

# 计算机图形学基础

## 作业 3：网格场景的局部光照与光线追踪渲染

Name ID

---

### 实验目的

渲染基础的三角网格场景。具体而言，在作业 2 的基础上，实现更丰富的渲染算法，具体包括：

- Blinn-Phong 局部光照；
- PBR 点光源与 IBL 环境光（包括漫反射与镜面反射）；
- 光线追踪（路径追踪）渲染器，包括 Lambertian 漫反射、镜面反射与玻璃折射，以及 Blinn-Phong 局部光照近似。

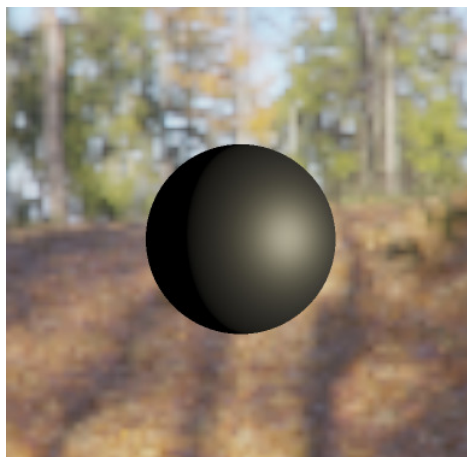
### 方法

场景在 Blender 2.93 中构建，导出为 Wavefront OBJ 格式。加载时，会同时加载模型 .obj 与材质 .mtl，但是一些特殊材质（如玻璃和镜面）的参数由物体名称确定。详见源代码 `src/scene_loader.rs`。

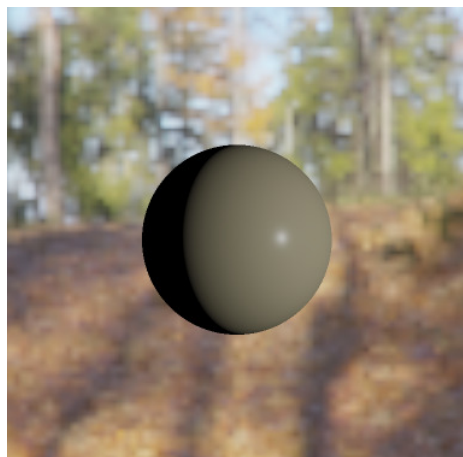
程序使用的 OpenGL 版本为 3.3。

接下来简要介绍一些部分的细节。

#### 1 PBR 点光源



金属度 = 0.9，粗糙度 = 0.6



金属度 = 0.05，粗糙度 = 0.25

对于物体表面的一个点  $P$  以及从  $P$  出发指向半球表面的单位向量  $\omega_o$ ，希望计算某个时刻所有方向射向  $P$  最终被物体以  $\omega_o$  方向反射的辐射率  $L_o(\omega_o)$ 。这可以由反射率方程（the reflectance equation）计算：

$$L_o(\omega_o) = \int_{\Omega} d\omega_i L_i(\omega_i) f_r(\omega_i, \omega_o) (n \cdot \omega_i),$$

其中  $\Omega$  是以  $P$  为中心、入射光线所在的半空间中的单位半球领域。  $L_i(\omega_i)$  表示从  $\omega_i$  方向入射的辐射率。  $f_r(\omega_i, \omega_o)$  为双向反射分布函数（BRDF），表示从  $\omega_i$  方向入射的光中有多少比例被反射到  $\omega_o$  方向。  $n$  表示物体表面的单位法向量。

这里采用 Cook-Torrance BRDF

$$f_r(\omega_i, \omega_o) = k_d \frac{c}{\pi} + k_s \frac{DG}{4(\omega_i \cdot n)(\omega_o \cdot n)}.$$

其中第一项为 Lambertian 漫反射，即半球内各向反射光强相等。\$k\_d\$ 是漫反射系数，刻画了入射光中有多大比例被漫反射。\$c\$ 为材质反照率。\$\frac{1}{\pi}\$ 为归一化因子，因为 \$\int\_{\Omega} d\omega\_i (n \cdot \omega\_i) = \pi\$（相当于求单位圆面积）。

第二项为 Cook-Torrance 镜面反射项。其中 \$k\_s\$ 是镜面反射系数，规定入射光中有多大比例被镜面反射。这通过 Fresnel 方程的 Schlick 近似公式计算

$$k_s(\omega_i, \omega_o) = F_0 + (1 - F_0) (1 - \cos \theta)^5,$$

其中 \$F\_0\$ 为平行法线入射光的反射比例，由材质性质决定。这里取 \$F\_0 = 0.04 + (c - 0.04)\beta\$，\$\beta\$ 为金属度参数。\$\theta\$ 为入射光与法线之间的夹角，即 \$\cos \theta = \omega\_i \cdot n\$。但在 Cook-Torrance 这样的微平面模型中，认为 \$n\$ 不是固定不变而是在宏观上服从某个随机分布的，这时在漫反射计算中用半程向量替换法向量：\$\cos \theta = \omega\_i \cdot h\$，\$h = \frac{\omega\_i + \omega\_o}{\|\omega\_i + \omega\_o\|}\$。注意到 \$\omega\_i\$ 与 \$\omega\_o\$ 关于 \$h\$ 对称，所以用 \$\omega\_o \cdot h\$ 计算也能得到相同的结果。

