# 光线追踪算法理论

段元兴

2019 年 5 月 9 日

# Content

# 1 摘要

作为Computer Graphics的圣杯—光线追踪,是一个十分古老但是热门的算法. 接下来将简要的介绍实时光线追踪算法的基本实现.

# 2 基本原理

## 2.1 概述

真实世界的几何光学模型是光线从光源发出,经过折射,镜面反射,漫反射等方式进入人眼.而现有计算机图形学的基本渲染方法却是光栅化,即利用空间投影变换将各种基本图形元素(如三角形)投影到屏幕坐标然后上色等等.然而这种做法有先天性的巨大缺陷:在实现折射反射及阴影等效果的时候需要利用各种所谓的技巧才能不太精确的模拟真实情况.

而模拟实际光线的光线追踪算法突破了这些传统图形学的局限性.

## 2.2 正反向光线追踪

一般来说光线追踪算法分两种:

1.正向光线追踪:完全按照真实世界的几何光学模型,模拟的光线从光源发出,经过折射反射等等作用进入人眼从而被看到.而由于一般只有少量从光源发出的光线被人眼看到, 所以这种方法浪费了巨大的计算能力.

2.反向光线追踪:利用几何光线的光路可逆性,将原来的光线的方向改变从人眼出发, 极大的降低了所需要的光线数量.但是在路径追踪里,通常也需要与正向追踪结合来加快寻找光源的速度.

## 2.3 物理模型

光线追踪里最基本的两个模型是光线和物体.人眼投射出光线,而光线在物体表面之间和内部传播,而物体则对其有镜面反射,漫反射,折射和体积吸收等效果.光线和物体的特征主要有几何特征和光学特征.光线的几何特征包括其初始点,方向和与物体碰撞时经过的距离; 而物体的几何特征则包括面元位置和表面法向.光线的光学特征有其颜色和占进入人眼光线的比例,物体的光学特征则包括其镜面反射率,漫反射率,透射率,体积吸收率和折射率.

光线追踪的主要任务有两个:

1.计算光线与物体的碰撞,即找到第一个与光线碰撞的物体.

2.计算光线与物体表面的相互作用,包括计算光线的颜色特征变化和几何特征变化.

对于任务1,最基本的想法是逐一对所有物体进行相交测试,找到距离最近的交点.不过如果合理利用物体之间的空间关系则可以大大加速这个过程,例如假如物体$a$在物体$b$内部,而光线未与$b$相交,则对$a$的相交测试完全可以省去.而我们又可以加入一些假象的物体, 即包围盒,将一些物体包围起来,使得未与包围盒相交的光线不可能与其内的物体相交.还可以对空间进行划分,使得光线背后的物体无需测试.总之有很多可以对测交过程加速的算法.我们这里选取BVH (Bounding volume hierarchy)加速算法

对于任务2,直接利用相关公式计算即可.

## 2.4 在计算机GPU上利用OpenGL实现

目前光线追踪算法的实现大多是在CPU上实现的,自然是可以利用高级编程语言带来的各种方便的语法.但是由于在这里的模型中光线与光线是无关的,故可以利用GPU的强大的并行性来加速.基本的想法是对于投影平面上的每一个像素点生成一根光线然后交给fragment着色器实现对其的追踪,获取的颜色则直接作为该像素点的颜色.追踪的过程包括:

1.从眼睛到投影屏幕之间投射一根光线,寻找第一个交点.若有交点,进行步骤2,否则返回天空盒子的颜色.

2.计算当前光线与物体之间的光学相互作用:

a.如果存在漫反射,计算光源与交点是否有物体遮挡,如果没有则计算光源对交点的光照;

b.如果存在折射,计算折射光线并将其压栈,计算折射对镜面反射率的影响;

c.如果存在镜面反射,计算反射光线并将其设为当前光线.

3.计算当前光线的第一个交点,若有交点,进行步骤2,否则退栈(若未栈空)或结束追踪(栈空).

当然栈深度是有限的,所以在达到栈深的时候即使有折射或反射光线也不能压栈.

