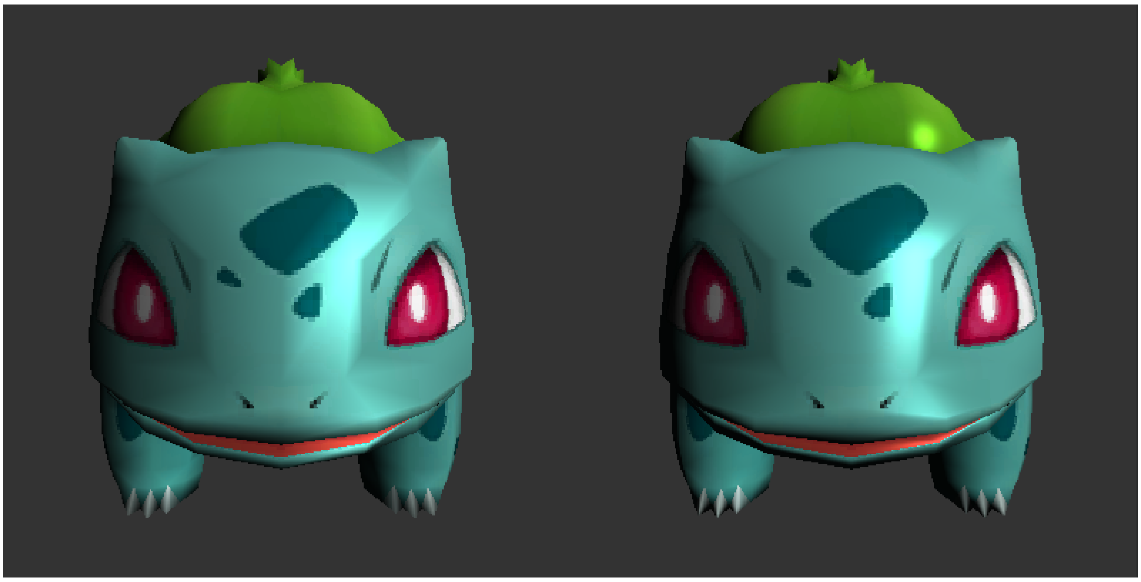
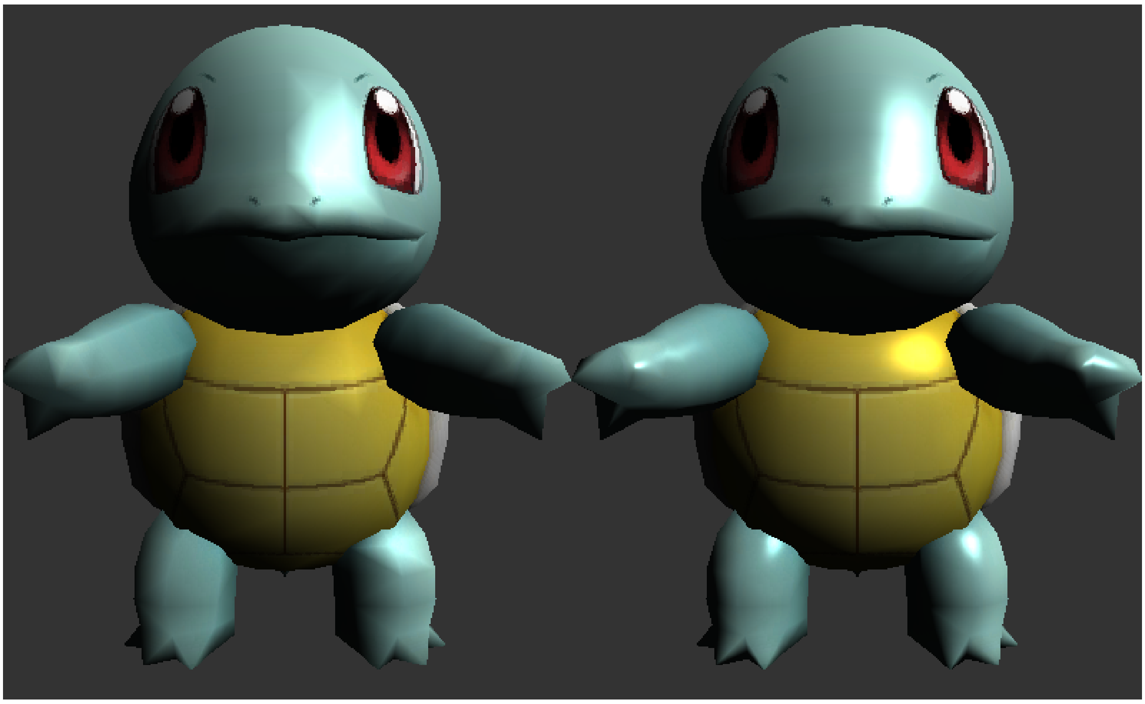
