

作业二：光照与纹理映射

代码目录结构如下：

```
code/
├── main.py                    # 主程序
├── framework/
│   ├── __init__.py
│   ├── renderer.py          # 渲染器
│   ├── geometry.py          # 几何体生成
│   ├── texture_loader.py     # 纹理加载
│   └── camera.py            # 相机控制
├── shaders/
│   ├── task1_sphere.vert     # 任务1: 球面顶点着色器
│   ├── task1_sphere.frag     # 任务1: 纹理采样
│   ├── task2_antialias.frag  # 任务2: 反走样
│   ├── task3_flat.vert      # 任务3: Flat Shading
│   ├── task3_flat.frag      # 任务3: Flat Shading
│   ├── task3_gouraud.vert    # 任务3: Gouraud Shading
│   ├── task3_gouraud.frag    # 任务3: Gouraud Shading
│   ├── task3_phong.vert      # 任务3: Phong Shading
│   ├── task3_phong.frag     # 任务3: Phong Shading
│   ├── task4_bump.frag       # 任务4: Bump Mapping
│   ├── task5_shadow.vert     # 任务5: 阴影计算
│   └── task5_shadow.frag     # 任务5: 阴影计算
├── resources/
│   ├── earthmap.jpg
│   ├── earthbump.jpg
│   └── chessboard.jpg
└── requirements.txt
```

主要考核：光照计算、纹理映射、凹凸纹理、阴影图等。具体任务要求如下：

1. 球面三角化与纹理映射 (2分):
 - 在顶点着色器中正确处理球面顶点变换(shaders\task1_sphere.vert);
 - 在片段着色器中实现纹理采样(shaders\task1_sphere.frag)
2. 纹理贴图反走样 (2分):
 - 在片段着色器中实现 Mipmap 或 多重采样(shaders\task2_antialias.frag)
3. Blinn-Phong 光照模型(3分):

- Flat Shading: 修改顶点着色器 (shaders\task3_flat.vert) 和片段着色器 (shaders\task3_flat.frag)
- Gouraud Shading: 修改顶点着色器 (shaders\task3_gouraud.vert) 和片段着色器 (shaders\task3_gouraud.frag)
- Phong Shading: 修改顶点着色器 (shaders\task3_phong.vert) 和片段着色器 (shaders\task3_phong.frag)

4. Bump Mapping (2 分):

- 在片段着色器中读取高度图, 扰动法线并应用到光照计算 (shaders\task4_bump.frag)

5. 阴影贴图 (1 分):

- 在着色器中实现阴影计算 (shaders\task5_shadow.vert, shaders\task5_shadow.frag)

6. 自由创作 (可选, 2 分):

- 在现有的框架下, 重写一个顶点着色器和片元着色器, 渲染场景可以不变, 用着色器自定义渲染效果, 例如实现金属、玻璃等不同材质效果, 添加天空盒或环境反射等。没有具体要求, 自由发挥即可。

Ps: 为了更好地对比效果, 可以修改 py 文件里的一些参数, 例如相机位置、球面几何精细度等。

作业提交要求:

- i) 代码 + 报告
- j) 报告里需简要说明每个考核点的实现过程, 并给出最终渲染的结果图像。

k) 会根据代码实现细节和渲染的结果图进行给分。

l) 文件命名：学号+姓名+作业二.zip 上传地址为

<https://bhpan.buaa.edu.cn/link/AA8E946B09A94642D497151463D5F53ED7>

文件夹名：作业二

有效期限：2025-12-15 23:59

提取码：iOZt