**CS550 – Fall Quarter 2016**

**Introduction to computer Graphics**

**Final Project Report**

**Solar System**

Instructor: Prof. Mike Bailey

Xiao Tan

December 4 2016

**Proposal:**

In this final project, I am going to simulate the solar system, which include nine planets, with OpenGL. Each planet, except Sun, have scale just as the real solar system, and each planet’s orbital period is proportional as real life.

Requirements:

* Animation:

All plants will rotate and revolve as they in the space. The orbital period can’t the same as real life, but the I will give them a good scale. In addition, I can use keyboard to control the animation stop or move.

* Texture:

There is no doubt every planet will get an appropriate texture. The textures also can be controlled.

* Lighting:

The sun is the only light in this solar system, each planet will have a bright side and a dark side shows in the window. The light can be controlled and it can work on textures or non-textures.

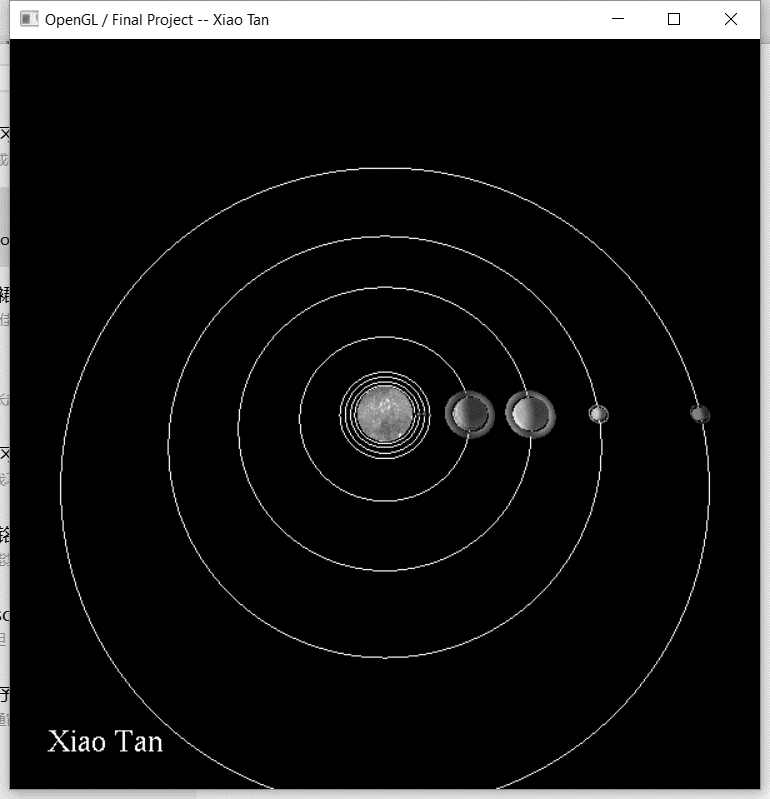
* Others:

Besides the requirements above, it also need a good outside viewing option where we can see everything.

**Implement:**

In this project, I simulated each main planet in our solar system with OpenGL, which include Sun and eight planets, Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus, Neptune.

First of all, I implement each planet with MjbSphere() function, with glTranslatef() move to the the right place. The following picture shows the result.



Because Mercury, Venus, Earth and Mars are too small to the others planets, so we can see their orbits, and when we zoom in, we can see them correctly.

Then we need to do the texture for each planet. In this step, Because I use the mjbsphere, so texture coordinates are already correctly mapped. What I need to do is load texture files and setup them to the right place. With the functions, void

LoadTexture(char\* filename, GLuint\* texture) and void

init\_texture(), I can load the bmp files and transfer them to texture data and bind in Graphic memory right place, waiting for program call them. This function also give textures their parameters.