计算得到 \$k\_s\$ 后，由能量守恒性质 \$k\_d + k\_s = 1\$ 可得漫反射系数 \$k\_d\$。

\$D\$ 用于描述物体表面微平面法线的随机分布，表示为概率密度函数。\$D(n, h) dh\$ 表示法向量为 \$n\$ 的表面上有多少比例面积微平面的法向量在半程向量 \$h\$ 的邻域内，即可以将 \$\omega\_i\$ 方向的入射光反射至 \$\omega\_o\$ 方向。这里采用 Trowbridge-Reitz 分布即 GGX 分布

$$D(n, h) = \frac{\alpha^2}{\pi ((n \cdot h)^2 (\alpha^2 - 1) + 1)^2},$$

其中 \$\alpha\$ 是材质粗糙度参数的平方 \$r\_{\text{mat}}^2\$。

\$G\$ 用于刻画微平面对光线的遮蔽。\$G(n, \omega\_i, \omega\_o)\$ 表示法向量为 \$n\$ 的表面上有多少比例的法向量等于半程向量 \$h\$ 的微平面未被其余微平面遮蔽。这里采用 Smith 近似将未遮蔽比例分解为入射与出射两部分之乘积：

$$G(n, \omega_i, \omega_o) = \frac{n \cdot \omega_i}{(n \cdot \omega_i)(1 - k) + k} \cdot \frac{n \cdot \omega_o}{(n \cdot \omega_o)(1 - k) + k},$$

其中 \$k = \frac{(r\_{\text{mat}} + 1)^2}{8}\$，\$r\_{\text{mat}}\$ 为粗糙度。采取此映射是因为上式实际上是 Schlick 在 Smith 模型中应用 Beckmann 法线分布（而非 GGX）得到的结果，为保持粗糙度参数（GGX 中的 \$\alpha\$ 与 Beckmann 中的 \$k\$）对结果的影响基本一致，在 Schlick 式中取 \$k = \frac{\alpha}{2} = \frac{r\_{\text{mat}}^2}{2}\$；同时为避免反射光过亮，采取了 Disney 的改进，将粗糙度在计算前重映射为 \$r\_{\text{mat}} \leftarrow \frac{r\_{\text{mat}} + 1}{2}\$。

最后，BRDF 表达式中的 \$\frac{1}{4(\omega\_i \cdot n)(\omega\_o \cdot n)}\$ 是归一化因子。

将上述所有 Cook-Torrance BRDF 中的公式代入反射率方程即可计算点光源直接光照强度。点光源的入射辐射率可表示为 \$L\_i(\omega\_i) = \delta(\omega\_i - \iota) L\$，其中 \$\iota\$ 是点光源的入射方向单位向量，\$\delta\$ 是广义 Dirac delta 函数，\$L\$ 是光源的强度。由此，对于任何一个出射方向 \$\omega\_o\$，只需计算 \$\omega\_i = \iota\$ 单点的被积函数值，此即为整个积分的值。

## 2 PBR 环境光, IBL



金属度 = 0.9, 粗糙度 = 0.6



金属度 = 0.05, 粗糙度 = 0.25

同样采用 Cook-Torrance BRDF. 将 BRDF 方程代入反射率方程得

$$L_o(\omega_o) = \int_{\Omega} d\omega_i L_i(\omega_i) \left( k_d \frac{c}{\pi} + k_s \frac{DG}{4 (\omega_i \cdot n)(\omega_o \cdot n)} \right) (n \cdot \omega_i).$$

此积分可拆分为漫反射与镜面反射两项:

$$L_d(\omega_o) = \int_{\Omega} d\omega_i L_i(\omega_i) k_d \frac{c}{\pi} (n \cdot \omega_i), \quad L_s(\omega_o) = \int_{\Omega} d\omega_i L_i(\omega_i) k_s \frac{DG}{4 (\omega_i \cdot n)(\omega_o \cdot n)} (n \cdot \omega_i).$$

考虑漫反射项. 其中漫反射系数  $k_d$  与反照率  $c$  都是固定的, 计算式可化为

$$L_d(\omega_o) = k_d \frac{c}{\pi} \int_{\Omega} d\omega_i L_i(\omega_i) (n \cdot \omega_i) \triangleq k_d \frac{c}{\pi} I(\omega_o).$$

对于每个  $\omega_o$  预计算积分  $I(\omega_o)$ , 作为纹理存储, 需要时直接从纹理采样即可. 所得结果称为辐照度贴图 (irradiance map). 预计算实际上是一个球面上的二维卷积过程, 可用朴素的均匀采样完成.