所以光线追踪实现主要包括:对栈和递归的模拟(由于glsl不支持递归),对包围盒的遍历,相交测试的计算,光学效果的计算.

考虑到实现实时最大的问题:性能,自然要对shader程序优化(效果可以非常明显,例如从compute shader移植到 fragment shader之后性能提高了160%).当然这样下来我们画出来的图片并不是无偏的,由于在漫反射时仅仅计算了对光源的漫反射而未考虑其他物体的影响.这可以通过分布式蒙特卡洛光线追踪解决,但是考虑到性能和后期降噪处理的复杂度,这里略去.所以物体之间的漫反射和焦散效果等等这里是看不到的.

# 3 理论推导

## 3.1 基本模型的数学描述

由于计算机只认识数,所以我们要将当前的各种物理模型转换成数学语言.这里涉及到基本的矢量计算.

### 3.1.1 颜色模型

虽然自然生活中光的波长是连续分布的,但是在这里采用显示器的描述方式,即用$(R,G,B)$的混合来描述千千万万种个颜色. 而一个物体的光学属性则十分复杂,这里仅仅考虑其表面的自发光颜色$g$(glow),表面的反射颜色$r$(reflect),透射颜色$t$(transmission),对光源的漫反射系数$d$(diffuse)和折射率$n$.所以用类$Color:\{r,t,g,d,n\}$来储存颜色信息. 而这些信息有可能随着在几何图元上的位置的不同而变化,因此有必要建立一个纹理坐标, 使OpenGL能够从纹理数组读取这些信息.

### 3.1.2 光线模型

一根光线由出发点$\overrightarrow{p_0}$,方向$\overrightarrow{n_0}$(已经归一化),颜色构成.为了方便GPU优化计算,这里的 $\overrightarrow{p_0}$加上一个$w=1$分量构成$p^\mu=(\overrightarrow{p_0},1)$.

而颜色一般由$(R,G,B)$三个分量来表示.由于我们用的是反向光线追踪, 所以还需要一个强度因子$(k_R,k_G,k_B)$来表示当前光线占总光线强度的比例.

光线经过的所有点可以由:

$$r^\mu=p_0^\mu+t\overrightarrow{n_0},t>0 \tag{1}$$

来表述.

### 3.1.3 点光源模型

一个点光源由其位置和颜色表示.

### 3.1.4 平面模型

这里平面的定义为一个无穷大的三维中的二维平面,由三维笛卡尔坐标方程来描述:

$$Ax+By+Cz+W=0. \tag{2}$$

为简化描述,我们用

$$n_p^\mu=(\overrightarrow{n_p},W)=(A,B,C,W) \tag{3}$$

来描述这个平面,其中$\overrightarrow{n_p}$也是该平面的法向(保证已经归一化).

### 3.1.5 三角形模型

这里三角形用空间中三个点$\overrightarrow{p_0}, \overrightarrow{c}, \overrightarrow{p_2}$来定义, 并且法向$\overrightarrow{n_p}$由顺序$p_0, c, p_2$按照右手螺旋法则定义为正方向.由这些顶点可以计算出$\overrightarrow{n_p}$(已归一化)和该三角形所在的平面$n_p^\mu$.而在三角形平面内的点可以用仿射坐标$\overrightarrow{t'} = (u', v')$来表示,并且该坐标的单位向量为 $\overrightarrow{e_0} = \overrightarrow{p_0c}, \overrightarrow{e_1} = \overrightarrow{p_0p_2}$. 三角形的三个顶点又具有其在纹理图片上的纹理坐标

$$\begin{cases} \overrightarrow{t_0} = (u_0, v_0) \\ \overrightarrow{t_1} = (u_1, v_1) \\ \overrightarrow{t_2} = (u_2, v_2) \end{cases} \tag{4}$$

从$(u', v')$计算实际纹理坐标$\overrightarrow{t}$可由以下公式得出:

$$\overrightarrow{t} = u'\overrightarrow{t_1} + v'\overrightarrow{t_2} + (1 - u' - v')\overrightarrow{t_0}. \tag{5}$$

