

基于物理的光照 (PBR) 在场景渲染方面的应用

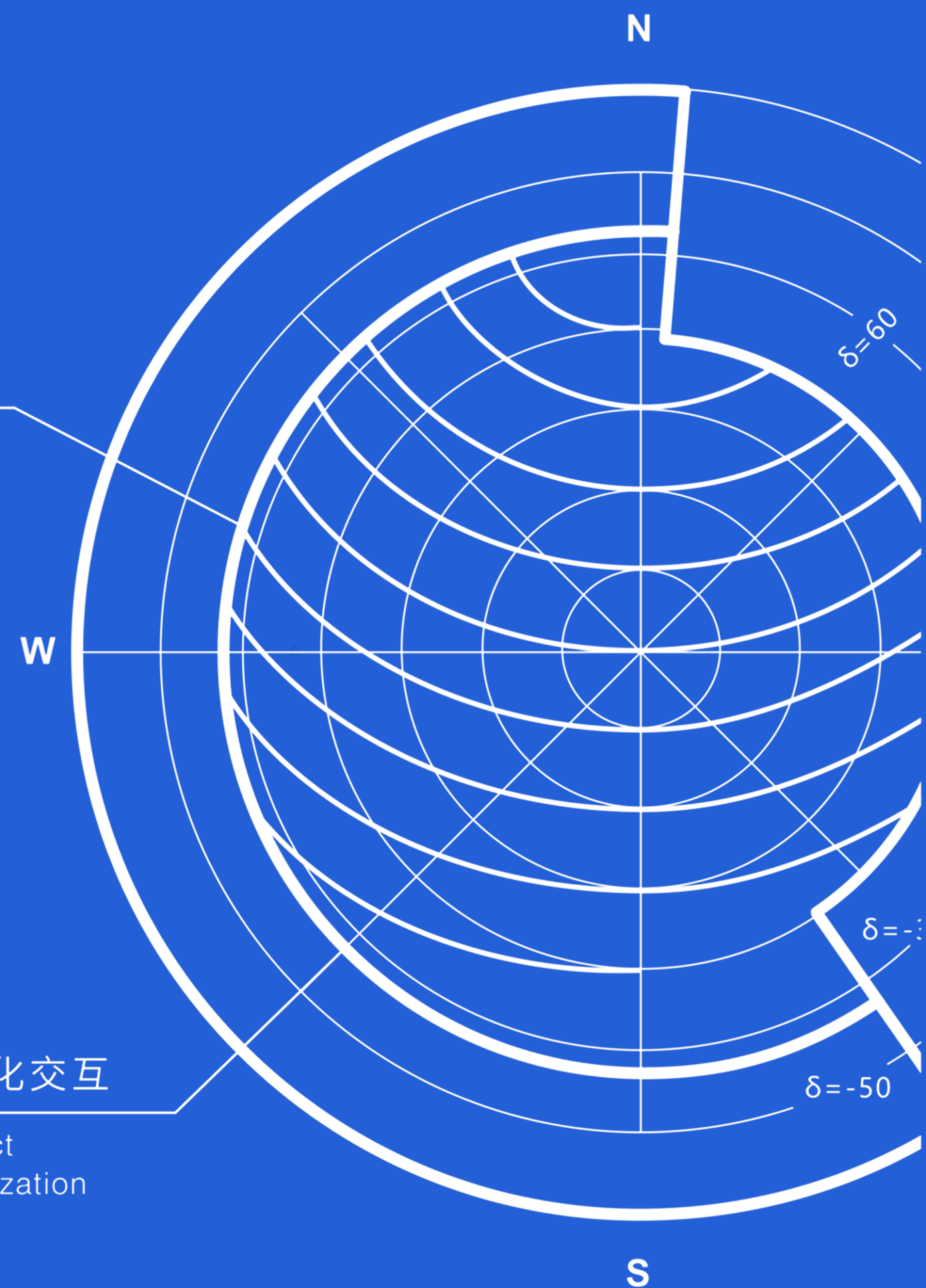
杨石兴(FreeSouth)

GIS 时空大数据

Spatiotemporal
Data For GIS

可视化交互

Interact
Visualization



目录

01



传统光照

02



物理光照

03



原理与实现

04



持续阅读

传统光照

Phong Lighting Model 冯氏光照模型

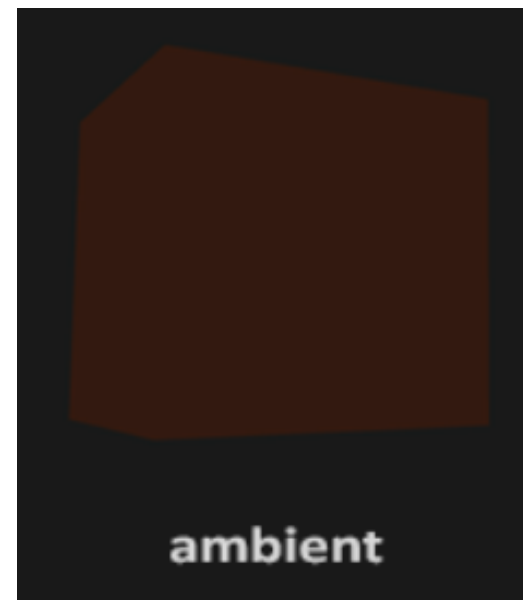


```
vec3 result = (ambient + diffuse + specular) * objectColor;  
vec3 result = (ka*ambient + kd*diffuse + ks*specular) * objectColor;
```

注意：颜色相乘是的结果的意义是正片叠加，以一个颜色为玻璃，拿另一个颜色照其上

传统光照

Phong Lighting Model 冯氏光照模型



环境光：没有什么物体是看不见的，哪怕在黑夜里，我们借助月光、星光仍然可以看见物体。环境光是指这种全局的无处不在的光源。

```
float ambientStrength = 0.1;  
vec3 ambient = ambientStrength * lightColor;  
vec3 result = ambient * objectColor;
```

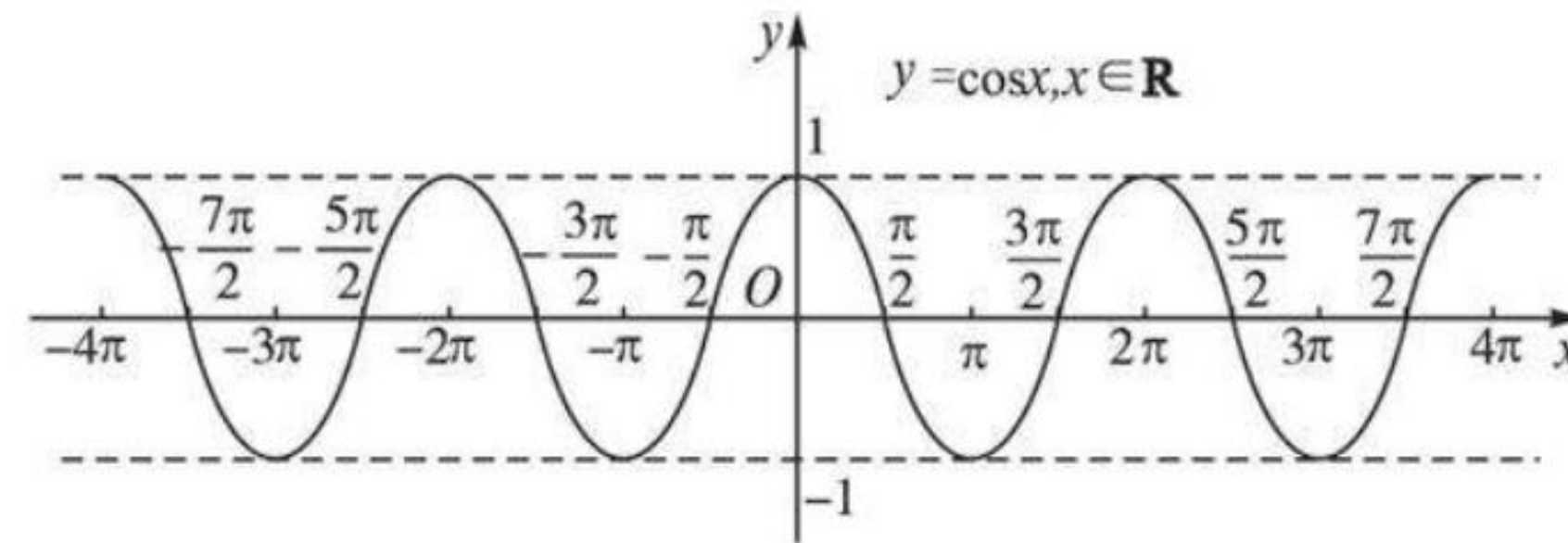
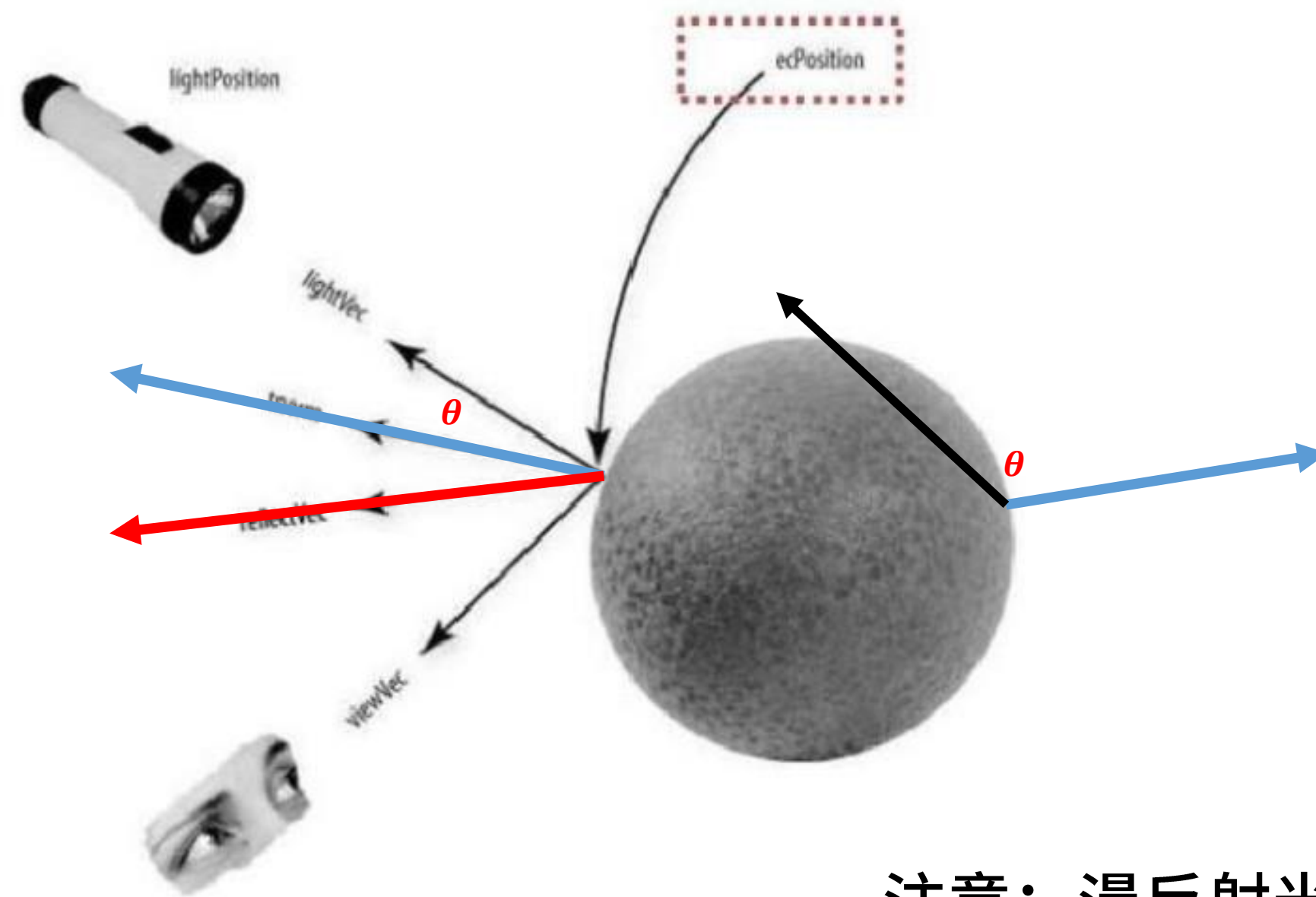
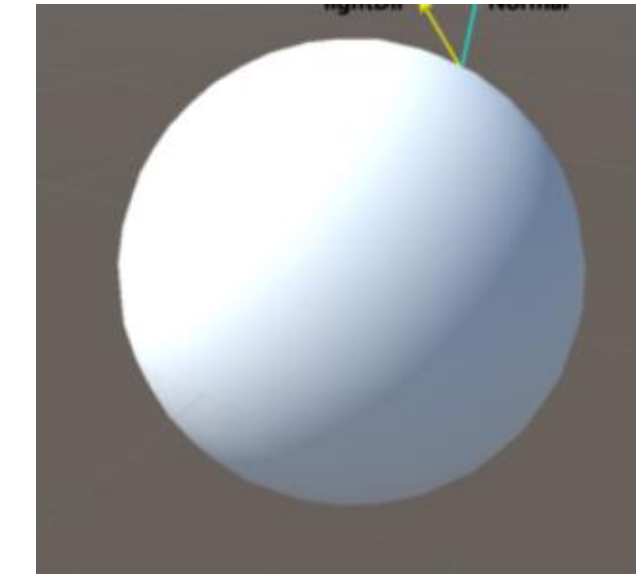
注意：环境光和入射向量、观察方向、物体表面法向量都没有关系

