COMP3069 Computer Graphics Project Report

Telephone Box Scene

Jaesung Park

20121762

2020-12-21

Shyjp1@nottingham.edu.cn

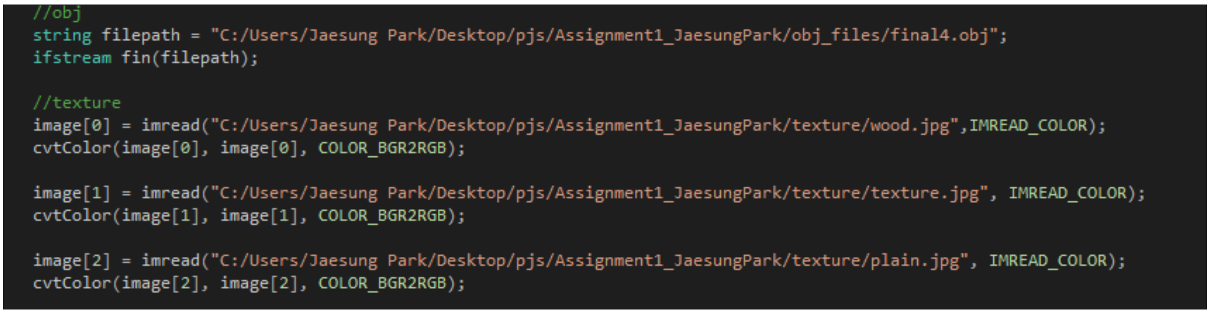
School of Computer Science University of Nottingham Ningbo China

AIM

The aim of this project is to create a 3D scene to demonstrate computer graphics techniques taught in COMP3069 lectures and labs. My personal aim for this project is to demonstrate the telephone box scene, which has a sensational mood.

ENVIRONMENT

The project is implemented on Visual studio 2013 32bit and used C language. Most of the objects are implemented by using Blender. The project also includes the OpenCV library to import textures. Libraries are not included in submission due to memory limit, so *opencv2/OpenCV.hpp* has to be included on a running computer. Plus, paths for object and image files have to be changed. There is a total of four paths to be changed in the main function, three texture files, and one object file as shown in Figure 0.



```
//obj
string filepath = "C:/Users/Jaesung Park/Desktop/pjs/Assignment1_JaesungPark/obj_files/final4.obj";
ifstream fin(filepath);

//texture
image[0] = imread("C:/Users/Jaesung Park/Desktop/pjs/Assignment1_JaesungPark/texture/wood.jpg",IMREAD_COLOR);
cvtColor(image[0], image[0], COLOR_BGR2RGB);

image[1] = imread("C:/Users/Jaesung Park/Desktop/pjs/Assignment1_JaesungPark/texture/texture.jpg", IMREAD_COLOR);
cvtColor(image[1], image[1], COLOR_BGR2RGB);

image[2] = imread("C:/Users/Jaesung Park/Desktop/pjs/Assignment1_JaesungPark/texture/plain.jpg", IMREAD_COLOR);
cvtColor(image[2], image[2], COLOR_BGR2RGB);
```

Figure 0. Four paths that should be changed

IMPLEMENTATION

<u>Camera</u>

The camera functionality is implemented on my own and it is using via two classes in source code, *void processNormalKeys,* and *void processSpeicalKeys.* The included keys are the following: Up, Right, Left, Down, Page up, Page down, and Esc. The Up and Down buttons are simply moving forward and backward on the scene and the Page up and down is for moving in y coordinates. For the Left and Right buttons, there is only an angle included looking around on the current position. Lastly, the Esc button is to exit from the scene. The camera able us to provide a different viewpoint of the scene

Implementation

The scene consists of the moon, a restaurant building, and a telephone box as seen in Figure 1. The moon is one of two moving objects and since the moon is moved via *translatef* from the origin to the position and rotated, it is moving around the restaurant slowly in a circular way. To move in every second, the *timer* class was also implemented to change the angle each second.

