Project Report

Project Title : 3D Classroom

Course Code : CSE-4110

Submitted To : **Dr. Md Manjur Ahmed**
Associate Professor
Department of Computer Science and Engineering,
University of Barishal.

Submitted By

Student 1

Name : Afrin Hayat

ID : 19CSE001 (Exam Roll : CSE044/7)

Session : 2018-19

Department : Computer Science and Engineering

Student 2

Name : Md Moonaz Rahman

ID : 19CSE023 (Exam Roll : CSE014/7)

Session : 2018-19

Department : Computer Science and Engineering

Date of Submission : 20 March , 2024

Rubric:

Course Outcome	**Notes**	**Marks**
CO1: Demonstrate the concept of computer graphics and ability to use the computer graphics technology.		25 marks
<u>Overall program structure:</u> - Free from error	5 marks	
<u>OpenGL Function:</u> - Apply functions of keyboard handling a. Rotation b. Translation c. Scaling -Apply functions of texture/color mapping	15 marks 5 marks	
CO2: Construct 2D and 3D graphics by implementing concepts of computer graphics and computer graphics programming.		25 marks
<u>Object:</u> -All required objects are provided	10 marks	
<u>Texture Mapping:</u> - Apply functions of texture/color mapping for appropriate color	5 marks	
<u>Camera / Lighting:</u> -Appropriate technique is implemented and suitable with the environment	10 marks	

CO3: Respond to instruction by listening actively and give feedback			25 Marks
-Camera moving -Animation -Answer the questions regarding basic of computer graphics		10 marks 10 marks 5 marks	
CO4: Work together effectively to achieve the same goal by building a good relationship and interaction among team members.			25 Marks
Student 1 **Name:** **Afrin Hayat** ID: 19CSE001	Objects (% of contribution) = 50%	5 Marks	
	Camera (% of contribution) = 50%	5 Marks	
	Animation (% of contribution) =50%	5 Marks	
	Report (% of contribution) =50%	10 Marks	
Student 2 **Name:** **Md. Moonaz Rahman** ID: 19CSE023	Objects (% of contribution) =50%	5 Marks	
	Camera (% of contribution) =50%	5 Marks	
	Animation (% of contribution) =50%	5 Marks	
	Report (% of contribution) =50%	10 Marks	

Table of Contents

INTRODUCTION

The 3D Classroom project aims to create a virtual classroom environment using computer graphics techniques. This project demonstrates various concepts of computer graphics and provides an interactive platform for users to explore a simulated classroom setting.

OBJECTIVES OF PROJECT

The objective of this project is to create a 3D visualization of a classroom environment using OpenGL. The scene includes various elements such as desks, chairs, a teacher's podium, a board, a door, windows, and even a rotating fan. The project aims to demonstrate basic 3D modeling and rendering techniques using OpenGL, allowing users to interact with the scene by adjusting the camera view, scaling, translating, and rotating objects, as well as animating certain elements like the fan. Overall, the project provides a visual representation of a classroom setting, showcasing the capabilities of OpenGL for 3D graphics programming.

FEATURE AND SCOPE

The project has following features:

- **3D Room Environment:** A virtual classroom is constructed with walls, floor, and ceiling to simulate a real-world setting.
- **Dynamic Objects:**
    - **Teacher's Podium:** Using the keyboard, the podium
        - Can be **Translated** along **x,y and z axes**;
        - Can be **Rotated** along **x,y and z axes**;
        - Can be **Scaled** along **x,y and z axes**;
    - **Fan:** There is a fan that is placed in the center of the room that is moving automatically, but we can control the on/off function with keyboard command.
    - **Door:** Users can interactively open and close it using keyboard controls.
- **Camera Control:** Users can rotate the camera view to explore different angles of the classroom.

OBJECTS OF 3D CLASSROOM

1. **Teacher's Podium :** Standard table dimensions with rectangular surface and four legs.
2. **Student desks :** There are 4 tables for the students that have the same dimensions as the teacher's podium, but they are different in size.
3. **Chairs :** There are 4 chairs for students, which are placed according to the corresponding student desks. Ecah chair has four legs , one seat, and one back support.
4. **Windows :** There are two windows in the classroom, which are positioned to the left and right walls of the room.
5. **Blackboard:** One blackboard is placed in the front wall of the room.
6. **Door :** The door of the classroom is positioned in the back wall of the room
7. **Fan:** Fan is attached at the ceiling of the room

FUNCTIONS OF OBJECTS

1. **Teacher's Podium :** Translation , rotation and scaling is enabled through the keyboard.
2. **Fan :** Rotating automatically , can also be controlled manually.
3. **Door :** Door can be opened and closed using keyboard.