传统光照

Phong Lighting Model 冯氏光照模型



漫反射光：模拟光源位置对光照的影响，
比如正常的电灯、手电等



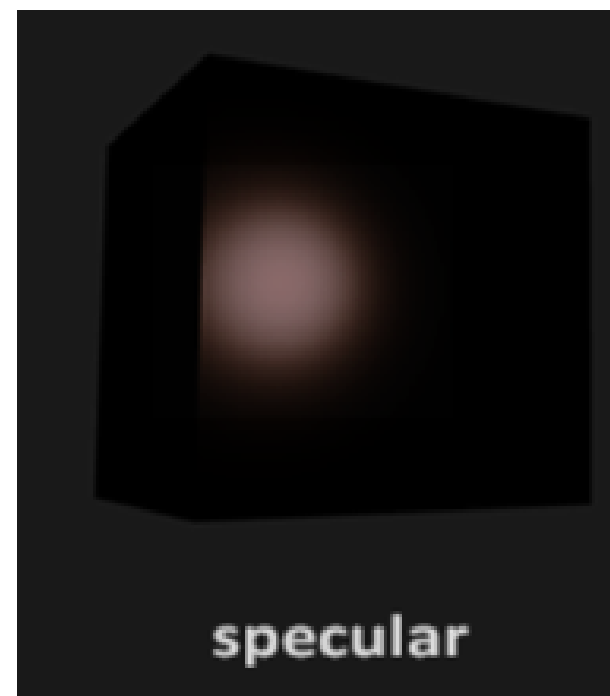
$$\vec{a} \bullet \vec{b} = x_1 x_2 + y_1 y_2 = |\vec{a}| |\vec{b}| \cos \theta$$

```
float diff = max(dot(norm, lightDir), 0.0);  
vec3 diffuse = diff * lightColor;
```

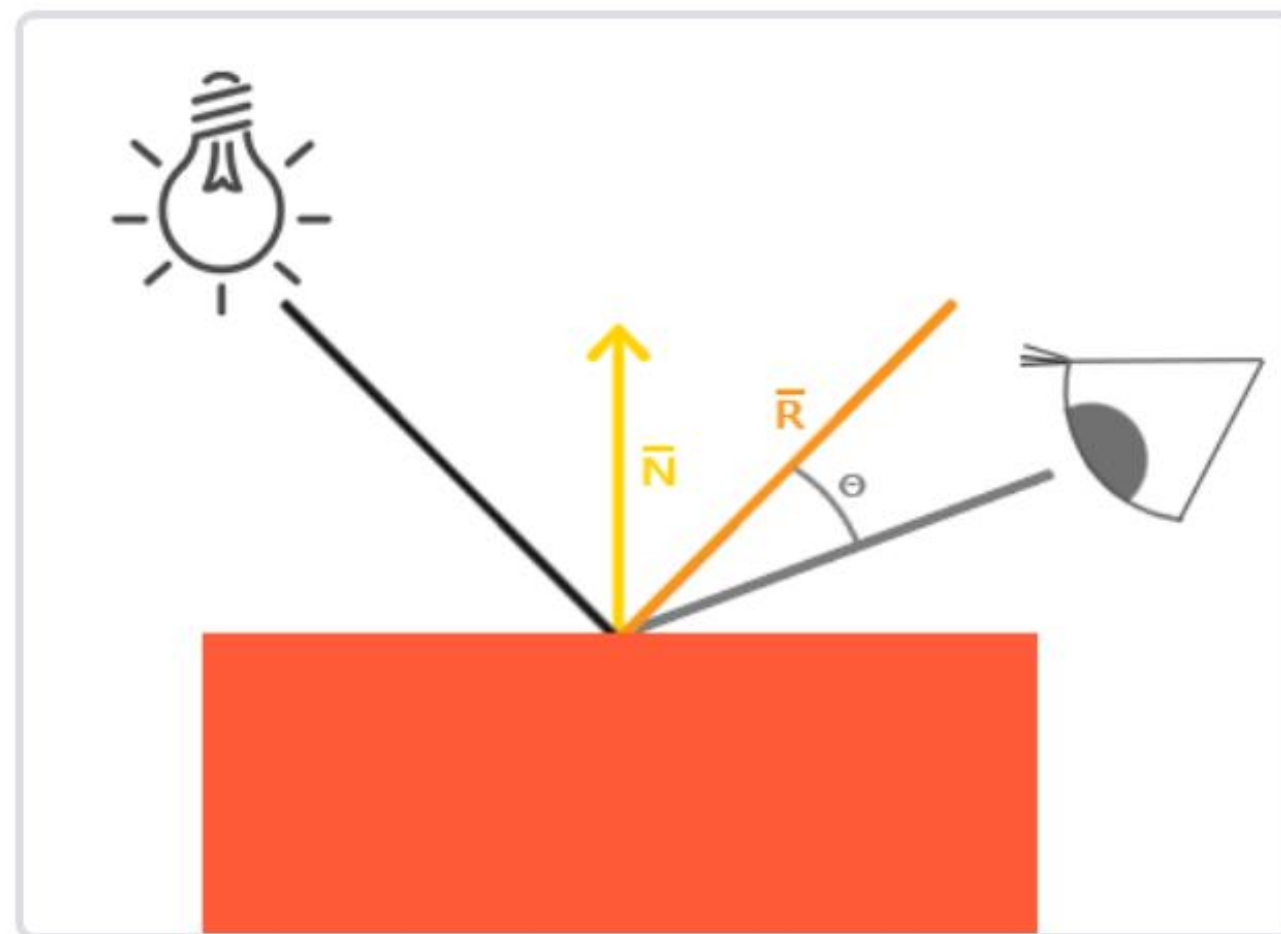
注意：漫反射光和物体表面法向量以及灯的入射向量有关，与观察向量无关

传统光照

Phong Lighting Model 冯氏光照模型



镜面光：模拟光滑物体反射的高光效果



```
float spec = pow(max(dot(viewDir, reflectDir), 0.0), 32);  
vec3 specular = specularStrength * spec * lightColor;
```