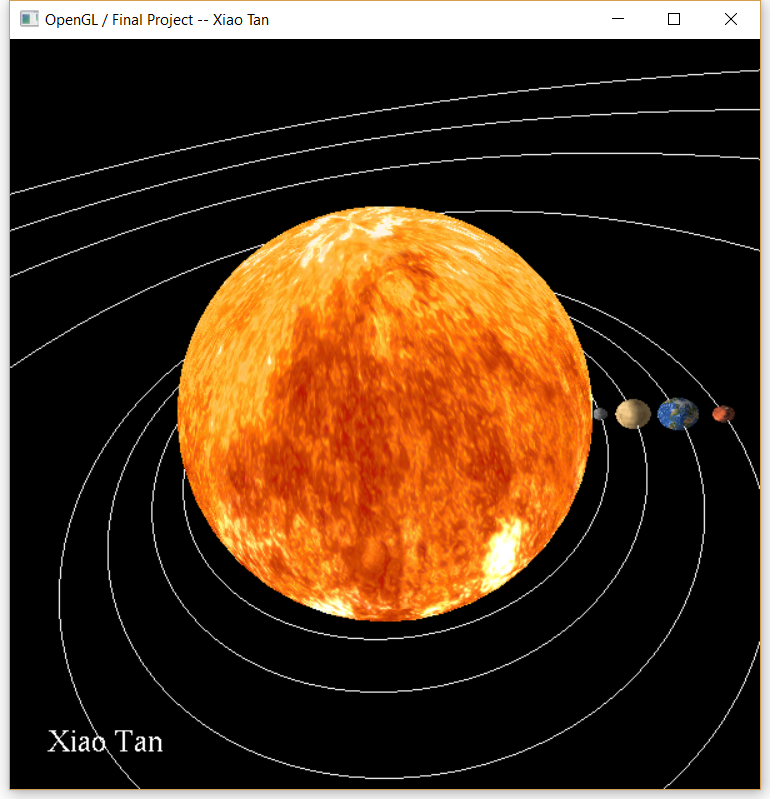
glTexParameteri(GL\_TEXTURE\_2D, GL\_TEXTURE\_WRAP\_S, GL\_MODULATE);

glTexParameteri(GL\_TEXTURE\_2D, GL\_TEXTURE\_WRAP\_T, GL\_MODULATE);

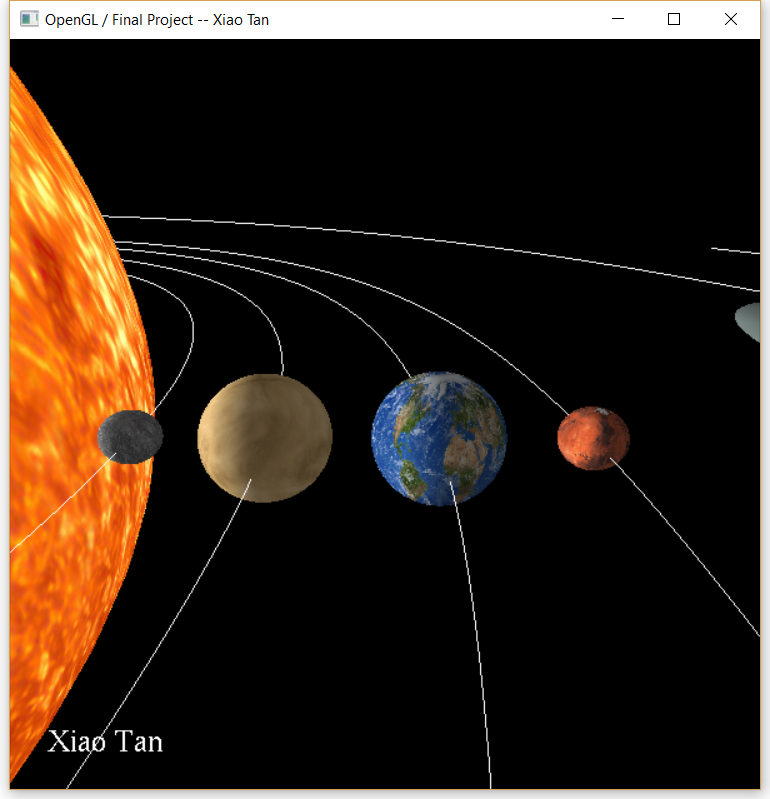
glTexParameteri(GL\_TEXTURE\_2D, GL\_TEXTURE\_MAG\_FILTER, GL\_LINEAR);

glTexParameteri(GL\_TEXTURE\_2D, GL\_TEXTURE\_MIN\_FILTER, GL\_LINEAR);

Here I use GL\_MODULATE, it works for lighting. The following picture shows partial of the planets textures.