SYSTEM REQUIREMENTS

- **Operating System:** Windows / Linux / macOS
- **Compiler:** GCC or any compatible compiler that supports OpenGL libraries
- **Library:** OpenGL Library
- **IDE:** Codeblocks

PROGRAM STRUCTURE

The project is structured as a single file and organized into distinct sections for :

1. **Object rendering** : Using the display() function, object rendering is completed.

2. **Animation and user interaction**: The special() function is used to accomplish animation and interaction.

3. **Camera control:** The gluLookAt() function is used inside the display() method to accomplish this.

Keyboard Controlling/Manual

Scaling Teacher Desk:
Use 'x' for Increase Teacher Desk about x-axis scale
Use 'X' for decrease Teacher Desk about x-axis scale
Use 'y' for increase Teacher Desk about y-axis scale
Use 'Y' for decrease Teacher Desk about y-axis scale
Use 'z' for increase Teacher Desk about z-axis scale
Use 'Z' for decrease Teacher Desk about z-axis scale

Translation Teacher Desk:
Use '4' for Translate along the (+ve) x-axis
Use '5' for Translate along the (-ve) x-axis
Use '6' for Translate along the (+ve) y-axis
Use '7' for Translate along the (-ve) y-axis
Use '8' for Translate along the (+ve) z-axis
Use '9' for Translate along the (-ve) z-axis

Rotation Teacher Desk :
Use 'a' for Rotating around x-axis positive
Use 's' for Rotating around x-axis negative
Use 'd' for Rotating around y-axis positive
Use 'f' for Rotating around y-axis negative
Use 'g' for Rotating around z-axis positive
Use 'h' for Rotating around Z-axis positive

Animation :
Use 'F' to rotate the fan and Stop fan.

Keyboard Control :
Use 'o' to Open and Close Door.

SOURCE CODE

In this section, we provide key code snippets from our 3D Classroom project to illustrate its functionalities and implementations:

Keyboard controlling code:

```
void myKeyboardFunc( unsigned char key, int x, int y )
{
   switch ( key )
   {

   case 'r':                    // Rotate Camera view Right to Left
      rot++;
      break;

   case 'R':                    // Rotate Camera View Left to Right
      rot--;
      break;

   case 'u':                    // Camera View Up
      eyeY++;
      break;

   case 'U':                     // Camera view Down
      eyeY--;
      break;
```

```
case '-':                       // Zoom Out
   eyeX=0;
   eyeZ++;
   lookZ++;
   break;

case '+':                       // Zoom In
   eyeZ--;
   break;

case 'x':                       // Increase x-axis scale
       scale_factor_x += 0.1;
       break;
case 'X':                       // Decrease x-axis scale
       scale_factor_x -= 0.1;
       break;
case 'y':                       // Increase y-axis scale
       scale_factor_y += 0.1;
       break;
case 'Y':                       // Decrease y-axis scale
       scale_factor_y -= 0.1;
       break;
case 'z':                       // Increase z-axis scale
       scale_factor_z += 0.1;
       break;
case 'Z':                       // Decrease z-axis scale
       scale_factor_z -= 0.1;
       break;


case '4':                        // Translate along the positive x-axis
   translate_x += 0.1;
   break;
case '5':                       // Translate along the negative x-axis
   translate_x -= 0.1;
   break;
case '6':                       // Translate along the positive y-axis
   translate_y += 0.1;
   break;
case '7':                       // Translate along the negative y-axis
   translate_y -= 0.1;
   break;
```

```
    case '8':                       // Translate along the positive z-axis
        translate_z += 0.1;
        break;
    case '9':                       // Translate along the negative z-axis
        translate_z -= 0.1;
        break;


    case 'F':
        fRotate = !fRotate;
        axis_x=0.0;
        axis_y=1.0;
        break;

    case 'a':                       // Rotate around x-axis positive
        rotate_x += 5.0;
        break;
    case 's':                       // Rotate around x-axis negative
        rotate_x -= 5.0;
        break;
    case 'd':                       // Rotate around y-axis positive
        rotate_y += 5.0;
        break;
    case 'f':                       // Rotate around y-axis negative
        rotate_y -= 5.0;
        break;
    case 'g':                       // Rotate around z-axis positive
        rotate_z += 5.0;
        break;
    case 'h':                       // Rotate around z-axis negative
        rotate_z -= 5.0;
        break;
    case 'o':                       // Open door
        rotate_door += 180.0;
        break;

    }
}
```

