

0.1 算法部分

描述一个完整的基于物理的渲染（PBR），一般意义上所使用的反射率方程为：

$$L_o(p, \omega_o) = \int_{\Omega} (k_d \frac{c}{\pi} + k_s \frac{DFG}{4(\omega_o \cdot n)(\omega_i \cdot n)}) L_i(p, \omega_i) n \cdot \omega_i d\omega_i$$

其中 p 为要着色的点的坐标， ω_o 为观察方向， ω_i 为入射方向向量

0.1.1 Unity中的BRDF积分

参考Unity中的PBR，BRDF所使用的DFG分别为：

D 使用TrowbridgeReitz GGX方法：

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

其中 h 表示用来与平面上为平面作比较用的中间向量， α 表示表面粗糙度。

F 使用Fresnel - Schlick方法：

$$F = F_0 + (1 - F_0)(1 - (n \cdot \omega_o)^5)$$

其中 F_0 为平面的基础反射率。

G 参考Eric Heitz的方法：

$$G = \frac{1}{2(\lambda_o + \lambda_i)} \begin{cases} \lambda_o = n \cdot \omega_i \sqrt{(1 - \alpha^2)(n \cdot \omega_o)^2 + \alpha^2} \\ \lambda_i = n \cdot \omega_o \sqrt{(1 - \alpha^2)(n \cdot \omega_i)^2 + \alpha^2} \end{cases}$$

我们知道在对场景应用PBR时，场景中的所有点都需要执行一遍该计算过程，而为了实现更符合实际的效果，所使用的BRDF可能会更为复杂，则在计算BRDF积分时就会产生较大的效率消耗。

0.1.2 使用LTC近似BRDF积分

为了简化BRDF积分的计算，Eric Heitz等人提出了使用线型变换余弦分布(Linearly Transformed Cosines，简称LTC) 拟合BRDF积分的方法，即通过对原始球面余弦分布应用不同的变换矩阵拟合出不同的球面分布。该方法可以对球面或半球面上任意多边形进行分析整合，快速计算出多边形在球面或

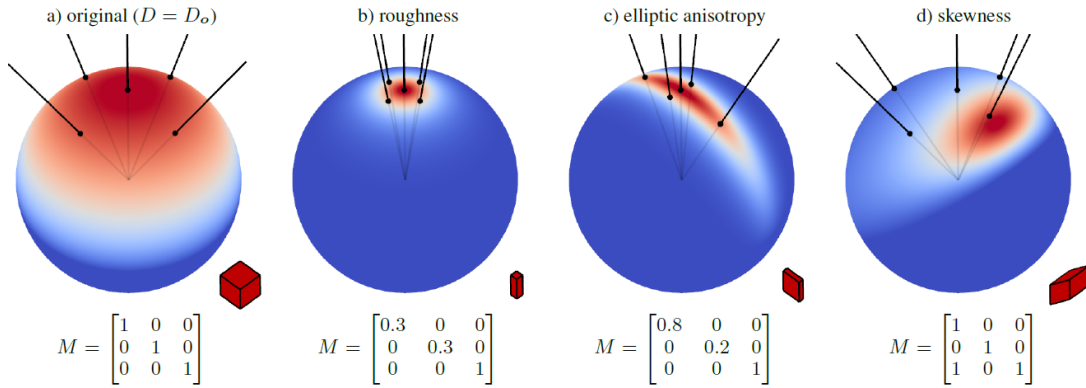


Figure 1: 使用LTC参数化的球面分布。通过对基本余弦分布应用不同的矩阵拟合新的分布情况：(a)原始球面余弦分布；(b)粗糙度；(c)各向异性；(d)偏斜方向。

半球面上的积分结果。通过对原始球面余弦分布的方向向量应用一个变换矩阵 (3×3)，使得变换后的球面分布近似于我们需要的球面分布，该矩阵的参数会因粗糙度、各向异性、偏斜方向的不同而不同，如Fig-1 所示。

使用LTC拟合BRDF，可以看作对球面余弦分布加上一个权重来拟合光线方向上BRDF球形函数：

$$D \approx \rho(\omega_v, \omega_i) \cos \theta_i$$

对于一个各向异性材料，BRDF取决于入射方向 $\omega_v = (\sin \theta_v, 0, \cos \theta_v)$ 和粗糙度 α 。因此对于任意的 (θ_v, α) 组合，使用LTC拟合出BRDF，得到最适合的一个矩阵 M ， M 可以表示为：

$$M = \begin{bmatrix} a & 0 & b \\ 0 & c & 0 \\ d & 0 & 1 \end{bmatrix} \quad (1)$$

因此只需要4个参数，Fig-2中展示了部分组合下的拟合效果。针对不同的入射角度 θ_v 和粗糙度 α ，预计算一组与GGX微平面的BRDF拟合良好的线性变换余弦值，拟合的结果分布中其实已经隐含了各向异性和偏斜两个属性。