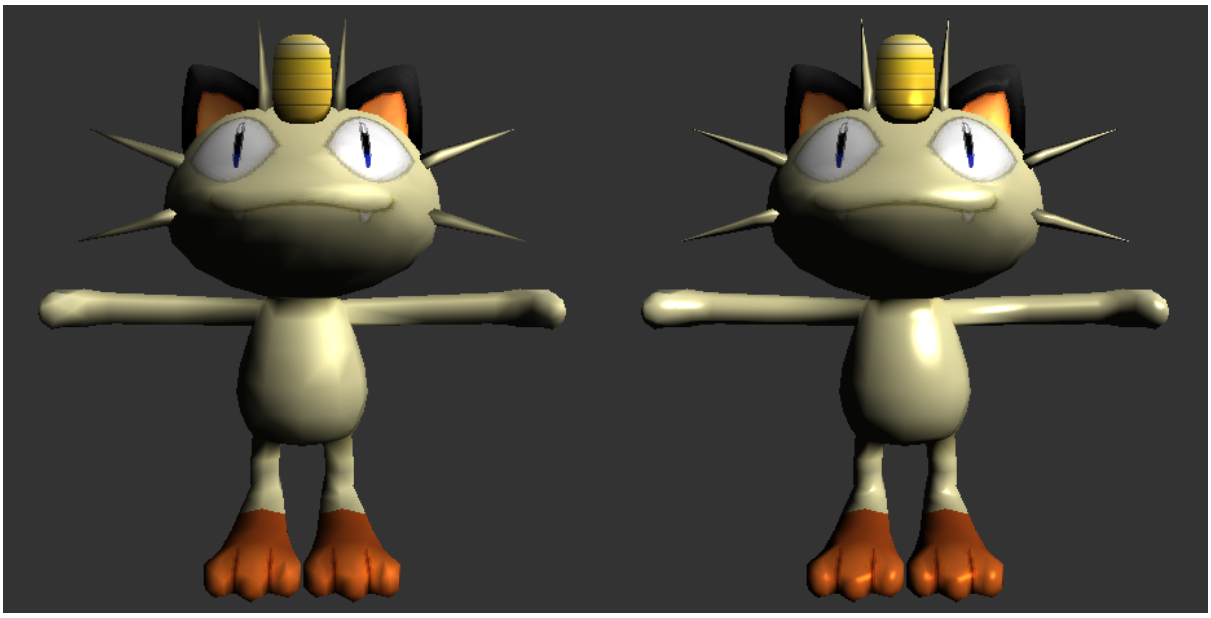
HW3_Report

