



计算机图形学

第八章 真实感图形绘制

颜波

复旦大学计算机科学技术学院
byan@fudan.edu.cn



本章概述

- 简单光照明模型
- 多边形绘制方法
- 透明
- 整体光照明模型
- 光线跟踪算法

真实感图形绘制

- 光照模型(Illumination Model): 计算某一点的光强度的模型
- 真实感图形的特点
 - 能反映物体表面颜色和亮度的细微变化
 - 能表现物体表面的质感
 - 能通过光照下的物体阴影, 极大地改善场景的深度感和层次感, 充分体现物体间的相互遮挡关系
 - 能模拟透明物体的透明效果和镜面物体的镜面效果

影响观察者看到的表面颜色的因素

1. 物体的几何形状
2. 光源：位置、距离、颜色、数量、强度、种类
3. 环境：遮挡关系、光的反射与折射、阴影
4. 视点位置
5. 物性：材料、颜色、透明度、折射性
6. 表面光洁度

- 几何性质
 - 点光源
 - 线光源
 - 面光源
- 光谱组成
 - 白色光等能量的各种波长可见光的组合
 - 彩色光
 - 单色光

光照模型

- 光照模型 (lighting model 或 illumination model) 用于计算光的强度:
 - 局部光照明 (Local Lighting)
 - 关注物体直接受到光源影响所产生的光照效果
 - 全局光照明 (Global Lighting)
 - 关注光线在场景中其他物体表面发生折射、反射等作用后, 间接照射到物体表面上产生的影响

光照模型的历史

- 1967年, Wylie 等人第一次在显示物体时加入了光照明效果, 认为光的强度与物体到光源的距离成反比关系
- 1970年, Bouknight 提出了第一个光反射模型:
 - Lambert 漫反射光 + 环境光
 - 发表于 Communication of ACM

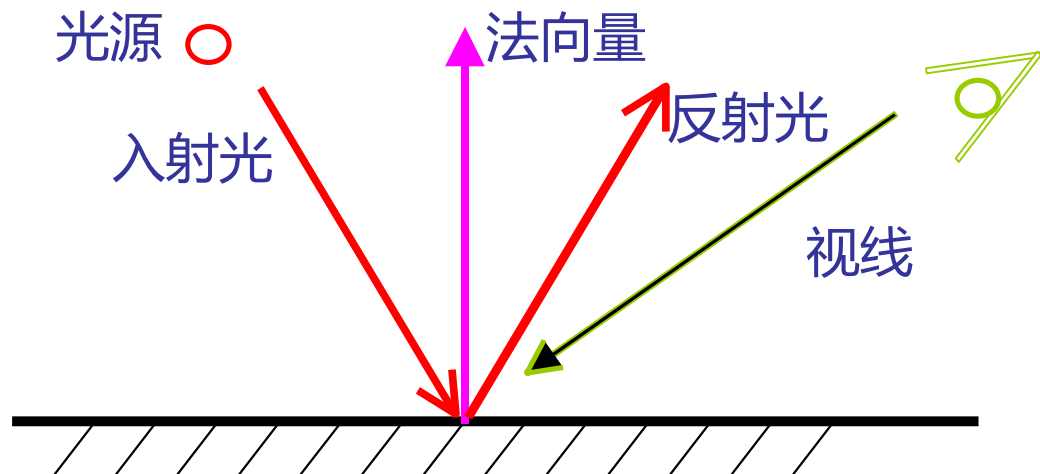
光照模型的历史

- 1971年, Gouraud 提出了漫反射模型加插值的思想:
 - Lambert 漫反射光 + Barycentric 插值
 - 发表于 IEEE transactions on Computers
- 1975年, Phong 提出了图形学中第一个有影响也是最有影响的光照模型: Phong 模型
 - 漫反射 (diffuse light) + 环境光 (ambient light) + 高光 (specular light)
 - 发表于 Communication of ACM

相关物理知识

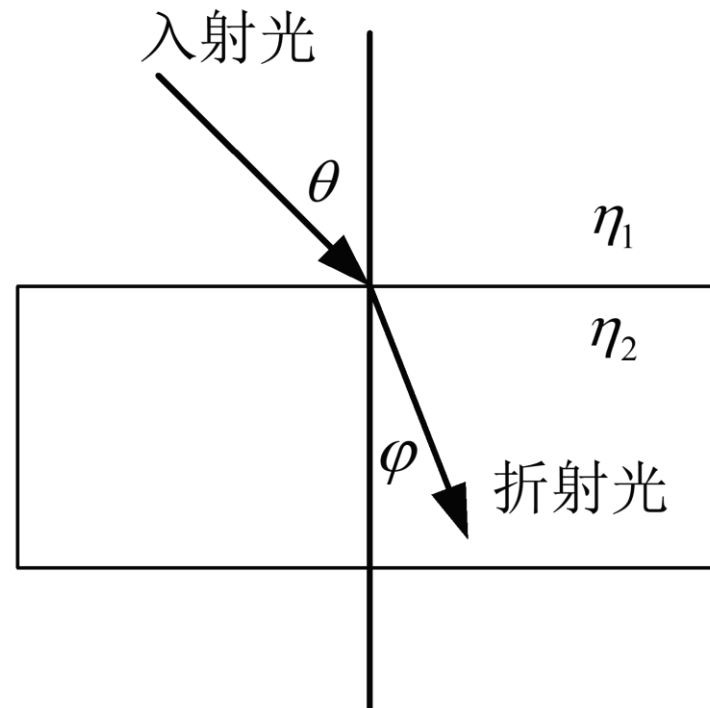
- 光的传播

- 反射定律：入射角等于反射角，而且反射光线、入射光线与法向量在同一平面上

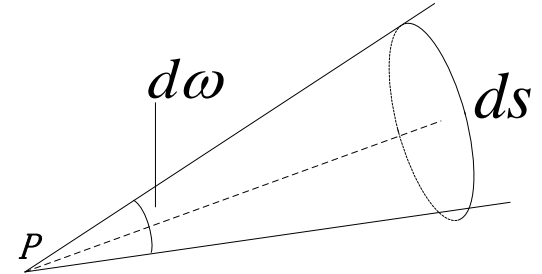


折射定律

- 折射定律：折射线在入射线与法线构成的平面上，折射角与入射角满足



光的度量方法



- 立体角 (Solid Angle):

- 衡量物体相对于某一视点 P 的视角大小: $d\omega = \frac{ds}{r^2}$
- 最大立体角为 (球体) ? 4π

- 光通量 (Luminous Flux):

- 单位时间内通过面元 dS 的光能量, 记为 dF

- 发光强度 (Radiance):

- 发光强度定义为: 单位立体角内的光通量, 记为 I

能量关系

- 在光的反射和折射现象中的能量分布：

$$I_i = I_d + I_s + I_t + I_v$$

- 下标为i,d,s,t,v的能量项分别表示为入射光强，漫反射光强，镜面反射光强，透射光强，吸收光强

入射光强 = 漫反射光强 + 镜面反射光强 + 透射光强 + 吸收光强

- 能量是守恒的

光照模型

- 光照模型 (lighting model 或 illumination model) 用于计算光的强度:
 - 局部光照明 (Local Lighting)
 - 关注物体直接受到光源影响所产生的光照效果
 - 全局光照明 (Global Lighting)
 - 关注光线在场景中其他物体表面发生折射、反射等作用后, 间接照射到物体表面上产生的影响

简单光照明模型

光照射到物体表面，主要发生：

- 反射
- 透射（对透明物体）
- 部分被吸收成热能

反射光，透射光决定了物体所呈现的颜色

简单光照明模型-环境光

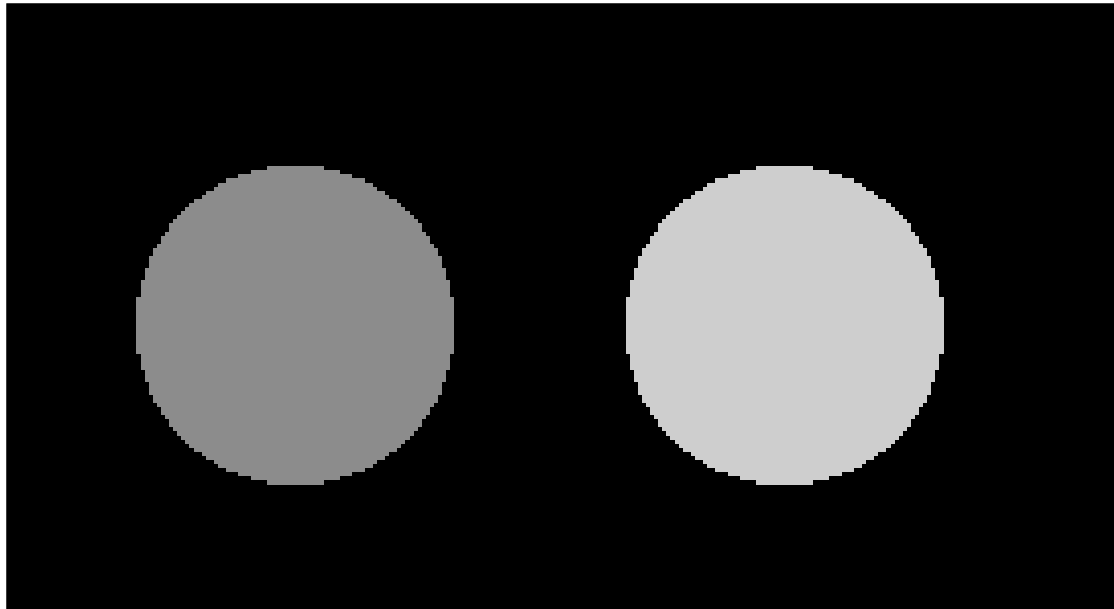
假定物体是不透明的（即无透射光）

- 环境光：在空间中近似均匀分布，即在任何位置、任何方向上强度一样，记为 I_a
- 环境光反射系数 K_a ：在分布均匀的环境光照射下，不同物体表面所呈现的亮度未必相同，因为它们的环境光反射系数不同。
- 光照明方程（仅含环境光）： $I_e = K_a I_a$