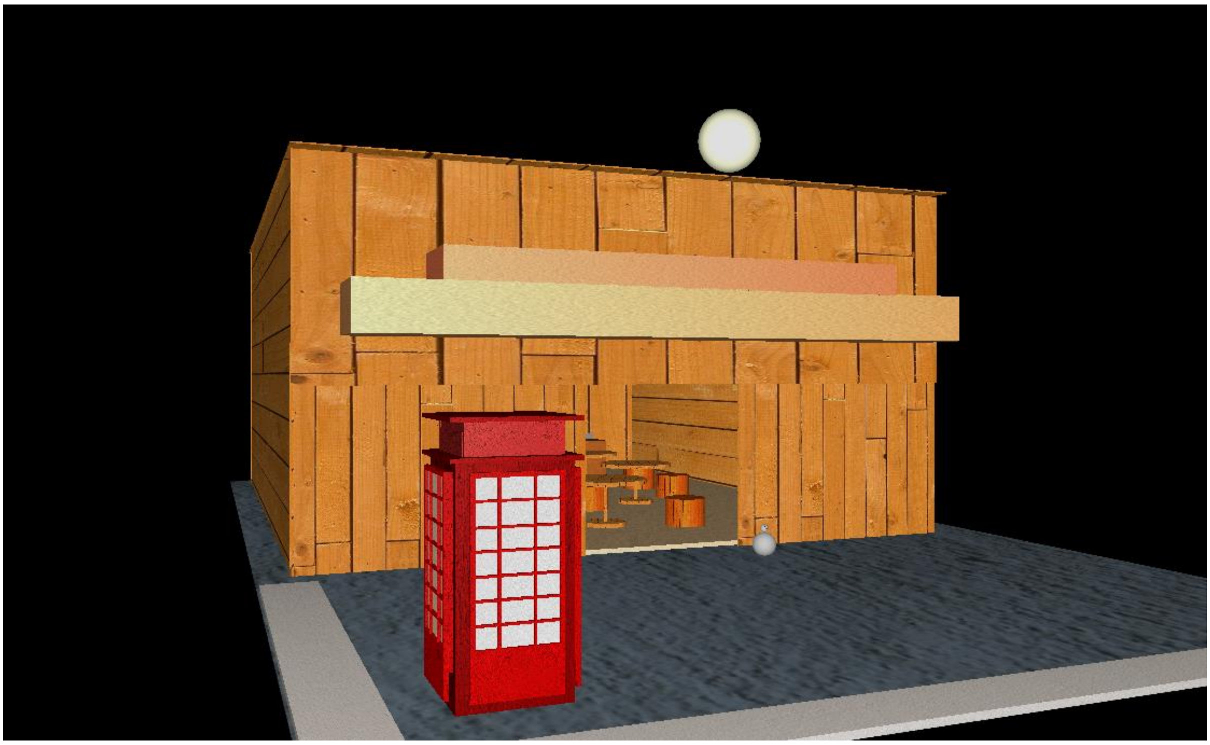


Figure 1. Overview of the scene

The purpose of putting a moon is to put a more sensational mood in the scene that I first planned. Along with the same purpose, the background is not covered with any texture because the pure black color is more adequate for the scene.

