

蒙特卡洛光线跟踪报告

一、实现功能

- 加载 obj 和 mtl 格式文件并整合场景；
- 蒙特卡洛跟踪：光线跟踪、漫反射随机采样、金属材质不均匀采样；
- 线程加速。

二、实验环境

- 操作系统：Windows 10 企业版
- 处理器：Intel(R) Core(TM) i7-6700k CPU @ 4.00GHz
- 显示适配器：NVIDIA GeForce GTX 1070
- 编译平台：Visual Studio 2015
- 库调用：Eigen

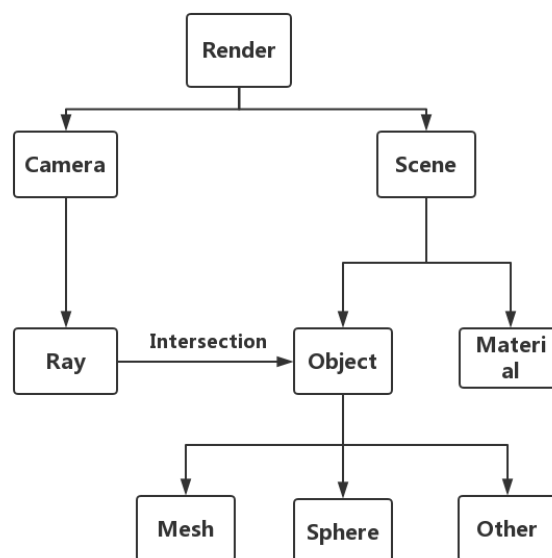
三、执行操作

通过改变 GlobalControl.h 头文件内的参数值，实现切换场景、改变采样和光线跟踪深度（路径设置在 main.h 里）。

```
#define RAY_TRACE_DEPTH 7 //光线跟踪深度
#define SAMPLE_NUM 1000 //采样数
#define LIGHT_POWER 1 //光强
#define SCENE_NUM 2 //场景：1、2、3
```

四、实现思路

4.1 程序结构



3.2 源文件介绍

- GlobalControl.h/cpp 全局参数调控
- Model.h/cpp 定义 mesh 结构以及导入、求交相关操作
- ShapeDefine.h/cpp 定义球体等自定义物体结构以及求交
- Material.h/cpp 定义材质结构
- Scene.h/cpp 整合场景物体
- Camera.h/cpp 定义相机结构
- Render.h/cpp 定义渲染过程

3.3 场景构建

场景通过 Scene.h 来调用 Model.h 导入 Mesh 结构，调用 ShapeDefine.h 里的 Sphere 实现球体，以此保证球面的光滑度和运算的速度。

3.4 求交计算

对于 mesh 物体，通过克拉默法则进行射线与三角形面片的求交判断，其中矩阵利用 Eigen 库计算。对于球体，直接利用球心坐标和半径进行相关计算。

3.5 随机取样

- 漫反射的随机取样使用蒙特卡洛法进行随机取样，具体如下。

```
double r1 = 2 * M_PI * drand48(), r2 = drand48(), r2s = sqrt(r2);
Double3 w = normal_, u = normalize((fabs(w.x) > .1 ? Double3(0, 1) : Double3(1)) % w), v = w % u;
Double3 diff = normalize(u * cos(r1) * r2s + v * sin(r1) * r2s + w * sqrt(1 - r2));
```

- 金属材质的随机采样使用半球面加权采样，具体如下。

```
Double3 w = reflect;
Double3 u = normalize((w.z != 0) ? Double3(-w.z, 0, w.x) : Double3(0, -w.z, w.y));
Double3 v = normalize(u % w);

double r1 = drand48(), r2 = drand48();
double theTa = acos(pow(r1, 1.0 / (ns + 1)));
double phy = 2 * M_PI * r2;

Double3 reflect = normalize(sin(theTa)*cos(phy) * u + sin(theTa) * sin(phy) * v + cos(theTa) * w);
```

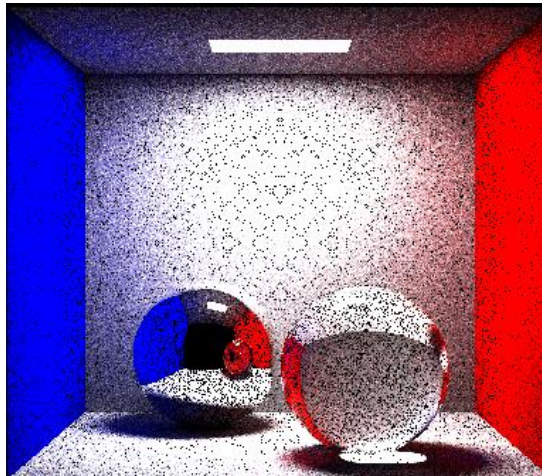
3.6 线程加速

通过 Thread 将 Rendering 根据显示屏高度平均四等分，以此提速四倍。

```
void Render::renderingThread()
{
    std::thread t1(&Render::rendering, this, 0, 100, SAMPLE_NUM);
    std::thread t2(&Render::rendering, this, 100, 200, SAMPLE_NUM);
    std::thread t3(&Render::rendering, this, 200, 300, SAMPLE_NUM);
    std::thread t4(&Render::rendering, this, 300, 400, SAMPLE_NUM);
    t1.join();
    t2.join();
    t3.join();
    t4.join();
}
```

五、 实验结果

场景 1: (深度, 采样, 光强) 分别为 (1000, 7, 200), 耗时约 18 分钟。



场景 2: (深度, 采样, 光强) 分别为 (1000, 7, 1), 耗时约 25 分钟。

