Chavangkon Kanjanakuldit (Ken) 6681371

<https://github.com/Ken19149/Graphics/tree/main/week_7>

#define STB\_IMAGE\_IMPLEMENTATION

#include "stb\_image.h"

#include <GL/glut.h>

#include <stdio.h>

#include <stdlib.h>

GLuint textureID;

GLuint loadTexture(const char\* filename) {

int width, height, channels;

unsigned char\* image = stbi\_load(filename, &width, &height, &channels, STBI\_rgb);

if (!image) {

printf("Error: Could not load texture file: %s\n", filename);

printf("Make sure the file exists in the current directory\n");

return 0;

}

printf("Successfully loaded texture: %s (%dx%d, %d channels)\n", filename, width, height, channels);

GLuint texture;

glGenTextures(1, &texture);

glBindTexture(GL\_TEXTURE\_2D, texture);

glTexImage2D(GL\_TEXTURE\_2D, 0, GL\_RGB, width, height, 0, GL\_RGB, GL\_UNSIGNED\_BYTE, image);

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

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

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

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

stbi\_image\_free(image);

return texture;

}

// Function to draw a textured cube manually

void drawTexturedCube(GLuint textureID) {

glBindTexture(GL\_TEXTURE\_2D, textureID);

glBegin(GL\_QUADS);

// Front face

glTexCoord2f(0.0f, 0.0f); glVertex3f(-0.5f, -0.5f, 0.5f);

glTexCoord2f(3.0f, 0.0f); glVertex3f(0.5f, -0.5f, 0.5f);

glTexCoord2f(3.0f, 3.0f); glVertex3f(0.5f, 0.5f, 0.5f);

glTexCoord2f(0.0f, 3.0f); glVertex3f(-0.5f, 0.5f, 0.5f);

// Back face

glTexCoord2f(1.0f, 0.0f); glVertex3f(-0.5f, -0.5f, -0.5f);

glTexCoord2f(1.0f, 1.0f); glVertex3f(-0.5f, 0.5f, -0.5f);

glTexCoord2f(0.0f, 1.0f); glVertex3f(0.5f, 0.5f, -0.5f);

glTexCoord2f(0.0f, 0.0f); glVertex3f(0.5f, -0.5f, -0.5f);

// Top face

glTexCoord2f(0.0f, 1.0f); glVertex3f(-0.5f, 0.5f, -0.5f);

glTexCoord2f(0.0f, 0.0f); glVertex3f(-0.5f, 0.5f, 0.5f);

glTexCoord2f(1.0f, 0.0f); glVertex3f(0.5f, 0.5f, 0.5f);

glTexCoord2f(1.0f, 1.0f); glVertex3f(0.5f, 0.5f, -0.5f);

// Bottom face

glTexCoord2f(1.0f, 1.0f); glVertex3f(-0.5f, -0.5f, -0.5f);

glTexCoord2f(0.0f, 1.0f); glVertex3f(0.5f, -0.5f, -0.5f);

glTexCoord2f(0.0f, 0.0f); glVertex3f(0.5f, -0.5f, 0.5f);

glTexCoord2f(1.0f, 0.0f); glVertex3f(-0.5f, -0.5f, 0.5f);

// Right face

glTexCoord2f(1.0f, 0.0f); glVertex3f(0.5f, -0.5f, -0.5f);

glTexCoord2f(1.0f, 1.0f); glVertex3f(0.5f, 0.5f, -0.5f);

glTexCoord2f(0.0f, 1.0f); glVertex3f(0.5f, 0.5f, 0.5f);

glTexCoord2f(0.0f, 0.0f); glVertex3f(0.5f, -0.5f, 0.5f);

// Left face

glTexCoord2f(0.0f, 0.0f); glVertex3f(-0.5f, -0.5f, -0.5f);

glTexCoord2f(1.0f, 0.0f); glVertex3f(-0.5f, -0.5f, 0.5f);

glTexCoord2f(1.0f, 1.0f); glVertex3f(-0.5f, 0.5f, 0.5f);

glTexCoord2f(0.0f, 1.0f); glVertex3f(-0.5f, 0.5f, -0.5f);

glEnd();

}

void display() {

glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT);

glLoadIdentity();

// Set camera position

gluLookAt(3.0, 3.0, 3.0, // Eye position

0.0, 0.0, 0.0, // Look at point

0.0, 1.0, 0.0); // Up vector

// Draw textured cube in the center

glEnable(GL\_TEXTURE\_2D);

drawTexturedCube(loadTexture("C:\\Users\\Student\\Ken 6681371\\Graphics\\week\_7\\textures\\hello\_adjust.jpg")); // center

// drawTexturedCube();

glDisable(GL\_TEXTURE\_2D);

// Draw additional colored cubes around the center

glEnable(GL\_TEXTURE\_2D);

glPushMatrix();

glTranslatef(2.0f, 0.0f, 0.0f);

// glColor3f(0.0f, 1.0f, 0.0f); // Green color

// glutSolidCube(0.5);

drawTexturedCube(loadTexture("C:\\Users\\Student\\Ken 6681371\\Graphics\\week\_7\\textures\\ken\_adjust.jpg")); // near