Functions of keyboard handling (Rotation, Translation, Scaling):

```
void teacher()
{
   float length = 0.5;
   float height = 3;
   float width = 0.5;

   glPushMatrix();
   glTranslatef(length / 2 + translate_x, translate_y, translate_z);
   glRotatef(rotate_x, 1, 0, 0);
   glRotatef(rotate_y, 0, 1, 0);
   glRotatef(rotate_z, 0, 0, 1);
   glScalef(scale_factor_x, scale_factor_y, scale_factor_z);
   glScalef(length, height, width);
   glTranslatef(0, 0, 0);
   cube(0.0, 0.0, 0.0);                    // 1st right leg for the desk
   glPopMatrix();

   glPushMatrix();
   glTranslatef(length / 2 + translate_x, translate_y, translate_z);
   glRotatef(rotate_x, 1, 0, 0);
   glRotatef(rotate_y, 0, 1, 0);
   glRotatef(rotate_z, 0, 0, 1);
   glScalef(scale_factor_x, scale_factor_y, scale_factor_z);
   glScalef(length, height, width);
   glTranslatef(4, 0, 0);
   cube(0.0, 0.0, 0.0);                    //1st left leg for desk
   glPopMatrix();

   glPushMatrix();
   glTranslatef(length / 2 + translate_x, translate_y, translate_z);
   glRotatef(rotate_x, 1, 0, 0);
   glRotatef(rotate_y, 0, 1, 0);
   glRotatef(rotate_z, 0, 0, 1);
   glScalef(scale_factor_x, scale_factor_y, scale_factor_z);
   glScalef(length, height, width);
   glTranslatef(0, 0, 3);
   cube(0.0, 0.0, 0.0);                    // 2nd right leg for desk
   glPopMatrix();
```

```
    glPushMatrix();
    glTranslatef(length / 2 + translate_x, translate_y, translate_z);
    glRotatef(rotate_x, 1, 0, 0);
    glRotatef(rotate_y, 0, 1, 0);
    glRotatef(rotate_z, 0, 0, 1);
    glScalef(scale_factor_x, scale_factor_y, scale_factor_z);
    glScalef(length, height, width);
    glTranslatef(4, 0, 3);
    cube(0.0, 0.0, 0.0);                    // 2nd left leg for desk
    glPopMatrix();

    glPushMatrix();
    glTranslatef(length / 2 + translate_x, translate_y, translate_z);
    glRotatef(rotate_x, 1, 0, 0);
    glRotatef(rotate_y, 0, 1, 0);
    glRotatef(rotate_z, 0, 0, 1);
    glScalef(scale_factor_x, scale_factor_y, scale_factor_z);
    glScalef(length * 6, height / 6, width * 5);
    glTranslatef(0, 6, 0);
    cube(1, 0.8, 0.4);                      // top of the desk
    glPopMatrix();
}
```

Functions of keyboard handling (Door open & close):

