3D MODELLING OF TAJ MAHAL

Under the Guidance of
Dr. Somnath Dey

Apoorva Joshi
150001002
Keshav Goyal
150001014

ABSTRACT

ABSTRACT

The Taj Mahal also known as "the jewel of Muslim art in India " is a perfect model for rendering the various computer graphics techniques being taught us in the class. As the course project of Computer Graphics (CS 352),we attempt at creating and transforming a 3D prototype of Taj Mahal.

FEATURES

FEATURES

1. *a closed 3d model of Taj Mahal created using gl Functions*
2. *Lighting added to the model (specular and diffused light)*
3. *Rotation of the model using keyboard functions*
4. *Shadow of the model shown (with the source light rotating with time)*
5. *Zooming using mouse functions*
6. *Reflection of the model in lake shown*

Files Included

1. main_shadow.cpp- This is the main file you need to execute. It includes main function and it calls to all other functions.

2. draw_face.hpp- This file include functions which draw all the faces of TAJ MAHAL, when called.

3. shadow.hpp- This file includes all the methods which are used to render shadow effects with the help of stencil buffer.

4. main_draw.hpp- This file includes the main drawing function which calls to function in other files to draw the complete TAJ MAHAL.

5. draw_compo.hpp- This file includes functions which upon calling draws all small components of TAJ MAHAL. (e.g. doors, windows, walls etc.)

6. Lighting.hpp- This file is responsible for the lighting effects included in our project.

7. Keyboard_mouse_handle.hpp- The functions included in this file controls the moving speed of light source, zoom in, zoom out operations.

FILES IN DETAIL

draw_compo.hpp

This file includes the various components used for drawing the final main faces, doors and windows.

1.void drawend():- draws the base of Taj Mahal

2. void drawBase():-creates a cuboid by calling drawbase() and drawend()

3.void displayCIRCLE(float centerx,float centery,float R ,int zz,int xx,float cons) :- creates a 2D disc(semicircle) of radius R with center (centerx,centery,cons) if flag zz is set to 1 else the center is set to be (cons,centery,centerx). The points are determined by incrementing the angle from 0 to 90 degrees. (Note the use of MPT or Bresenham algorithm for determination of points could not be used because of the very small value of radius(less than 1.0f))

The points are drawn by calling drawpoints(float centerx,float centery,float x,float y,int zz,int xx,float cons) function.

draw_compo.hpp

4. void drawpoints(float centerx,float centery,float x,float y,int zz,int xx,float cons) :-draws a line prependicular to y axis in xy plane if flag zz is set to 1 or in xz plane if flag xx is set to 1

5.void draw_door_y(int yy) :- creates a black coloured door in yz plane by first drawing a rectangle and then a semicircle by calling drawCIRCLE function . The flag yy helps in drawing the door along the positive or negative x direction.

6.void draw_door_z(int zz) :-creates a black coloured door in xy plane by first drawing a rectangle and then asemicircle by calling drawCIRCLE function . The flag zz helps in drawing the door along the positive or negative z direction.

7.void draw_window_z(int zz):-creates a black coloured window in xy plane by first drawing a rectangle and then a semicircle by calling drawCIRCLE function . The flag zz helps in drawing the door along the positive or negative z direction.

draw_compo.hpp

8.void draw_window_y(int yy):-creates a black coloured window in yz plane by first drawing a rectangle and then a semicircle by calling drawCIRCLE function . The flag yy helps in drawing the door along the positive or negative x direction.

9.void draw_window_slnt(int xx ,int zz,int xyx):-creates a black coloured window along the face joining the faces in yz and xy planes by first drawing a rectangle and then a hemisphere. The flags xx,zz,xyx helps in actually defining the correct coordinates of the point that lie along a line.

10.void draw_cylinder(float r1 , float r2, float h):- creates a cylinder along y axis (first rotated the matrix) with r1 as the base radius , r2 as the top radius and h as height.

11.void draw_hemisphere(float r1,float r2) :- creates a hemisphere along y axis with radius as r1 by using glutSolidSphere function. The output is further clipped by using r2 as cutting plane (y>r2 only shown)

draw_face.hpp

1.void draw_face(int zz) :- draw main face along xy plane . The flag zz is used to help in deciding the direction of z axis.

