王蕙钰 2100018733

**可视计算与交互概率 期末大作业报告**

**一．选题**：C. Rendering 5.Path Tracing

Case1：实现路径追踪，一种基于渲染方程的全局光照渲染方法。在界面提供UI接口以调整参数和显示进度。使用bvh包围盒结构加速。

Case2:在路径追踪基础上，通过刻意使用不合理的采样生成艺术纹理效果。

**二．实现思路**：

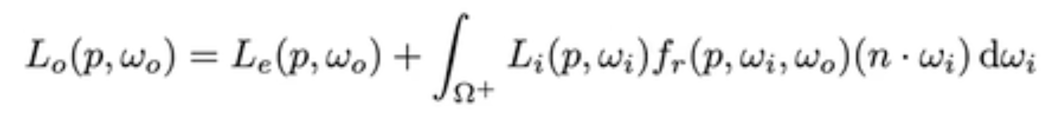
1. 文件结构：

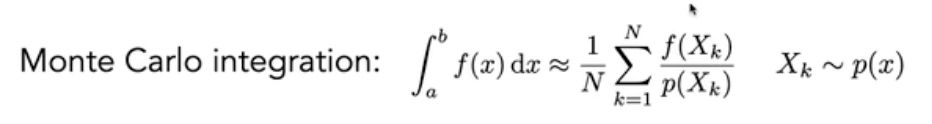
花了比较多的时间理解lab框架的博大精深。本作业基于lab0的框架（运行请输入xmake, xmake run lab0），case1沿用CaseFixed.cpp和CaseFixed.h，功能实现上添加了PathTracing.cpp和PathTracing.h；case2沿用CaseResizable.cpp和Case Resizable.h，功能实现上添加了Painterly.cpp和Painterly.h。

1. 路径追踪：

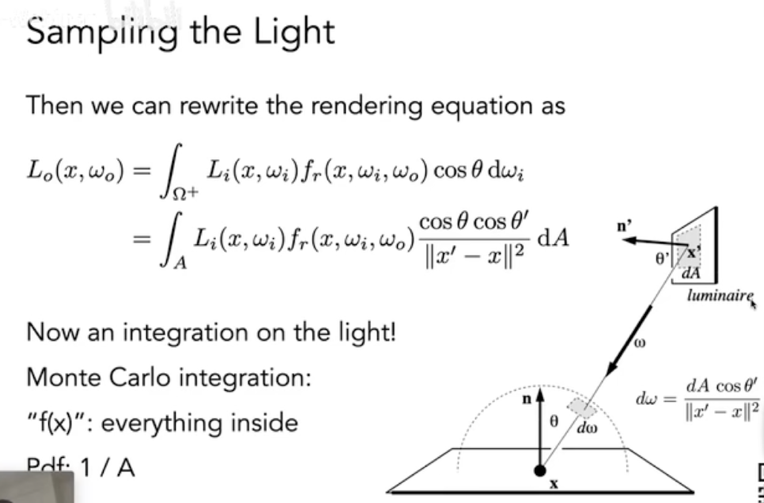
参考了smallpt[[1]](#footnote-1)和对它的解读[[2]](#footnote-2)与改进[[3]](#footnote-3)。原理参考了GAMES101 Lecture16[[4]](#footnote-4)。