I_e 为物体表面所呈现的亮度。

简单光照明模型- 环境光例子

- 具有不同环境光反射系数的两个球

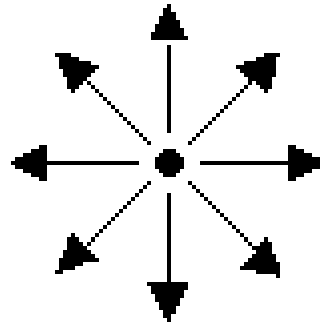


- 缺点：虽然不同的物体具有不同的亮度，但是同一物体的表面的亮度是一个恒定的值，没有明暗的自然过度。

简单光照明模型

考虑引入点光源。

- 点光源：几何形状为一个点，位于空间中的某个位置，向周围所有的方向上辐射等强度的光。记其亮度为 I_p



- 点光源的照射：在物体的不同部分其亮度也不同，亮度的大小依赖于物体的朝向及它与点光源之间的距离。

简单光照明模型：-漫反射角度余弦的推导



● 漫反射

— 粗糙、无光泽物体（如粉笔）表面对光的反射

— 光照明方程 $I_d = I_p K_d \cos \theta$ $\theta \in [0, \frac{\pi}{2}]$

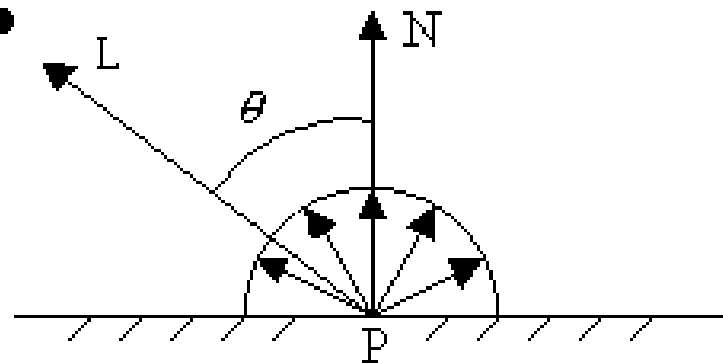
I_d 漫反射的亮度

I_p 点光源的亮度

K_d 漫反射系数

θ 入射角

点光源



漫反射光的强度只与入射角有关

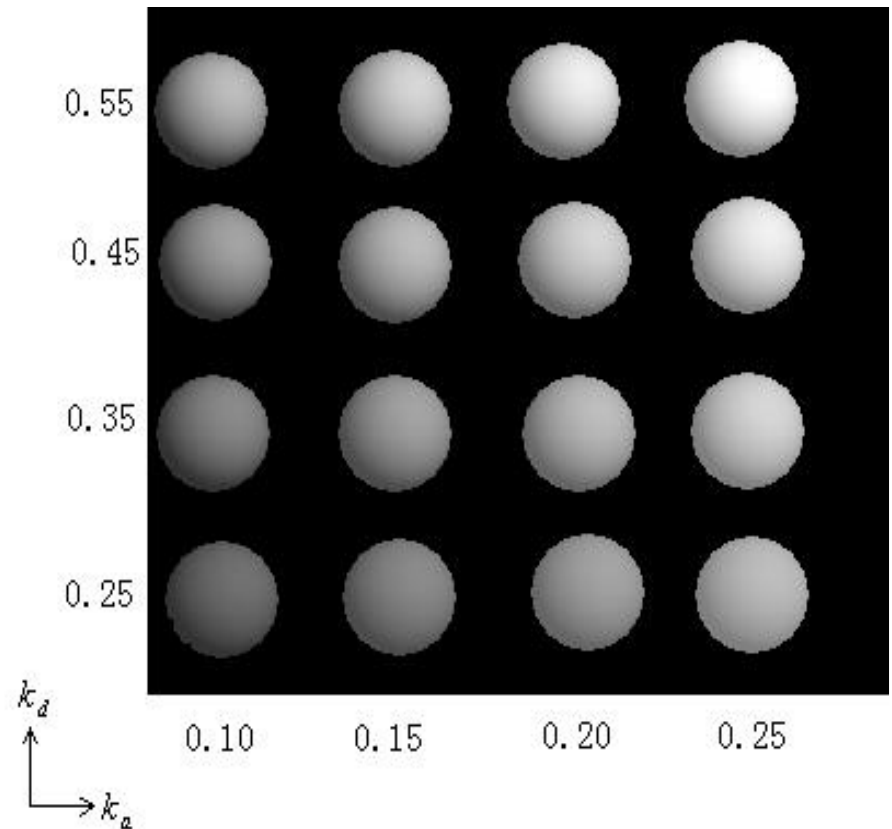
简单光照明模型-漫反射

- 将环境光与漫反射结合起来

$$I = I_e + I_d = I_a K_a + I_p K_d (L \cdot N)$$

一般取 $I_a = (0.02 \sim 0.2) I_d$

- 例子



简单光照明模型-漫反射

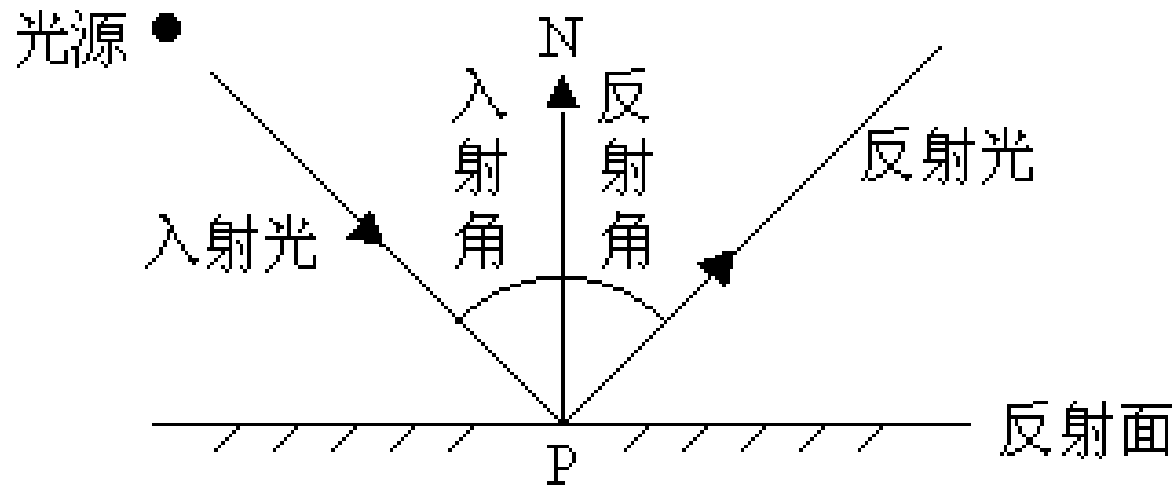
缺点：对于许多物体，使用上式计算其反射光是可行的，但对于大多数的物体，如擦亮的金属、光滑的塑料等是不适用的，原因是这些物体还会产生镜面发射。