考虑漫反射系数  $k_d = 1 - k_s$ . 镜面反射系数  $k_s$  同样可由 Fresnel 公式计算, 但可采取进一步近似, 用法线与出射光的夹角  $n \cdot \omega_o$  替代法线与入射光的夹角  $n \cdot \omega_i$ , 以使系数与入射方向  $\omega_i$  无关. 另外, 注意到粗糙非金属表面的 Fresnel 间接反射有过强的趋势, 需在原本使用的 Schlick 近似中引入粗糙度修正. 修正后的反射系数为

$$k_s^*(\omega_o, r_{\text{mat}}) = F_0 + (\max(1 - r_{\text{mat}}, F_0) - F_0) (1 - (n \cdot \omega_o))^5.$$

漫反射系数  $k_d$  由此可用  $k_d^* = (1 - k_s^*)(1 - m_{\text{mat}})$  作近似, 其中  $m_{\text{mat}}$  为材质金属度参数.

考虑镜面反射项. Unreal Engine 4 提出采用下述近似将原积分式拆分为两个积分的乘积, 称为 split sum approximation. 记  $f(\omega_i, \omega_o) = \frac{DG}{4 (\omega_i \cdot n)(\omega_o \cdot n)}$ , 其中  $G$  项取消此前的粗糙度重映射, 取  $k = \frac{r_{\text{mat}}^2}{2}$ :

$$\begin{aligned} L_s(\omega_o) &\approx \frac{\int_{\Omega} d\omega_i L_i(\omega_i) f(n, \omega_o) (n \cdot \omega_i)}{\int_{\Omega} d\omega_i f(n, \omega_o) (n \cdot \omega_i)} \cdot \int_{\Omega} d\omega_i (k_s f(\omega_i, \omega_o) (n \cdot \omega_i)) \\ &\triangleq R(\omega_o) \cdot C(\omega_o). \end{aligned}$$

$R(\omega_o)$  可借助重要性采样的 quasi-Monte Carlo 方法预计算. 针对若干个  $r_{\text{mat}}$  取值各自计算一张贴图, 保存为一系列 mipmap. Chetan Jags 提出, 在  $r_{\text{mat}}$  较大时采样环境贴图的 mipmap 而不是采样原贴图本身, 可以显著减少预计算结果中的亮点状图案.

对于  $C(\omega_o)$  部分, 将 Schlick 的 Fresnel 近似式代入得

$$\begin{aligned} C(\omega_o) &= \int_{\Omega} d\omega_i \left( F_0 + (1 - F_0) (1 - \omega_o \cdot h)^5 \right) f(\omega_i, \omega_o) \\ &= F_0 \int_{\Omega} d\omega_i f(\omega_i, \omega_o) (1 - (1 - \omega_o \cdot h)^5) (n \cdot \omega_i) + \\ &\quad \int_{\Omega} d\omega_i f(\omega_i, \omega_o) (1 - \omega_o \cdot h)^5 (n \cdot \omega_i) \\ &\triangleq F_0 C_1(n \cdot \omega_o) + C_2(n \cdot \omega_o). \end{aligned}$$

$C_1(n \cdot \omega_o)$  与  $C_2(n \cdot \omega_o)$  也都可预计算得出, 也可采用重要性采样的 quasi-MC 方法. 由于它们还与粗糙度  $r_{\text{mat}}$  有关, 因此需要针对不同的  $n \cdot \omega_o$  和  $r_{\text{mat}}$  各自预计算一张二维的查找表.

结合上述所有近似与预计算结果得 IBL 的渲染时计算式

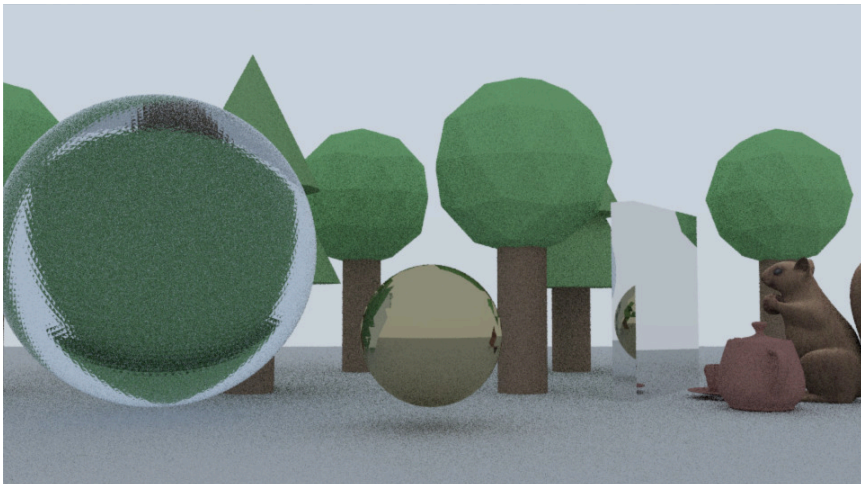
$$L_o(\omega_o) = k_d^* \frac{c}{\pi} I(\omega_o) + R(\omega_o) (F_0 C_1(n \cdot \omega_o) + C_2(n \cdot \omega_o)).$$

### 3 光线追踪渲染器

光线追踪实现了环境光和带阴影测试的 Blinn-Phong 模型拟合镜面光照. 漫反射为 Lambertian 模型, 通过随机化的路径追踪采样实现. 由于拟合会影响物理上的精确性, 为尽可能清晰地展示, 这一部分中使用的示意图关闭了 Blinn-Phong 镜面光照.