要解渲染方程：

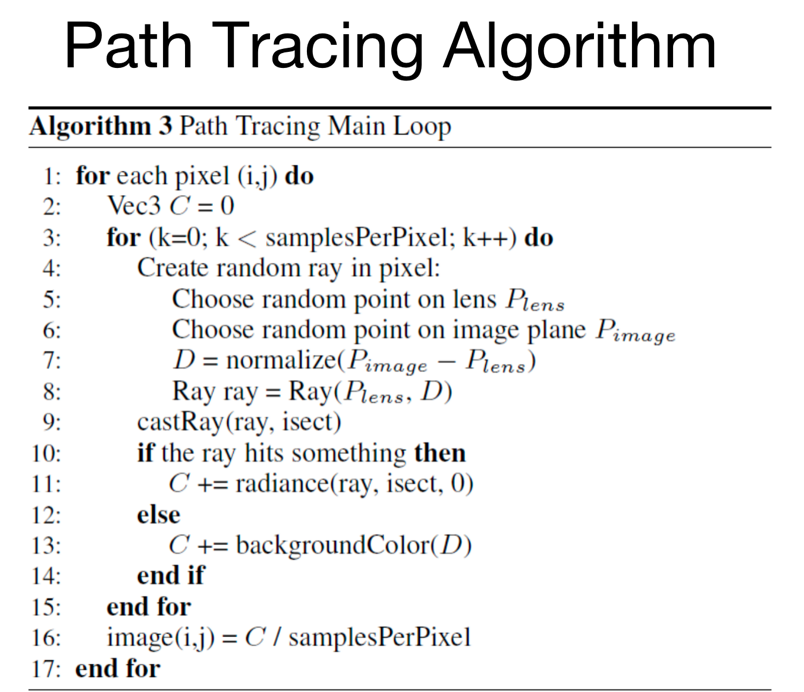


计算积分使用蒙特卡洛方法。

对于每一个着色点，均匀随机选取一个方向发射一条光，如果达到光源则为直接光照，添加光源贡献；若达到其他物体，计算间接光照，递归求交点反射过来的光。

1）为防止光束数量呈指数级增长，只随机选取一个方向发射光，但穿过一个像素可以发射多条光减少噪声。2）递归终止：使用俄罗斯轮盘赌RR，随机决定终止。以一定概率P发射光线，得到的结果/P保证期望不变。3）提升发射光束的效率（打到光源），直接光照从光源采样（将直接光照和间接光照分开计算）

路径追踪代码流程如下：



1. bvh包围盒结构：

通过构建bvh包围盒加速intersect。具体步骤：首先构建包围盒，由于aabb与坐标轴平行，参数只需引入坐标最小的点和最大的点。对当前所有物体先构建大的包围盒，再给物体按照一根坐标轴上的位置排序，在中点进行分割，递归构建子树，直到只剩一个物体。求光线与物体焦点时，先判断光线与包围盒是否相交，如果是叶节点再测试光线是否与球体相交，减少需要求交的物体数。代码编写中特别注意需要创建一个新的索引indices记录物体排序结果，不能和物体本身的序号弄错了。

本项目中由于场景比较简单，添加了包围盒结构后速度没有明显提升(甚至觉得更慢了)。因此将带包围盒(finalproject\_bvh)和不带包围盒(finalproject)的工程文件都提交了。

