USTC-CG/2024 课程作业 实验报告

实验 6	Shaders
马天开	PB21000030 (ID: 08)
Due: 2024.04.14	Submitted: 2024.04.14

功能实现 Features Implemented

作业要求部分 Required Features

实现 Blinn-Phong 着色模型

TBN 矩阵的计算：

```
vec3 normalmap_value = texture2D(normalMapSampler, vTexcoord).xyz;
normal = normalize(vertexNormal);

vec3 edge1 = dFdx(vertexPosition);
vec3 edge2 = dFdy(vertexPosition);
vec2 deltaUV1 = dFdx(vTexcoord);
vec2 deltaUV2 = dFdy(vTexcoord);

tangent = edge1 * deltaUV2.y - edge2 * deltaUV1.y;

if(length(tangent) < 1E-7) {
    vec3 bitangent = -edge1 * deltaUV2.x + edge2 * deltaUV1.x;
    tangent = normalize(cross(bitangent, normal));
}

tangent = normalize(tangent - dot(tangent, normal) * normal);
bitangent = normalize(cross(tangent,normal));
```

Blinn-Phong 着色模型的计算：

```
vec3 lightDir = normalize(lights[i].position - pos);
vec3 viewDir = normalize(camPos - pos);
vec3 halfwayDir = normalize(lightDir + viewDir);
vec3 light_color = lights[i].color;
vec3 ambient_color = 0 * light_color;
vec3 diffuse_color = texture2D(diffuseColorSampler, uv).xyz;

float diffuse = abs(dot(normal, lightDir));
float specular = abs(dot(normal, halfwayDir));

if (diffuse == 0.0) {
    specular = 0.0;
```

```
    } else {
        specular = pow(specular, shininess);
    }

    Color = vec4(ambient_color + diffuse * light_color * diffuse_color + specular *
    light_color, 1.0);

    int shadow_map_id = lights[i].shadow_map_id;
    float shadow_map_value = texture(shadow_maps, vec3(uv, shadow_map_id)).x;
```

实现 **Shadow Mapping** 算法

参考 Learn OpenGL CN

Render Graph

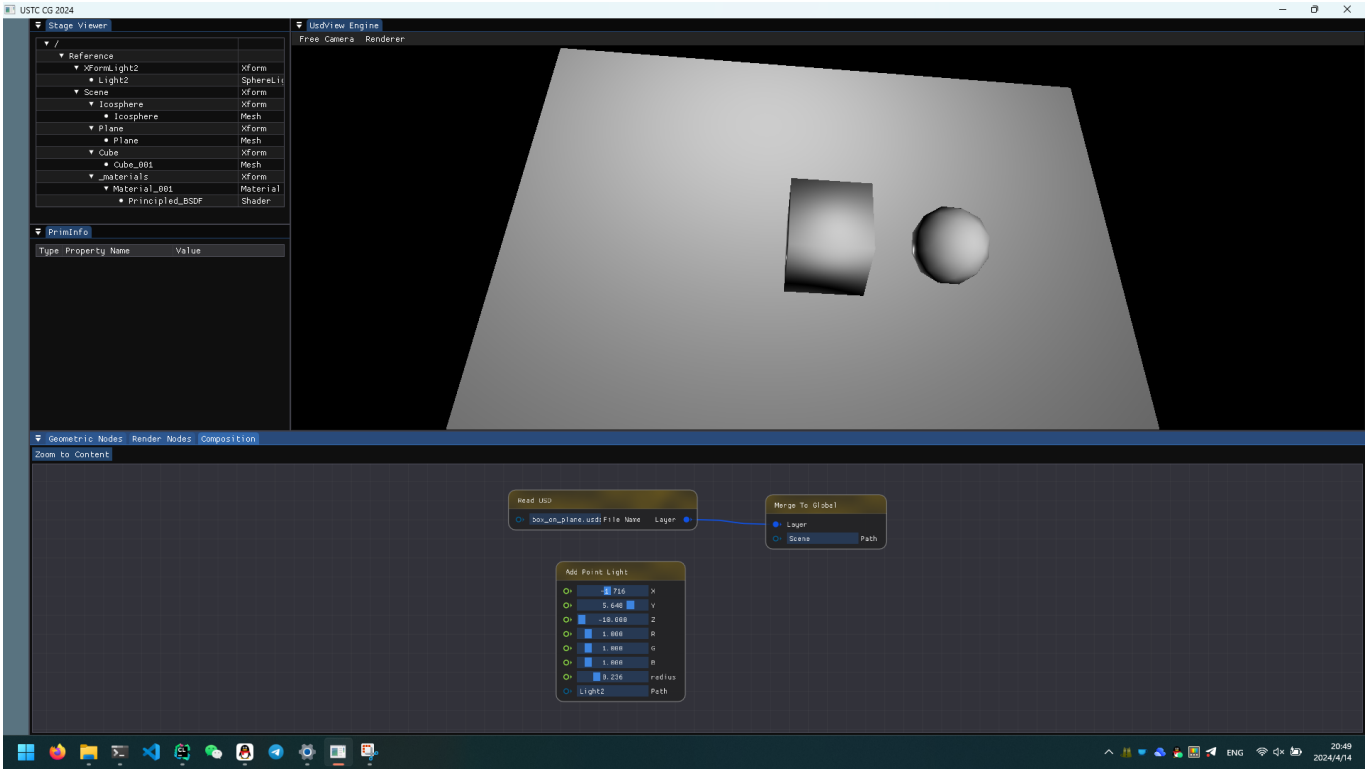
```
graph LR
    C[Scene Camera] --> R[Rasterize]
    M[Scene Meshes] --> R
    Mt[Scene Materials] --> R
    L[Scene Lights] --> S[Shadow Maping]
    M --> S
    S --> D[Deferred Lighting]
    R --> D
    D --> P[Present]
```

Composition Graph

```
graph TD
R[Read USD] --> M[Merge to Global]
A[Add Point Light]
```

额外功能 Extra Features

运行截图 Screenshots


USTC CG 2024
Stage Viewer
/
Reference
XFormLight2 Xform
Light2 SphereLi
Scene Xform
Icosphere Xform
Icosphere Mesh
Plane Xform
Plane Mesh
Cube Xform
Cube_001 Mesh
_materials Xform
Material_001 Material
Principled_BSDF Shader
PrimInfo
Type Property Name Value
UsdView Engine
Free Camera Renderer
Geometric Nodes Render Nodes Composition
Zoom to Content
Read USD
box_on_plane.usd File Name Layer
Merge To Global
Layer
Scene Path
Add Point Light
-1.716 X
5.648 Y
-18.000 Z
1.000 R
1.000 G
1.000 B
0.236 radius
Light2 Path
ENG
20:49
2024/4/14