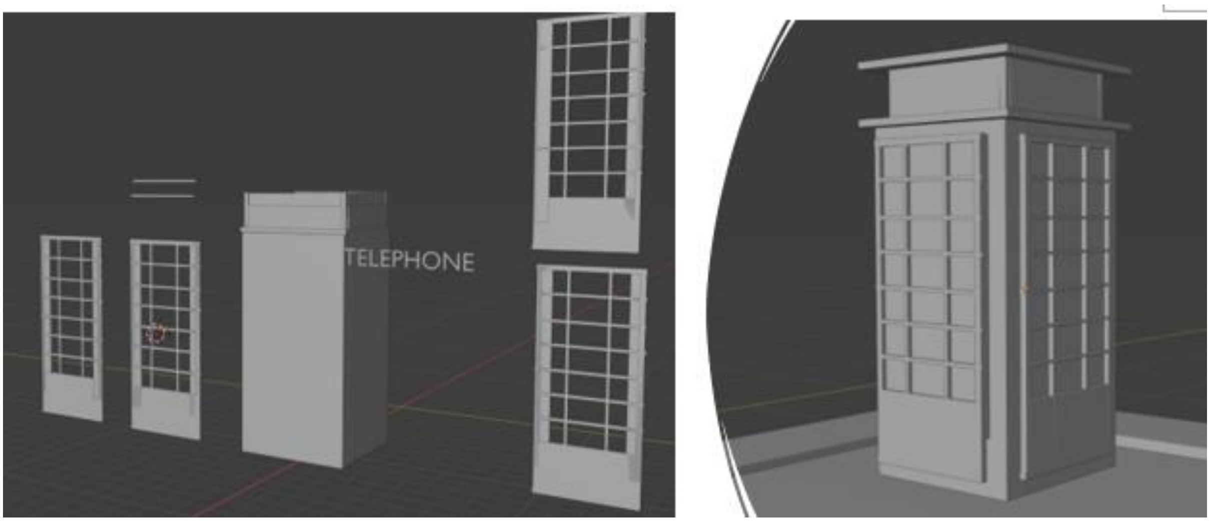
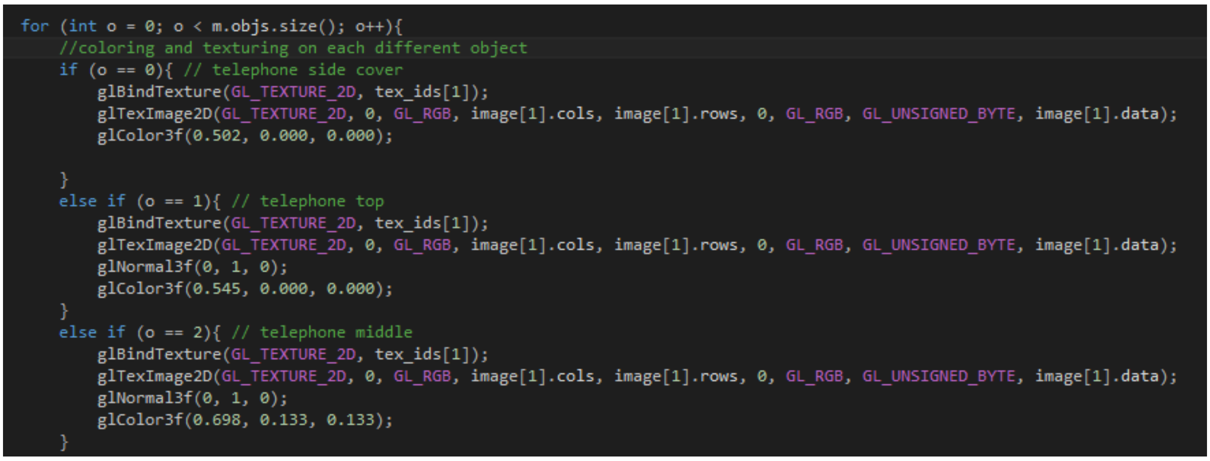


Figure 2. Telephone box object from Blender



```
for (int o = 0; o < m.objs.size(); o++){
    //coloring and texturing on each different object
    if (o == 0){ // telephone side cover
        glBindTexture(GL_TEXTURE_2D, tex_ids[1]);
        glTexImage2D(GL_TEXTURE_2D, 0, GL_RGB, image[1].cols, image[1].rows, 0, GL_RGB, GL_UNSIGNED_BYTE, image[1].data);
        glColor3f(0.502, 0.000, 0.000);

    }
    else if (o == 1){ // telephone top
        glBindTexture(GL_TEXTURE_2D, tex_ids[1]);
        glTexImage2D(GL_TEXTURE_2D, 0, GL_RGB, image[1].cols, image[1].rows, 0, GL_RGB, GL_UNSIGNED_BYTE, image[1].data);
        glNormal3f(0, 1, 0);
        glColor3f(0.545, 0.000, 0.000);
    }
    else if (o == 2){ // telephone middle
        glBindTexture(GL_TEXTURE_2D, tex_ids[1]);
        glTexImage2D(GL_TEXTURE_2D, 0, GL_RGB, image[1].cols, image[1].rows, 0, GL_RGB, GL_UNSIGNED_BYTE, image[1].data);
        glNormal3f(0, 1, 0);
        glColor3f(0.698, 0.133, 0.133);
    }
```