简单光照明模型-镜面反射

- 镜面反射
 - 光滑物体（如金属或塑料）表面对光的反射
- 高光
 - 入射光在光滑物体表面形成的特别亮的区域

简单光照明模型-镜面反射

- 理想镜面反射

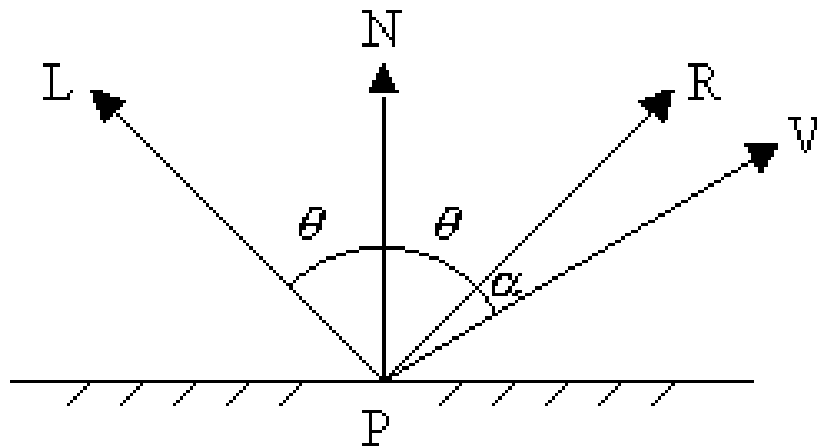


- 观察者只能在反射方向上才能看到反射光，偏离了该方向则看不到任何光。

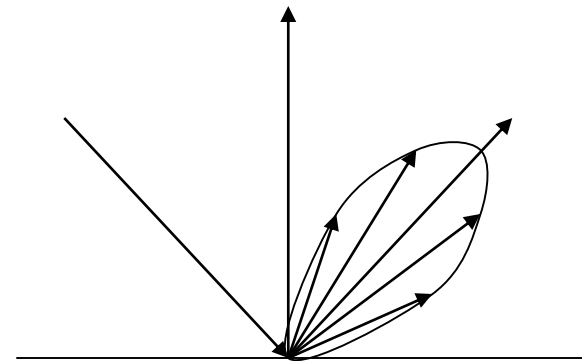
简单光照明模型-镜面反射

● 非理想镜面反射

$$I = I_p K_s \cos^n a$$



镜面

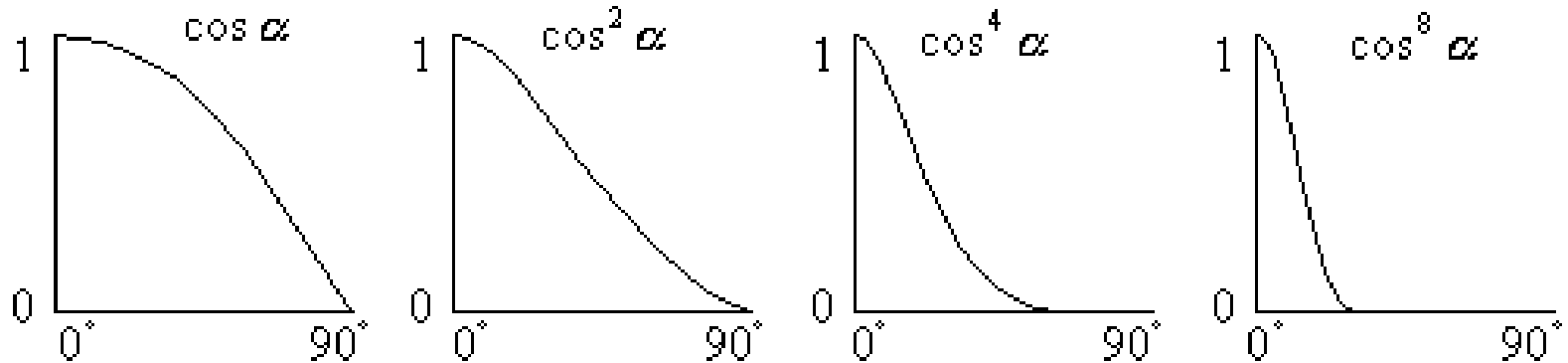


光滑平面

- P为物体表面上一点，L为从P指向光源的单位矢量，N为单位法矢量，R为反射单位矢量，V为从P指向视点的单位矢量

简单光照明模型-镜面反射

- 镜面反射 $I_s = I_p K_s \cos^n \alpha$ $I_s = I_p K_s (V \cdot R)^n$
- I_s 为镜面反射光强。 I_p 点光源的亮度
 - K_s 是与物体有关的镜面反射系数。 n 为镜面反射指数, n 越大, 则 I_s 随 α 的增大衰减的越快。
 - n 的取值与表面粗糙程度有关。
 - ① n 越大, 表面越平滑 (散射现象少, 稍一偏离, 明暗亮度急剧下降)
 - ② n 越小, 表面越毛糙 (散射现象严重)



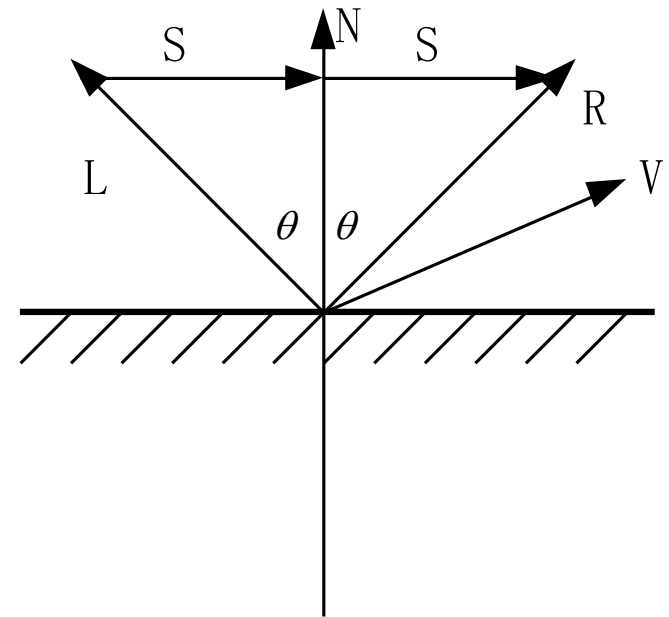
简单光照明模型-镜面反射

- 反射方向计算
- L 在 N 上的投影矢量为 $N \cos \theta$, 则 $S+L= N \cos \theta$

记矢量 $S= N \cos \theta - L$

则有 $R= N \cos \theta + S$

$$R = 2 N \cos \theta - L = 2 N (N \cdot L) - L$$



简单光照明模型-Phong光照明模型

- 简单光照明模型模拟物体表面对光的反射作用, 光源为点光源
- 反射作用分为
 - 物体间作用用环境光(Ambient Light)
 - 漫反射(Diffuse Reflection)
 - 镜面反射(Specular Reflection)

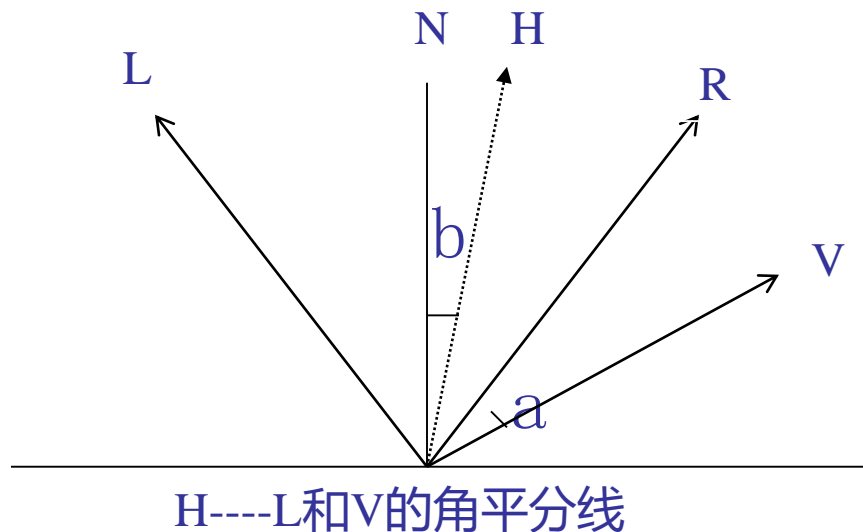
简单光照明模型-Phong光照明模型

- Phong光照明模型的综合表述：由物体表面上一点 P 反射到视点的光强 I 为环境光的反射光强 I_e 、理想漫反射光强 I_d 、和镜面反射光 I_s 的总和。

$$\begin{aligned} I &= I_e + I_d + I_s \\ &= I_a K_a + I_p [K_d (L \cdot N) + K_s (V \cdot R)^n] \end{aligned}$$

简单光照明模型-Phong光照明模型的实现

- 对物体表面上的每个点 P ，均需计算光线的反射方向。为了减少计算量，假设：
 - 光源在无穷远处， L 为常向量
 - 视点在无穷远处， V 为常向量
 - $(H \cdot N)$ 近似 $(R \cdot V)$ ， H 为 L 与 V 的平分向量