### 3.1.6 球形模型

设球心的位矢为$\overrightarrow{c}$,半径为$R$.为方便GPU储存与计算,用$c^\mu = (\overrightarrow{c}, R^2)$来储该球的几何特征. 纹理坐标这么表示:定义一个球的z方向$\overrightarrow{e_2}$(类似于地球的极轴,从地心指向北极点为正方向),指定一个方向$\overrightarrow{e_0}$作为x正方向, 根据右手系建立球坐标系$(r, \theta, \phi)$.而纹理坐标$(u, v)$由$\theta$和$\phi$经过归一化得到:

$$\begin{cases} u = \frac{\phi}{2\pi} \\ v = 1 - \frac{\theta}{\pi} \end{cases} . \tag{6}$$

### 3.1.7 圆盘模型

类似于所有平面图形,圆盘由其所在平面$p^\mu = (\overrightarrow{n_0}, W)$,圆心坐标和圆半径$c^\mu = (\overrightarrow{c}, R^2)$组成.而为了计算纹理坐标,我们还要破坏其轴对称性,即在圆盘平面内架一个$uv$坐标系.因此首先在平面内指定一个方向为$u$坐标正方向$\overrightarrow{e_0}$,然后由法向$\overrightarrow{n_0}$计算出另一个坐标的单位向量$\overrightarrow{e_1}$.设焦点相对圆心的位置为$\overrightarrow{d}$, 则该点的纹理坐标为:

$$\begin{cases} u = \frac{\overrightarrow{d}\cdot\overrightarrow{e_0}}{2R} + \frac{1}{2} \\ v = \frac{\overrightarrow{d}\cdot\overrightarrow{e_1}}{2R} + \frac{1}{2} \end{cases} . \tag{7}$$

### 3.1.8 圆柱模型

这里的圆柱只描述其侧面(底面和顶面都是由圆来描述).圆柱可以由一个圆柱底面圆心坐标$\overrightarrow{c}$,圆柱轴线方向$\overrightarrow{n}$(从底面圆心到顶面圆心为正方向),圆柱半径$R$和圆柱高度$l$这些参数表述.而侧面方程为:

$$|(\overrightarrow{r} - \overrightarrow{c}) \times \overrightarrow{n}| = R, \tag{8}$$

且满足:

$$0 \leqslant (\overrightarrow{r} - \overrightarrow{c}) \cdot \overrightarrow{n} \leqslant l. \tag{9}$$

至于纹理坐标,可以建立以下纹理坐标:以$\overrightarrow{c}$为坐标原点,以$\overrightarrow{n}$为z轴,再指定一个方向$\overrightarrow{e_0}$为 x轴正方向,建立右手柱坐标系$(r, \theta, z)$,$\overrightarrow{e_1}$为y轴.然后纹理坐标为:

$$\begin{cases} u = \frac{\theta}{2\pi} \\ v = \frac{z}{l} \end{cases} . \tag{10}$$

### 3.1.9 圆锥模型

这里的圆锥只描述其侧面(底面由圆来描述).圆锥可以由一个锥尖坐标$\overrightarrow{c}$,圆锥轴线方向$\overrightarrow{n_0}$(从锥尖到底面圆心为正方向),半顶角$\alpha$和母线长度$l$这些参数描述.而侧面方程为:

$$(\overrightarrow{r} - \overrightarrow{c}) \cdot \overrightarrow{n} = \cos\alpha\, |\overrightarrow{r} - \overrightarrow{c}| \tag{11}$$

且满足:

$$0 \leqslant (\overrightarrow{r} - \overrightarrow{c}) \cdot \overrightarrow{n} \leqslant lcos(\alpha). \tag{12}$$

而纹理坐标可以这么建立:以底面圆心为坐标原点,以$-\overrightarrow{n_0}$为z轴,选定一个方向$e_0$作为x轴,按照右手系建立极坐标系$(,\theta,z)$,而纹理坐标为:

$$\begin{cases} u = \frac{\theta}{2\pi} \\ v = \frac{z}{lcos(\alpha)} \end{cases}. \tag{13}$$

#### 3.1.10 包围盒模型

为了加速相交测试,我们采用矩形包围盒模型(类似的还有球形包围盒等等).一个矩形包围盒包括其坐标最小点$\overrightarrow{p_{min}}$和坐标最大点$\overrightarrow{p_{max}}$.