Figure 3. Implementation of coloring and texturing on different objects

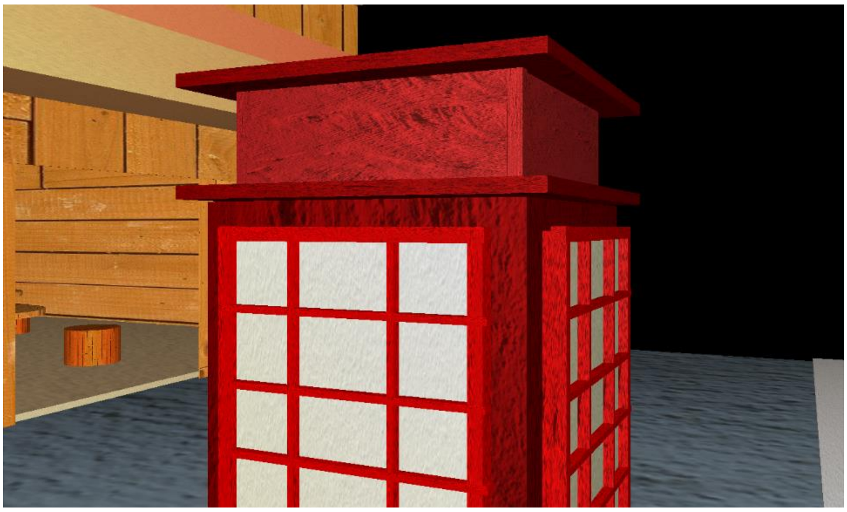
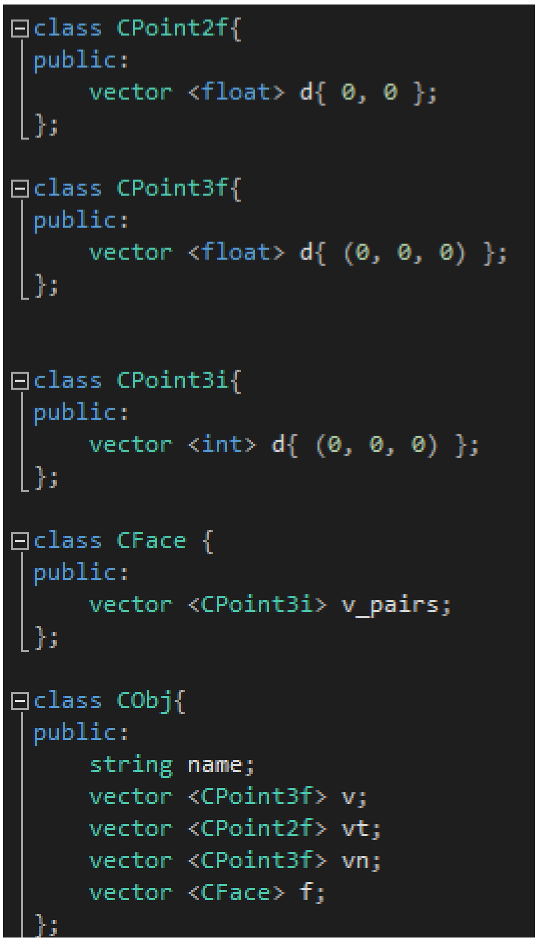


Figure 3.1 Different red colors on the telephone box

Figure 2 shows the telephone-box object imported from Blender. There is more than one object is included in a telephone box to give different colors and textures. As shown in Figure 3, each telephone box objects are imported separately and colored in a different color. Figure 3.1 is showing different colors on the telephone box. It is meant to give more diversity to one whole object.



```
class CPoint2f{
public:
    vector <float> d{ 0, 0 };
};

class CPoint3f{
public:
    vector <float> d{ (0, 0, 0) };
};


class CPoint3i{
public:
    vector <int> d{ (0, 0, 0) };
};

class CFace {
public:
    vector <CPoint3i> v_pairs;
};

class CObj{
public:
    string name;
    vector <CPoint3f> v;
    vector <CPoint2f> vt;
    vector <CPoint3f> vn;
    vector <CFace> f;
};
```

Figure 4. Object created

The way I imported blender objects is to firstly create Obj class including name, vector *v*, vector *vt*, and vector *vn* and pairs of followings as shown in Figure 4. Each line of obj textfile had a different set of numbers so it is captured with different *CPoint* classes. By reading each line of obj text files, each *v*, *vt*, and *vn* are placed into vectors for later reading. With these file readings, I could easily handle each object's color and texture. Detailed implementation can be found in class *Display* and *CModel*.

