HW2
109550134 梁詠晴

How I do my homework:

功能簡介: 按1可以讓貓變形，按2可以改變貓的顏色，按三可以增加光照+調整亮度+變形，按0可以回到初始狀態。

Status控制:

```
id keyCallback(GLFWwindow* window, int key, int scancode, in

  if (key == GLFW_KEY_ESCAPE && action == GLFW_PRESS)
      glfwSetWindowShouldClose(window, true);
  if (glfwGetKey(window, GLFW_KEY_1) == GLFW_PRESS) {
      status = 1;
      cout << "model deformation\n";
  }
  if (glfwGetKey(window, GLFW_KEY_2) == GLFW_PRESS) {
      status = 2;
      cout << "change color\n";
  }
  if (glfwGetKey(window, GLFW_KEY_3) == GLFW_PRESS) {
      status = 3;
      cout << "model deformation+change color\n";
  }
  if (glfwGetKey(window, GLFW_KEY_0) == GLFW_PRESS) {
      status = 0;
      cout << "init model\n";
  }
```

```
if (status == 1) {//deformation
    c_fsKey = 1;
    b_fsKey = 1;
    c_vsKey = 2;
    b_vsKey = 1;
}
else if (status == 2) {//change color
    c_fsKey = 2;
    b_fsKey = 1;
    c_vsKey = 1;
    b_vsKey = 1;
}
else if (status == 3) {//color+deformation
    c_fsKey = 3;
    b_fsKey = 3;
    c_vsKey = 3;
    b_vsKey = 3;
}
else if (status == 0) {

    c_fsKey = 1;
    b_fsKey = 1;
    c_vsKey = 1;
    b_vsKey = 1;
}
```

偵測按鍵改變status(初始=0)，並依照目前status改變cat和box各自傳入vertex shader和 fragment shader的參數。

Vertex shader:

```
layout (location = 0) in vec3 aPos;
layout (location = 1) in vec3 aNormal;
layout (location = 2) in vec2 aTexCoord;

uniform mat4 M;
uniform mat4 V;
uniform mat4 P;
uniform int key;

out vec2 texCoord;
out vec4 worldPos;
out vec3 normal;

void main()
{
    if(key==2){
        gl_Position = P * V * M * vec4(aPos.x*0.5, aPos.y*2, aPos.z, 1.0);
    }
    else if(key==3){
        gl_Position = P * V * M * vec4(aPos.x*1.5, aPos.y*0.5+1, aPos.z*0.5, 1.0);
    }
    else{
        gl_Position = P * V * M * vec4(aPos, 1.0);
    }

    texCoord = aTexCoord;
    worldPos = M * vec4(aPos, 1.0);
    mat4 normal_transform = transpose(inverse(M));
    normal = normalize((normal_transform * vec4(aNormal, 0.0)).xyz);;
```

用收到的key判斷status，改變aPos求出不同的gl_position

Fragment shader:

```
#version 330 core

in vec2 texCoord;
in vec4 worldPos;
in vec3 normal;

uniform sampler2D ourTexture;
uniform int fsKey;


out vec4 FragColor;

void main()
{
    vec3 lightPos = vec3(5, 10, 10);
    vec3 n_normal = normalize(normal);
    vec3 light = (normalize(vec4(lightPos, 1.0) - worldPos)).xyz;

    float diffuse = max(dot(light, n_normal), 0.0);
    if(fsKey==2){
        FragColor = 1.0-texture(ourTexture, texCoord);
    }
    else if(fsKey==3){
        FragColor = 3*diffuse*texture(ourTexture, texCoord);
    }
    else{
        FragColor = texture(ourTexture, texCoord);
    }

}
```

用收到的key判斷status，求出不同的FragColor(eg. 負片效果)、加入光照

另外，使用ModelVAO、DrawModel兩個function處理VAO、畫出model
ModelVAO: 3個VBO分別存model的positions、normals、texcoords資訊
DrawModel: 判斷目前是cat或box，bind相對應的VAO並畫出來

Problems:
因為傳入了各種key到shader裡，處理Loc傳送時key參數混淆而一直無法跑出預期的結果。在調整傳送之後就出現正確結果了