2.void draw_facey(int yy):-draw main face along yz plane . The flag zz is used to help in deciding the direction of x axis.

3.void draw_smallface(int zz,int xy):-draw face with less height along xy plane . The flag zz is used to help in deciding the direction of z axis. The flag xy is used for translation purposes.

4.void draw_smallfacey(int yy, int xy):-draw face with less height along yz plane . The flag yy is used to help in deciding the direction of x axis. The flag xy is used for translation purposes.

5.void draw_slnt(int xx,int yy):- used to draw face joining the faces in xy and yz planes.

main_draw.hpp

This file contains the main draw function that calls all the other functions from main_compo.hpp and main_face.hpp in an order to give the desired results.

1.void draw():-first draw the various faces , then doors and windows , chhatris , the onion dome and minarets.

Keyboard_mouse_handle.hpp

GLOBAL VARIABLES:-

1.float lightAngle ,lightHeight :- for defining the properties of light source for creating shadow

2.float angle, angle2:- for rotating the model along y and x axis respectively.

3.int moving,startx, starty :-to keep track of location with the mouse

4.int lightStartX, lightStartY:- helps in calculating lightAngle on changing mouse coordinates

FUNCTIONS:-

1.static void mouse(int button, int state, int x, int y) :- defines the effect of moving the mouse pointer on the light source for shadow and lighting

2.static void motion(int x, int y) :-changing parameters wrt to mouse cursor

3.static void idle():- defines what happens when the cursor is not moved.

Keyboard_mouse_handle.hpp

4.static void normalKeyboard(unsigned char c, int x, int y) :- the keyboard handler function defines the effect of pressing certain keys

Lighting.hpp

GLOBAL VARIABLES:-

1.static GLfloat lightPosition[4]- This variable stores the position of all light sources.(initialised in main_shadow.hpp)
2.static GLfloat lightColor[4]- Stores the values of red, green, blue, alpha.(Color of each light source)

FUNCTIONS :-
1.void Lighting()- This function enables all the lighting sources, define initialise their color, position, type using glLightfv() function.
2.void Ediffuselight()- This function enables the diffuse lighting source, when called.
3.void Especularlight()- This function enables the specular light source, when called.

shadow.hpp

GLOBAL VARIABLES:-

1.static int offsetShadow- This variable is set to 1 if we to enable want to enable offset shadow 0 otherwise.
2.static GLfloat floorshadow[4][4]- This variable stores the shadow matrix, which is used to project shadow of an object.

FUNCTIONS:-

1.void CshaMatrix():- This function takes the position of light and floor coordinates as input and gives shadow Matrix as output.
2.void update_stencilB():- This function takes a flag as input if set to one fill the stencil buffer with specific value, at specific positions.
3.void shadow():- Renders shadow effect and draws the shadow of object at specific position where stencil buffer is set.

main_shadow.cpp

GLOBAL VARIABLES:-

1.static int renderShadow:- Renders shadow if variable is set to 1, otherwise 0.

2.static int renderScene:- Renders 3D TAJ MAHAL if variable is set 1, otherwise 0.

3.static int renderReflection:- Renders reflection of TAJ MAHAL in lake if variable is set to 1, otherwise 0.

4.static GLfloat floorVertices[4][3]:- Stores the coordinates of main plane on which whole TAJ MAHAL is being constructed.

5.static GLfloat floorVertices[4][3]:- Stores the coordinates of lake on which the reflection is being constructed.

FUNCTIONS:-

1.static void drawfloor1(void):- This function renders the lake in the project.

2.static void drawfloor(void):- This function renders the main plane on which whole TAJ MAHAL is being constructed.

3.static void redraw():- This function is the main function in our project, it integrates most of the components in our project to render the TAJ MAHAL.

It sets the position of light source dynamically. It calls CshaMatrix() function to generate shadow matrix. It renders the reflection in lake using stencil buffer and render the lake with lighting effects. It renders the floor. Then it renders the TAJ MAHAL with lighting effects. Afterwards it renders shadow effect by calling shadow() function. At last we use swapbuffers() as we are using dual buffers.

