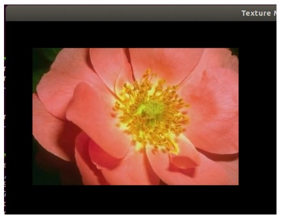
CS460

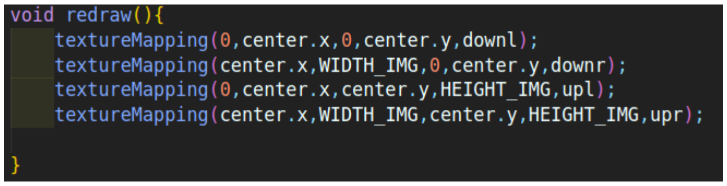
Homework 4 Report

Sinem Ozden

1. Problem Statement:
    1.1. Creating a deformed rectangle with flower.bmp image's texture.
    1.2. Rotating the rectangle that contains flower image's texture.
    1.3. Loading and rotating a 3D image.
2. Algorithm Design:
    2.1.To display the image right click to screen and select image from the menu.



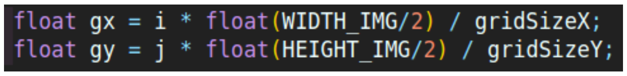
I used the rectangle deformation model in hint 2.3. I filled each rectangle with the texture array and create small textures. I tired to achieve reverse mapping. So first changed the center of the image coordinated with the center of the object as I get the input from the mouse. I set the begin and end points of rectangles in function redraw and call each rectangles filling function one after another. Then I send the beginning and ending points of the new rectangle to texture mapping function.



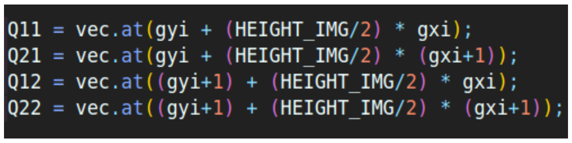
```
void redraw(){
	textureMapping(0,center.x,0,center.y,downl);
	textureMapping(center.x,WIDTH_IMG,0,center.y,downr);
	textureMapping(0,center.x,center.y,HEIGHT_IMG,upl);
	textureMapping(center.x,WIDTH_IMG,center.y,HEIGHT_IMG,upr);

}
```

In texture mapping function I created a grid as big as the new rectangle and scale it to the small texture rectangle. Because the texture array I created is linear and not 2D I need to convert the 2D dimensions to 2D.

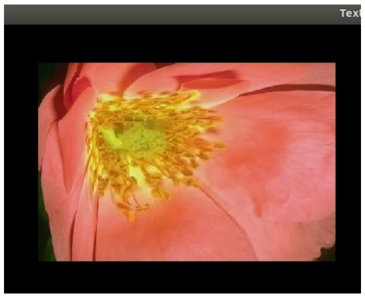


```
float gx = i * float(WIDTH_IMG/2) / gridSizeX;
float gy = j * float(HEIGHT_IMG/2) / gridSizeY;
```



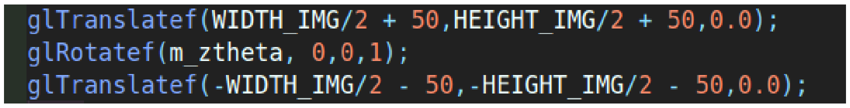
```
Q11 = vec.at(gyi + (HEIGHT_IMG/2) * gxi);
Q21 = vec.at(gyi + (HEIGHT_IMG/2) * (gxi+1));
Q12 = vec.at((gyi+1) + (HEIGHT_IMG/2) * gxi);
Q22 = vec.at((gyi+1) + (HEIGHT_IMG/2) * (gxi+1));
```

After getting the color relative to texture rectangle I sent the coordinates of four points around the point that I want to interpolate and the color of these points to computebilinearinterpolation function. This function does the calculations to find the new color for the point. After getting the color I draw it.

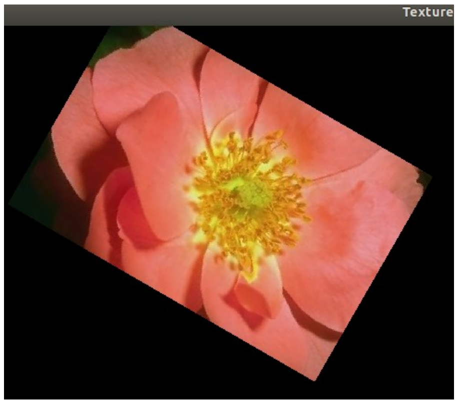


2.2. For rotating the rectangle you have to right click the screen to open the menu and select rotate when image is visible on the screen. You can control the rotation from up and down buttons too.