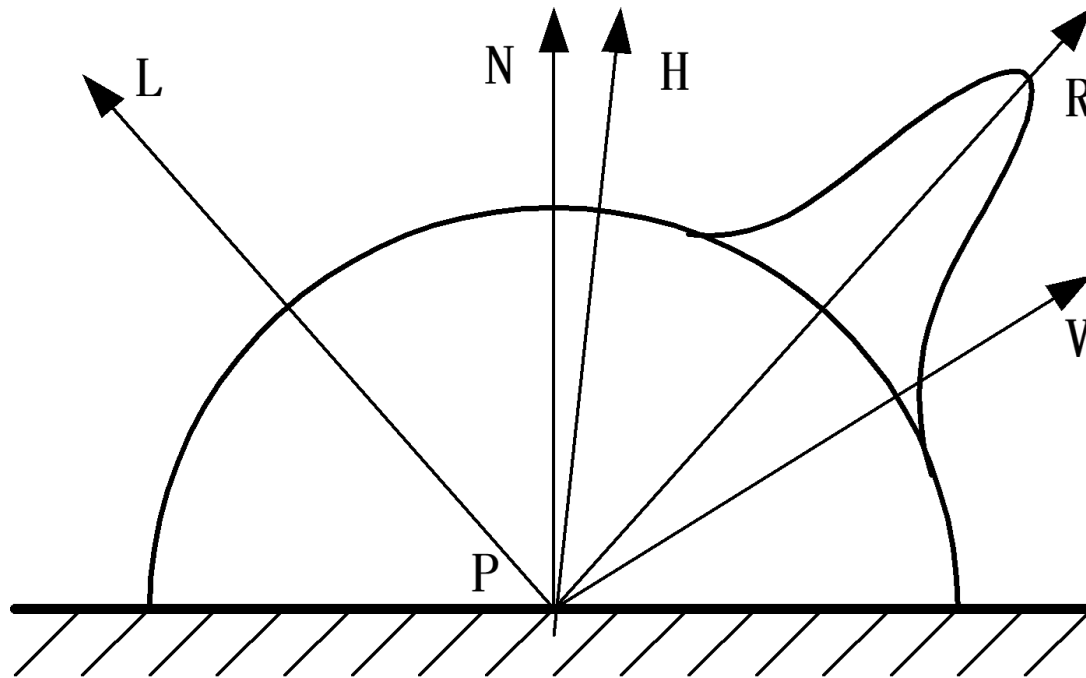


对所有的点总共只需计算一次 H 的值，节省了计算时间

简单光照明模型-Phong光照明模型



- Phong模型几何

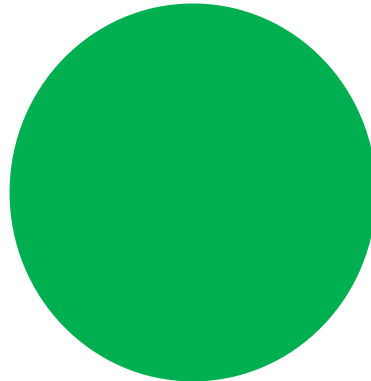


Phong 模型示例



漫反射光

+



环境光

+

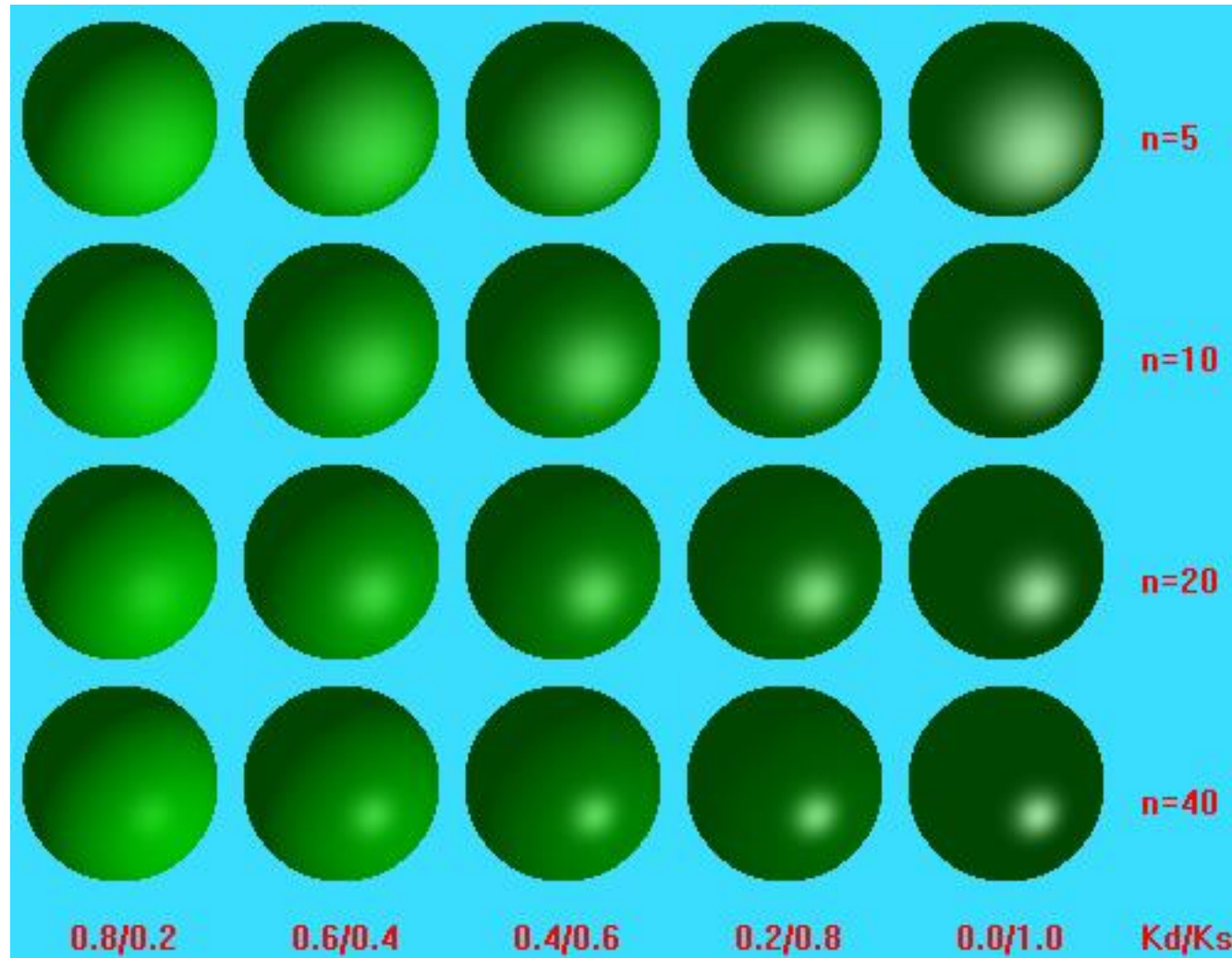


镜面反射光

=



Phong 模型示例



简单光照明模型-光的衰减

- 光的衰减

两个阶段：

- ① 从光源到物体表面的过程中的衰减
- ② 从物体表面到人眼过程中的衰减

总的效果：物体表面的亮度降低

- 光照明方程

- ① 有效衰减函数的加入
- ② 深度暗示技术的加入

简单光照明模型-光的衰减

- 光的衰减

- 光在光源到物体表面过程中的衰减
- 光强按 $1/d^2$ 进行衰减:

缺点: 当 d 很大时, 变化很小; 当 d 很小时, 变化很大。

- 衰减函数

$$f(d) = \min\left(\frac{1}{c_0 + c_1 d + c_2 d^2}, 1\right)$$

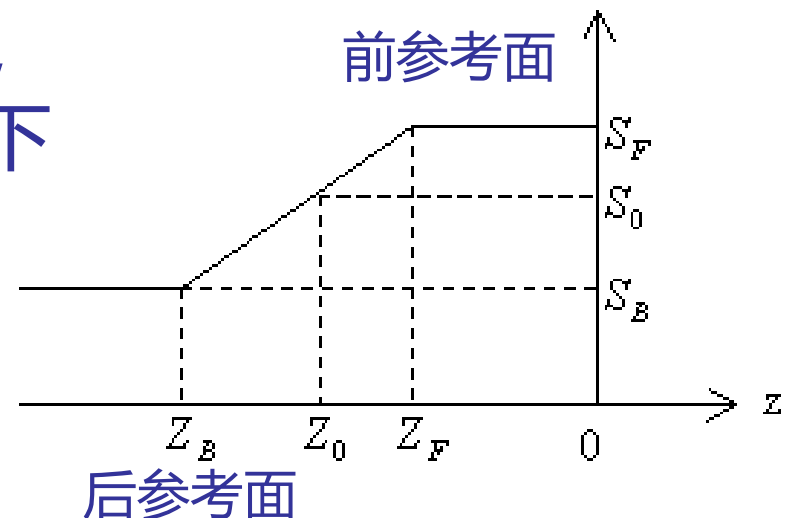
- 光照明方程

$$I = I_a K_a + f(d) I_p [K_d (L \cdot N) + K_s (V \cdot R)^n]$$

简单光照明模型-光的衰减

- 光在物体表面到人眼过程中的衰减
 - 深度暗示 (Depth Cueing) 技术: 最初用于线框图形的显示, 使距离远的点比近的点暗一些。经过改进, 这种技术同样适用于真实感图形显示。
 - 设前参考面 $Z=Z_f$, 后参考面 $Z=Z_b$; 其比例因子分别为 S 和 S_b (S 和 S_b 属于 $[0,1]$)。

给定物体上一点的深度值 Z_0 ,
该点对应的比例因子 S_0 按如下
方式确定



简单光照明模型-光的衰减

- 当 $Z_0 > Z_f$ 时, 取 $S_0 = S_f$
- 当 $Z_0 < Z_b$ 时, 取 $S_0 = S_b$
- 当 $Z_0 \in [Z_b, Z_f]$ 时, 取

$$S_0 = S_b + \frac{S_f - S_b}{Z_f - Z_b} (Z_0 - Z_b)$$

- 原亮度 I 按比例 S_0 与融和亮度 I_{dc} 混合, 目的是获得最终用于显示的亮度 I' , I_{dc} 由用户指定,

$$I' = S_0 I + (1 - S_0) I_{dc}$$

简单光照明模型-光的衰减

$$I' = S_0 I + (1 - S_0) I_{dc}$$

特例：取 $S_f=1, S_b=0, I_{dc}=0$

- 则当物体位于参考面之前时， $S_0=S_f=1, I' = I$ ，即亮度没有被衰减。
- 当物体位于后参考面之后时， $S_0=S_b=0, I' = I_{dc}=0$ ，即亮度衰减为0。
- 而当 Z_0 属于 $[Z_b, Z_f]$ 时， $I' = S_0 I$ ，亮度被部分衰减。由此可以产生较好的效果。

简单光照明模型-彩色场景的产生

- 产生彩色
 - 选择合适的颜色模型----RGB模型
 - 为颜色模型中的每一种基色建立光照明方程

$$\begin{cases} I_R = I_{aR}K_{aR} + f(d)I_{pR}[K_{dR}(L \cdot N) + K_{sR}(V \cdot R)^n] \\ I_G = I_{aG}K_{aG} + f(d)I_{pG}[K_{dG}(L \cdot N) + K_{sG}(V \cdot R)^n] \\ I_B = I_{aB}K_{aB} + f(d)I_{pB}[K_{dB}(L \cdot N) + K_{sB}(V \cdot R)^n] \end{cases}$$

简单光照明模型-彩色场景的产生

— 系数分解

$$\begin{bmatrix} K_{aR} \\ K_{aG} \\ K_{aB} \end{bmatrix} = K_a \begin{bmatrix} C_{dR} \\ C_{dG} \\ C_{dB} \end{bmatrix} \quad \begin{bmatrix} K_{dR} \\ K_{dG} \\ K_{dB} \end{bmatrix} = K_d \begin{bmatrix} C_{dR} \\ C_{dG} \\ C_{dB} \end{bmatrix} \quad \begin{bmatrix} K_{sR} \\ K_{sG} \\ K_{sB} \end{bmatrix} = K_s \begin{bmatrix} C_{sR} \\ C_{sG} \\ C_{sB} \end{bmatrix}$$