注意：镜面光与法向量、入射向量、观察向量都有关系

物理光照

传统光照的缺点：

- 1、效果细节不够丰富。
- 2、与现实有差距。
- 3、设计师与程度之间没有统一的标准。

物理光照的优点：

- 1、写实的，客观的，设计师有客观设计标准，设计与程序实现解耦。
- 2、效果与现实一致，细节丰富。

传统光照的优点：

- 1、速度快。
- 2、效果上也能够满足大多数要求。

物理光照的缺点：

- 1、消耗性能、存储、GPU计算。
- 2、设计工作量大。

物理光照

概念

基于与现实世界的物理原理更相符的基本理论所构成的渲染技术的集合

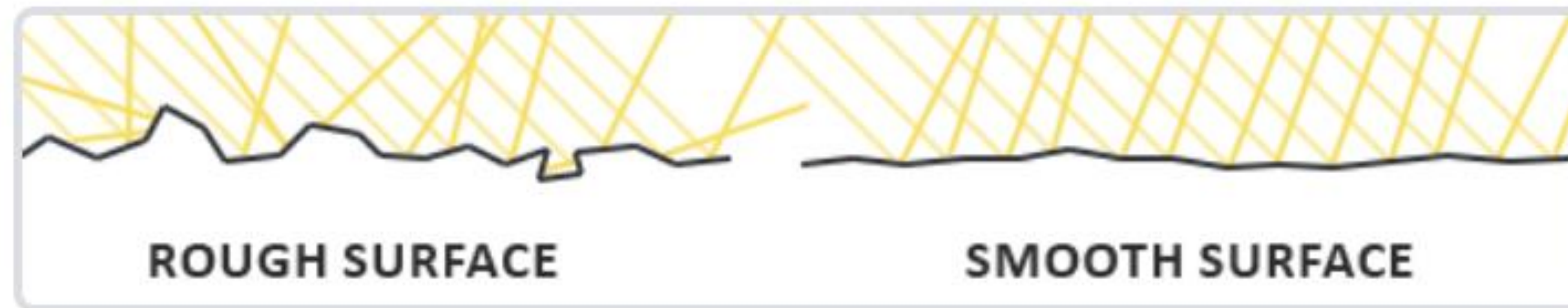
满足以下三个条件：

1. 基于微平面(Microfacet)的表面模型。
2. 能量守恒。
3. 应用基于物理的BRDF。

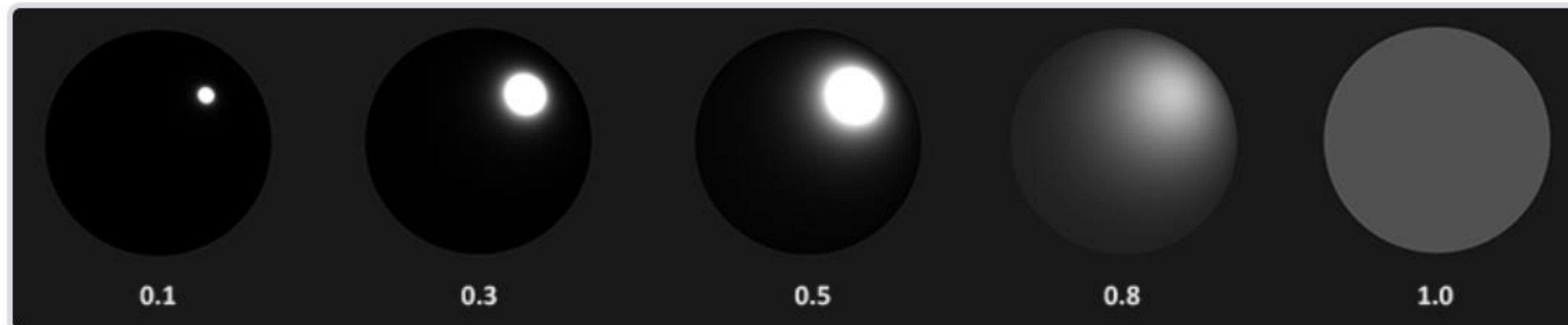


物理光照

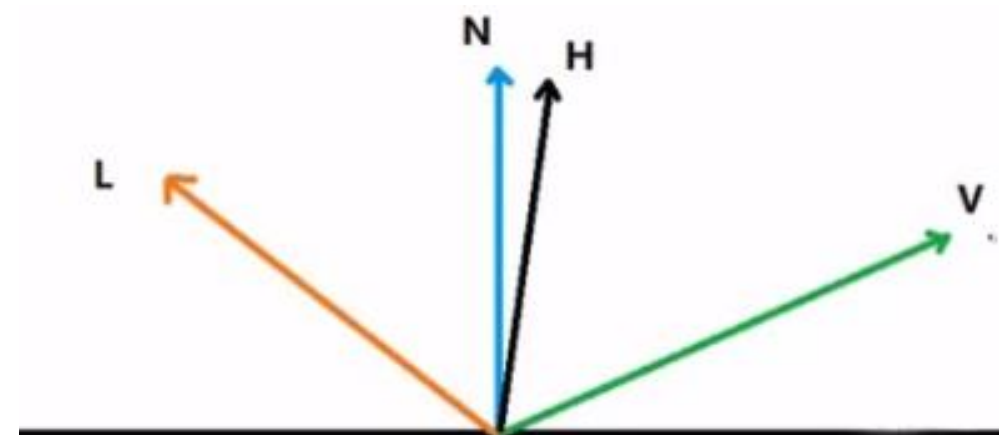
微平面理论



任何平面放大的足够大都是由无数个小平面构成的，其小无内



使用粗糙度来恒量微平面的一致性，使用半角向量辅助计算

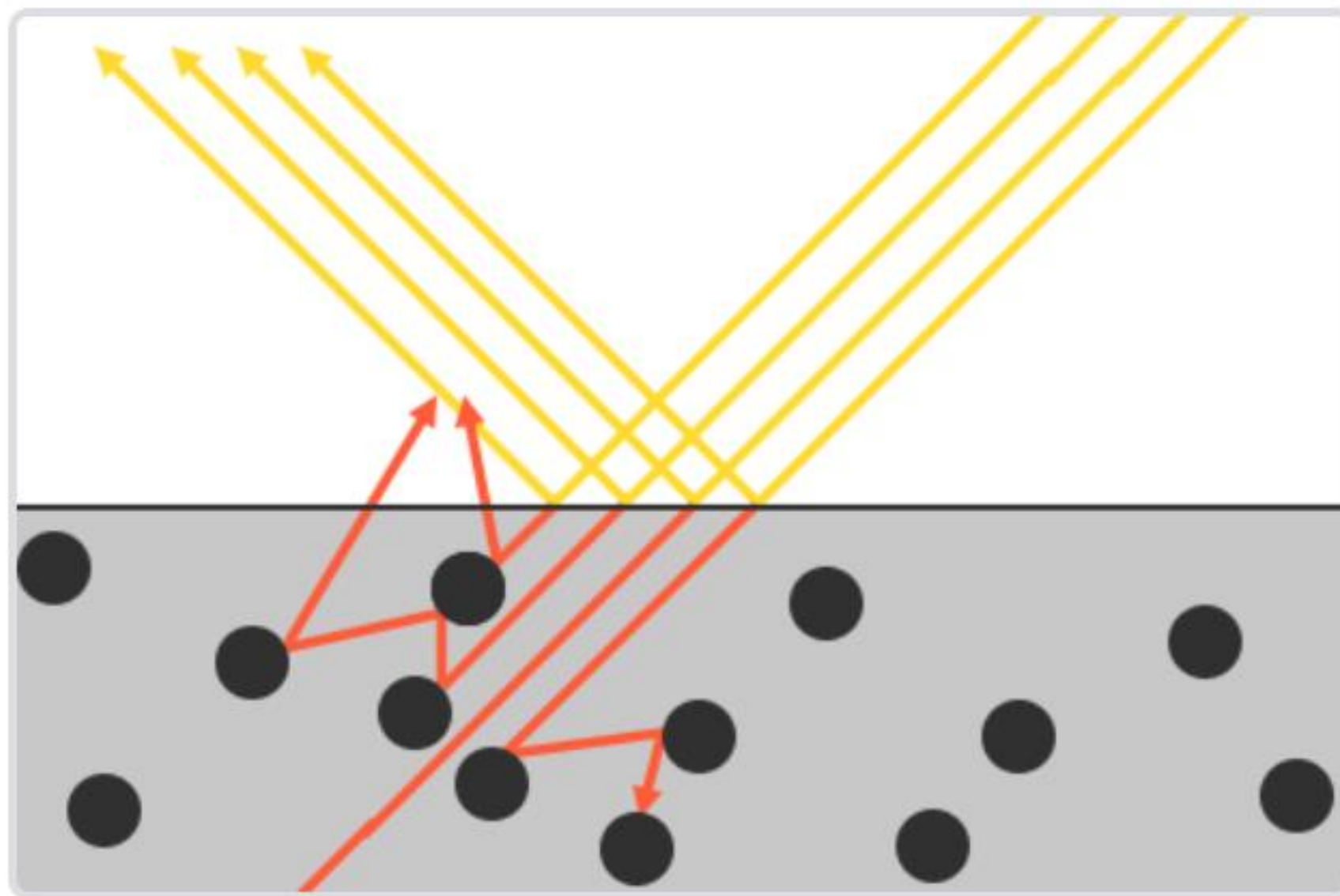


$$h = \frac{l + v}{\|l + v\|}$$

物理光照

能量守恒

出射光线的能量永远不能超过入射光线的能量（发光体除外）



```
float ks = calculateSpecularComponent(...); // 反射/镜面 部分
```

```
float kd = 1.0 - ks; // 折射/漫反射 部分
```