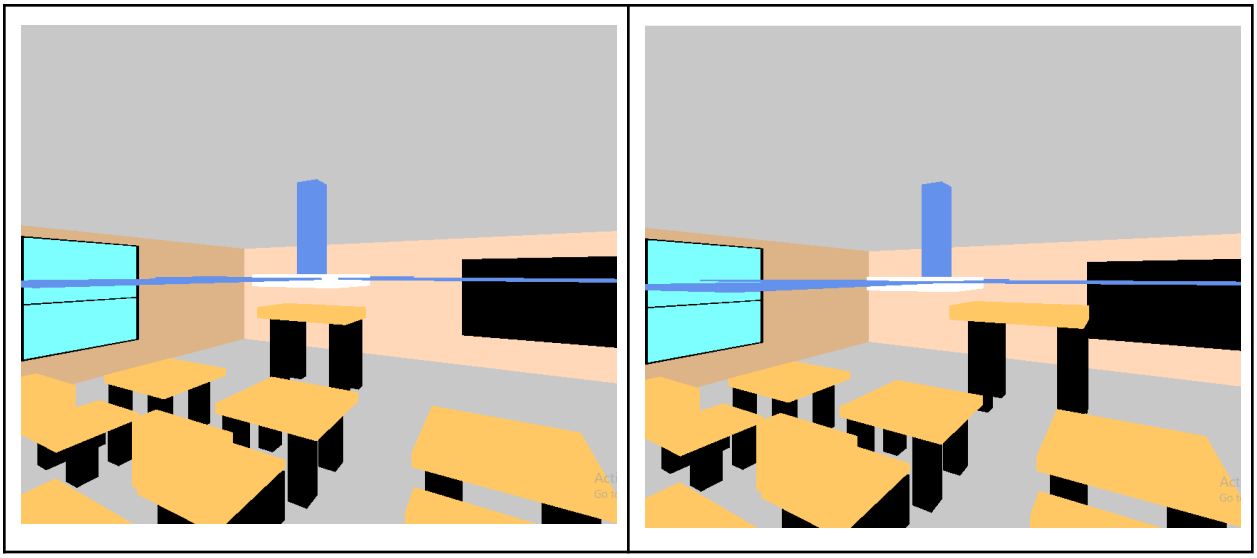
```
GLfloat rotate_door = 0.0;
void door()
{
     glPushMatrix();
    glScalef(3.8,10,.2);
    glTranslatef(2.3,0,-75);
    cube(0, 0, 0);
    glPopMatrix();

    glPushMatrix();
    glScalef(3.8,10,.19);
    glTranslatef(2.2,0,-75);
     glRotatef(rotate_door, 0, 1, 0);          // Rotate around y-axis
    cube(.8, .7, .5);
    glPopMatrix();

}
```

Animation for fan spin:

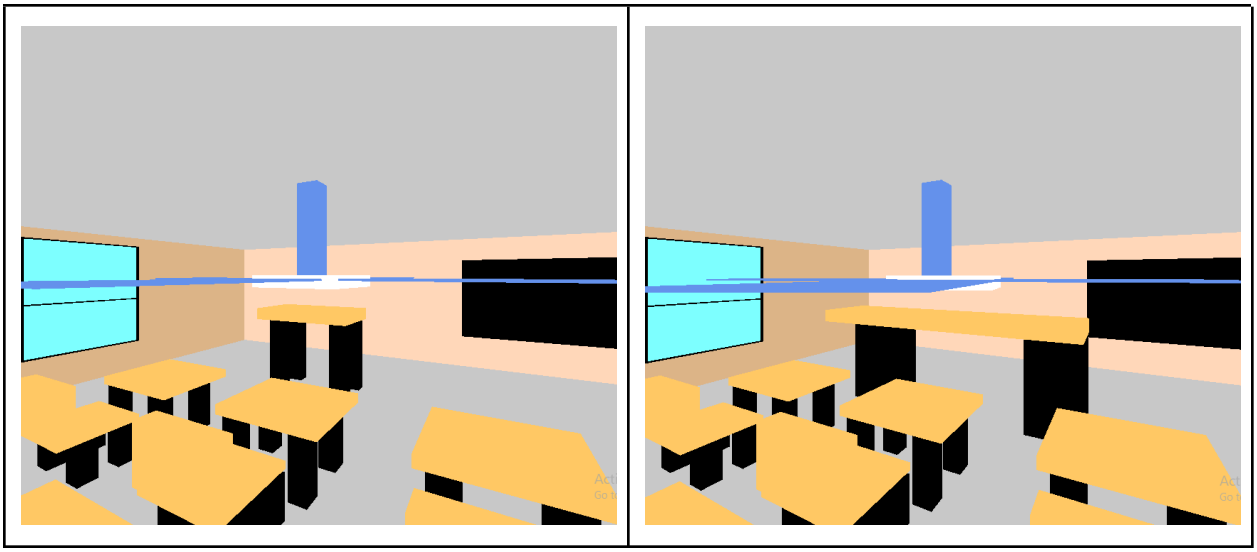
```
void animate()
{
   if (bRotate == true)
   {
      theta += 0.1;
      if(theta > 360.0)
         theta -= 360.0*floor(theta/360.0);
   }


   if (fRotate == true)
   {
      alpha += 0.8;
      if(alpha > 360.0)
         alpha -= 360.0*floor(alpha/360.0);
   }

   glutPostRedisplay();

}
```