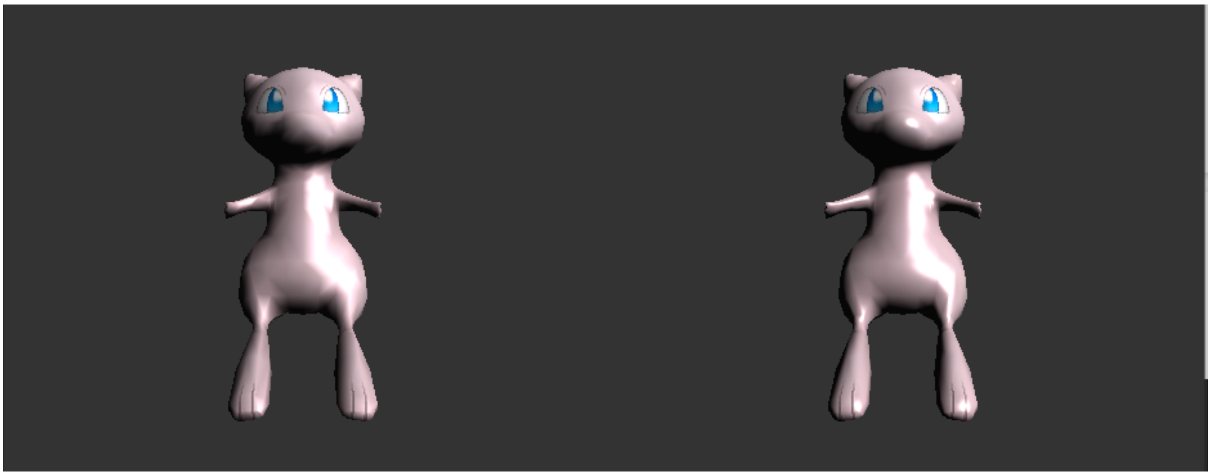
109065539 韓承翰

Demo：



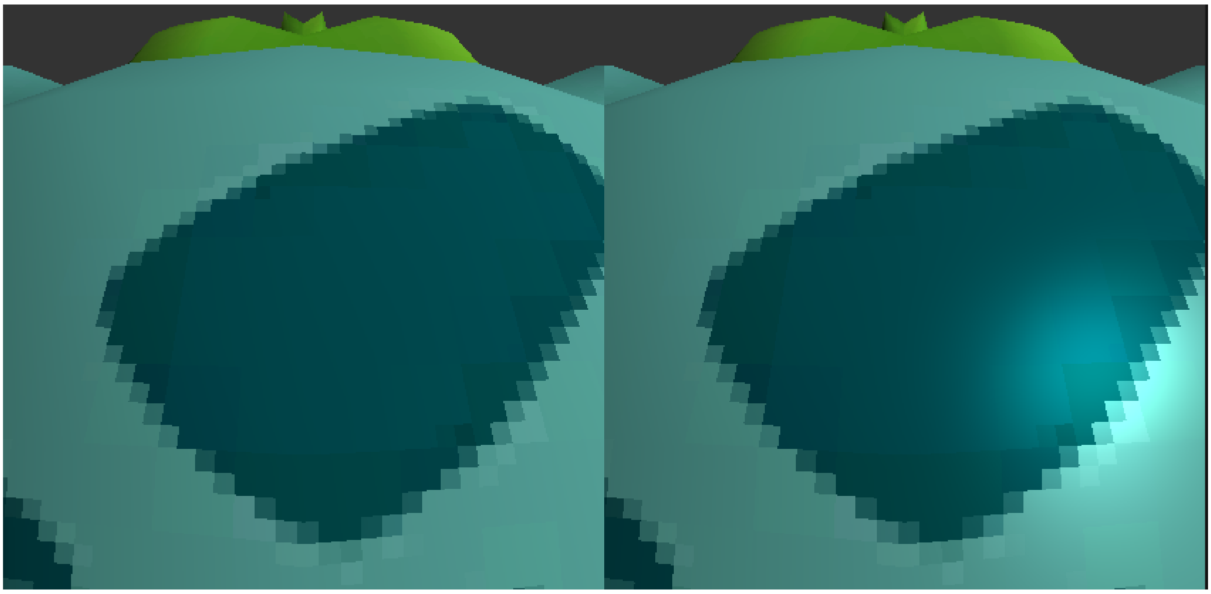