### 3.2 相交测试的推导

如果我们想要求光线与几何图元的相交点,则必须求出方程1中的$t$,相应的纹理坐标也能计算得出.而为了进一步计算反射折射光线,还要计算出该位置图元的法向$\overrightarrow{n_{hit}}$.

#### 3.2.1 平面

方程2即可以写成:

$$r^{\mu} n_0^{\mu} = 0, \tag{14}$$

代入方程1,故

$$(p_0^{\mu} + t\overrightarrow{n_0}) n_0^{\mu} = 0. \tag{15}$$

由此可以解出$t$:

$$t = -\frac{p_0^{\mu} n_p^{\mu}}{\overrightarrow{n_0} \cdot \overrightarrow{n_p}}. \tag{16}$$

法向$\overrightarrow{n_{hit}}$就是$\overrightarrow{n_p}$.

#### 3.2.2 三角形

光线若与一个三角形相交,则必先与该三角形所在的平面相交.由方程16可求得 $t$,由方程1可知位置矢量$r^{\mu}$.而为判断该交点是否在三角形内部,我们首先要计算该焦点在平面内的仿射坐标$(u', v')$.设交点$\overrightarrow{r}$到三角形顶点$c$的位矢为:

$$\overrightarrow{d} = \overrightarrow{r} - \overrightarrow{c} = u'\overrightarrow{e_1} + v'\overrightarrow{e_2}. \tag{17}$$

则:

$$\begin{cases} \overrightarrow{d} \cdot \overrightarrow{e_1} = |\overrightarrow{e_1}|^2 u' + (\overrightarrow{e_1} \cdot \overrightarrow{e_2})\, v' \\ \overrightarrow{d} \cdot \overrightarrow{e_2} = (\overrightarrow{e_1} \cdot \overrightarrow{e_2})\, u' + |\overrightarrow{e_2}|^2 v' \end{cases}, \tag{18}$$

由此可以解出:

$$\begin{cases} u' = \overrightarrow{k_1} \cdot \overrightarrow{d} \\ v' = \overrightarrow{k_2} \cdot \overrightarrow{d} \end{cases}, \tag{19}$$

其中:

$$\begin{cases} \overrightarrow{k_1} = \frac{|\overrightarrow{e_2}|^2 \overrightarrow{e_1} - (\overrightarrow{e_1} \cdot \overrightarrow{e_2}) \overrightarrow{e_2}}{s} \\ \overrightarrow{k_2} = \frac{|\overrightarrow{e_1}|^2 \overrightarrow{e_2} - (\overrightarrow{e_1} \cdot \overrightarrow{e_2}) \overrightarrow{e_1}}{s} \end{cases}, \tag{20}$$

$$s = |\overrightarrow{e_1} \times \overrightarrow{e_2}|^2. \tag{21}$$

而判断交点是否在三角形内则可由以下条件判断:

$$\begin{cases} u' \geqslant 0 \\ v' \geqslant 0 \\ u' + v' \leqslant 1 \end{cases}. \tag{22}$$

由此可以看出在少量的预计算下三角形的相交测试可以非常简单.而这些预计算又可以由GPU以并行化的方式快速完成,达到加速的目的.

法向$\overrightarrow{n_{hit}}$就是$\overrightarrow{n_p}$,纹理坐标$(u,v)$可以由方程5得出.

考虑到由三角形构成的立体物体表面看起来十分的不光滑,一般会采用一种叫法向插值的技术使交点处的法向不是固定为三角形所在平面的法向而是取决于三个顶点法向的插值,即法向插值.故有:

$$\overrightarrow{n_{hit}} = u'\overrightarrow{n_1} + v'\overrightarrow{n_2} + (1-u'-v')\overrightarrow{n_0}, \tag{23}$$

其中$\overrightarrow{n_0}$,$\overrightarrow{n_1}$,$\overrightarrow{n_2}$分别是三个顶点的法向,一般可以由包含该顶点的三角形法向平均计算得到.