// drawTexturedCube();

glPopMatrix();

glDisable(GL\_TEXTURE\_2D);

glEnable(GL\_TEXTURE\_2D);

glPushMatrix();

glTranslatef(-2.0f, 0.0f, 0.0f);

drawTexturedCube(loadTexture("C:\\Users\\Student\\Ken 6681371\\Graphics\\week\_7\\textures\\pattern.jpg")); // back

glPopMatrix();

glDisable(GL\_TEXTURE\_2D);

glEnable(GL\_TEXTURE\_2D);

glPushMatrix();

glTranslatef(0.0f, 2.0f, 0.0f);

drawTexturedCube(loadTexture("C:\\Users\\Student\\Ken 6681371\\Graphics\\week\_7\\textures\\top.jpg")); // top

glPopMatrix();

glDisable(GL\_TEXTURE\_2D);

glEnable(GL\_TEXTURE\_2D);

glPushMatrix();

glTranslatef(0.0f, -2.0f, 0.0f);

glScalef(0.5f, 0.5f, 0.5f);

drawTexturedCube(loadTexture("C:\\Users\\Student\\Ken 6681371\\Graphics\\week\_7\\textures\\creeper.jpg")); // bottom

glPopMatrix();

glDisable(GL\_TEXTURE\_2D);

glutSwapBuffers();

}

void reshape(int w, int h) {

glViewport(0, 0, w, h);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

gluPerspective(45.0, (double)w / (double)h, 1.0, 20.0);

glMatrixMode(GL\_MODELVIEW);

}

void keyboard(unsigned char key, int x, int y) {

switch (key) {

case 27: // ESC key

exit(0);

break;

}

}

void init() {

glClearColor(0.0f, 0.0f, 0.0f, 1.0f);

glEnable(GL\_DEPTH\_TEST);

// Enable lighting

glEnable(GL\_LIGHTING);

glEnable(GL\_LIGHT0);

// Set light properties

GLfloat lightPos[] = {5.0f, 5.0f, 5.0f, 1.0f};

GLfloat lightAmbient[] = {0.3f, 0.3f, 0.3f, 1.0f};

GLfloat lightDiffuse[] = {0.8f, 0.8f, 0.8f, 1.0f};

//

glLightfv(GL\_LIGHT0, GL\_POSITION, lightPos);

glLightfv(GL\_LIGHT0, GL\_AMBIENT, lightAmbient);

glLightfv(GL\_LIGHT0, GL\_DIFFUSE, lightDiffuse);

// Enable color material

glEnable(GL\_COLOR\_MATERIAL);

glColorMaterial(GL\_FRONT, GL\_AMBIENT\_AND\_DIFFUSE);

// Load texture from file

printf("Loading texture.jpg...\n");

textureID = loadTexture("C:\\Users\\Student\\Ken 6681371\\Graphics\\week\_7\\textures\\hello\_adjust.jpg");

// textureID = loadTexture(".\\week\_7\\textures\\texture.jpg");

if (textureID == 0) {

printf("Creating fallback texture instead...\n");

// Create a simple fallback texture

const int texSize = 64;

GLubyte texture[texSize][texSize][3];

for (int i = 0; i < texSize; i++) {

for (int j = 0; j < texSize; j++) {

texture[i][j][0] = (i \* 4) % 256; // Red gradient

texture[i][j][1] = (j \* 4) % 256; // Green gradient

texture[i][j][2] = 128; // Blue constant

}

}

glGenTextures(1, &textureID);

glBindTexture(GL\_TEXTURE\_2D, textureID);

glTexImage2D(GL\_TEXTURE\_2D, 0, GL\_RGB, texSize, texSize, 0, GL\_RGB, GL\_UNSIGNED\_BYTE, texture);

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

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

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

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

}

}

int main(int argc, char\*\* argv) {

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_DOUBLE | GLUT\_RGB | GLUT\_DEPTH);

glutInitWindowSize(600, 600);

glutCreateWindow("GLUT Textured Cube with texture.jpg");

init();

glutDisplayFunc(display);

glutReshapeFunc(reshape);

glutKeyboardFunc(keyboard);

printf("\n=== Texture Loading Info ===\n");

printf("Looking for: texture.jpg\n");

printf("Current directory contents should include:\n");

system("ls -la \*.jpg \*.jpeg \*.png 2>/dev/null || echo 'No image files found in current directory'");

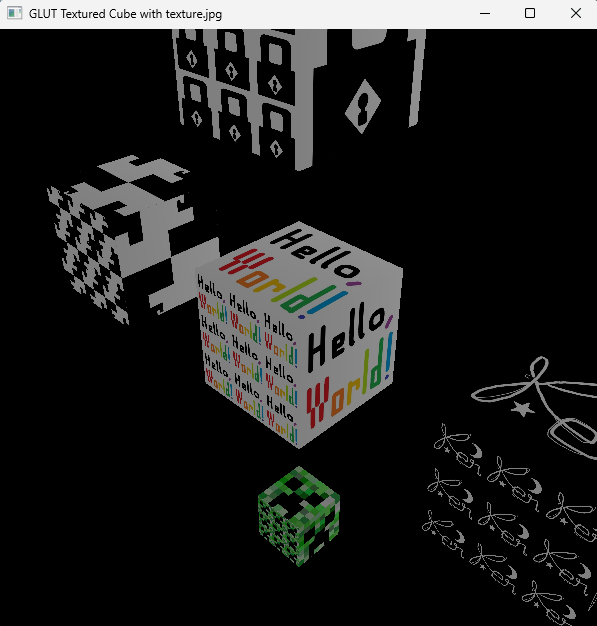
printf("============================\n\n");

glutMainLoop();

return 0;

}

Output:



I tried adding Gluint textureID in the drawTexturedCube function so I can add multiple image textures to each cube. Most textures I created in MS Paint except for the creeper head from Minecraft which I scaled down by half. I also tried experimenting with coordinate mapping and got this result. I still don’t quite understand how it works yet but I will play with it more to understand it better.