RESULTS

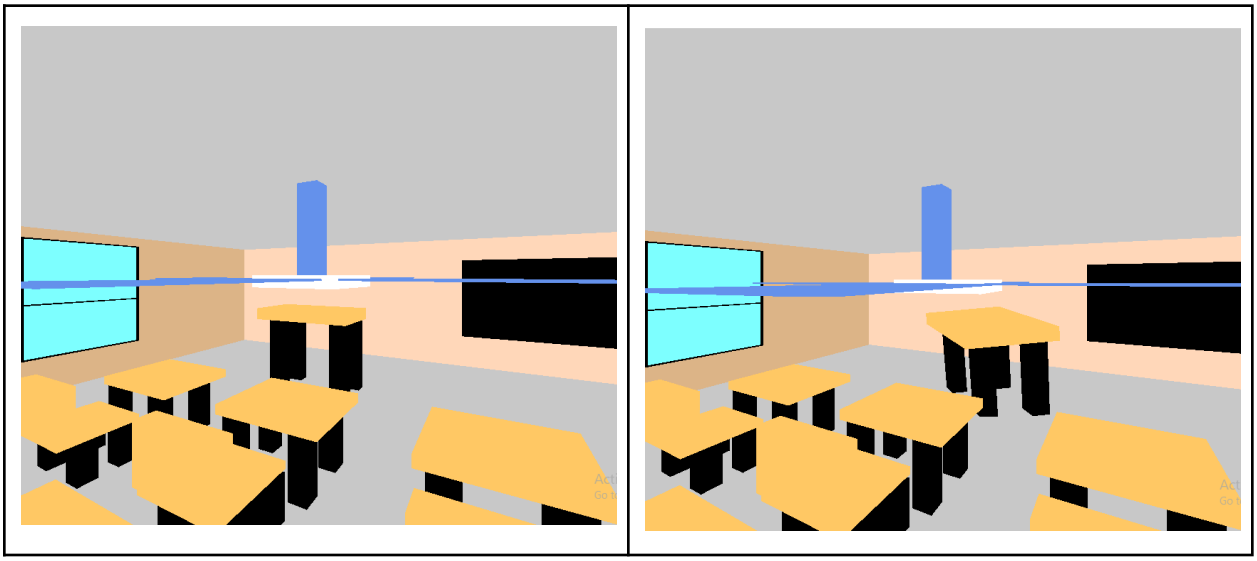
Translation on Teacher Desk



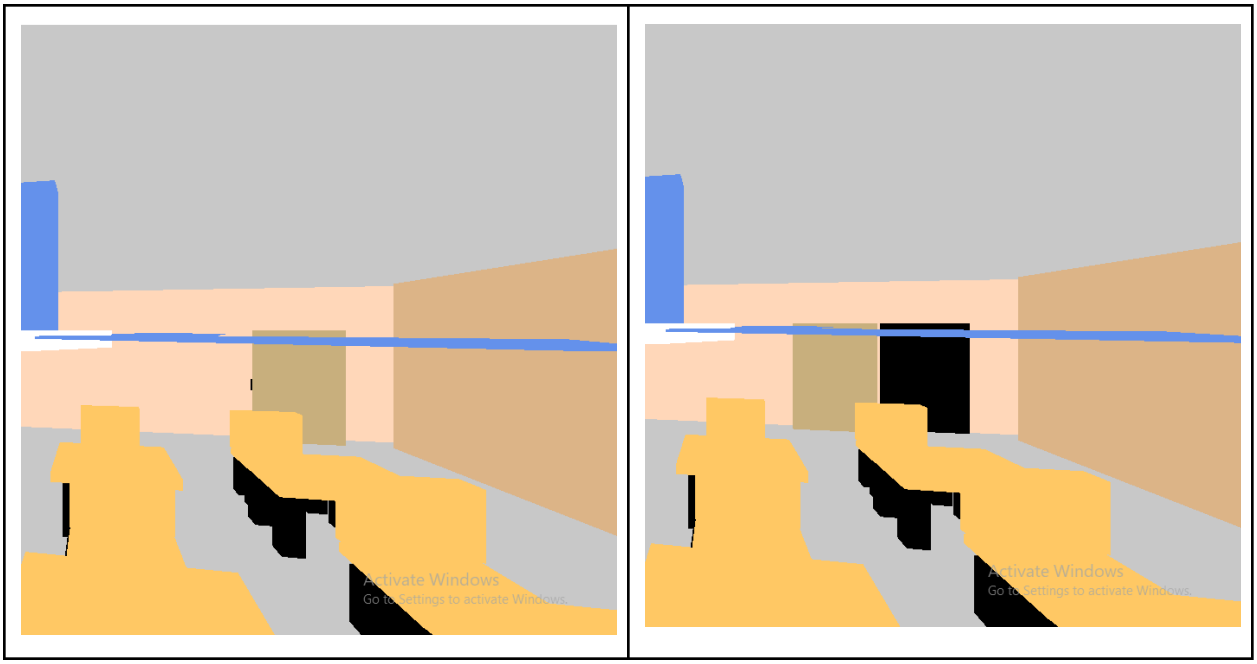
Scaling on Teacher Desk



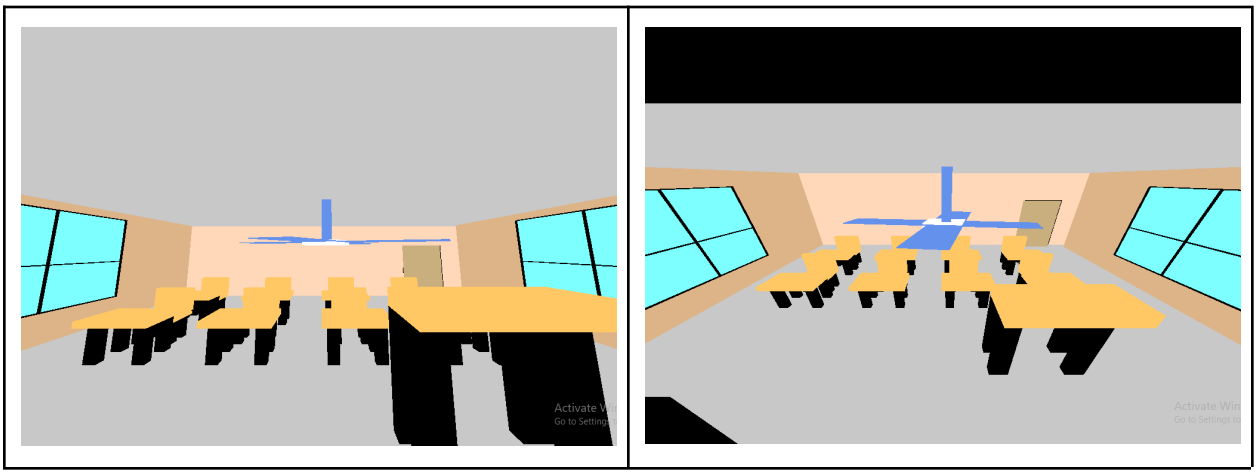
Rotation on Teacher Desk



Door closed vs opened



Front view vs Top view



FUTURE ENHANCEMENTS

Possible future enhancements for the project include:

1. Texture Mapping: Apply textures to objects for a more realistic appearance.

2. User Interface (UI): Implement a graphical user interface for easier interaction.

3. Additional Objects: Add more objects to the room for increased complexity and realism.

4. Lighting Effects: Incorporate advanced lighting techniques for improved visual quality.

CONCLUSION

The 3D Classroom project successfully demonstrates the application of computer graphics techniques to create a virtual classroom environment. By implementing interactive controls with keyboard, animation, and simulations, the project provides an engaging and immersive experience for users. This project enhances our understanding of computer graphics concepts and serves as a practical application of theoretical knowledge.