### 3.2.3 球形

光线若与一个球相交,则该直线到球心的距离小于$R$.而这个距离平方可以表示为 $|(\overrightarrow{c} - \overrightarrow{p_0}) \times \overrightarrow{n_0}|^2$. 令$\overrightarrow{d} = \overrightarrow{c} - \overrightarrow{p_0}$, 则有:

$$s^2 = R^2 - \left|\overrightarrow{d} \times \overrightarrow{n_0}\right|^2 \geqslant 0. \tag{24}$$

球的方程为:

$$\left|\overrightarrow{r} - \overrightarrow{c}\right|^2 = R^2, \tag{25}$$

代入方程1可得:

$$\left|\overrightarrow{n_0}t - \overrightarrow{d}\right|^2 = R^2, \tag{26}$$

由此可以解出:

$$\begin{cases} t_1 = \overrightarrow{n_0} \cdot \overrightarrow{d} + \sqrt{R^2 - \left(\left|\overrightarrow{d}\right|^2 - \left(\overrightarrow{n_0} \cdot \overrightarrow{d}\right)^2\right)} \\ t_2 = \overrightarrow{n_0} \cdot \overrightarrow{d} - \sqrt{R^2 - \left(\left|\overrightarrow{d}\right|^2 - \left(\overrightarrow{n_0} \cdot \overrightarrow{d}\right)^2\right)} \end{cases}. \tag{27}$$

由于$\left|\overrightarrow{d} \times \overrightarrow{n_0}\right|^2 = \left|\overrightarrow{d}\right|^2 - \left(\overrightarrow{n_0} \cdot \overrightarrow{d}\right)^2$, 故:

$$\begin{cases} t_1 = \overrightarrow{n_0} \cdot \overrightarrow{d} + s \\ t_2 = \overrightarrow{n_0} \cdot \overrightarrow{d} - s \end{cases}. \tag{28}$$

当然只有其中一个解是最近的相交点,于是最小的正解即为我们所求的$t$. 而法向:

$$\overrightarrow{n_{hit}} = \frac{\overrightarrow{r} - \overrightarrow{c}}{R}. \tag{29}$$

计算$\overrightarrow{n_{hit}}$在$xOy$平面上的投影:

$$\cos\theta = \overrightarrow{n_{hit}} \cdot \overrightarrow{e_2}; \tag{30}$$

$$\overrightarrow{n_{xy}} = normalize\left(\overrightarrow{n_{hit}} - \overrightarrow{e_2}\cos\theta\right), \tag{31}$$

则:

$$\theta = \arccos\left(\overrightarrow{n_{hit}} \cdot \overrightarrow{e_2}\right); \tag{32}$$

$$\phi = \begin{cases} \arccos(\overrightarrow{n_{xy}} \cdot \overrightarrow{e_0}), & \overrightarrow{n_{xy}} \cdot \overrightarrow{e_1} \geqslant 0 \\ 2\pi - \arccos(\overrightarrow{n_{xy}} \cdot \overrightarrow{e_0}), & \overrightarrow{n_{xy}} \cdot \overrightarrow{e_1} < 0 \end{cases}. \tag{33}$$

纹理坐标则可以用6计算得到.

### 3.2.4 圆盘

同三角形,首先计算出相交位置$r^{\mu}$.然后判断是否在圆盘内:

$$\overrightarrow{d} = \overrightarrow{r} - \overrightarrow{c},\ \left|\overrightarrow{d}\right|^2 < R^2. \tag{34}$$

纹理坐标按照7计算即可,法向$\overrightarrow{n_{hit}}$即$\overrightarrow{n_p}$.

