

基于 Bézier 曲面的 3D 图像真实感绘制 实验报告

张益玮 2016011319

yw_zhangthu@163.com

1 绘制算法 (Renderer.cpp)

我使用了 PPM(Progressive Photon Mapping) 算法进行 3D 图像真实感绘制。

光线追踪 (PASS1): 对每个像素进行光线追踪, 获得碰撞点图, 碰撞点与漫反射表面相关联。

发射光子 (PASS2): 从光源发射一定数量的光子。每当光子打在漫反射表面, 在以 KDTree 形式构建的碰撞点图中找到半径内包含光子的碰撞点, 更新碰撞点的光通量和光子计数器。

更新碰撞点图的信息: 更新碰撞点的累积光通量和半径, 更新以 KDTree 形式构建的碰撞点图的节点维护的最大半径。

估计辉度值: 根据论文中的公式进行计算。

2 附加功能

2.1 参数曲面求交 (Bezier.cpp)

使用具有 16 个控制点的双三次 Bezier 曲面进行参数曲面的造型。使用了包围盒和牛顿迭代法求解参数曲面求交问题。

1. 四分曲面：建立一个队列，每次将队首的参数曲面进行四分，将子参数曲面中曲面的包围盒与光线有交的加入到队列中，直到队列为空或者曲面足够小。
2. 牛顿迭代法：对队列中的可能存在交点的小曲面使用牛顿迭代法，并且在所有交点中选择距离最近的一个。取牛顿迭代法的迭代次数为 50 次。

2.2 包围盒加速 (Bezier.cpp&utils.cpp)

使用了最简单的 AABB 包围盒对求交加速，以 KDTree 的形式来构建碰撞点图，加速了询问过程。

2.3 OpenMP(Renderer.cpp)

使用了 OpenMP 进行加速，48 核并行加速明显。

2.4 景深 (Camera.cpp)

修改 Camera 类发射光线的函数，对光线的出发点在光圈上做一个随机扰动。

2.5 软阴影 & 抗锯齿

使用 PPM 算法之后，软阴影和抗锯齿问题得到解决，并且随着 PPM 迭代轮数的增加，碰撞点的半径减小，图像更加接近于真实效果。

2.6 UV 纹理贴图 (Sphere.cpp&Plane.cpp&NormalSphere.cpp)

对球体构造球坐标系映射，对平面进行无缝拼接，对参数曲面同样构造映射。使用了双二次线性插值进行贴图的平滑。

2.7 法线贴图 (NormalSphere.cpp)

同样进行球坐标系映射，将存储在 rgb 图片中的 Normal Map 信息通过 rgb 值归一化之后乘 2 减 1 得到法线信息。根据三个正交向量的权值得到新的法向量。