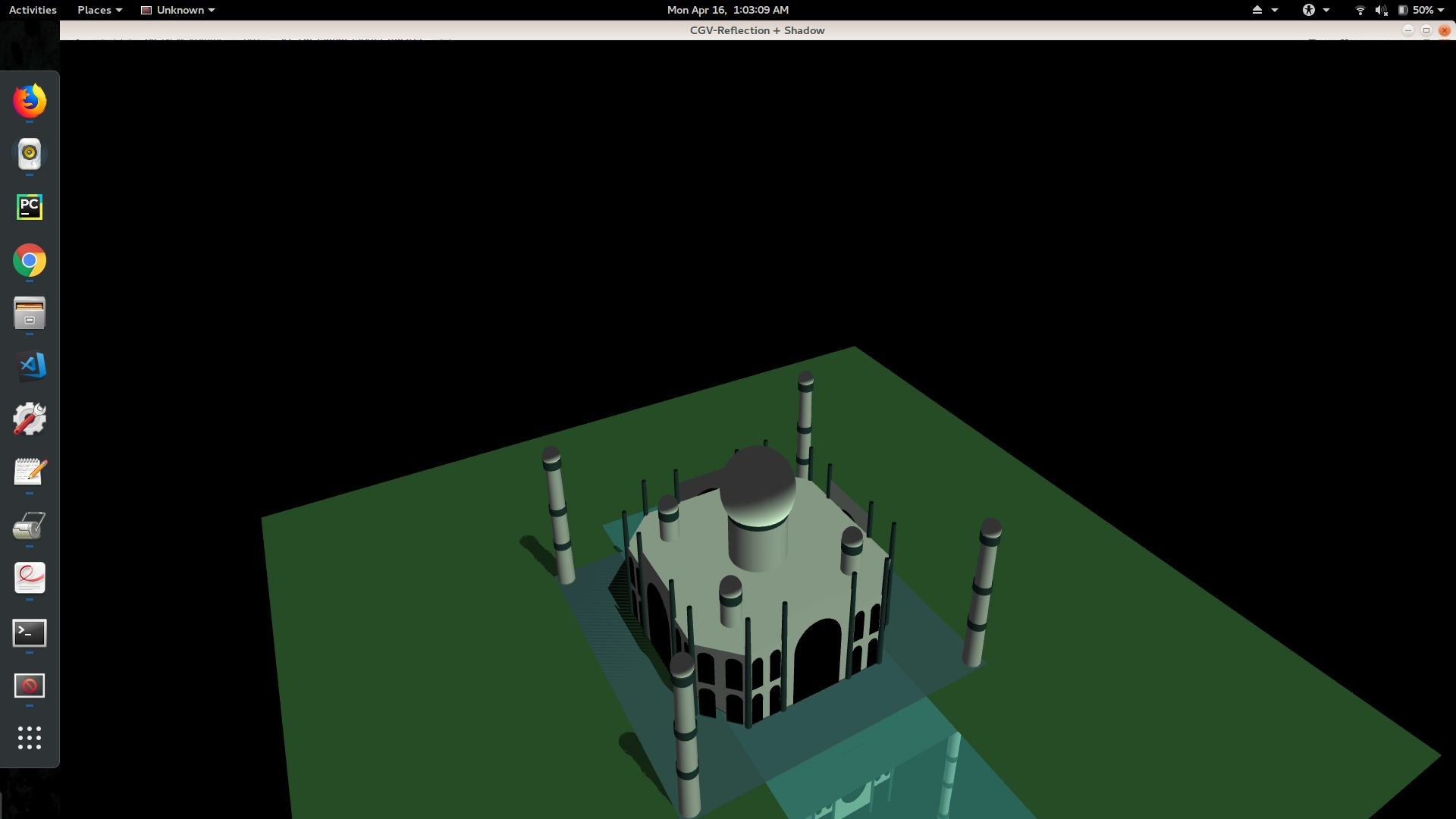
4.int main():- It sets the window creation display mode. It sets the size of window in which our whole project will render. It creates a top-level window with the name "CGV-Reflection + Shadow". Then gl-functions to call redraw(), mouse(),motion(),idle() and normalKeyboard are used. Finally glutMainLoop() is called.

5.void initRendering():-sets the defaults for various lighting and matrix options.