物理光照

基于物理的BRDF

BRDF，或者说双向反射分布函数，它接受入射（光）方向 ω_i ，出射（观察）方向 ω_o ，平面法线 n 以及一个用来表示微平面粗糙程度的参数 a 作为函数的输入参数。**BRDF可以近似的求出每束光线对一个给定了材质属性的平面上最终在某个方向上的反射出来的光线所作出的贡献程度。**

$$f_r = k_d f_{\text{lambert}} + k_s f_{\text{cook-torrance}}$$

漫反射

镜面反射

主要劝退点

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

原理-辐射度学基本量

用辐射度学来对光的行为进行解释

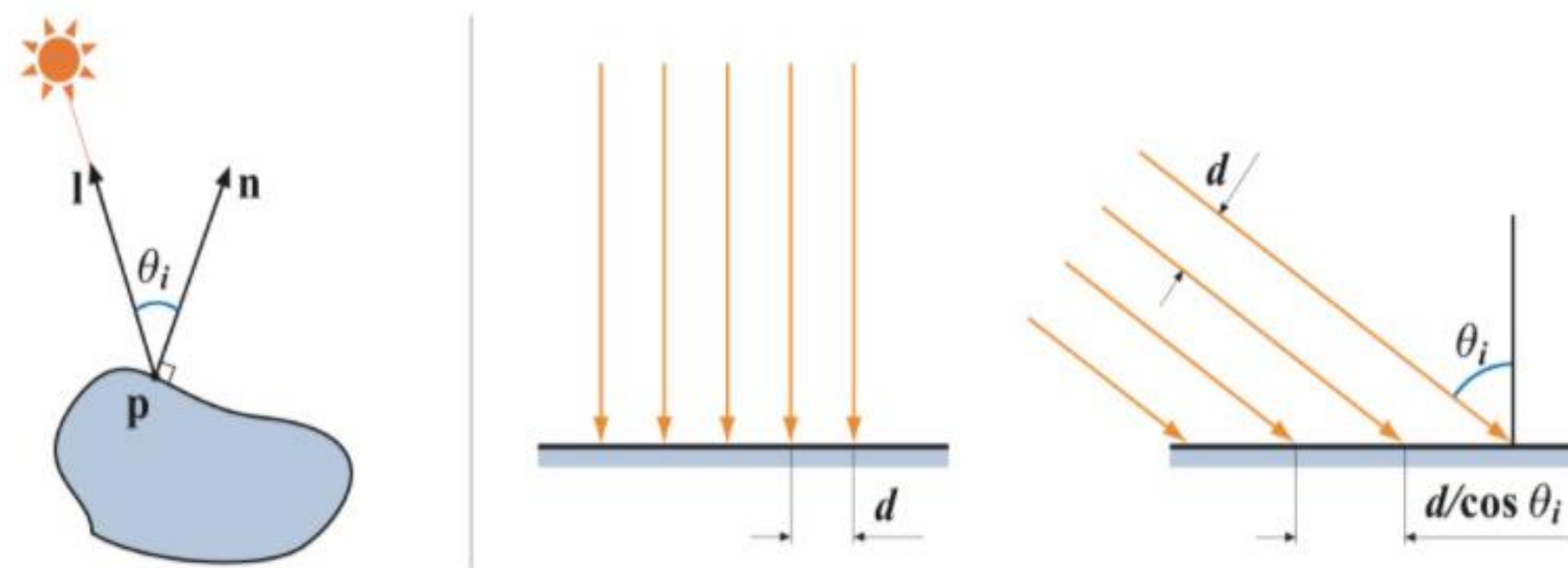
光的能量：能量(Energy)，用符号 Q 表示，单位是焦耳(J)，每个光子都具有一定的能量，和频率相关，频率越高，能量也越高。

功率（辐射通量）：功率(Power)，单位瓦特(Watts)，或者焦耳/秒(J/s)。辐射度学中，辐射功率也被称为辐射通量(Radiant Flux)或者通量(Flux)，指单位时间内通过表面或者空间区域的能量的总量，用符号 Φ 表示，定义 $\Phi = \frac{dQ}{dt}$

原理-辐射度学基本量

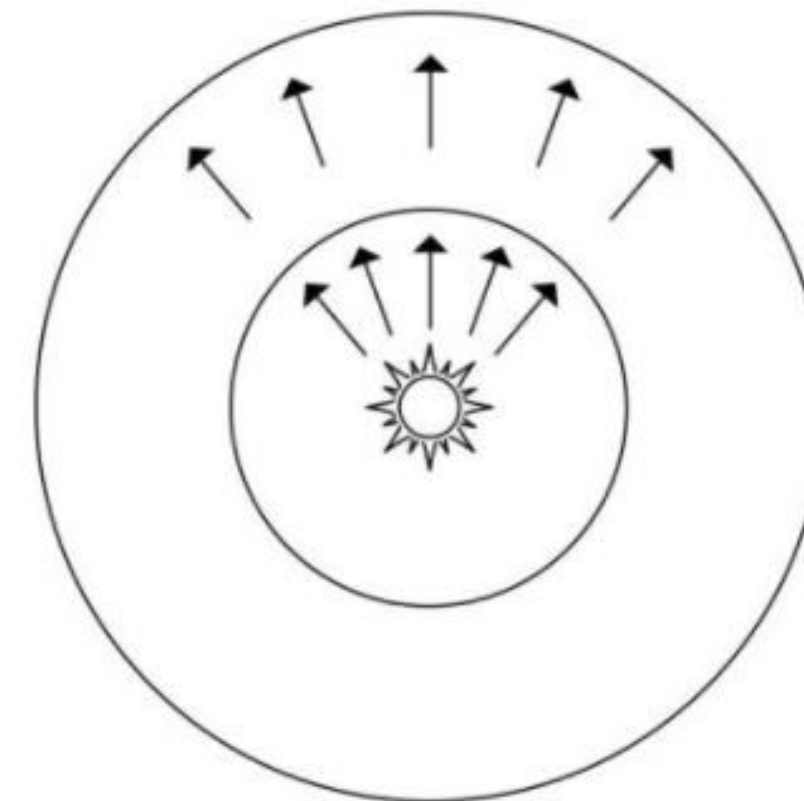
用辐射度学来对光的行为进行解释

辐照度和辐出度：辐照度(Irradiance)，指单位时间内到达单位面积的辐射通量，辐出度是离开的辐射通量。 $E = \frac{d\Phi}{dA}$ 单位是 W/m^2



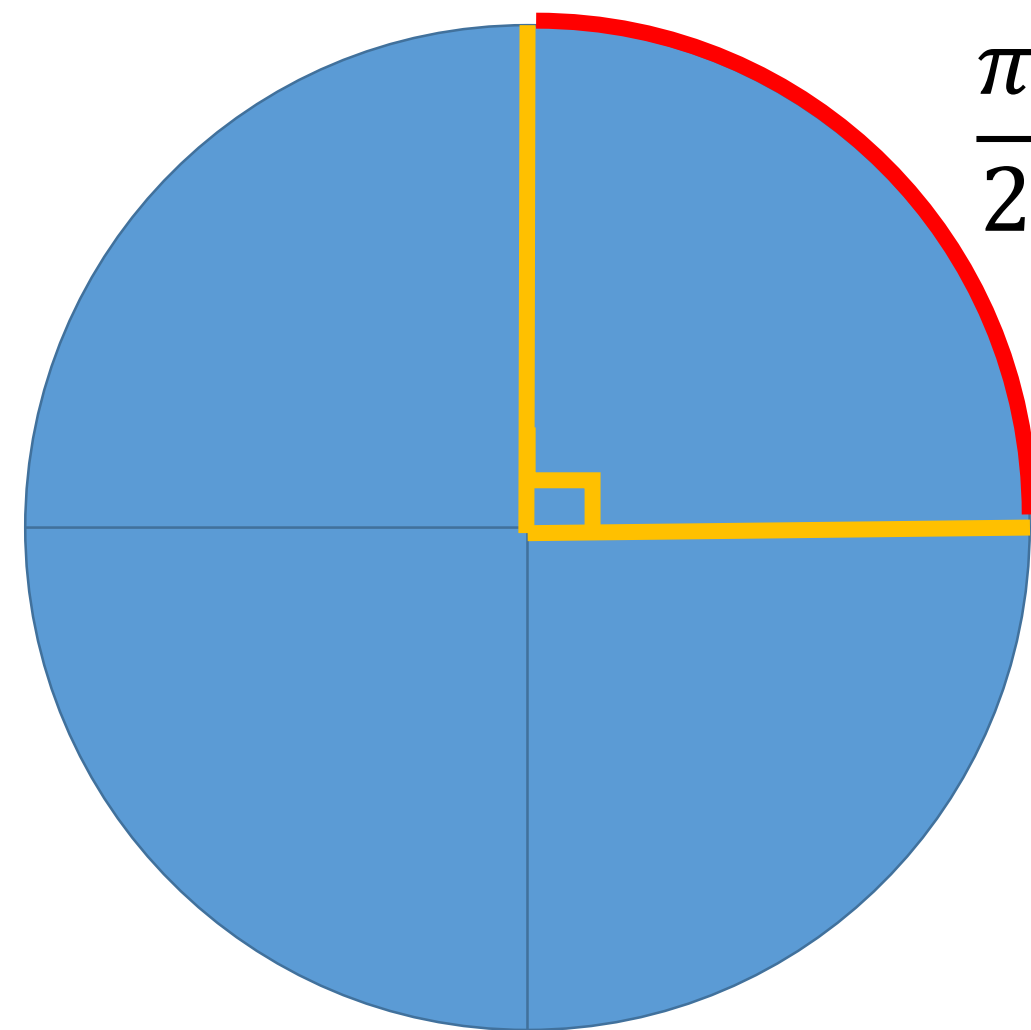
$$E_L = \frac{\Phi}{A^\perp} = \frac{\Phi}{A \cos \theta_i}$$

点光源向四周辐射能量，假想以点光源为中心不同半径的球包围着点光源，穿过这些球的辐射通量是相同的，均为 Φ ，而球的表面积是 $4\pi r^2$ ，可得通量密度 $E = \frac{\Phi}{4\pi r^2}$ ，**也就是说通量密度与距离的平方成反比**，离光源越远，通量密度越低，这也是为什么光的衰减与距离的平方成正比。