- 上述各等式中，右端的矢量用来控制表面的基本颜色，当选定了物体表面的颜色之后，它们就固定不变了。
- 用户通过调节 K_a , K_d , K_s 来改变表面的反射率。

简单光照明模型-彩色场景的产生

— 新的光照明方程

$$\begin{cases} I_R = K_a C_{dR} I_{aR} + f(d) I_{pR} [K_d C_{dR} (L \cdot N) + K_s C_{sR} (V \cdot R)^n] \\ I_G = K_a C_{dG} I_{aG} + f(d) I_{pG} [K_d C_{dG} (L \cdot N) + K_s C_{sG} (V \cdot R)^n] \\ I_B = K_a C_{dB} I_{aB} + f(d) I_{pB} [K_d C_{dB} (L \cdot N) + K_s C_{sB} (V \cdot R)^n] \end{cases}$$

— 统一表示 $\lambda = R, G, B$

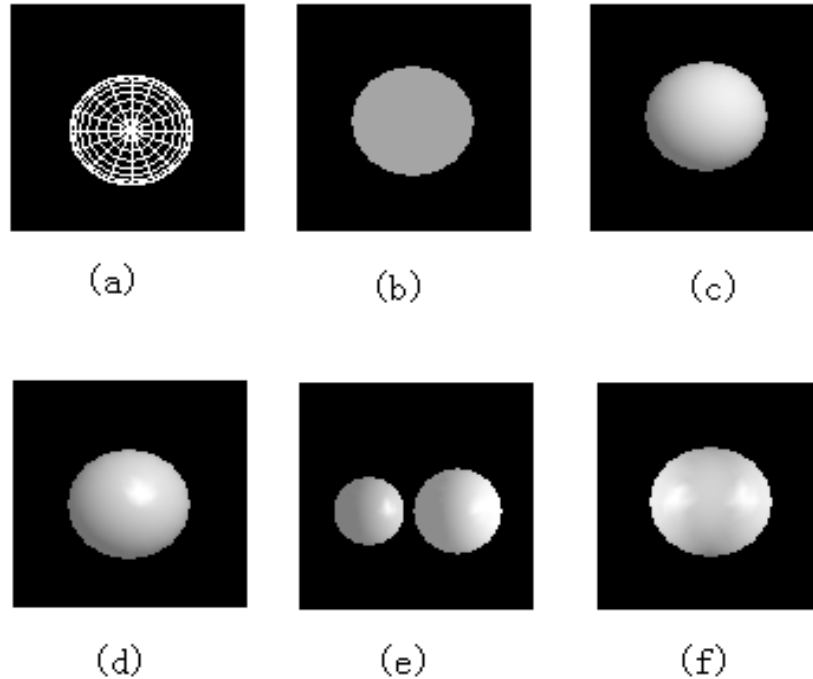
$$I_\lambda = K_a C_{d\lambda} I_{a\lambda} + f(d) I_{p\lambda} [K_d C_{d\lambda} (L \cdot N) + K_s C_{s\lambda} (V \cdot R)^n]$$

简单光照明模型-多个光源

- 采用多个光源
 - 采用m个光源的光照明方程

$$I_{\lambda} = K_a C_{d\lambda} I_{a\lambda} + \sum_{i=1}^m f(d_i) I_{p_i\lambda} [K_d C_{d\lambda} (L_i \cdot N) + K_s C_{s\lambda} (V \cdot R_i)^n]$$

简单光照明模型-多个光源



a图：线框图 b图：环境光

c图：增加漫反射 d图：增加镜面反射

e图：增加光的衰减 f图：两个点光源

Phong光照模型的不足

- Phong光照模型是真实感图形学中提出的第一个有影响的光照明模型
- 经验模型，Phong模型存在不足：
 - 显示出的物体象塑料，无质感变化
 - 没有考虑物体间相互反射光
 - 镜面反射颜色与材质无关
 - 镜面反射大入射角失真现象

多边形绘制方法

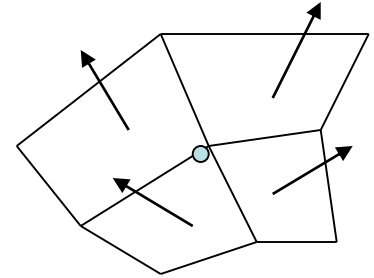
- 分类：均匀着色与光滑着色
- 均匀着色：任取多边形上一点，利用光照明方程计算出它的颜色，用这个颜色填充整个多边形

适用场合：

- 1) 光源在无穷远处；
- 2) 视点在无穷远处；
- 3) 多边形是物体表面的精确表示；

多边形绘制方法

- 缺点：产生的图形效果不好。
- 如图：相邻两个多边形的法向不同，计算出来的颜色也不同，因此造成整个物体表面的颜色过渡不光滑。
- 如何解决？
- 光滑着色，亦称插值着色
 - Gouraud着色方法
 - Phong着色方法



Gouraud着色方法

- Gouraud于1971年提出，又被称Gouraud明暗处理
- 基本思想：在每个多边形顶点处计算颜色，然后在各个多边形内部进行线性插值，得到多边形内部各点颜色。即它是一种颜色插值着色方法。
- 注意：Gouraud着色方法并不是孤立的处理单个多边形，而是将构成一个物体表面的所有多边形（多边形网格）作为一个整体来处理。

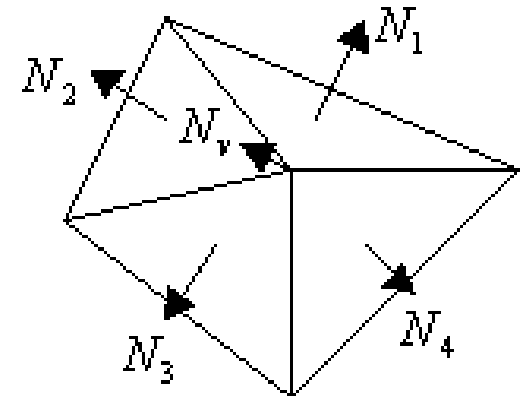
Gourand着色方法

- 对多边形网格中的每一个多边形，Gourand着色处理分为如下四个步骤：
 - 步骤
 - 1、计算多边形的单位法矢量
 - 2、计算多边形顶点的单位法矢量

Gouraud着色方法-顶点法向计算

- 与某个顶点相邻的所有多边形的法向平均值近似作为该顶点的近似法向量

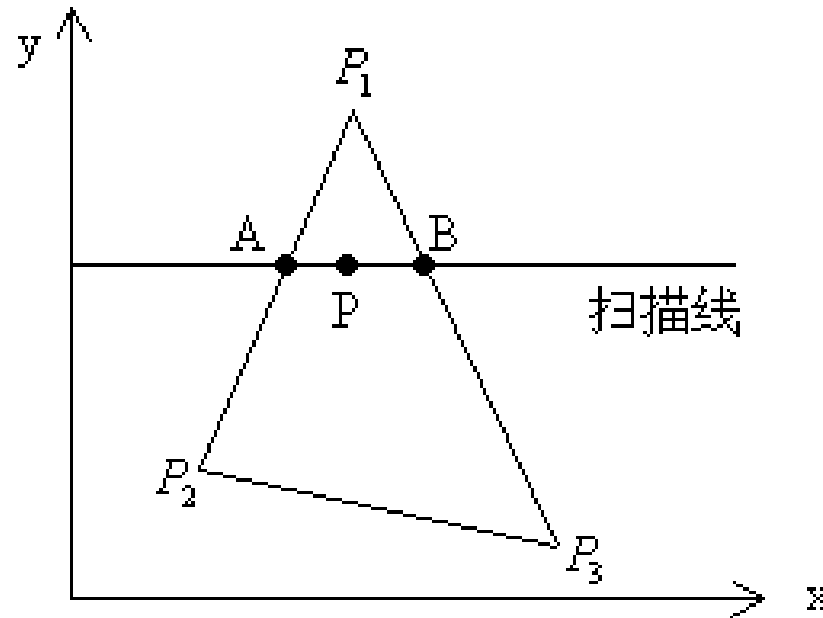
$$N_v = \frac{\sum_{i=1}^n N_i}{\left| \sum_{i=1}^n N_i \right|}$$