### 3.2.5 圆柱

将1代入8并展开可得:

$$\left[1 - (\overrightarrow{n_0} \cdot \overrightarrow{n})\right] t^2 + 2\left(\overrightarrow{n_0} \cdot \overrightarrow{j}\right) t + \overrightarrow{d} \cdot \overrightarrow{j} - R^2 = 0, \tag{35}$$

其中:

$$\overrightarrow{j} = \overrightarrow{d} - \left(\overrightarrow{n} \cdot \overrightarrow{d}\right) \overrightarrow{n}. \tag{36}$$

令$a = 1 - (\overrightarrow{n_0} \cdot \overrightarrow{n})$, $b = \overrightarrow{n_0} \cdot \overrightarrow{j}$, $c = R^2 - \overrightarrow{d} \cdot \overrightarrow{j}$; 由此可以解得:

$$\begin{cases} t_1 = \frac{-b-\sqrt{b^2+ac}}{a} \\ t_2 = \frac{-b+\sqrt{b^2+ac}}{a} \end{cases}. \tag{37}$$

考虑到约束9并代入$\overrightarrow{r}$可得:

$$z = \overrightarrow{d} \cdot \overrightarrow{n} + (\overrightarrow{n_0} \cdot \overrightarrow{n})\, t; \tag{38}$$

$$0 \leqslant z \leqslant l. \tag{39}$$

于是最小的满足约束条件的正解即为我们所求的$t$. 法向$\overrightarrow{n_{hit}} = \frac{\overrightarrow{d}+\overrightarrow{n_0}t-\overrightarrow{n}z}{R}$; 而纹理坐标则可以由10得到,其中

$$\theta = \begin{cases} \arccos(\overrightarrow{n_{hit}} \cdot \overrightarrow{e_0}), & \overrightarrow{n_{hit}} \cdot \overrightarrow{e_1} \geqslant 0 \\ 2\pi - \arccos(\overrightarrow{n_{hit}} \cdot \overrightarrow{e_0}), & \overrightarrow{n_{hit}} \cdot \overrightarrow{e_1} < 0 \end{cases}. \tag{40}$$

### 3.2.6 圆锥

将1代入11并展开可得:

$$\left[(\overrightarrow{n_0} \cdot \overrightarrow{n}) - \cos^2\alpha\right] t^2 + 2\left(\overrightarrow{j} \cdot \overrightarrow{n_0}\right) t + \overrightarrow{j} \cdot \overrightarrow{d} = 0; \tag{41}$$

其中:

$$\overrightarrow{j} = \left(\overrightarrow{d} \cdot \overrightarrow{n}\right) \overrightarrow{n} - \overrightarrow{d} \cos^2\alpha. \tag{42}$$

令$a = (\overrightarrow{n_0} \cdot \overrightarrow{n}) - \cos^2\alpha$, $b = \overrightarrow{j} \cdot \overrightarrow{n_0}$, $c = \overrightarrow{j} \cdot \overrightarrow{d}$; 则解为:

$$\begin{cases} t_1 = \frac{-b-\sqrt{b^2-ac}}{a} \\ t_2 = \frac{-b+\sqrt{b^2-ac}}{a} \end{cases}. \tag{43}$$

令$\overrightarrow{r_d} = \overrightarrow{r} - \overrightarrow{c}$,则:

$$\overrightarrow{n_{hit}} = \frac{\frac{\overrightarrow{r_d}}{|\overrightarrow{r_d}|}\cos\alpha - \overrightarrow{n}}{\sqrt{1-\cos^2\alpha}}. \tag{44}$$

而$\overrightarrow{n_{hit}}$在$xOy$平面上的投影

$$\overrightarrow{n_{xy}} = \frac{\frac{\overrightarrow{r_d}}{|\overrightarrow{r_d}|} - \overrightarrow{n}\cos\alpha}{\sqrt{1-\cos^2\alpha}}. \tag{45}$$

故:

$$\theta = \begin{cases} \arccos(\overrightarrow{n_{xy}} \cdot \overrightarrow{e_0}), & \overrightarrow{n_{xy}} \cdot \overrightarrow{e_1} \geqslant 0 \\ 2\pi - \arccos(\overrightarrow{n_{xy}} \cdot \overrightarrow{e_0}), & \overrightarrow{n_{xy}} \cdot \overrightarrow{e_1} < 0 \end{cases}; \tag{46}$$

$$v = 1 - \frac{|\overrightarrow{r_d}|}{l}. \tag{47}$$

#### 3.2.7 包围盒

给定一个线段$L = \{\overrightarrow{r} = \overrightarrow{p_0} + \overrightarrow{n_0}t | t \in (0, t_0]\}$, (当$t_0 < 0$时视为$t_0 = +\infty$原光线,即一条射线).光线与包围盒的相交的定义为$Box \cap L \neq \varnothing$.可以用如下方法判断:

