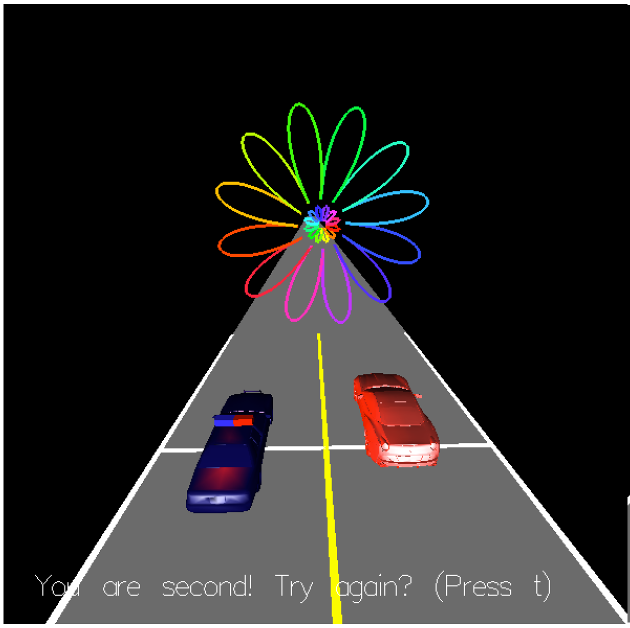


You Win!

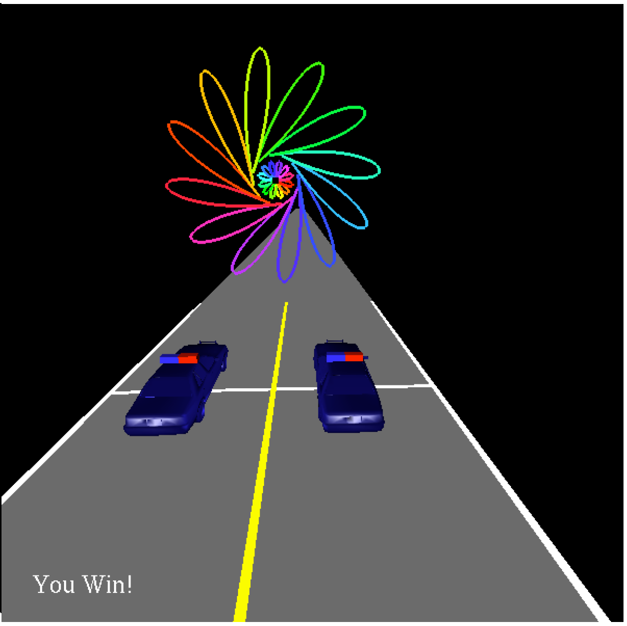








You are second! Try again? (Press t)



You Win!