- 计算出的平均法向一般与该多边形物体近似曲面的切平面比较接近

Gourand着色方法

3. 利用光照明方程计算顶点光强（颜色）
4. 对多边形顶点光强（颜色）进行双线性插值，获得多边形内部各点的光强（颜色）



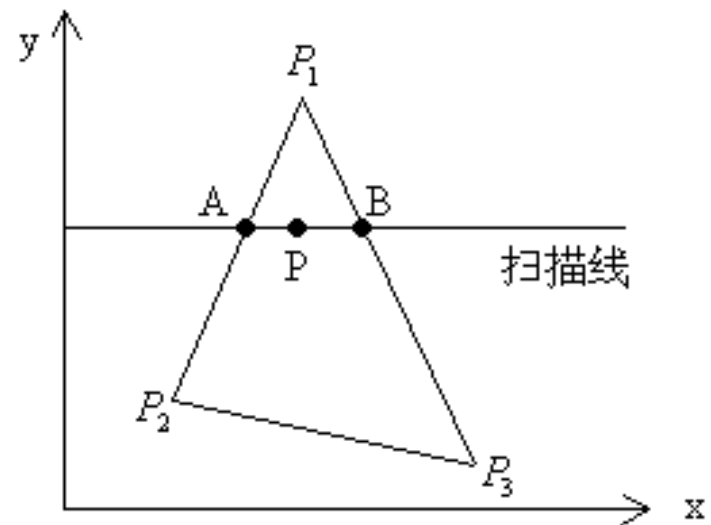
Gourand着色方法-光强插值

- 双线性光强插值：假设待绘制的三角形投影为 $P_1P_2P_3$ ， P_i 的坐标为 (x_i, y_i) ， $i=1,2,3$ ；
- 一条扫描线与三角形的两条边分别交于 $A(x_A, y_A)$ ， $B(x_B, y_B)$ 两点。 $P(x, y)$ 是 AB 上的一点。
- A 点的颜色 I_A 由 P_1 、 P_2 点的颜色 I_1 、 I_2 线性插值得到

$$I_A = \frac{y_A - y_2}{y_1 - y_2} I_1 + \frac{y_1 - y_A}{y_1 - y_2} I_2$$

$$I_B = \frac{y_B - y_3}{y_1 - y_3} I_1 + \frac{y_1 - y_B}{y_1 - y_3} I_3$$

$$I_P = \frac{x_B - x}{x_B - x_A} I_A + \frac{x - x_A}{x_B - x_A} I_B$$



Gourand着色方法-增量算法

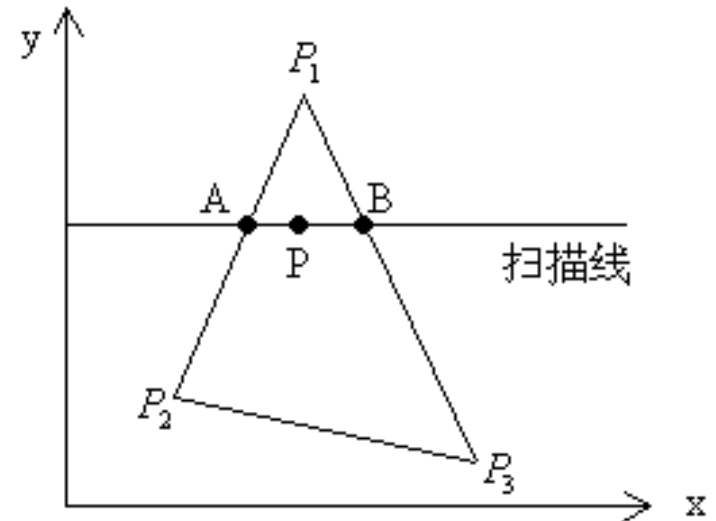
采用增量算法可以加速计算。

① 当扫描线 y 递增一个单位变为 $y+1$ 时, I_A 、 I_B 的增量分别为 ΔI_A , ΔI_B , 即

$$I_{A,y+1} = I_{A,y} + \Delta I_A \quad I_{B,y+1} = I_{B,y} + \Delta I_B$$

其中:

$$\Delta I_A = \frac{I_1 - I_2}{y_1 - y_2} \quad \Delta I_B = \frac{I_1 - I_3}{y_1 - y_3}$$



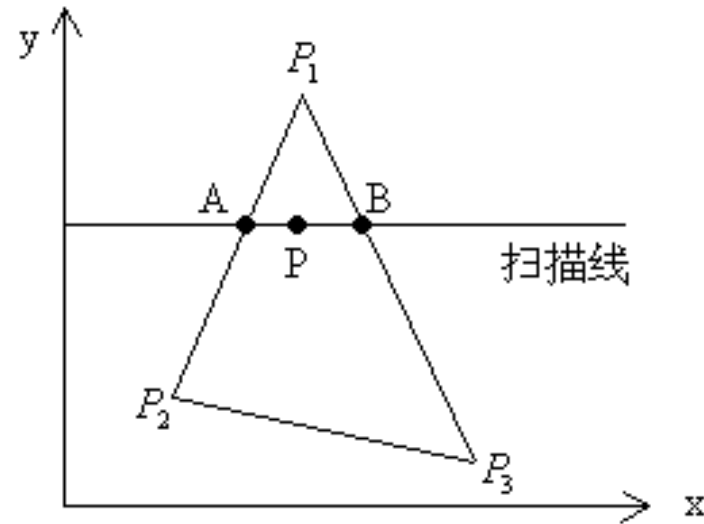
Gourand着色方法-增量算法

② 当x递增一个单位时, I_P 的增量为 ΔI_P 即

$$I_{P,x+1} = I_{P,x} + \Delta I_P$$

其中:

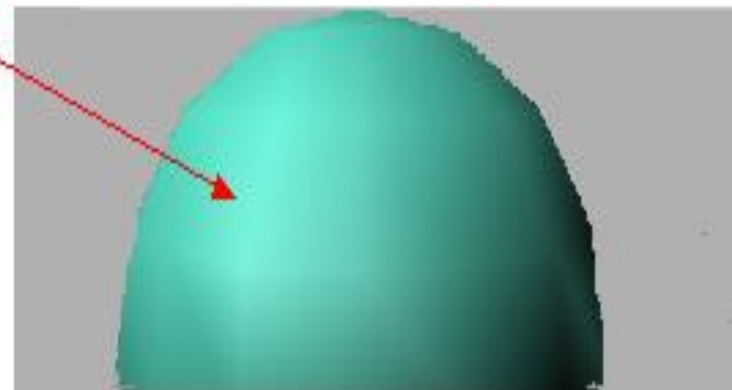
$$\Delta I_P = \frac{I_B - I_A}{x_B - x_A}$$



Gourand着色方法

- 优点：能有效的显示漫反射曲面，计算量小
- 缺点：
 - ① 高光有时会异常
 - ② 当对曲面采用不同的多边形进行分割时会产生不同的效果。
 - ③ Gouraud明暗处理会造成表面上出现过亮或过暗的条纹，称为马赫带（Mach_band）效应
- 改进 - Phong提出双线性法向插值，以时间为代价，解决高光问题

由于线性光强度插值产生的马赫带效应



Phong着色方法

- 基本思想：通过对多边形顶点的法矢量进行插值，获得其内部各点的法矢量，又称为法向插值着色方法，主要步骤：
 - ① 计算多边形单位法矢量
 - ② 计算多边形顶点单位法矢量
 - ③ 对多边形顶点法矢量进行双线性插值，获得内部各点的法矢量
 - ④ 利用光照明方程计算多边形内部各点颜色

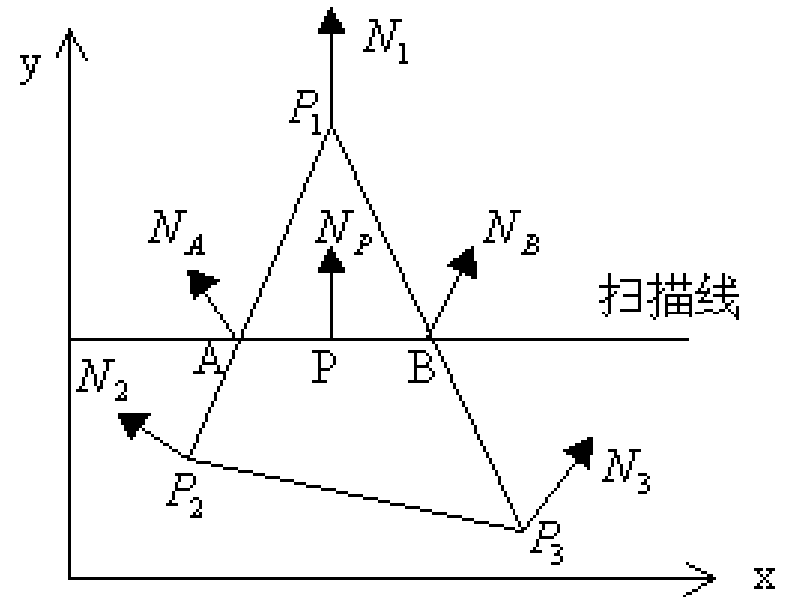
Phong着色方法-法向插值

N_A 由 N_1 、 N_2 线性插值得到:

$$N_A = \frac{y_A - y_2}{y_1 - y_2} N_1 + \frac{y_1 - y_A}{y_1 - y_2} N_2$$

$$N_B = \frac{y_B - y_3}{y_1 - y_3} N_1 + \frac{y_1 - y_B}{y_1 - y_3} N_3$$

$$N_P = \frac{x_B - x}{x_B - x_A} N_A + \frac{x - x_A}{x_B - x_A} N_B$$



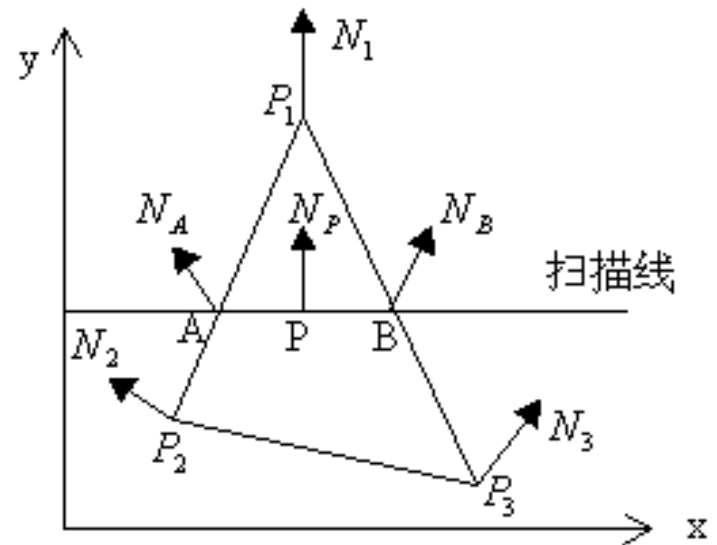
Phong着色方法-增量算法

采用增量算法可以加速计算。

① 当扫描线 y 递增一个单位变为 $y+1$ 时, N_A 、 N_B 的增量分别为 ΔN_A , ΔN_B , 即

$$N_{A,y+1} = N_{A,y} + \Delta N_A \quad N_{B,y+1} = N_{B,y} + \Delta N_B$$

$$\Delta N_A = \frac{N_1 - N_2}{y_1 - y_2} \quad \Delta N_B = \frac{N_1 - N_3}{y_1 - y_3}$$