1.若$\overrightarrow{p_0}$在包围盒内,则一定相交.即:

$$\begin{cases} min_x \leqslant p_{0_x} \leqslant max_x \\ min_y \leqslant p_{0_y} \leqslant max_y \\ min_z \leqslant p_{0_z} \leqslant max_z \end{cases} . \tag{48}$$

2.计算光线与包围盒的各个平面的六个交点获得三个相交区间: $(t_{x0}, t_{x1})$, $(t_{y_0}, t_{y_1})$, $(t_{z0}, t_{z1})$.

3.求这三个相交区间的交集$(t_{min}, t_{max})$.若交集非空且$t_{max} > 0$, 则当$t_{min} < t_0$时相交.

### 3.3 光线效果的模拟

为了达到真实的光学效果,需要计算光线传播过程中的镜面反射,与光源的漫反射和折射组成. 而且对于玻璃材质,需要考虑光学的菲涅尔公式才能真实模拟光线的透射率和反射率.设光线与某个几何图元相交,得到以下参数:

1.几何图元的法向$\overrightarrow{n_{hit}}$

2.光线入射方向$\overrightarrow{n_0}$

3.光线传播的距离$t$

4.几何图元的折射率$n$

5.几何图元的颜色信息$c$

之后需要据此计算出该光线的颜色$c$,其中包括$c_r$(镜面反射),$c_t$(透射), $c_d$(点光源漫反射)和$c_g$(面元自发光).得到颜色后考虑到光线在某些介质中传播时会发生损耗,故还要计算这个衰减.

#### 3.3.1 镜面反射

考虑法向$\overrightarrow{n_0}$的变化:

$$\overrightarrow{n_0}' = \overrightarrow{n_0} - 2\left(\overrightarrow{n_{hit}} \cdot \overrightarrow{n_0}\right)\overrightarrow{n_{hit}}. \tag{49}$$

而起始点为:

$$\overrightarrow{p_0}' = \overrightarrow{p_0} + \overrightarrow{n_0}\left(t - \delta\right). \tag{50}$$

为保证继续进行相交测试时不会重新与该物体在该点相交,故实际上要将$t$稍微减少$\delta$. 对于光学效果的模拟,要考虑两种情况:

1.没有折射,即无需计算菲涅尔效应带来的修正.

2.有折射,则需要计算该修正.

而修正为:

$$r' = \frac{r}{2}\left[\left(\frac{\cos i_1 - n\cos i_0}{n\cos i_0 + \cos i_1}\right)^2 + \left(\frac{n\cos i_1 - \cos i_0}{\cos i_0 + n\cos i_1}\right)^2\right]. \tag{51}$$

注意到当$i_1$和$i_0$都趋于0时,数值计算可能不精确,于是求极限:

$$\lim_{i_0, i_1 \to 0} \frac{1}{2}\left[\left(\frac{\cos i_1 - n\cos i_0}{n\cos i_0 + \cos i_1}\right)^2 + \left(\frac{n\cos i_1 - \cos i_0}{\cos i_0 + n\cos i_1}\right)^2\right] = \left(\frac{n-1}{n+1}\right)^2. \tag{52}$$

而我们所需要的$c_r$则可以由拟递归获取的反射光线颜色$c'$得到:

$$c_r = r'c'. \tag{53}$$

### 3.3.2 折射

由折射定律

$$\sin i_0 = n \sin i_1 \tag{54}$$

可以求出$i_1$.这里需要考虑$\overrightarrow{n_{hit}}$与$\overrightarrow{n_0}$ 的方向一致性.经过计算得到:

1.$\overrightarrow{n_{hit}} \cdot \overrightarrow{n_0} < 0$:

$$\overrightarrow{n_0}' = \frac{\overrightarrow{n_0} - (n \cos i_1 + \cos i_0) \overrightarrow{n_{hit}}}{n} \tag{55}$$