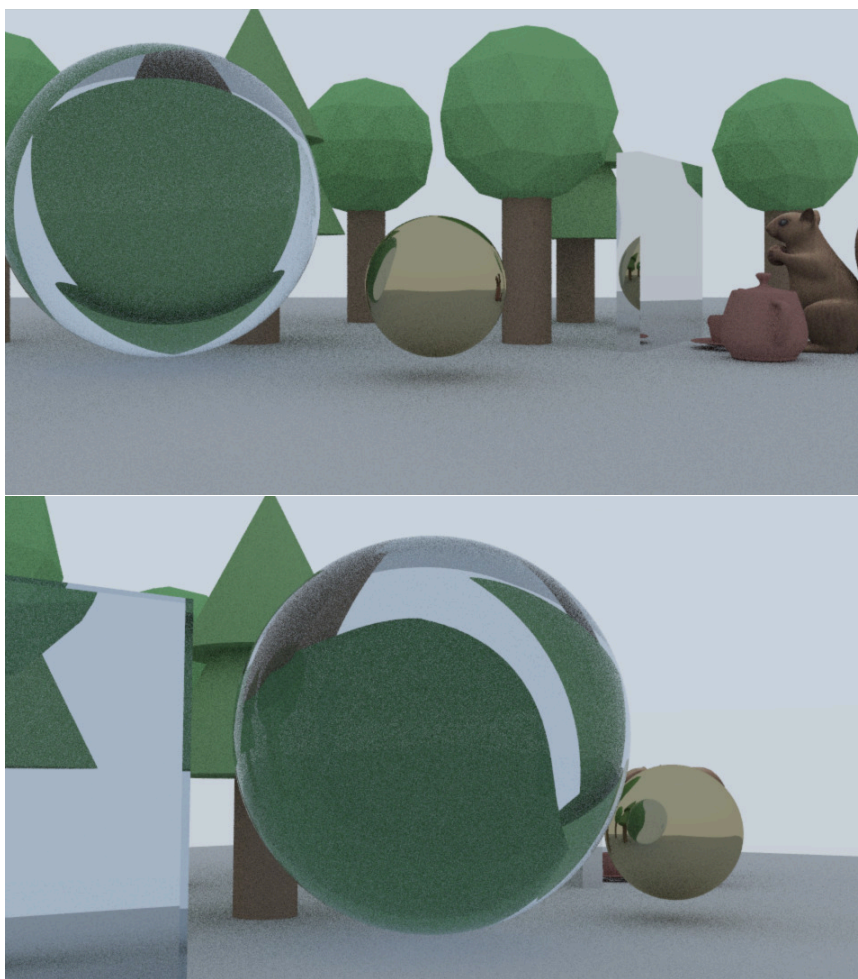
场景中包含大量三角形, 故采用了 Rust 语言的 Cargo 程序包 bvh 在场景创建时构建 BVH, 以排除不必要的射线与三角形求交运算.

场景中包含的球面表示为 icosphere, 即将正二十面体进行若干次细分, 每次将每个正三角形面细分为 4 个更小的正三角形面. 网格近似造成玻璃折射与金属反射效果与真实的球体有偏差, 如下图左侧玻璃球展示出凹凸不平的错觉 (实为三角形面片折射所致), 而中央铁球则清晰显示出三角形面片.

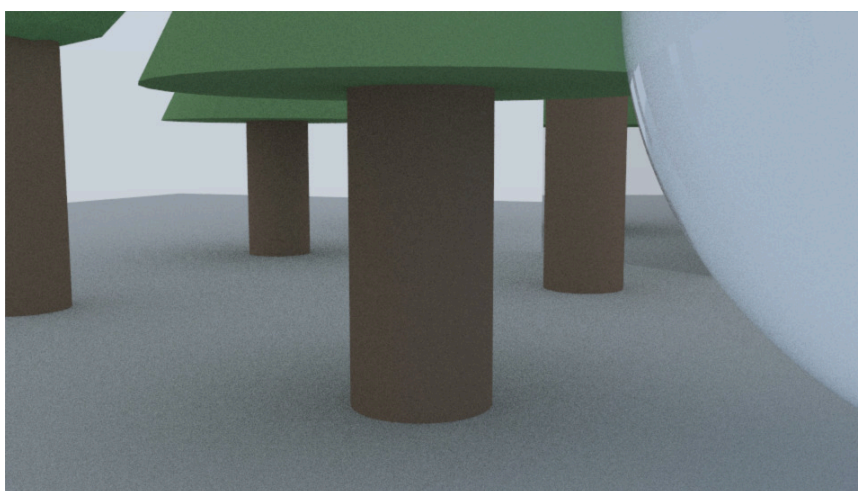




解决方法是使用法线插值。在 Blender 中选择“Shade Smooth”即可自动完成顶点法向量的插值；在计算反射或折射光线时，采用三个顶点的法向量直接以重心坐标加权计算线性组合。