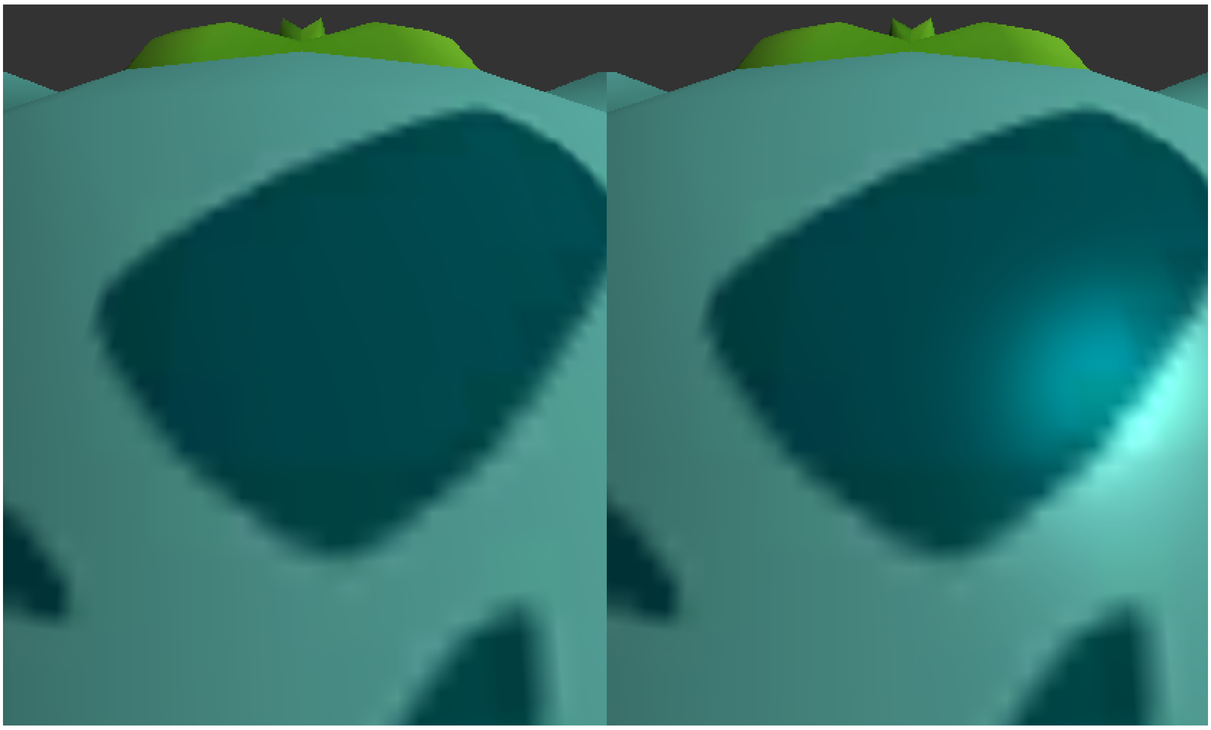


Magnification texture filtering mode:

- nearest sampling



- linear sampling

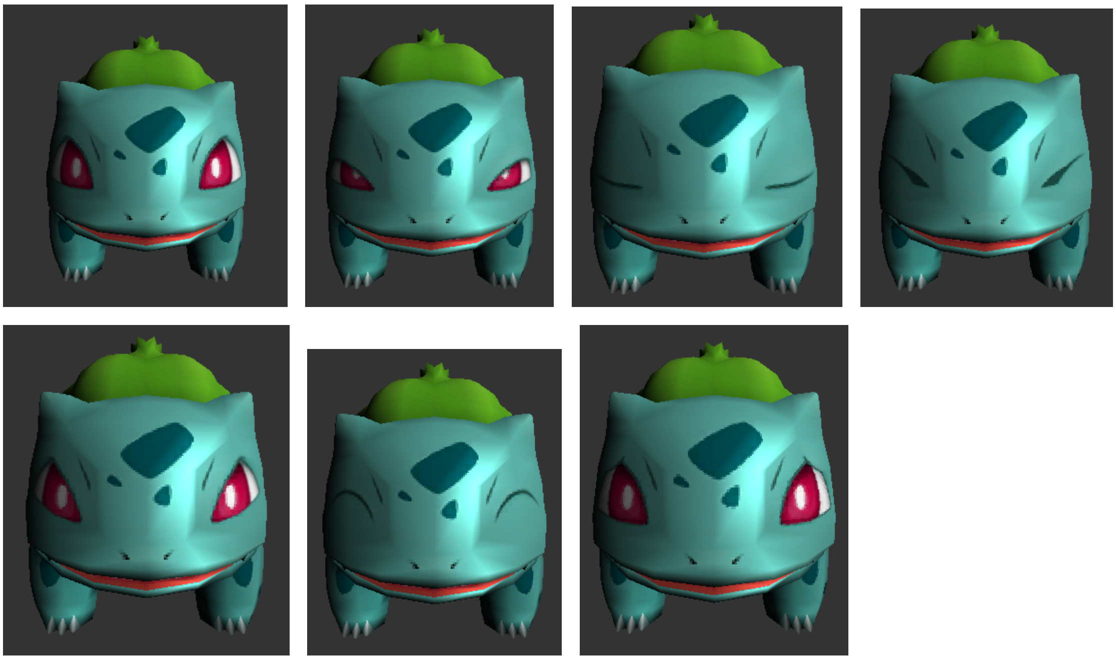


Magnification texture filtering mode：

- nearest sampling

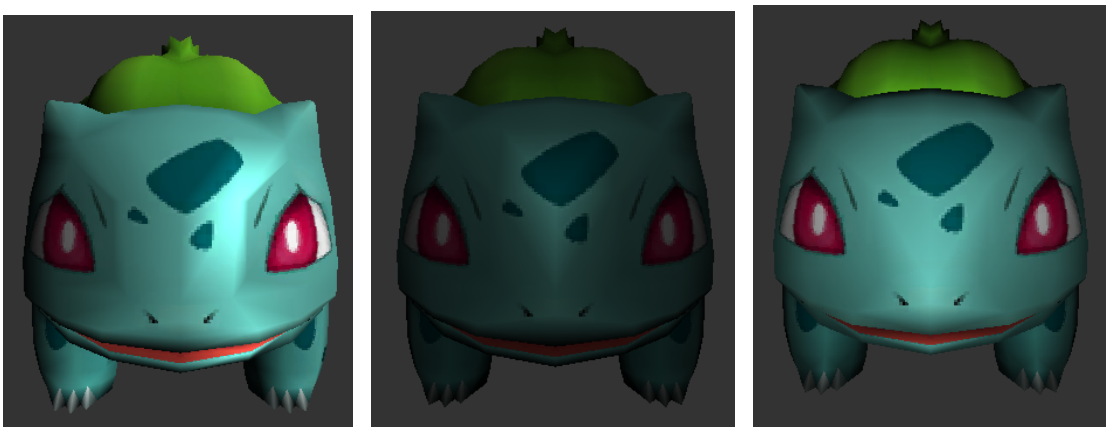
- linear_mipmap_linear

Texture transform on Pokemon models' eyes：**(1-7)**

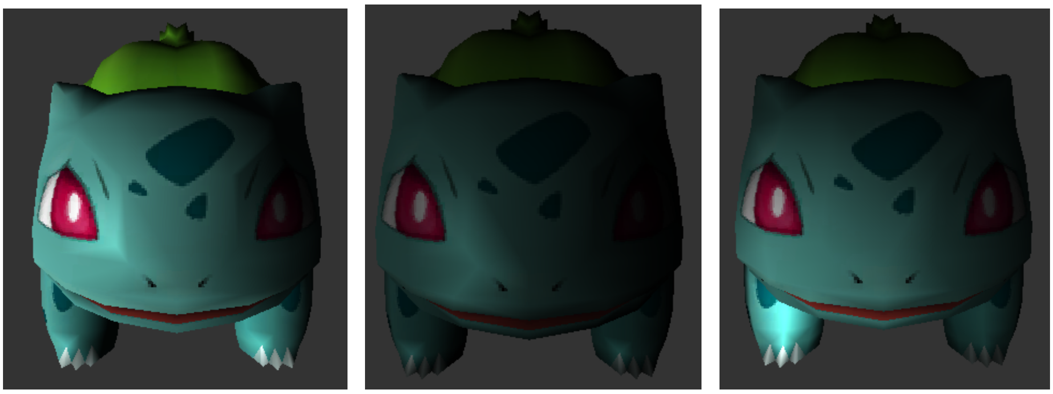


Lighting in previous Homework (HW2):