原理-立体角

弧度



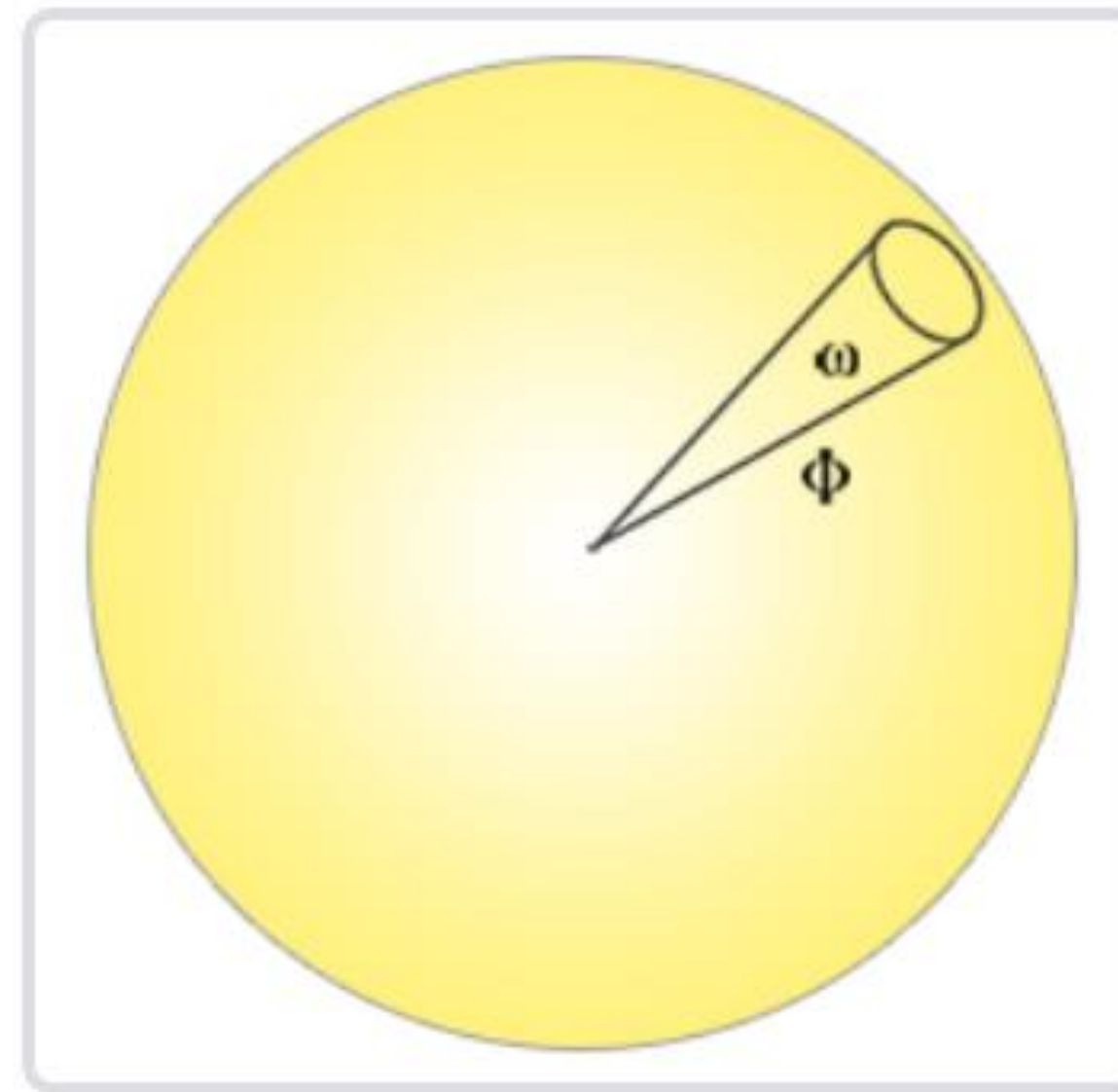
角度所对应单位圆的弧长

为什么引入立体角 ω ?

光线可以使用一个向量和一个立体角来表示，如下图所示。向量表示这束光线的方向，立体角表示这束光线在单位球上的面积，也就是光线的粗细

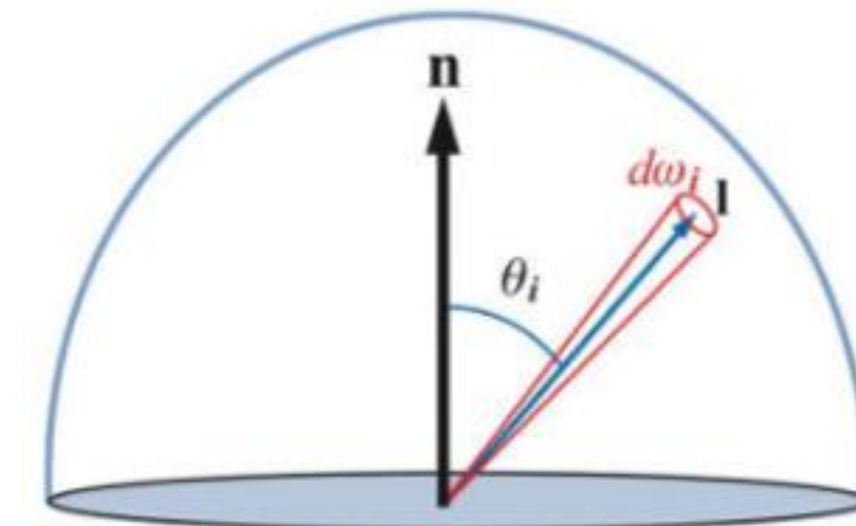
辐射强度：单位立体角的辐射通量。 $I = \frac{d\Phi}{d\omega}$

立体角（立体弧度）

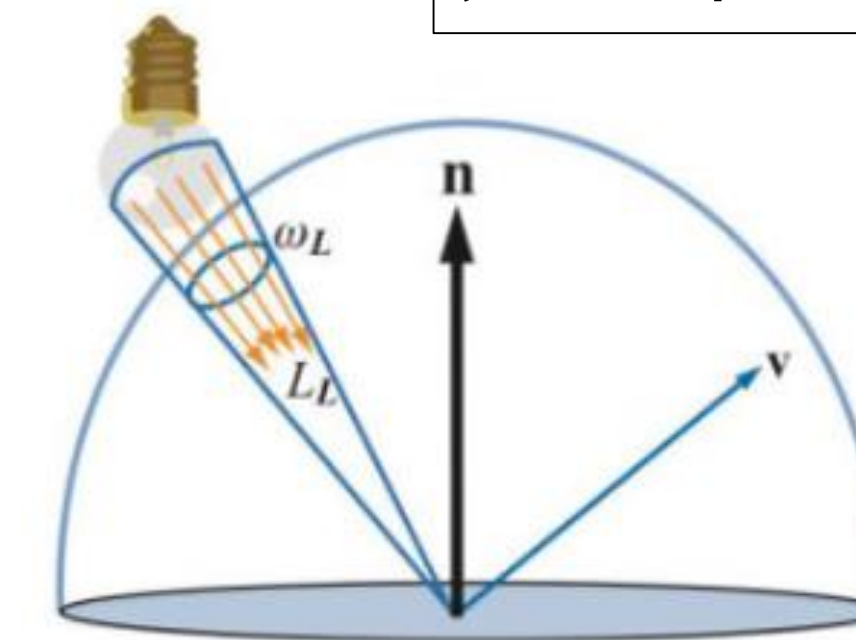


角度所对应单位球表面积

光的方向



光的粗细



为什么引入辐射强度?

因为有时候要度量通过一个点的通量密度，但是因为点的面积是0，无法使用辐照度（通过面积的通量密度），所以引入辐射强度。

原理-辐射率

辐射率:

指每单位面积每单位立体角的辐射通量密度

可以解释为从一个微小面积的表面出发，射向某个微小方向的通量（或者来自某个微小方向，照射到微小面积表面的通量）

$$L = \frac{d\Phi}{d\omega dA^\perp}$$

特性:

- 1、辐射率其实就是我们看到的物体的颜色，基于物理着色其实就是在计算它的辐射率
- 2、辐射率不会随距离的变化而衰减。这和我们日常观察一至，无论物体是近是远其颜色不会发生变化，这是因为随着距离拉远物体到达视网膜的通量密度会变小，而这块区域在视网膜表面上的立体角也会变小，相互抵消。

$$f_r = k_d f_{\text{lambert}} + k_s f_{\text{cook-torrance}}$$

镜面反射

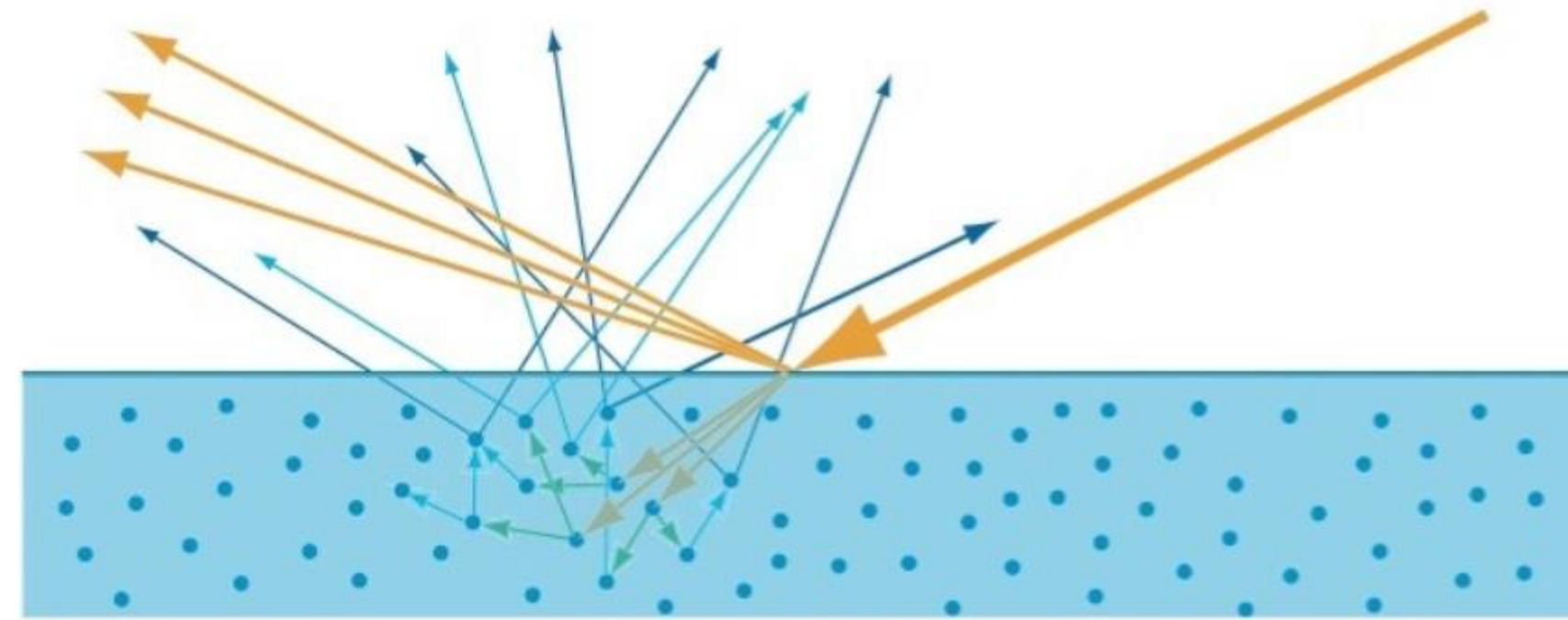
下一步：证明劝退点公式BRDF

漫反射

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

原理-BRDF的定义

一束光照在物体上，即发生反射也会被吸收，我们眼镜看到的光是无数光碰到物体发生的反射进我们眼里的反射光的总和



BRDF是一个比例关系：指定方向的反射光和入射光之间的关系。

$$f(l, v) = \frac{dL_0(v)}{dE(l)}$$

f 就是BRDF, l 是入射光的方向， v 是观察方向，也就是我们关心的反射光方向。

其中 $dL_0(v)$ 是表面反射到 v 方向的反射光的微分辐射率。表面反射到 v 方向的反射光的辐射率为 $L_0(v)$ ，来自于表面上半球所有方向的入射光线的贡献，也就是最终呈现的颜色。而微分辐射率 $dL_0(v)$ 特指来自方向 l 的入射光贡献的反射辐射率。

其中 $dE(l)$ 是表面上来自入射光方向 l 的微分辐照度，表面接收到的辐照度为 E ，来自上半球所有入射光线的贡献，而微分辐照度 $dE(l)$ 特指来自于方向 l 的入射光。

原理-怎么用BRDF来计算表面辐射率?