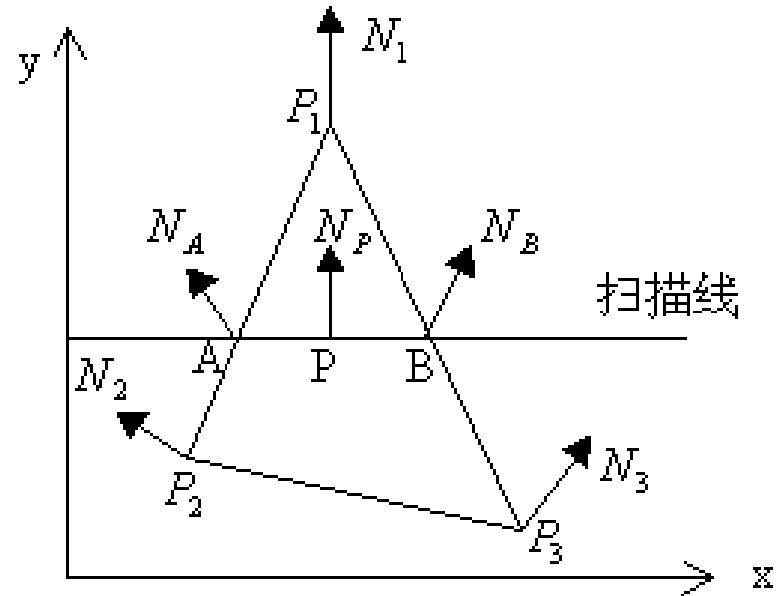
Phong着色方法-增量算法

② 当x递增一个单位时, I_P 的增量为 ΔI_P 即

$$N_{P,x+1} = N_{P,x} + \Delta N_P$$

其中:

$$\Delta N_P = \frac{N_B - N_A}{x_B - x_A}$$



Phong着色方法

优点:

Phong着色方法绘制的图形比Gouraud方法更真实, 体现在两个方面: 高光区域的扩散, 产生正确的高光区域

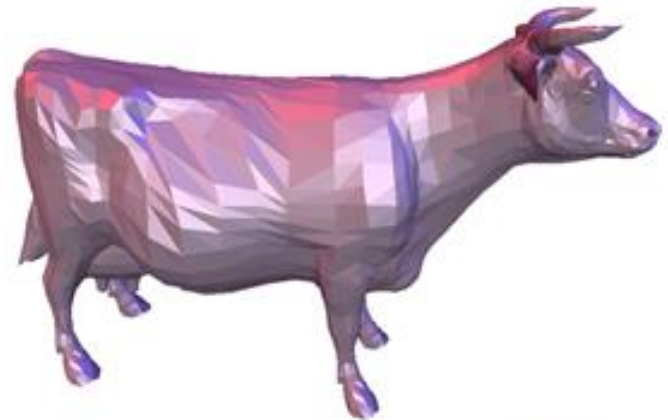
缺点:

- ① Phong着色方法计算量远大于Gouraud着色方法
- ② 在处理某些多边形分割的曲面时, Phong算法还不如Gouraud算法好。

增量式模型示例

牛的三角网格模型

- 用简单光照明模型显示



- 用增量式光照明模型显示



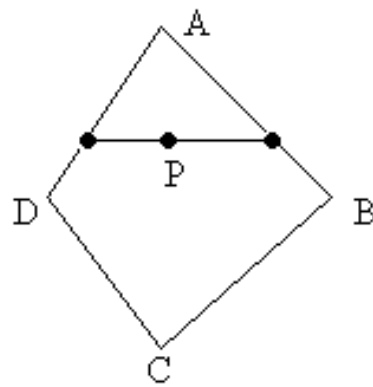
插值多边形绘制方法

- 着色方法存在的问题
 - 不光滑的物体轮廓：物体边缘轮廓是折线段而非光滑曲线



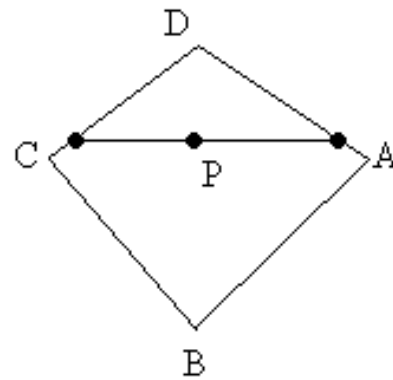
插值着色多边形绘制方法

– 方向依赖性、不连续性等



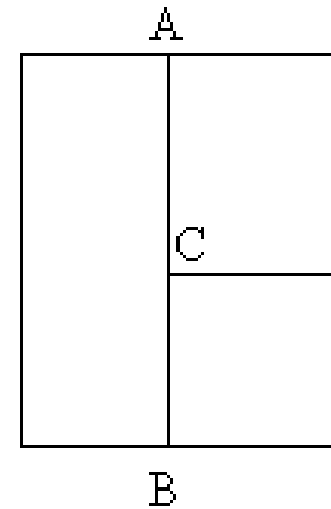
(a)

依赖ADB三点



(b)

依赖DCA三点



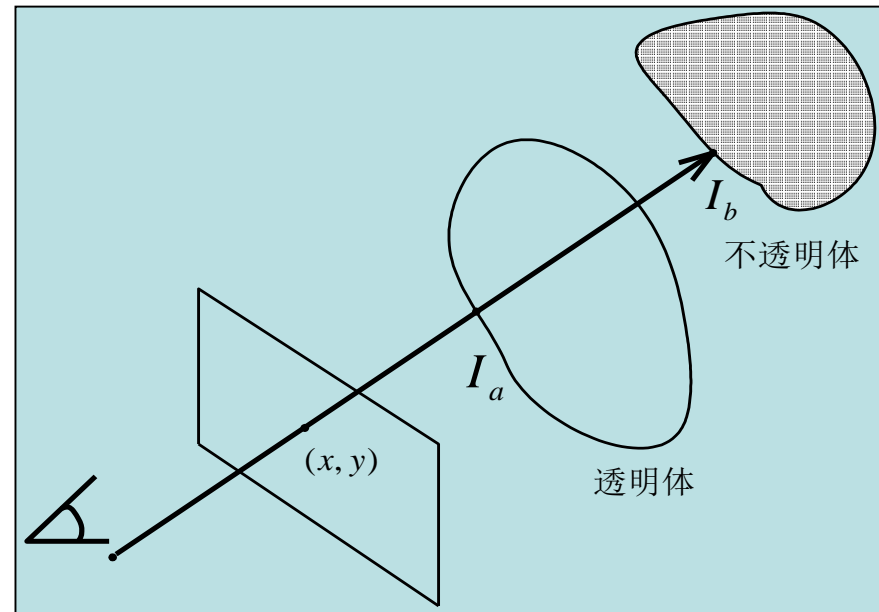
公共顶点处颜色不连续

透明

- 现实世界中有许多透明物体，如玻璃等。透过透明物体，可以观察到其后面的景物。如何模拟这种透明效果呢？
- 模拟透明的最简单的方法是忽略光线在穿过透明体时所发生的折射。
- 虽然这种模拟方法产生的结果不真实，但在许多场合往往非常有用。例如：我们有时希望能够看到透过某透明物体观察其后面的景物，而又不希望景物因为折射而发生变形。

透明效果的简单模拟

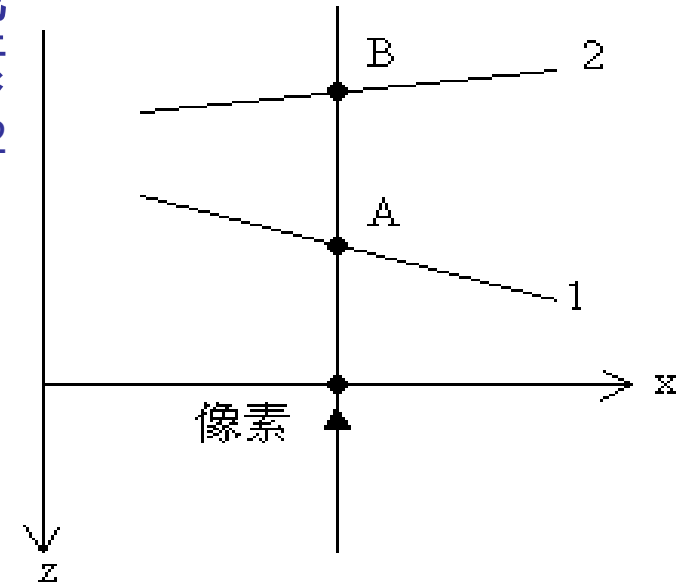
- 不考虑透明体对光的折射以及透明物体本身的厚度
- 光通过物体表面不改变方向
- 产生简单透明效果的方法
 - ① 插值透明方法
 - ② 过滤透明方法