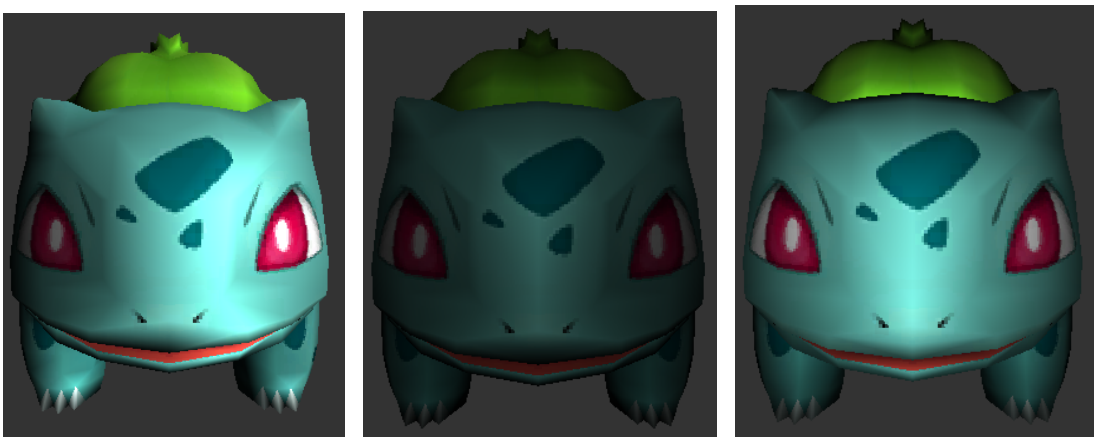
- Directional/Point/Spot light



- Light Editing Mode



- Shininess Editing Mode

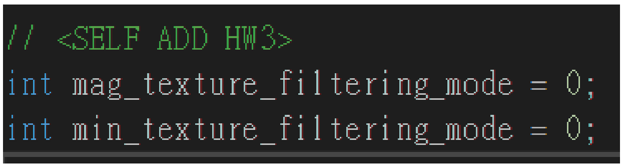


Code :

* Lightning部分與上次相同

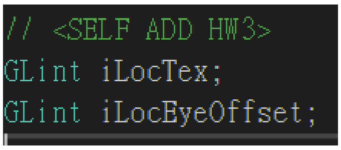
main.cpp

- 新增變數 :
  - ○ 紀錄目前texture filtering mode



```
// <SELF ADD HW3>
int mag_texture_filtering_mode = 0;
int min_texture_filtering_mode = 0;
```

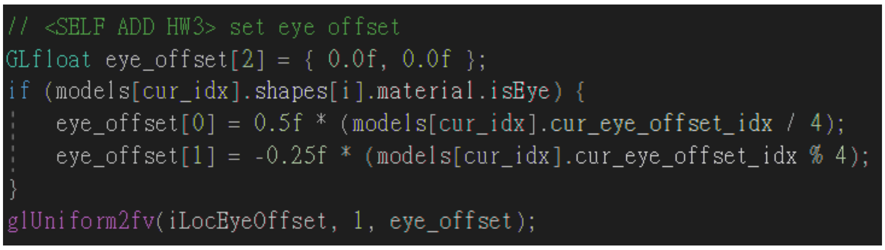
  - ○ 新增傳到shader的參數



```
// <SELF ADD HW3>
GLint iLocTex;
GLint iLocEyeOffset;
```

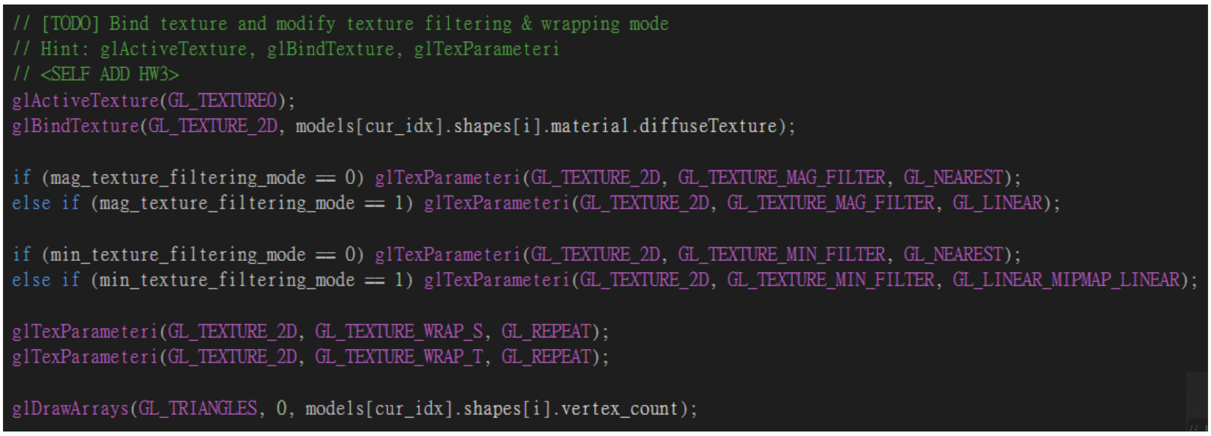
- 修改function :
  - ○ RenderScene() :