来自方向 l 的入射光辐射率 $L_i(l)$

$$L_i(l) = \frac{d\Phi}{d\omega_i dA^\perp} = \frac{d\Phi}{d\omega_i dA \cos \theta_i} = \frac{dE(l)}{d\omega_i \cos \theta_i}$$

则照射到表面来自于方向 l 的入射光贡献的微分辐照度

$$dE(l) = L_i(l) d\omega_i \cos \theta_i$$

符号 $\otimes$ 表示按向量的分量相乘，因为 f 和 L_i 都包含RGB三个分量。

表面反射到 v 方向的由来自于方向 l 的入射光贡献的微分辐射率为：

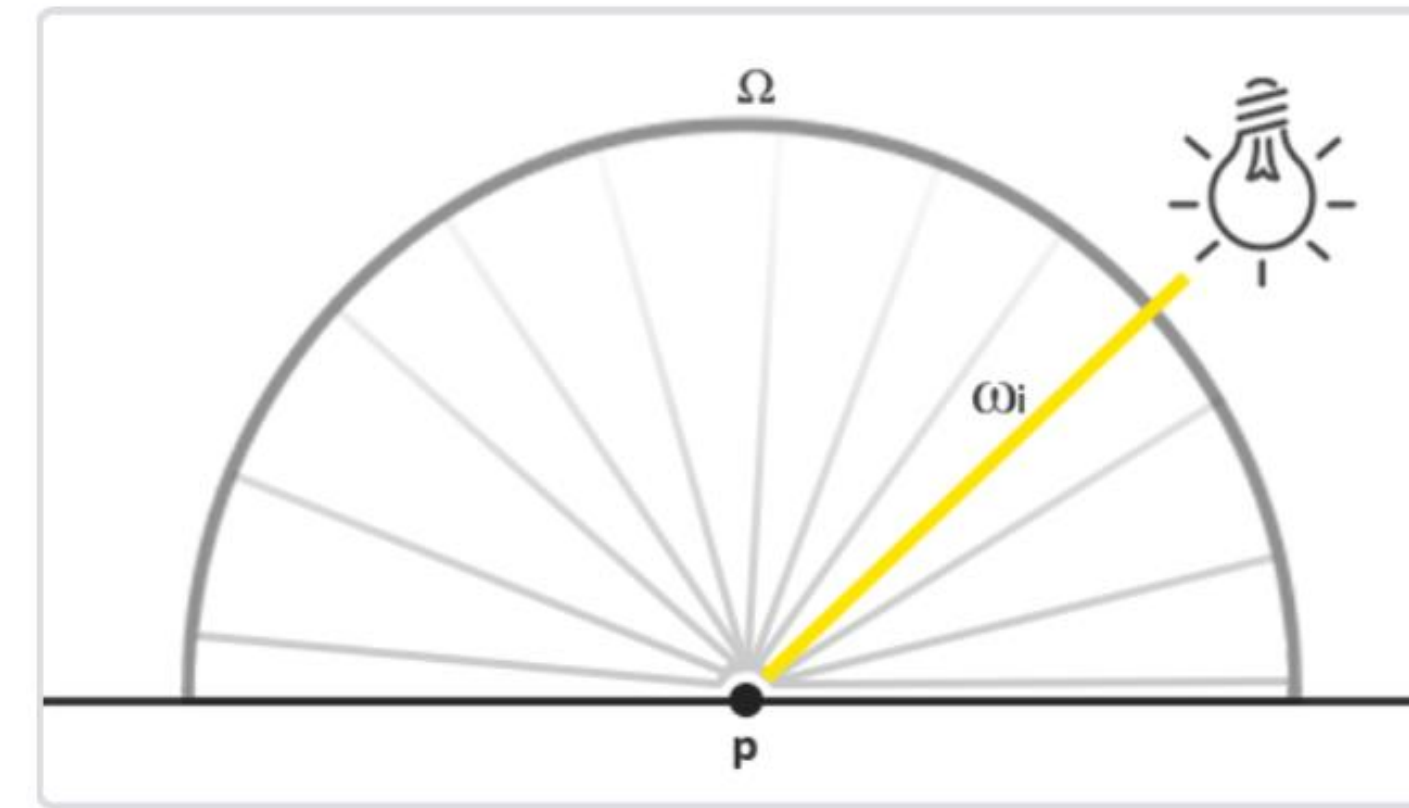
由BRDF定义 $f(l, v) = \frac{dL_o(v)}{dE(l)}$ 则 $dL_o(v) = f(l, v) \otimes dE(l) = f(l, v) \otimes L_i(l) d\omega_i \cos \theta_i$

要计算表面反射到 v 方向的来自上半球所有方向入射光线贡献的辐射率，可以将上式对半球所有方向的光线进行积分

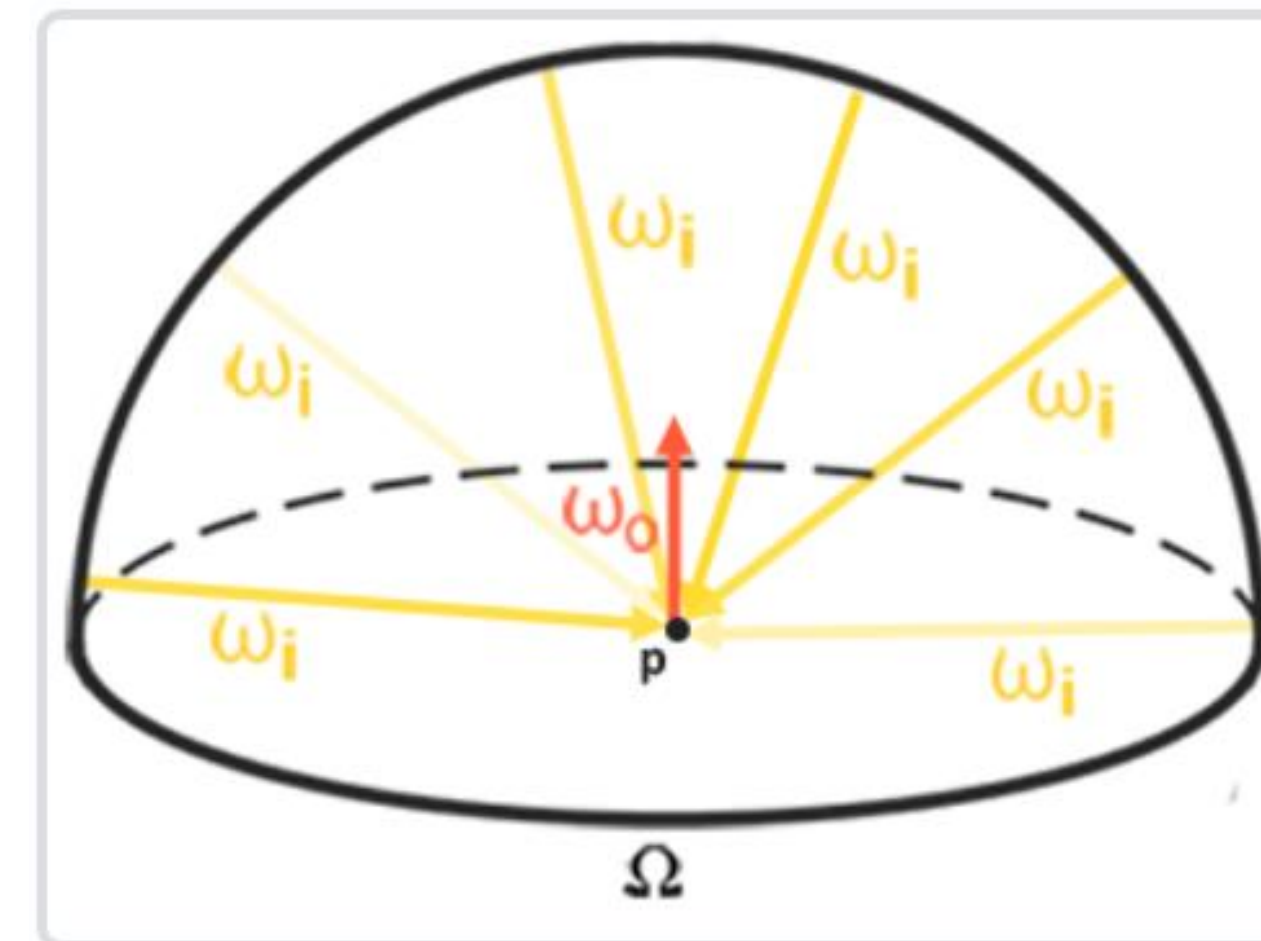
$$L_o(v) = \int_{\Omega} f(l, v) \otimes L_i(l) d\omega_i \cos \theta_i$$