2.$\overrightarrow{n_{hit}} \cdot \overrightarrow{n_0} > 0$:

$$\overrightarrow{n_0}' = \frac{\overrightarrow{n_0} + (n \cos i_1 - \cos i_0) \overrightarrow{n_{hit}}}{n} \tag{56}$$

于是综合得到:

$$\overrightarrow{n_0}' = \frac{\overrightarrow{n_0} + [sign\left(\overrightarrow{n_{hit}} \cdot \overrightarrow{n_0}\right) n \cos i_1 - \cos i_0] \overrightarrow{n_{hit}}}{n}. \tag{57}$$

而起始点为:

$$\overrightarrow{p_0}' = \overrightarrow{p_0} + \overrightarrow{n_0}\left(t + \delta\right). \tag{58}$$

为保证继续进行相交测试时不会重新与该物体在该点相交,故实际上要将$t$稍微增加$\delta$.

计算菲涅尔透射率:

$$t' = 2nt \cos i_0 \cos i_1 \left[\frac{1}{\left(n \cos i_0 + \cos i_1\right)^2} + \frac{1}{\left(\cos i_0 + n \cos i_1\right)^2}\right]. \tag{59}$$

同理,在$i_1$和$i_0$都趋于0时,计算极限:

$$\lim_{i_0, i_1 \to 0} 2nt \cos i_0 \cos i_1 \left[\frac{1}{\left(n \cos i_0 + \cos i_1\right)^2} + \frac{1}{\left(\cos i_0 + n \cos i_1\right)^2}\right] = n\left(\frac{2}{1+n}\right)^2. \tag{60}$$

而我们所需要的$c_t$则可以由拟递归获取的折射光线颜色$c'$得到:

$$c_t = t'c'. \tag{61}$$

### 3.3.3 点光源漫反射

这里采用简化的模型:若交点与点光源之间没有其他图元之间遮挡,则认为可以被光源照到.因此, 首先计算出相交位置 $\overrightarrow{p_0}' = \overrightarrow{p_0} + \overrightarrow{n_0}\left(t - \delta\right)$ ($t$减少$\delta$是为防止面元后侧的光源干扰)然后对所有光源逐一判断是否有图元遮挡.若对某个点光源没有遮挡关系,则在该面源处对该点光源可视,于是按点光源光强衰减和照朗伯体模型计算光照强度:

$$c_d = c_l \frac{\overrightarrow{r} \cdot \overrightarrow{n_h it}}{\left|\overrightarrow{r}\right|^3}. \tag{62}$$

其中$c_l$是点光源发光强度,$\overrightarrow{r}$是从面源到点光源的位矢.

### 3.3.4 自发光

这个属于物体固有属性,直接读取就行.一般发光体的自发光强与角度无关.

### 3.3.5 衰减

在透明物体中传播的光线会发生衰减现象.为更加真实的模拟,采用了指数衰减模型:

$$I = I_0 e^{-\alpha t}. \tag{63}$$

其中$\alpha$为衰减系数.

但是对衰减系数的获取是存在一定困难的,因为我们没有引入体积光模型,所以所有的光学信息都要从物体的表面

获取.这导致我们无法直接获得相机位置处的衰减系数.

一种解决办法是进行预计算,即从相机处发出一根光线,一直沿直线传播,将沿途的衰减系数相加减(用光线与法线的夹角判断是进入还是穿出物体)以获得最终的相机点的衰减系数.一般由于精度问题,这样计算出来的衰减系数可能会随着在空间中移动而突变.所以还需要随机发出多根光线来计算然后取最多的那个结果作为实际的衰减系数.

#### 3.3.6 杂项

考虑到在计算菲涅尔公式的时候计算的是光功率密度,而这个值与相机所在位置的折射率有关.但是真实情况是眼角膜所在的折射率是固定的(显然眼角膜不是直接暴露在空气中或水中),所以要对这个进行修正,这又要求获取当前相机位置处的折射率.同理可以由上面的方法解决.