另一个问题是阴影部分的计算速度和收敛速度都较慢。下图是每像素 40 采样得到的结果，可以观察到阴影部分噪点多于直接被环境光照明的部分。



因为漫反射部分没有采用局部光照模型对光照作近似，而是直接执行随机化的路径追踪，阴影部分的光线需要多次反射才能到达光源，而对于 Lambertian 漫反射而言，此过程的随机性相当强，故更难收敛。程序暂未针对此问题作优化。

## 程序细节

### 交互方式

**移动.** 移动鼠标以旋转摄像机视角. 按键盘 W/S 键或上下箭头按键以前后移动, A/D 键或左右箭头按键以左右移动, Q/Z 键以上下漂浮.

**后处理.** 按 Shift 键切换毛玻璃效果.

**PBR/IBL.** 按数字 1/2 键减小或增大金属度参数, 按数字 3/4 键减小或增大粗糙度参数. 按数字 0 键选择只展示 PBR 渲染的小球, 隐藏其余物体; 再次按此键恢复.

**光线追踪.** 按空格键切换光线追踪渲染器.

### 编译方式

使用 Rust 编译器, 参考版本为 rustc 1.58.0-nightly (29b124802 2021-10-25). 程序使用了一些只有 nightly 版本编译器支持的功能, 因此 stable 版本可能无法成功编译.

在所附的源代码目录下执行 `cargo run --release` 即可编译并运行.

### 性能

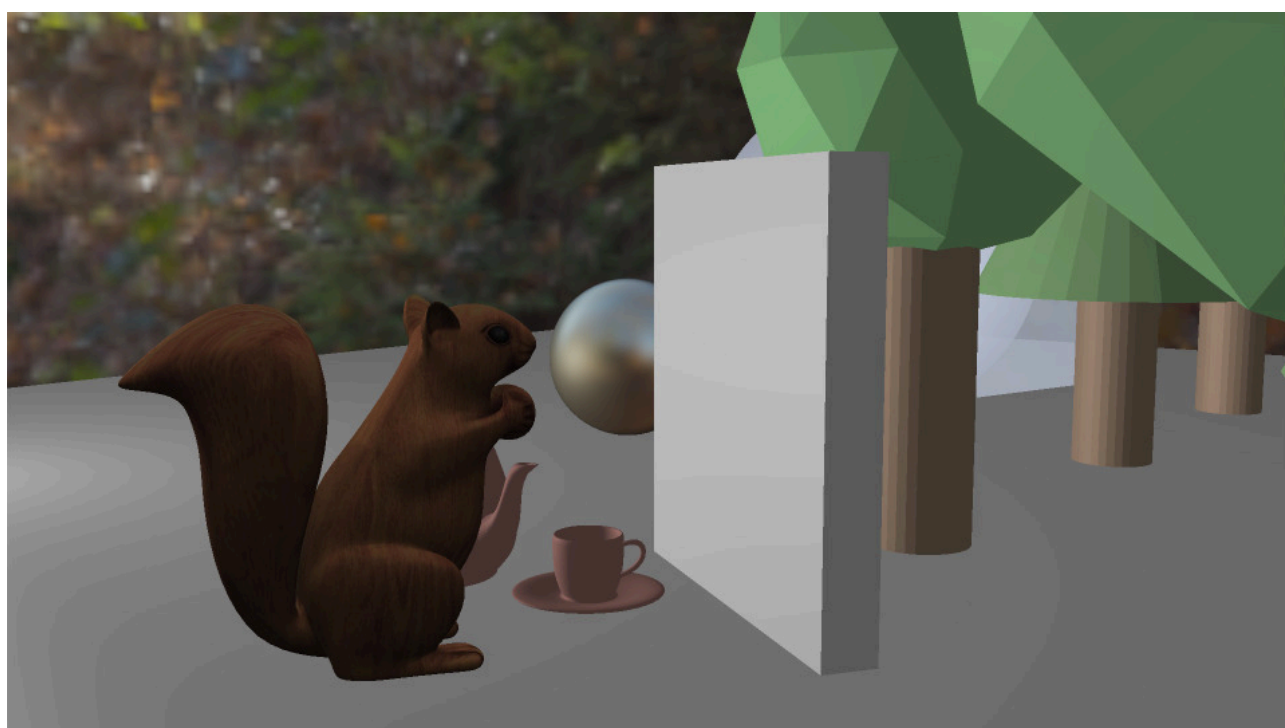
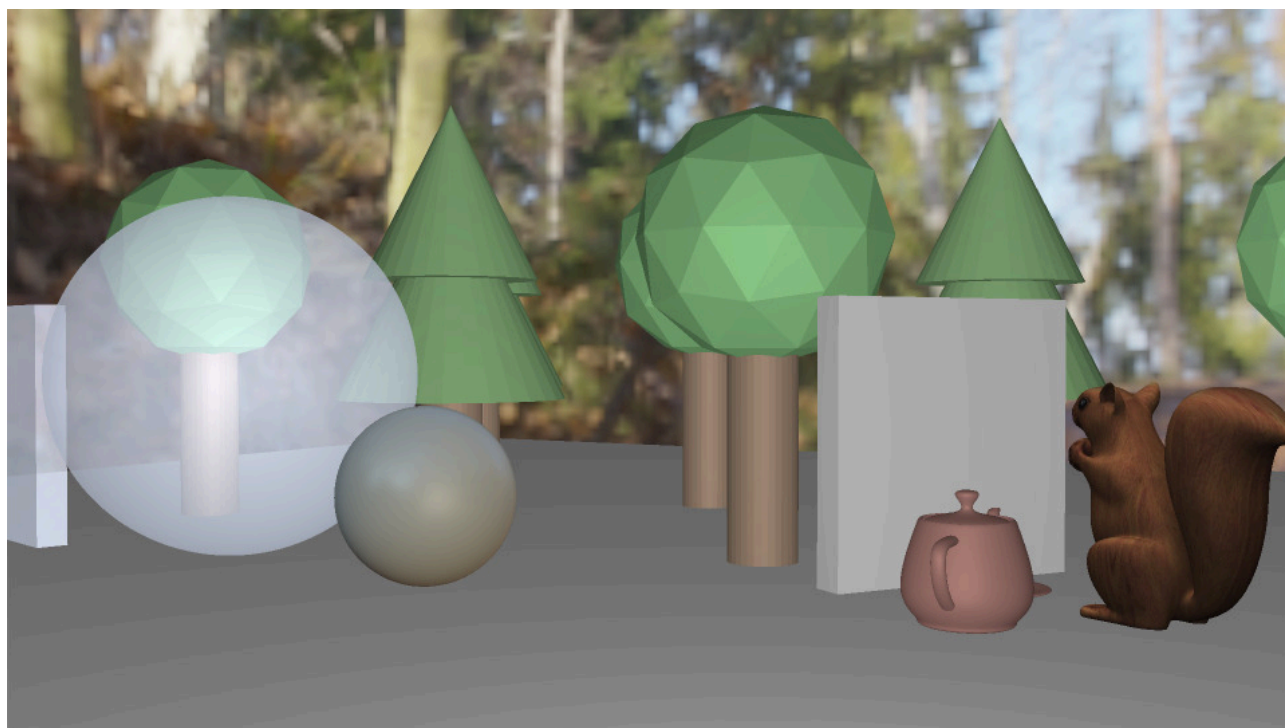
性能测试在 MacBook 上完成, CPU 为 3.1 GHz Intel Core i7, GPU 为 Intel Iris Graphics 6100 1536 MB. 操作系统版本为 macOS 10.14.6.