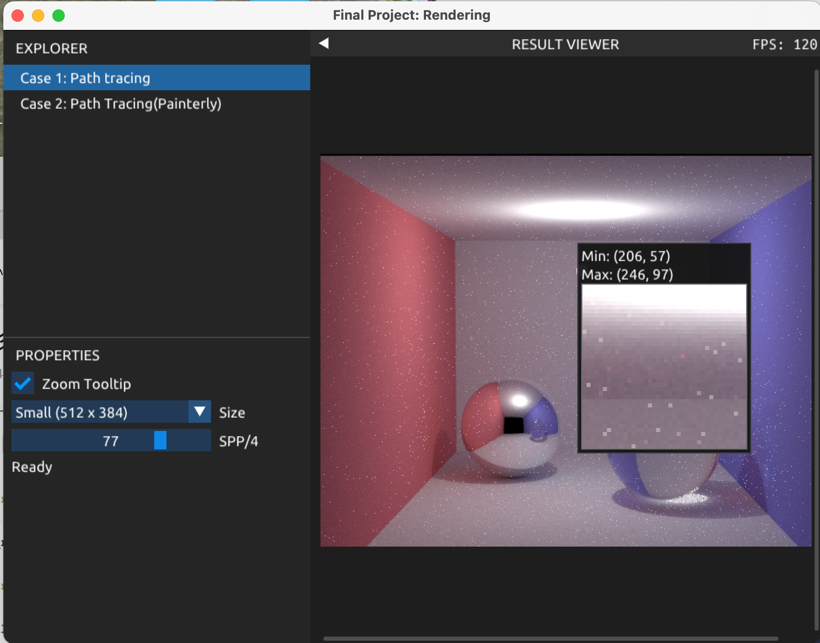
1. 艺术纹理效果

参考Smallpaint[[5]](#footnote-5)，在路径追踪基础上通过不合理的采样生成艺术纹理效果。研究smallpaint的代码可以发现，主要问题在于做路径追踪处理反射的hemisphere sampling不合理。hemisphere sampling用于给定一个表面的法向量，随机采样该法向量上半球中的一个方向。

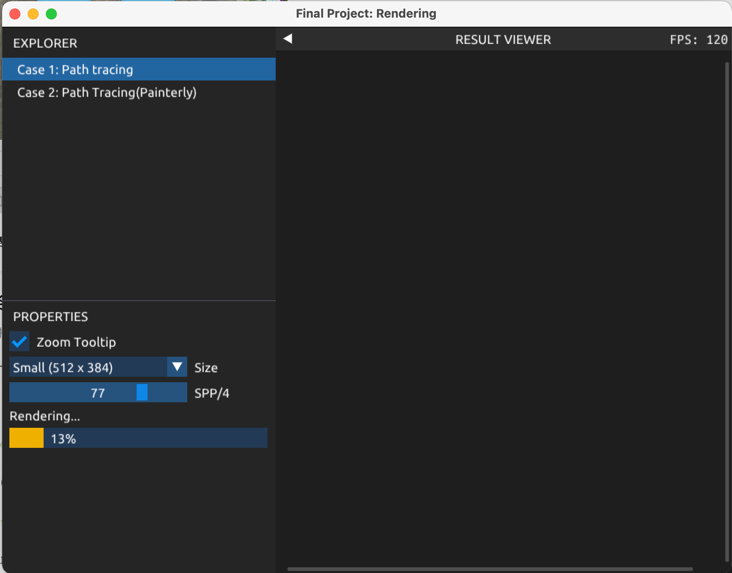
Smallpt中的正确做法是：以法线为z轴建立局部坐标系，生成两个随机数，使用余弦加权来随机采样半球方向，再转换回世界坐标系。而smallpaint中的做法是：以世界坐标系的z轴构成标准上半球采样，在直接与法线相加，导致采样并不均匀。case2中我采用了smallpaint中的采样方法，复刻扰动带来的纹理效果。

**三．效果：**

1. UI交互：
   1. 可以选择图像大小和SPP，zoom的功能可以查看细节



* 1. 渲染过程中有进度条提示渲染进度



1. 渲染结果
   1. Case 1: 路径追踪

|  |  |
| --- | --- |
|  |  |
| spp = 1 \* 4 | spp = 46 \* 4 |
|  | |
| spp=100 \* 4 | |

* 1. Case 2: 路径追踪（绘画效果）

|  |  |
| --- | --- |
|  |  |
| spp = 1 \* 4 | spp = 45 \* 4 |
|  | |
| spp = 100 \* 4 | |

1. http://www.kevinbeason.com/smallpt/ [↑](#footnote-ref-1)
2. https://drive.google.com/file/d/0B8g97JkuSSBwUENiWTJXeGtTOHFmSm51UC01YWtCZw/view?resourcekey=0-BSMES1GmPEtIIcv6EgBTjw [↑](#footnote-ref-2)
3. http://www.kevinbeason.com/smallpt/explicit.cpp [↑](#footnote-ref-3)
4. https://www.bilibili.com/video/BV1X7411F744?vd\_source=c27f680575ff5da483336a4a7f1602f8&p=16&spm\_id\_from=333.788.videopod.episodes [↑](#footnote-ref-4)
5. https://users.cg.tuwien.ac.at/zsolnai/gfx/smallpaint/ [↑](#footnote-ref-5)