由于线性变换不会改变积分结果，如Fig-3，因此有了这个矩阵 M ，计算多边形 P 在球面上的分布 D 的积分就相当于计算多边形 $P_o = M^{-1}P$ 在原始球面分布 D_o 的积分：

$$\int_P D(\omega) d\omega = \int_{P_o} D_o(\omega_o) d\omega_o$$

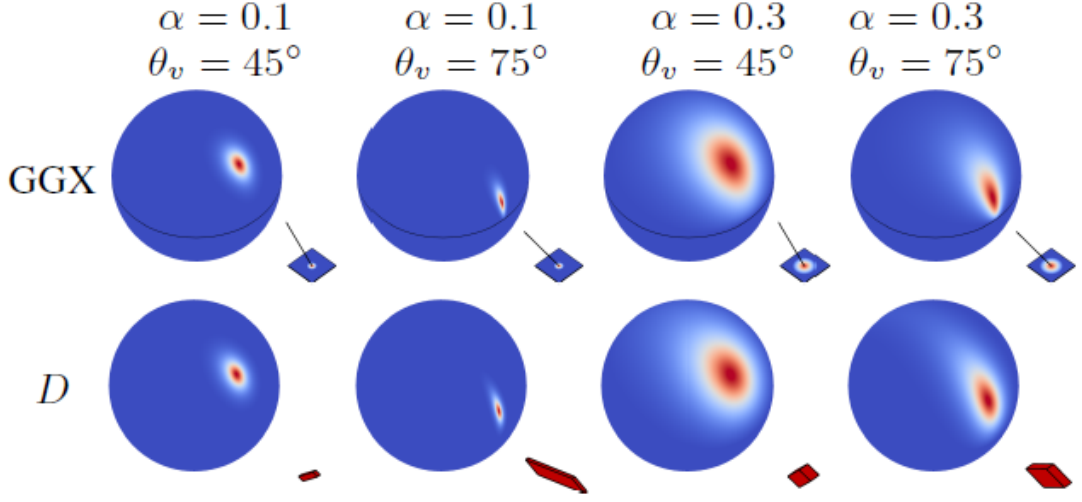


Figure 2: 使用LTC拟合BRDF，图中分别为不同的 (θ_v, α) 组合下拟合的效果

$$D(\omega) = D_o(\omega_o) \frac{\partial \omega_o}{\partial \omega} = D_o\left(\frac{M^{-1}\omega}{\|M^{-1}\omega\|}\right) \frac{|M^{-1}|}{\|M^{-1}\omega\|^3}$$

其中 $\frac{\partial \omega_o}{\partial \omega} = \frac{|M^{-1}|}{\|M^{-1}\omega\|^3}$ 为单位化向量 $\omega = M\omega_o / \|M\omega_o\|$ 变换的Jacobian行列式。

原始球面分布 D_o 可以看作 z 分量的一个截断:

$$D_o(\omega_o = (x, y, z)) = \frac{\max(0, z)}{\pi}$$

作者将预计算的这组矩阵的4个参数存储到分辨率为64x64的二维浮点数纹理中，因为上面用到的都是 M^{-1} ，因此这里存储的也是逆矩阵形式。其中使用的 $\theta_v \in [0, \frac{\pi}{2}]$ ， $\sqrt{\alpha} \in [0, 1]$ ，并利用线性插值获取。

多边形光源下对场景中某一点着色相当于计算多边形在该点的球形区域内的辐照度积分值：

$$I = \int_P L(\omega_i) \rho(\omega_v, \omega_i) \cos \theta_i d\omega_i \approx \int_P L(\omega_i) D(\omega_i) d\omega_i$$

由于该计算过程是基于多边形的各个顶点，因此该方法对于点光源的场景也应是适用的。而对于单个点光源，由于点光源的颜色是一个单一的值，因此 $L(\omega_i) = L$ 可以看作一个常量，有

$$I = \int_P L(\omega_i) D(\omega_i) d\omega_i = L \int_P D(\omega_i) d\omega_i = L \int_{P_o} D_o(\omega_o) d\omega_o$$

其中 P 为点光源的位置, $P_o = M^{-1}P$ 为经过矩阵变换后的位置。

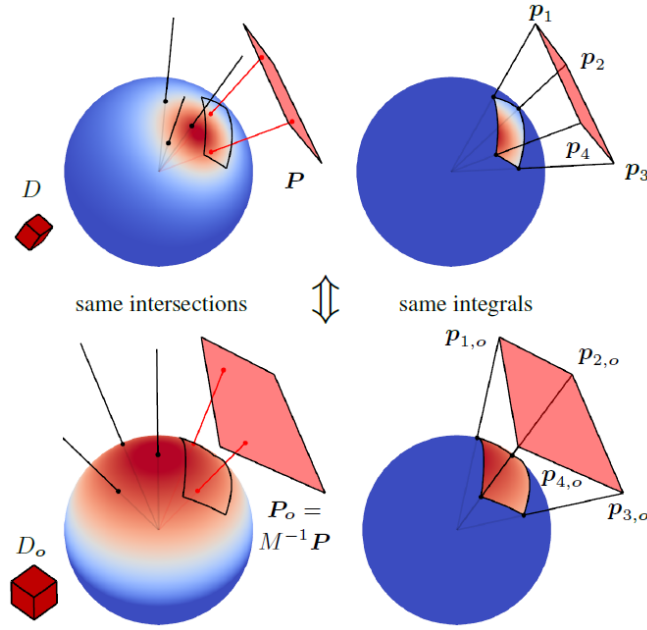


Figure 3: 多边形在球面积分的等价替换，图下方的多边形是由图上方的多边形乘上矩阵 M^{-1} ,由于线性变换不会改变直线与多边形的交点个数，因此在分布中生成的样本在图上方和图下方两种情况下具有相同的相交点个数，因此两种情况下的球面多边形积分是相同的。

0.1.3 LTC在计算单个点光源场景的应用

对于只有一个点光源的场景中，为场景中某一点着色时，漫反射部分仍然是：

$$I_{diffuse} = k_d \frac{c}{\pi} (\omega_i \cdot n)$$

但对于镜面反射部分会有所不同：

$$I_{specular} = k_s \frac{\max(0, (M^{-1}\omega_i) \cdot z)}{\|M^{-1}\omega_i\|} \frac{|M^{-1}\omega_i|}{\|M^{-1}\omega_i\|^3}$$

有

$$I = L \int_{P_o} (I_{diffuse} + I_{specular})$$