3 效果图及参数曲面模型

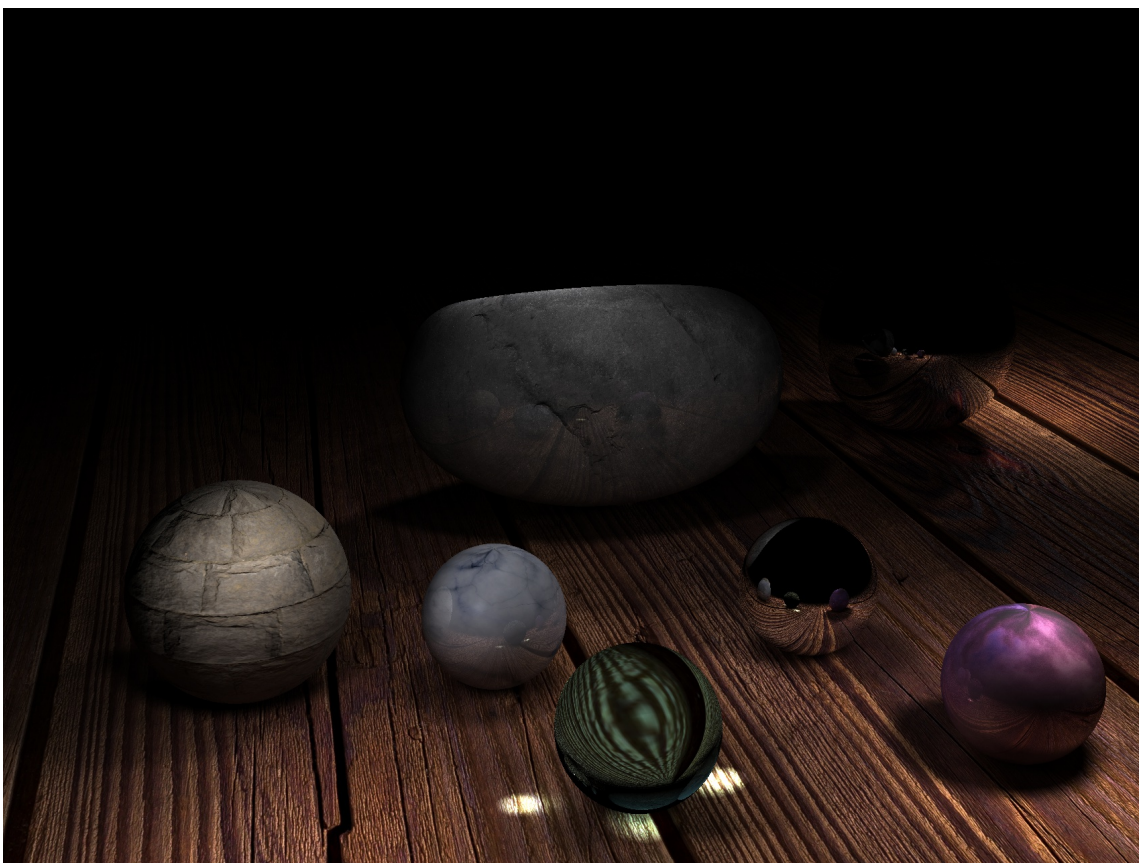


图 1: 最终效果图



图 2: 景深效果图

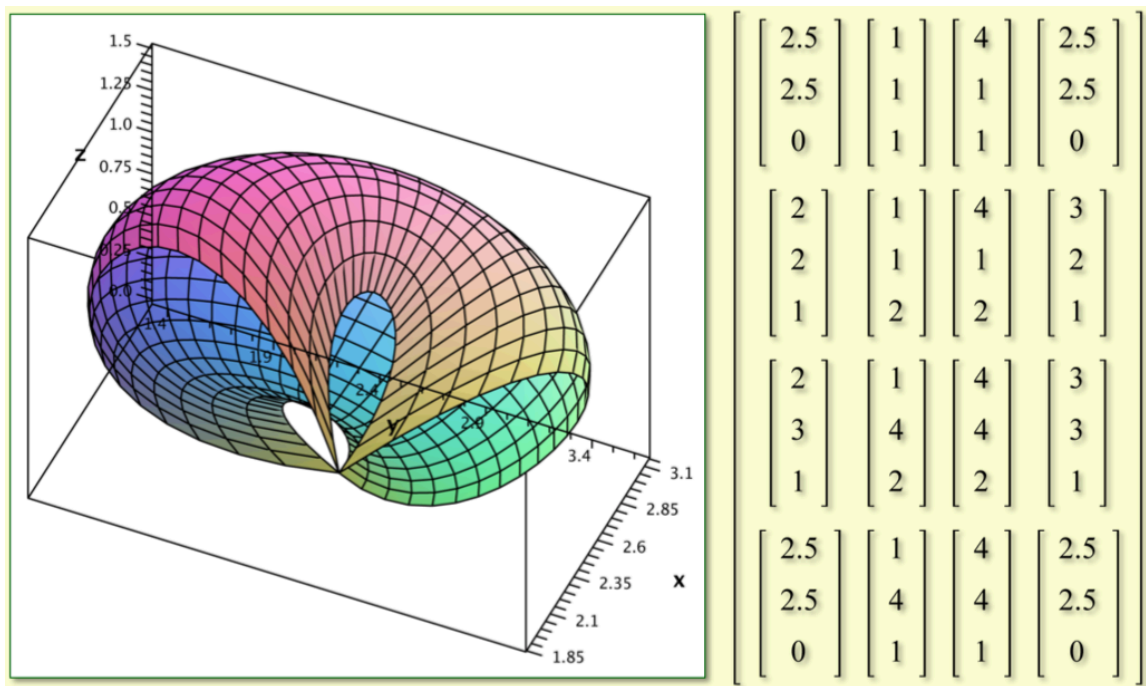


图 3: Bezier 曲面控制点及模型