將eye_offset傳進shader中



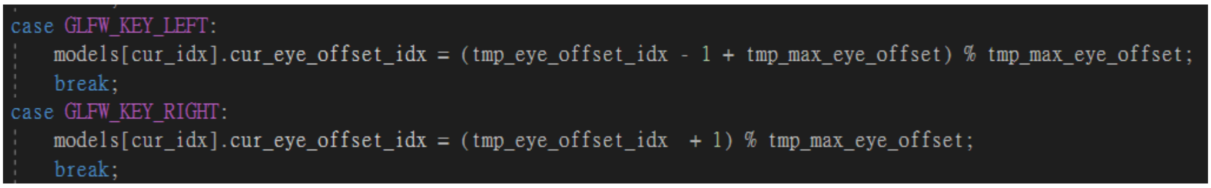
```
// <SELF ADD HW3> set eye offset
GLfloat eye_offset[2] = { 0.0f, 0.0f };
if (models[cur_idx].shapes[i].material.isEye) {
    eye_offset[0] = 0.5f * (models[cur_idx].cur_eye_offset_idx / 4);
    eye_offset[1] = -0.25f * (models[cur_idx].cur_eye_offset_idx % 4);
}
glUniform2fv(iLocEyeOffset, 1, eye_offset);
```

Bind texture



```
// [TODO] Bind texture and modify texture filtering & wrapping mode
// Hint: glActiveTexture, glBindTexture, glTexParameteri
// <SELF ADD HW3>
glActiveTexture(GL_TEXTURE0);
glBindTexture(GL_TEXTURE_2D, models[cur_idx].shapes[i].material.diffuseTexture);

if (mag_texture_filtering_mode == 0) glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, GL_NEAREST);
else if (mag_texture_filtering_mode == 1) glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, GL_LINEAR);

if (min_texture_filtering_mode == 0) glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_NEAREST);
else if (min_texture_filtering_mode == 1) glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_LINEAR_MIPMAP_LINEAR);

glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_S, GL_REPEAT);
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_WRAP_T, GL_REPEAT);

glDrawArrays(GL_TRIANGLES, 0, models[cur_idx].shapes[i].vertex_count);
```

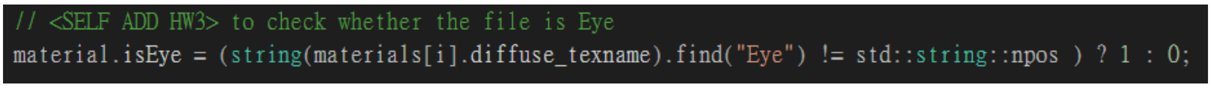
- KeyCallback(): 左右鍵



```
case GLFW_KEY_LEFT:
    models[cur_idx].cur_eye_offset_idx = (tmp_eye_offset_idx - 1 + tmp_max_eye_offset) % tmp_max_eye_offset;
    break;
case GLFW_KEY_RIGHT:
    models[cur_idx].cur_eye_offset_idx = (tmp_eye_offset_idx  + 1) % tmp_max_eye_offset;
    break;
```