原理-几个为什么

为什么是上半球？



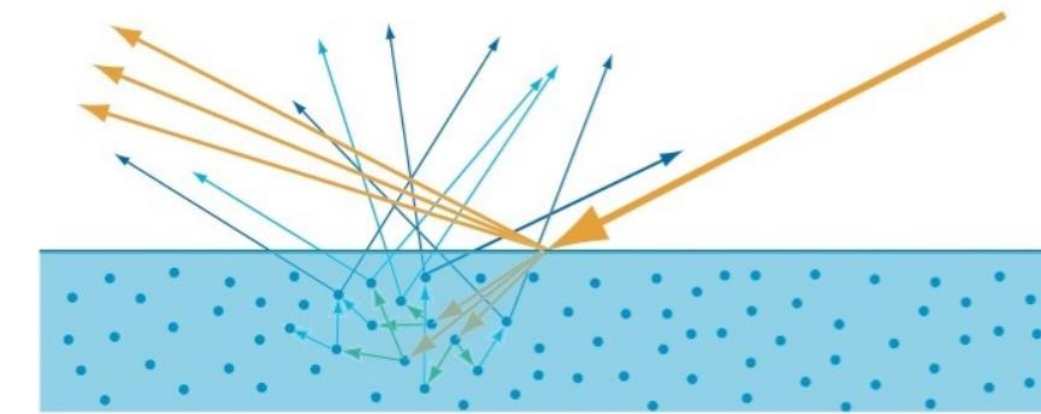
基于微平面光的入射与人观察的反射都是基于一个平面的，只考虑平面法线侧的半球方向的光的计算



原理-几个为什么

为什么BRDF要定义成辐射率和辐照度的比值？

$$f(l, v) = \frac{dL_0(v)}{dE(l)}$$

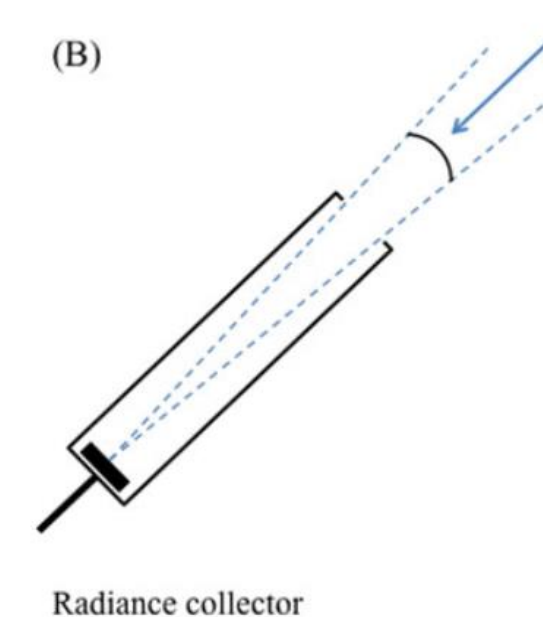
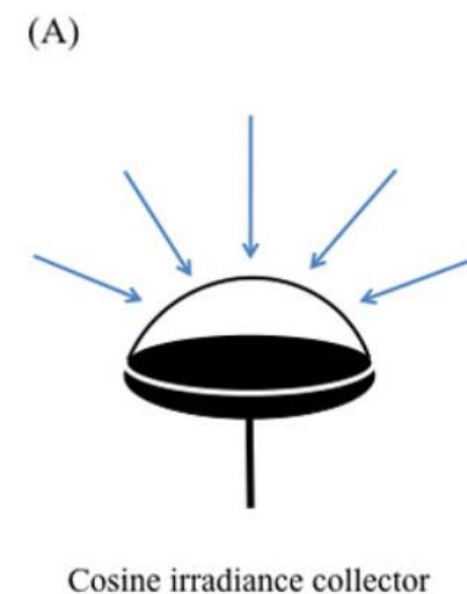


从数学角度，对于一束光，因为出射对观察方向的辐射率只占入射辐射率的一小部分

$$\lim_{\omega \rightarrow 0} \frac{dL_0}{L_i} = 0$$

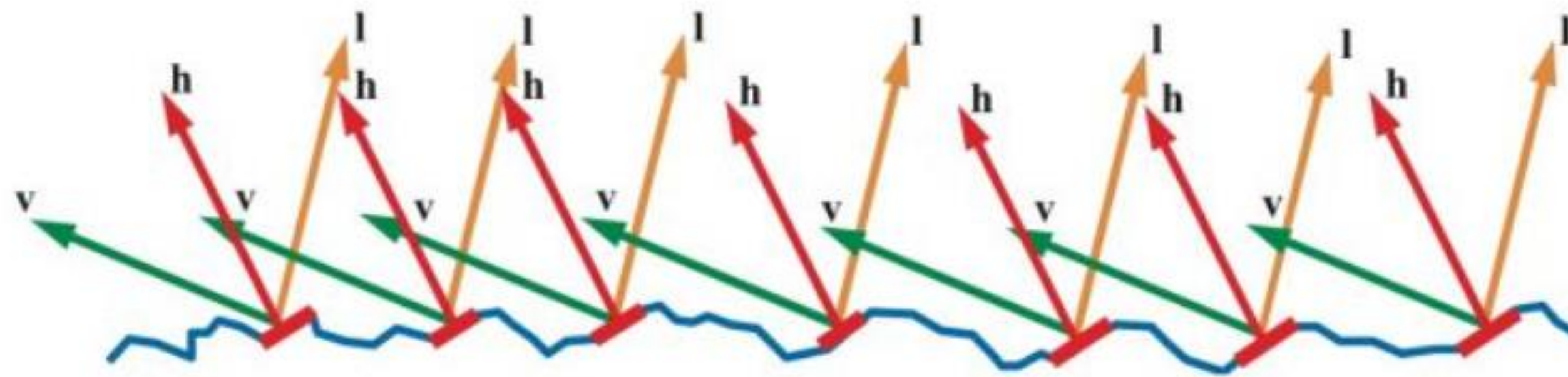
实际计算中是没有意义的，而除以辐照度则多了一个微分立体角，结果不是0，是有意义的

从物理角度，出射辐射率与入射辐射度都是可测的，而入射辐射率的测量难度很大几乎是不可能的



原理-利用BRDF来建立光照模型

1967年Torrance-Sparrow在Theory for Off-Specular Reflection From Roughened Surfaces中使用辐射度学和微表面理论建立了模拟真实光照的BRDF模型，1981年Cook-Torrance在A Reflectance Model for Computer Graphics中把这个模型引入到计算机图形学领域，现在这个模型已经成为基于物理着色的标准，被称为**Cook-Torrance模型**。

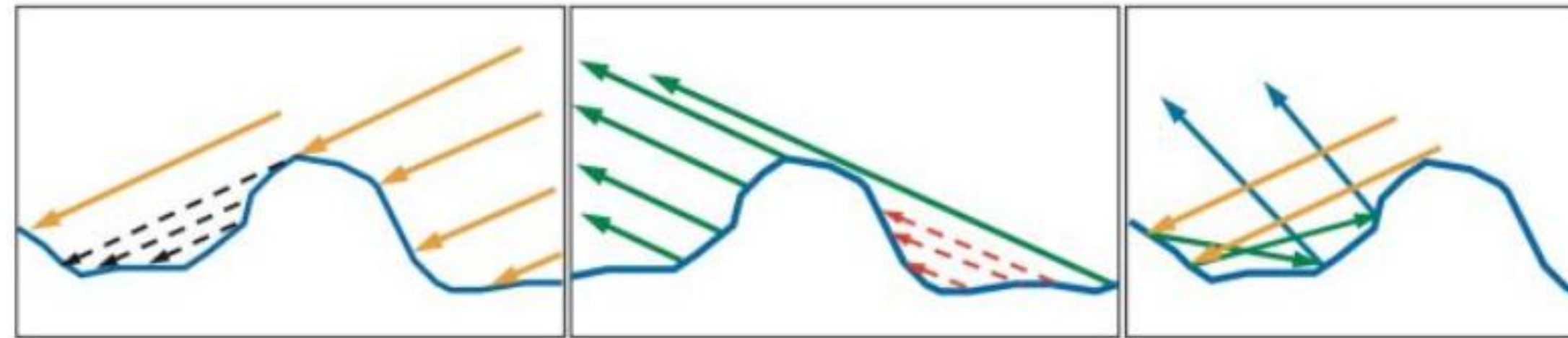


法线分布函数（Normal Distribution Function，简称为NDF） $D(h)$ 来描述组成表面一点的所有法线的分布概率。可以近似这么理解：向NDF输入一个朝向 h ，NDF会返回朝向是 h 的微表面数占微表面总数的比例，比如有1%的微表面朝向是 h ，那么就有1%的微表面可能将光线反射到 v 方向。

法线分布函数模拟的是有多少面朝向你关注的方向

原理-利用BRDF来建立光照模型

几何衰减因子（Geometrical Attenuation Factor） $G(l, v)$ 来建模，输入入射和观察光线方向，输出值表示光线未被遮蔽而能从 l 反射到 v 方向的比例。



几何衰减因子模拟的是朝你方向的面有多少是被各种挡住的

菲涅尔方程（Fresnel Equations） $F(l, h)$ ，表示光学平面并不会将所有光线都反射掉，而是一部分被反射，一部分被折射，反射比例符合菲涅尔方程（Fresnel Equations） $F(l, h)$

菲涅尔方程模拟的是即便光未被挡住，也有部分是反射有部分折射，计算出这个比例，只有反射才可能入眼

Torrance-Sparrow基于微表面理论，用上述三个函数建立了高光BRDF模型：

$$f(l, v) = \frac{F(l, h)G(l, v)D(h)}{4 \cos \theta_i \cos \theta_o} = \frac{F(l, h)G(l, v)D(h)}{4(n \cdot l)(n \cdot v)}$$

原理-推理Cook-Torrance公式

$$f(l, v) = \frac{F(l, h)G(l, v)D(h)}{4 \cos \theta_i \cos \theta_o} = \frac{F(l, h)G(l, v)D(h)}{4(n \cdot l)(n \cdot v)}$$

一束光，照到一组微平面上，入射光方向 ω_i ，观察方向 ω_o ，对反射到 ω_o 方向的反射光有贡献的微表面法线为半角向量 ω_h ，则这束光的微分通量

$$d\Phi_h = L_i(\omega_i)d\omega_i dA^\perp(\omega_h) = L_i(\omega_i)d\omega_i \cos \theta_h dA(\omega_h)$$

其中 $dA(\omega_h)$ 是法线为半角向量 ω_h 的微分表面面积， $dA^\perp(\omega_h)$ 为 $dA(\omega_h)$ 在入射光线方向的投影， θ_h 为入射光线 ω_i 和微表面法线 ω_h 的夹角。

Torrance-Sparrow将微分微表面面积 $dA(\omega_h)$ 定义为 $dA(\omega_h) = D(\omega_h)d\omega_h dA$