After finishing texture, I added lighting to the scene. For the solar system, lighting is very easy, because there is only one light in the solar system, which is Sun. The other planets only do reflecting. So I only open one light in Sun’s position is OK. There is one thing need to notice, which is lighting should turn on after drawing Sun, otherwise, Sun will be lost its brightness. The lighting result shows in the following picture, earth has an obvious bright side and a dark side.



Everything is placed correctly, but they are static. So, the next thing is to do animation. By searching the internet, I got each planet’s rotation and revolution period and direction. To do animation, I used

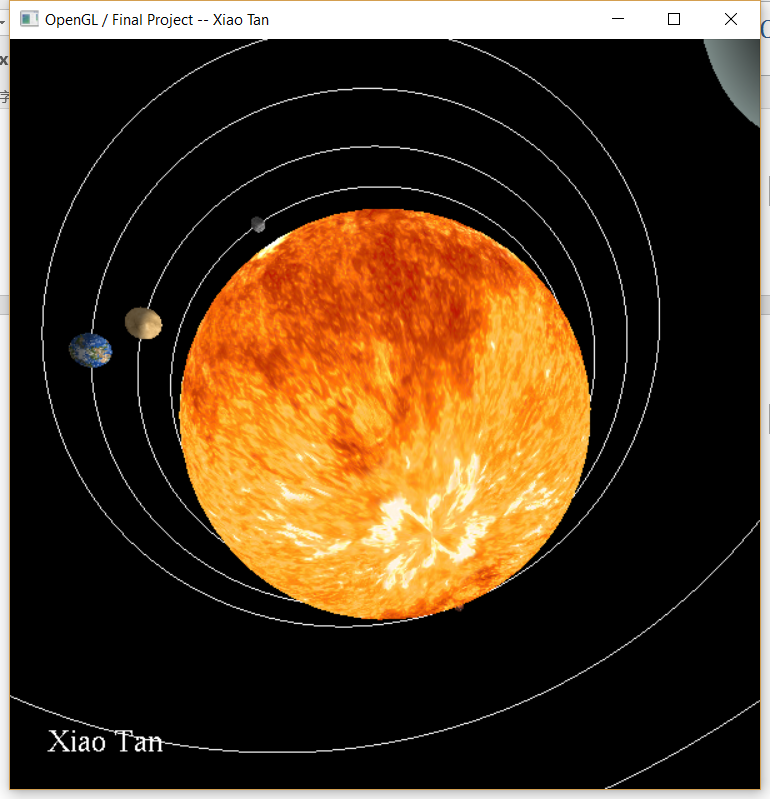
int ms = glutGet(GLUT\_ELAPSED\_TIME);

ms %= MS\_PER\_CYCLE;

Time = (float)ms / (float)MS\_PER\_CYCLE; // [0.,1.)

To get a periodic number Time. By calculating the rates of each planet’s rotation and revolution period, with glRotatef() and rates\*Time, that can make each object rotate in a right cycle.

The following picture shows, partial of the scene. Mercury, Venus, Earth, and Mars are changing their position.



After animation, I have already finish the core of my project, and I also did some other in this object. To see each planet after zoom in, I make it can change eye position and look at point by using keyboard. It makes me easier to adjust point of view to see the whole scene. In addition, I used two variables “isTexture”, “Light0On” to control textures and lighting.

**Things different from proposal:**

I did everything I planned to do, and I added keyboard control to make my program friendlier.

**Conclusion:**

After doing this project, I combined everything I learned from this class in my project. I learned something very basic, such as how to draw 3D objects in a blank scene, then I learned how to texturing and lighting. I also learned some advanced techniques, like shader, vulkan and the very popular thing, 3D printer. For this project, I still have some ideas to add shadow in the scene, and add some fire effects on Sun, make it looks more real. However, I don’t have enough time to learn these things, so I will learn more about these in the future, then make this scene better.