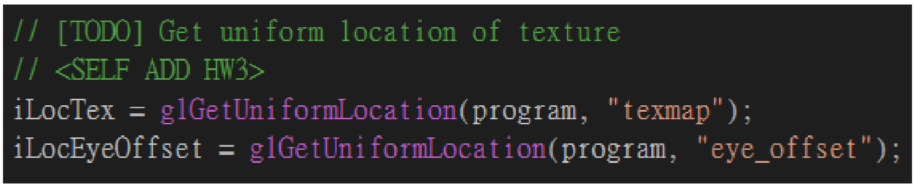
- LoadTexturedModels():

根據檔名確認是否是眼睛檔案，來設定material中的isEye



```
// <SELF ADD HW3> to check whether the file is Eye
material.isEye = (string(materials[i].diffuse_texname).find("Eye") != std::string::npos ) ? 1 : 0;
```

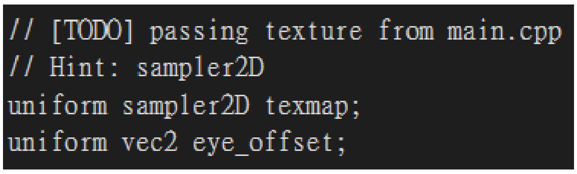
- setUniformVariables():



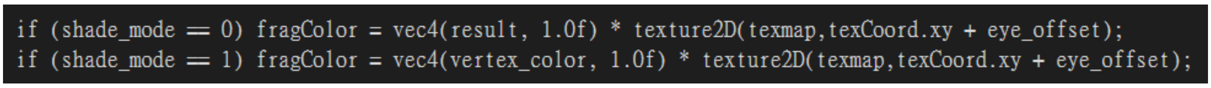
```
// [TODO] Get uniform location of texture
// <SELF ADD HW3>
iLocTex = glGetUniformLocation(program, "texmap");
iLocEyeOffset = glGetUniformLocation(program, "eye_offset");
```

shader.fs.glsl

本次新增/修改部分：



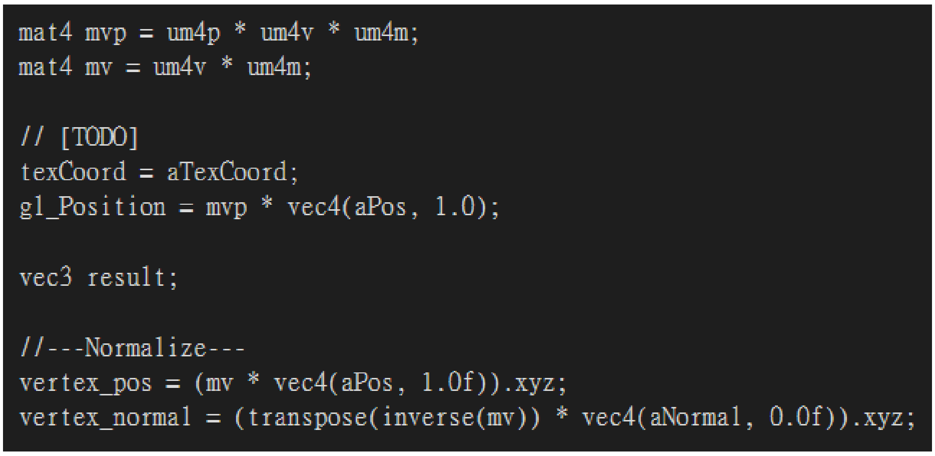
```
// [TODO] passing texture from main.cpp
// Hint: sampler2D
uniform sampler2D texmap;
uniform vec2 eye_offset;
```



```
if (shade_mode == 0) fragColor = vec4(result, 1.0f) * texture2D(texmap,texCoord.xy + eye_offset);
if (shade_mode == 1) fragColor = vec4(vertex_color, 1.0f) * texture2D(texmap,texCoord.xy + eye_offset);
```

shader.vs.glsl

本次新增/修改部分：



```
mat4 mvp = um4p * um4v * um4m;
mat4 mv = um4v * um4m;

// [TODO]
texCoord = aTexCoord;
gl_Position = mvp * vec4(aPos, 1.0);

vec3 result;

//---Normalize---
vertex_pos = (mv * vec4(aPos, 1.0f)).xyz;
vertex_normal = (transpose(inverse(mv)) * vec4(aNormal, 0.0f)).xyz;
```