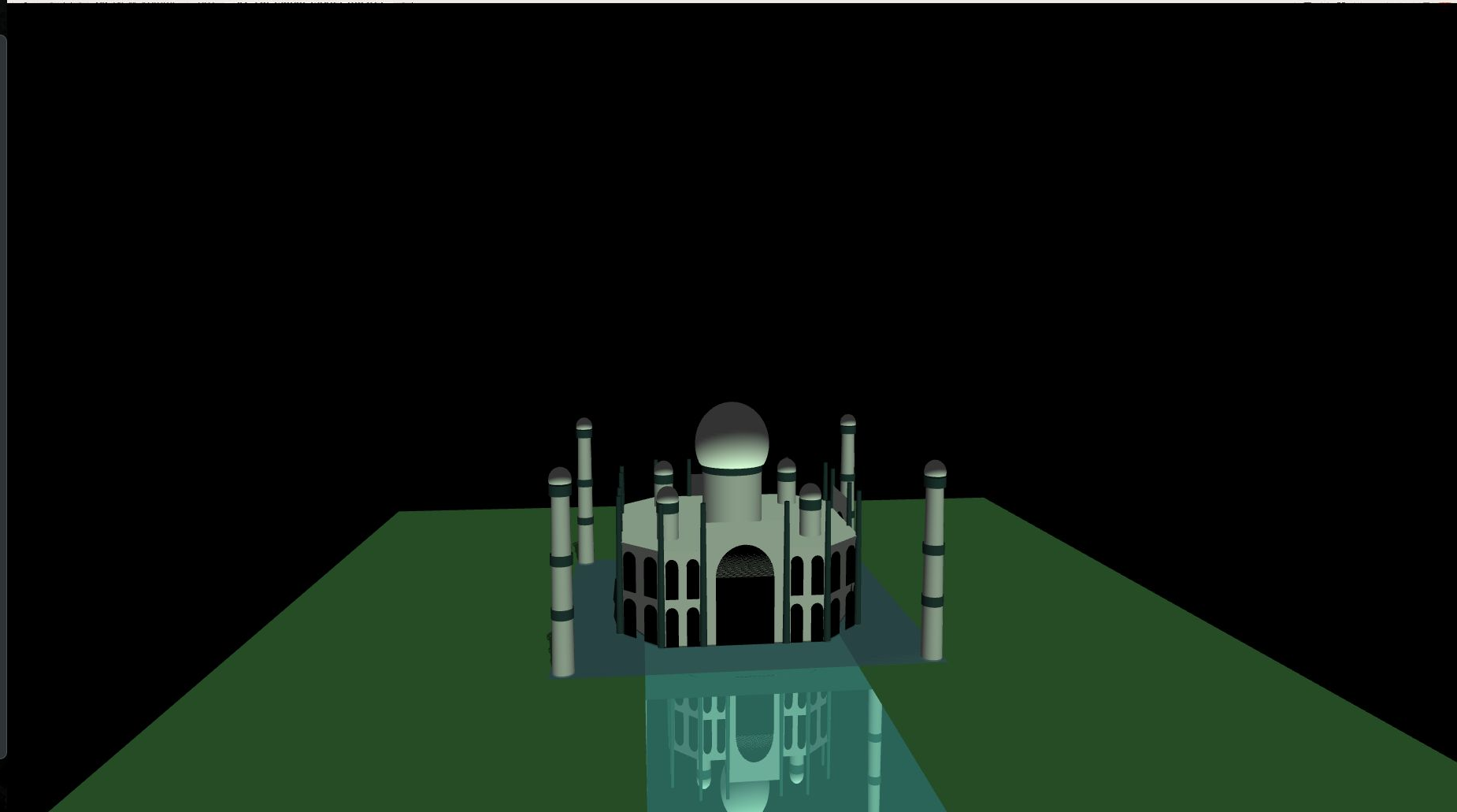
Steps to execute

EXECUTION

```
apoorva@lun4t1c:~/Downloads/150001014_150001002/Taj3$ g++ -o taj  main_shadow.cp
p -lglut -lGL -lGLU
```


Activities
Places
Unknown
Mon Apr 16, 1:03:09 AM
50%
CGV-Reflection + Shadow

CGV-Reflection + Shadow




Thank You