由以上两式可得：

$$d\Phi_h = L_i(\omega_i)d\omega_i \cos \theta_h D(\omega_h)d\omega_h dA$$

菲涅尔定理，则反射通量

$$d\Phi_o = F_r(\omega_o)d\Phi_h$$

$$dL_o(\omega_o) = \frac{d\Phi_o}{d\omega_o \cos \theta_o dA} = \frac{F_r(\omega_o)L_i(\omega_i)d\omega_i \cos \theta_h D(\omega_h)d\omega_h dA}{d\omega_o \cos \theta_o dA}$$

$$f_r(\omega_i, \omega_o) = \frac{dL_o(\omega_o)}{dE_i(\omega_i)} = \frac{dL_o(\omega_o)}{L_i(\omega_i) \cos \theta_i d\omega_i} = \frac{F_r(\omega_o) \cos \theta_h D(\omega_h)d\omega_h}{\cos \theta_o \cos \theta_i d\omega_o}$$

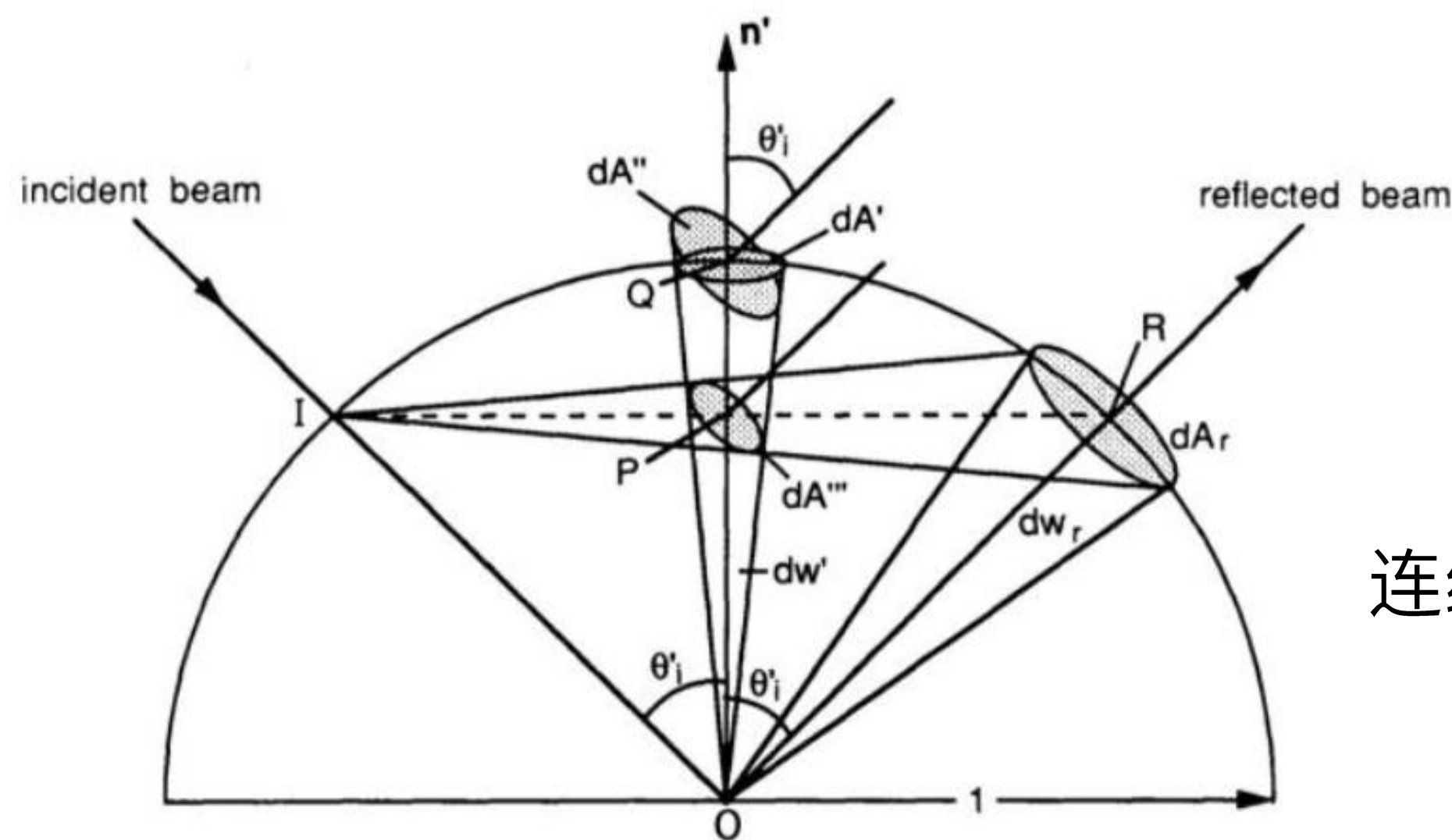
原理-推理Cook-Torrance公式

$$f(l, v) = \frac{F(l, h)G(l, v)D(h)}{4 \cos \theta_i \cos \theta_o} = \frac{F(l, h)G(l, v)D(h)}{4(n \cdot l)(n \cdot v)}$$

现在咱们推理到这了: $f_r(\omega_i, \omega_o) = \frac{dL_o(\omega_o)}{dE_i(\omega_i)} = \frac{dL_o(\omega_o)}{L_i(\omega_i) \cos \theta_i d\omega_i} = \frac{F_r(\omega_o) \cos \theta_h D(\omega_h) d\omega_h}{\cos \theta_o \cos \theta_i d\omega_o}$

只需要再证明 $\frac{d\omega_h}{d\omega_o} = \frac{1}{4 \cos \theta_h}$ 然后再考虑衰减因子G, 就可以了。

$$\frac{d\omega_h}{d\omega_o} = \frac{dA'}{dA_r}$$



连线IR与法线 n' 相交于点 P , 则 $IR = 2IP$

dA_r 与 dA''' 的半径的比值等于 $\frac{IR}{IP}$, 而面积为 πr^2 , 与半径的平方成正比, 所以 $dA_r = 4dA'''$

连线 OQ 长度为1, OP 长度为 $\cos \theta'_i$ 而 $dA'' = \frac{dA'}{\cos \theta'_i}$

$$\frac{dA'}{dA_r} = \frac{dA'' \cos \theta'_i}{4 dA'''} = \frac{1}{4 \cos \theta'_i}$$

实践-求解点光源的BRDF镜面反射部分

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

第一步：ks和kd其和为1，从菲涅尔方程得来：

$$F_{Schlick}(h, v, F_0) = F_0 + 1 - F_0(1 - (h \cdot v)^5)$$

求得的是ks， F_0 是从法线的方向上正看，其反射光线所占百分比，被称为基础反射率，对于非电解质来说，一般设置为0.04即可，h是v和l的半角向量，v是观察向量都是已知的

```
56.  vec3 fresnelSchlick(float cosTheta, vec3 F0)
57.  {
58.      return F0 + (1.0 - F0) * pow(1.0 - cosTheta, 5.0);
59.  }
```

实践-求解点光源的BRDF镜面反射部分

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

第二步：求D：

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

n是表面法线，可以从法线纹理中获取，是已知，h是l和v的半角向量，a是粗糙度可以从粗糙度纹理中获取

```
21. float DistributionGGX(vec3 N, vec3 H, float roughness)
22. {
23.     float a = roughness*roughness;
24.     float a2 = a*a;
25.     float NdotH = max(dot(N, H), 0.0);
26.     float NdotH2 = NdotH*NdotH;
27.
28.     float nom    = a2;
29.     float denom = (NdotH2 * (a2 - 1.0) + 1.0);
30.     denom = PI * denom * denom;
31.
32.     return nom / max(denom, 0.001); // prevent divide
    for roughness=0.0 and NdotH=1.0
33. }
```

实践-求解点光源的BRDF镜面反射部分

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

第三步：求G：

$$G(n, v, l, k) = G_{sub}(n, v, k) G_{sub}(n, l, k)$$

$$G_{SchlickGGX}(n, v, k) = \frac{n \cdot v}{(n \cdot v)(1 - k) + k}$$

$$k_{direct} = \frac{(\alpha + 1)^2}{8}$$

n是法线，从法线纹理中获取，v是观察向量，而k与粗糙度有个映射关系

```
35. float GeometrySchlickGGX(float NdotV, float roughness)
36. {
37.     float r = (roughness + 1.0);
38.     float k = (r*r) / 8.0;
39.
40.     float nom    = NdotV;
41.     float denom  = NdotV * (1.0 - k) + k;
42.
43.     return nom / denom;
44. }
45. // -----
46. float GeometrySmith(vec3 N, vec3 V, vec3 L, float roughness)
47. {
48.     float NdotV = max(dot(N, V), 0.0);
49.     float NdotL = max(dot(N, L), 0.0);
50.     float ggx2 = GeometrySchlickGGX(NdotV, roughness);
51.     float ggx1 = GeometrySchlickGGX(NdotL, roughness);
52.
53.     return ggx1 * ggx2;
54. }
```

实践-求解点光源的BRDF镜面反射部分

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

求解 $L_i(p, \omega_i)$

$$L_i(l) = \frac{c}{distance * distance}$$

```
79.         float attenuation = 1.0 / (distance * distance);  
80.         vec3 radiance = lightColors[i] * attenuation;
```

因为是点光源，所以半球积分只需要考虑光源和顶点这一条光线就可以，不会有其它光线，到这里其实BRDF的镜面反射部分已经全部得证并且代码OK了~

持续阅读

遗留的漫反射部分，还有关于IBL（基于图像的BRDF），关于反射探针等研究…..

请参考以下书籍和文章：

->首先：<https://learnopengl-cn.github.io/>

里面关于PBR的部分是本文的大部分内容，以及本文未介绍的也值得深读

->其次：<https://zhuanlan.zhihu.com/p/21376124>

里面关于BRDF的推导过程讲述的非常浅显易懂

->其它：《Ray Tracing In One Weekend》三部曲

对蒙特卡洛积分、光线追踪是很好的入门书籍，蒙特卡洛积分是实时PBR中也会使用的

《Real shading in Unreal Engine 4》

介绍了UE4中使用的PBR模型，是现实实践的首选模型

Thanks
For
Watching

www.henggetec.com

商务合作：陈亮宏

电话：13051650026