着色器的测试方式是连续绘制 1000 次场景 (包括 Blinn-Phong 照明和 PBR/IBL 照明的物体, 不包括天空盒), 每次绘制后调用 `glFlush()`, 测量耗时. 光线追踪渲染器测量 1 spp 的耗时. 测试中均按默认视角渲染场景, 每组设置至少测量 20 组数据.

长 × 宽	Shader × 10 <sup>3</sup> (s)	Ray tracing (s)
480 × 270	0.59823 ± 0.02336	0.6987 ± 0.0027
960 × 540	0.70193 ± 0.01674	2.7847 ± 0.0065
1920 × 1080	0.92269 ± 0.02094	11.040 ± 0.035
1920 × 1080	1084 fps	110 s/帧 @ 10 spp

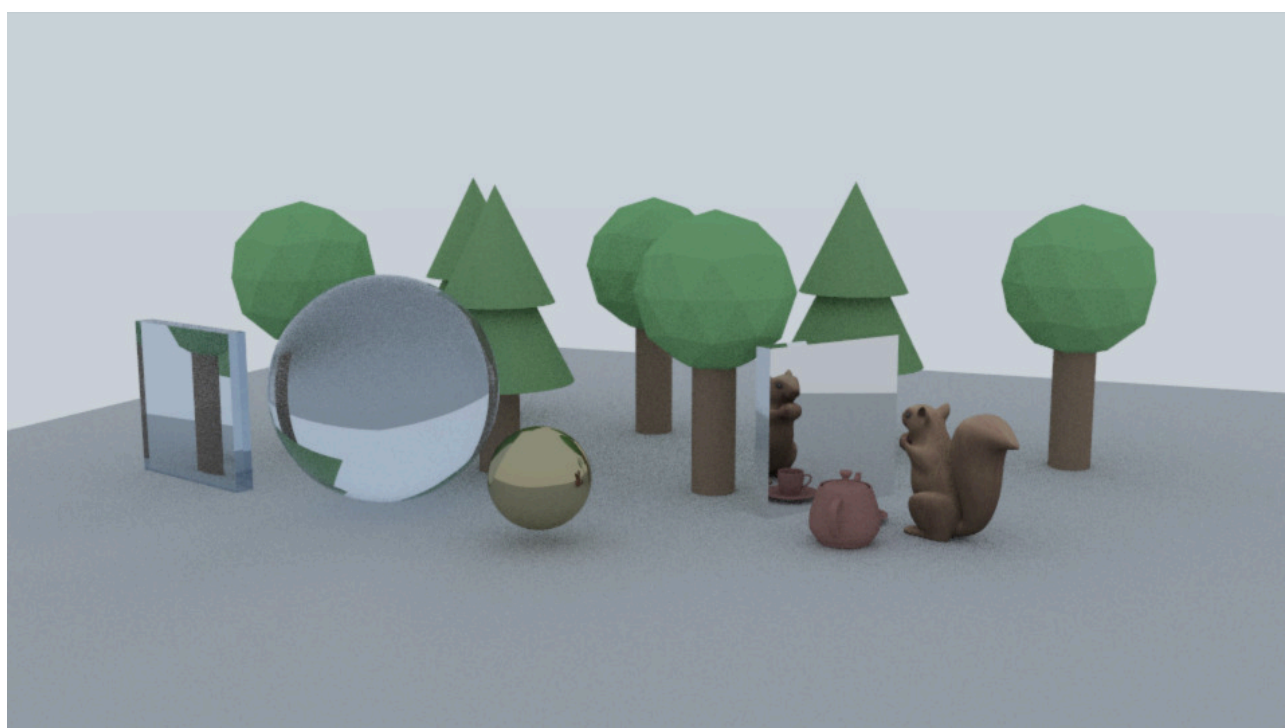
### 渲染效果图

下附渲染器输出的图像. 注意到加入局部光照近似的光线追踪渲染结果中, 球状树冠显示来自点光源的照明. 进一步加强点光源强度可以显示出更清晰的阴影, 但是损失了物理上的准确性, 视觉效果并不好, 故未在此展示.



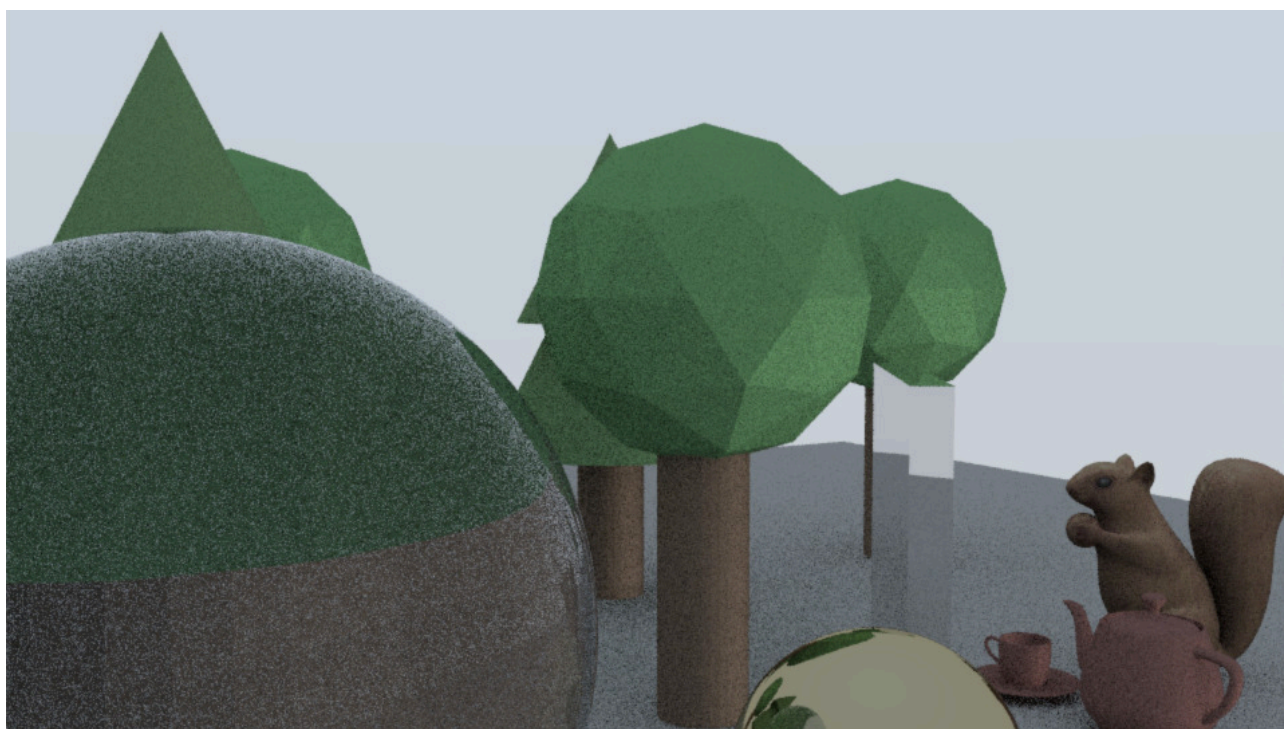
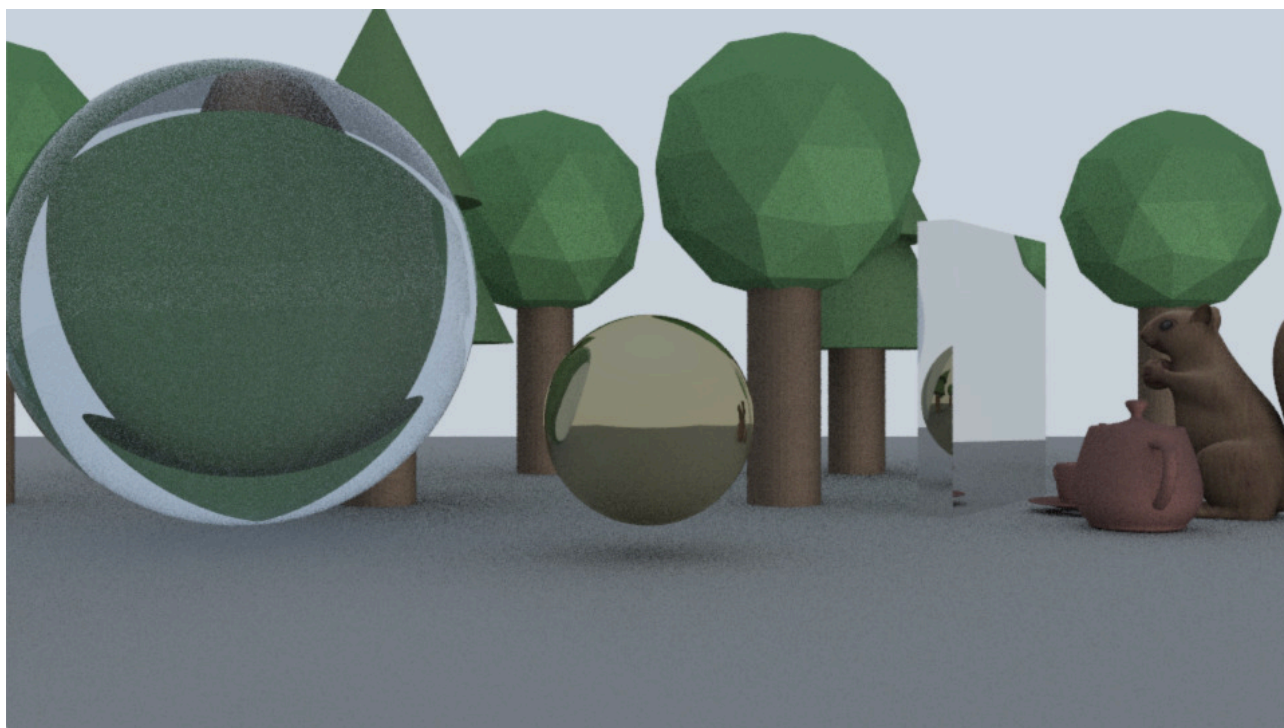
基于 OpenGL 着色器的 Blinn-Phong 局部光照