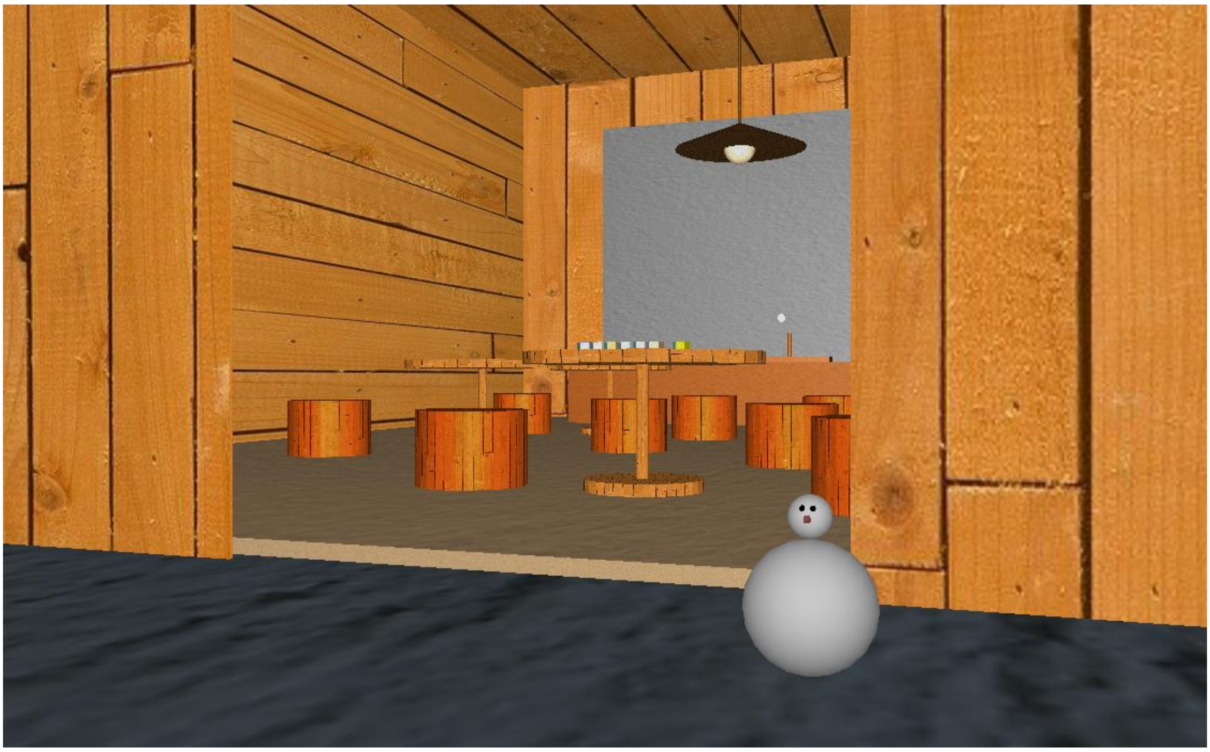


Figure 5. The entrance of the restaurants

There is a restaurant's snowman decoration at the entrance. The snowman is not an imported object from Blender, but it is implemented using all transformation of model techniques (scaling, translating, and rotating).

Figure 6. Inside of the restaurant

Inside the restaurant, there are desks, chairs, a light bulb, book decorations, a magic toy ball for decoration which is rotating, and more as shown in Figure 6. By inputting different colors on each book, the scene becomes more colorful and fulfilled. The magic ball between the books is meant to show the ball is moving because of the wind coming out of its stand. There is no scientific evidence that is possible (probably, it needs wider wind ejection (a stand) to blow wind to make it possible in real life) but it is only a decoration of the restaurant.

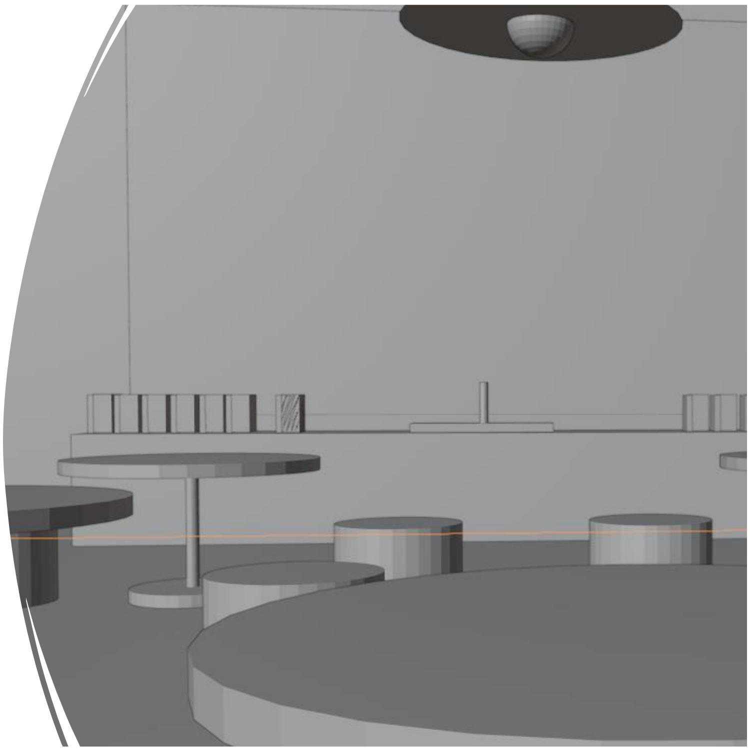


Figure 7. Blender Object of the restaurant.