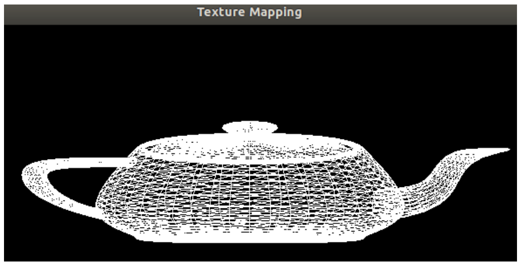
I first translate to the center of the rectangle. Then I call glRotate with the previously calculated angle. (I calculate the angle as the rotate from menu is clicked or the up down buttons pressed. You can find more information how I did it at assignment 3.)Then I translate back where I started. I do all these because if I don't translate to middle image will rotate from the 0,0 point of the window, I translate back to beginning because if I don't it will draw the rectangle to the where center should be and again rotate the square from old center spot.

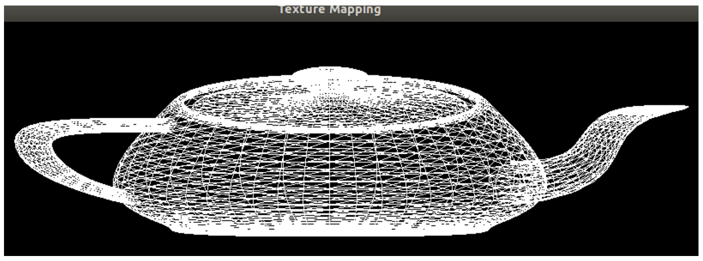


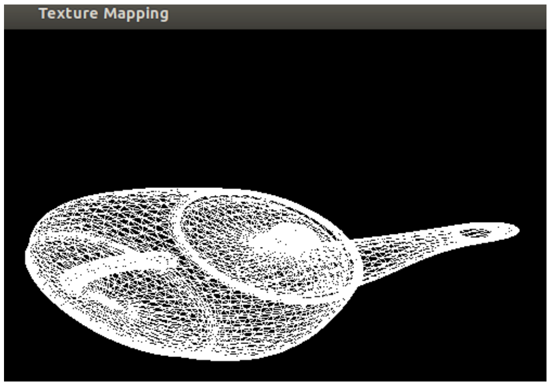
```
glTranslatef(WIDTH_IMG/2 + 50,HEIGHT_IMG/2 + 50,0.0);
glRotatef(m_ztheta, 0,0,1);
glTranslatef(-WIDTH_IMG/2 - 50,-HEIGHT_IMG/2 - 50,0.0);
```



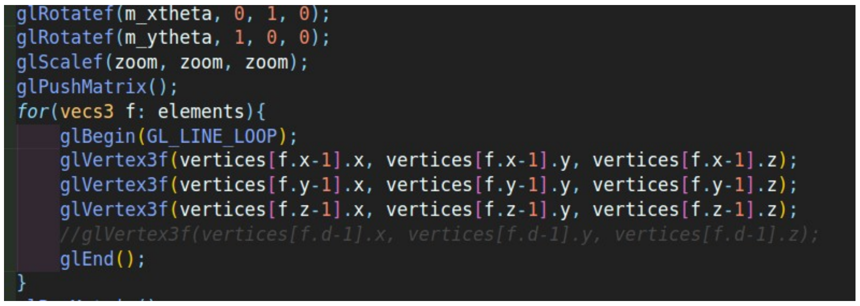
2.3. To display the object right click to screen and select the object from menu. You can display and zoom in to object with up and down buttons. If you also select the rotate from menu you can rotate the teapot object on x and y orientations.







To zoom in and rotate the object I did the same way as the rectangle object. To display I traverse at elements vector and get the verticies of the object then draw them with glLineLoop.



```
glRotatef(m_xtheta, 0, 1, 0);
glRotatef(m_ytheta, 1, 0, 0);
glScalef(zoom, zoom, zoom);
glPushMatrix();
for(vecs3 f: elements){
    glBegin(GL_LINE_LOOP);
    glVertex3f(vertices[f.x-1].x, vertices[f.x-1].y, vertices[f.x-1].z);
    glVertex3f(vertices[f.y-1].x, vertices[f.y-1].y, vertices[f.y-1].z);
    glVertex3f(vertices[f.z-1].x, vertices[f.z-1].y, vertices[f.z-1].z);
    //glVertex3f(vertices[f.d-1].x, vertices[f.d-1].y, vertices[f.d-1].z);
    glEnd();
}
```

3. To compile the first 3 parts of the assignment run these lines:

```
g++ -Wall -g -I/usr/include -c main.cpp -o main.o
g++ main.cpp -lglut -lGL -lGLU -lXxf86vm
./a.out
```