光线追踪（无局部光照近似），10 spp





光线追踪（有局部光照近似），10 spp（注意树冠颜色与上一页中的不同.）

## 参考资源与文献

- **LearnOpenGL CN: PBR**  
JoeyDeVries, J.moons, KenLee, flyingSnow, Krasjet, <https://learnopengl-cn.github.io/>
- **深入理解 PBR/基于图像照明 (IBL)**  
Ubp.a, <https://zhuanlan.zhihu.com/p/66518450>
- **DX12 渲染管线 (1) - 基于物理的渲染 (PBR)**  
moriya 苏蛙可, <https://zhuanlan.zhihu.com/p/61962884>  
上述三篇详细介绍了 PBR 和 IBL 的基础知识.
- **PBR Diffuse Lighting for GGX+Smith Microsurfaces**  
Earl Hammon, Jr., GDC 2017  
介绍通用 BRDF 公式背后的数学与物理原理.
- **Background: Physics and Math of Shading**  
Naty Hoffman, SIGGRAPH 2013 Course Notes  
(pp. 12-20) 详细解释 BRDF 镜面反射项的物理含义.
- **Real Shading in Unreal Engine 4**  
Brian Karis, ACM SIGGRAPH 2013  
介绍 UE4 选取的 BRDF,  $G$  中  $k = \frac{(r_{\text{mat}} + 1)^2}{8}$  (直接光照) 或  $\frac{r_{\text{mat}}^2}{2}$  (环境光照) 的理由, 以及 split sum approximation 的细节.
- **Geometrical Shadowing of a Random Rough Surface**  
Bruce G. Smith, IEEE Trans. Antennas Propag. 1967  
介绍镜面反射几何项  $G$  的 Smith 模型.
- **An Inexpensive BRDF Model for Physically-based Rendering**  
Christophe Schlick, Computer Graphics Forum 1994  
将 Beckmann NDF 应用于 Smith 模型, 即 UE4 所改进的  $G$  的原型.
- **Ray Tracing in One Weekend**  
Peter Shirley  
(章节 10.5) 介绍玻璃反射与折射中的 Fresnel 现象.