The way of the toy implementation is that the stand (wind ejection) between books is implemented via Blender as shown in Figure 7 and the ball itself is implemented on the code. Translate and rotate are used to place into the right above of the stand.

Light

All light components, ambient, diffuse, and specular, are also being used in the scene. The light ambient set as a low number since the scene is meant to be in the nighttime, but diffuse is getting set as a high number so that the restaurant components can be clearly viewed. Material components are also getting set for the same reason as light components, but there is a shininess of 25 to give a more dramatic view of the scene.

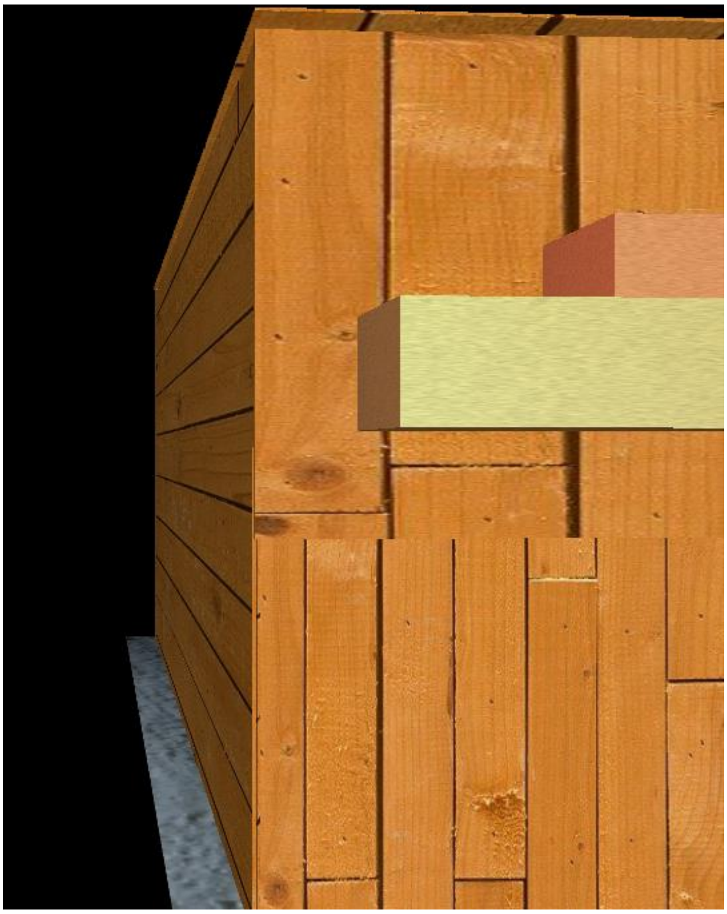


Figure 8. Difference in brightness

Figure 8 is showing the difference in brightness between the restaurant's sidewall and front wall. The same texture and color are set on both, but the scene is showing different color on the front and side because of the brightness.

Texturing

The texture is imported via OpenCV library and the jpeg image files are from Pixabay. I tried to place simple textures to maintain my mood in the scene. Restaurant walls, chairs, and desks are using the same wood texture, and the ground and restaurant floor are using rough texture. On other objects, I used a plain texture that does not have dynamic texture to maintain its own feel.

REFLECTION

The first obstacle I faced was importing multiple objects from Blender. I realized after importing only one object, if I import one single obj file, I can only use one color and texture. Since my code is implemented using pure C, it was a little hard to find a way to import objects from Blender. There was no specific example of loading obj files (on pure C) on the internet. By getting an idea from example code (C++) on our lectures labs and some video lectures from other universities (C), I managed to successfully import objects with C. After learning how to import object files, it was straight forward since our class lecture was well explained and ppt files were useful.

Before learning this module, I had no idea how to create a scene or a single line with OpenGL and Blender. I am very satisfied with my scene when I think of myself who could not do anything a few months ago. This coursework let my desire of how to build a 3D model or Computer graphic fulfilled and I enjoyed doing it since there was no strict guideline like "you should draw this or that". Honestly, this is the best module this semester that I learned so much and now I can create a scene from nothing. Thank you.