简单透明-插值透明

- 假设：多边形1是透明的，它位于观察者与不透明的多边形2之间。像素的颜色 I_λ 由A、B两点的颜色 $I_{\lambda 1}$ 和 $I_{\lambda 2}$ 插值产生，即

$$I_\lambda = (1 - K_{t_1})I_{\lambda 1} + K_{t_1}I_{\lambda 2}$$



其中 K_{t_1} 是多边形1的透射系数。

K_{t_1} 范围 (0, 1)

$K_{t_1} = 0$ 表示多边形完全不透明，所以 $I_\lambda = I_{\lambda 1}$

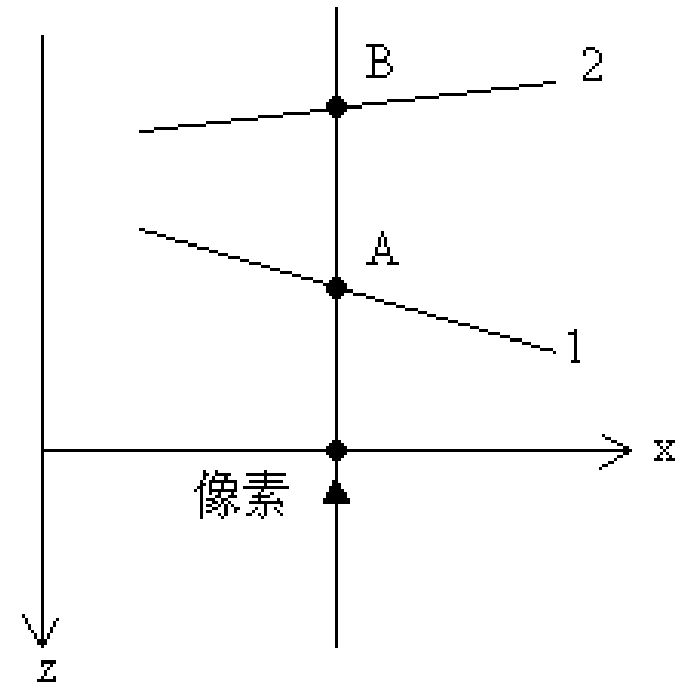
$K_{t_1} = 1$ 表示多边形完全透明，所以 $I_\lambda = I_{\lambda 2}$

简单透明-插值透明

为了产生逼真的效果，通常只对两个多边形表面颜色的**环境光分量**和**漫反射分量**采用

$$I_{\lambda} = (1 - K_{t_1})I_{\lambda 1} + K_{t_1}I_{\lambda 2}$$

进行计算，得到的结果再加上多边形1的镜面反射分量作为像素的颜色值。



简单透明-过滤透明

- 过滤透明方法将透明物体看作一个过滤器，有选择的允许某些光透过而屏蔽了其余的光。对右图有：

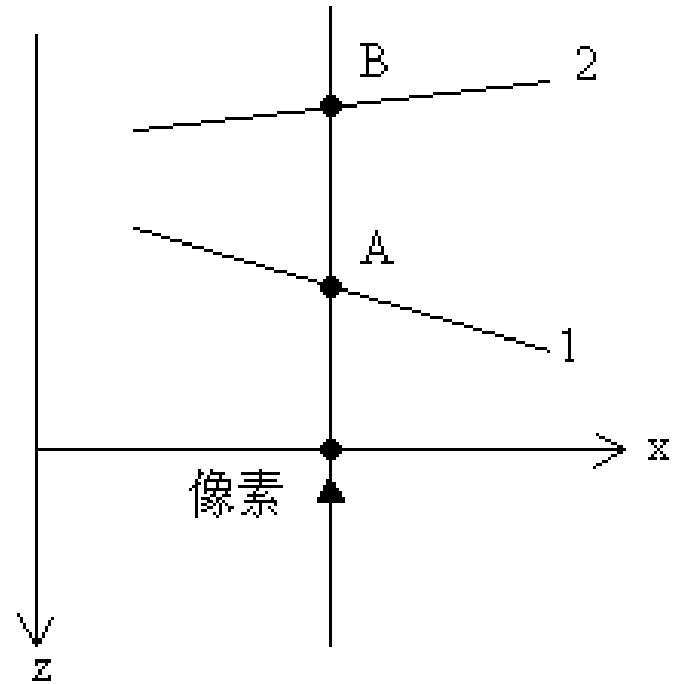
$$I_{\lambda} = I_{\lambda 1} + K_{t1} C_{t\lambda} I_{\lambda 2}$$

其中 K_{t1} 仍是多边形1的透射系数，但不再局限于(0~1)。

K_{t1} 越大，多边形2的颜色透过来的越多。

$C_{t\lambda}$ 对不同的颜色各不相同。

$C_{t\lambda} = 0$ 表示某种颜色的光不能透过多边形1。



简单透明

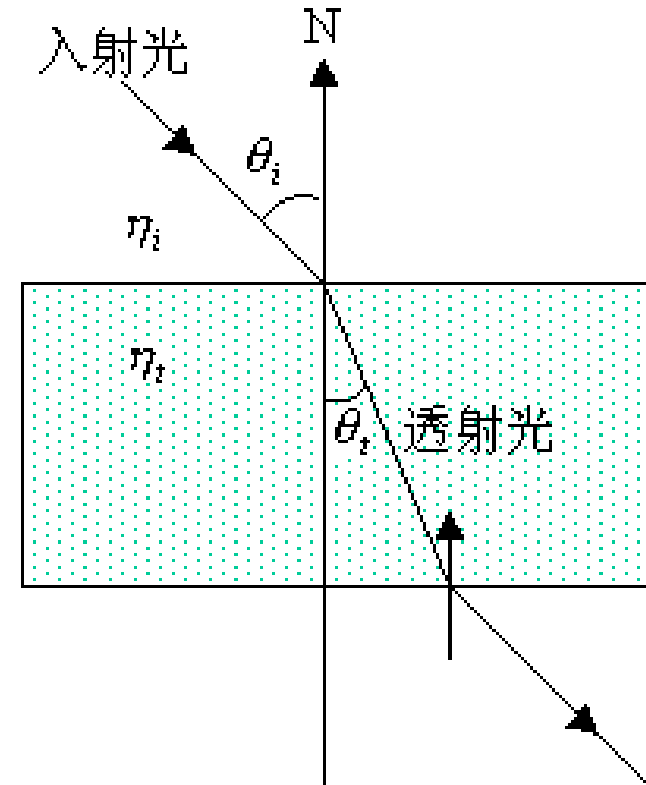
- 无论采用插值透明方法还是采用过滤透明方法，当多边形1之前还有其它的透明多边形时， I_1 都要递归计算。
- 简单透明比较容易结合到多边形绘制算法中。

考虑折射的透明

- 折射定律

$$\frac{\sin \theta_i}{\sin \theta_t} = \frac{\eta_t}{\eta_i}$$

- 其中： η_i , η_t 分别是入射光线在空气，物体中的折射率， θ_i 、 θ_t 分别是入射角和折射角



考虑折射的透明-透射矢量的计算

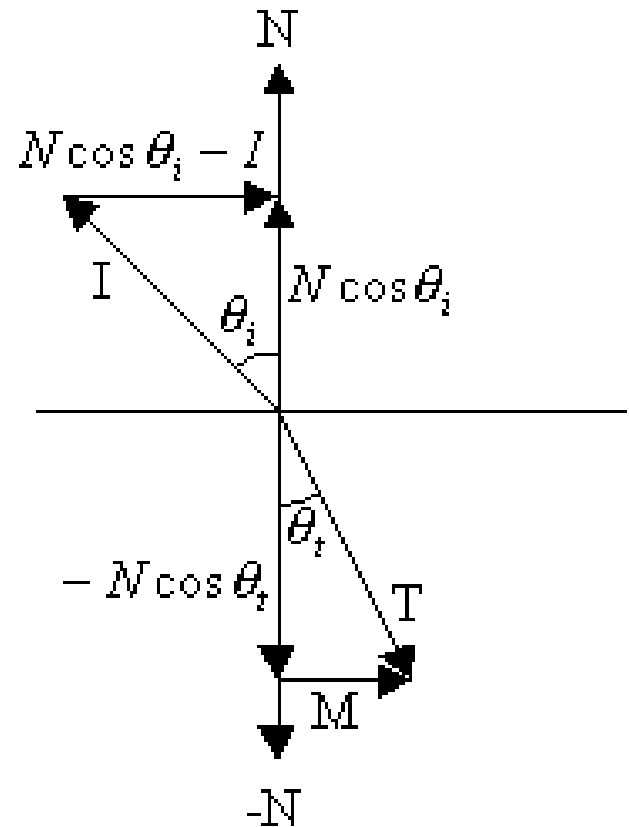
设单位入射光矢量为 I （方向与光线的入射方向相反），单位法矢量为 N ，单位透射光矢量为 T ，则

$$T = M - N \cos \theta_t$$

$$\frac{\sin \theta_i}{\sin \theta_t} = \frac{\eta_t}{\eta_i} \Rightarrow \frac{|N \cos \theta_i - I|}{|M|} = \frac{\eta_t}{\eta_i}$$

$$\text{令 } \eta = \frac{\eta_i}{\eta_t} \Rightarrow |M| = \eta |N \cos \theta_i - I|$$

$$T = (\eta \cos \theta_i - \cos \theta_t) N - \eta I$$



考虑折射的透明

- 当光线从高密度介质向低密度介质时, $n_i > n_t$, 即 $\theta_t > \theta_i$.
 - 如果入射角不断增大, 到一定的程度, 折射角 $\theta_t = 90^\circ$ 度, 此时透射光线沿着平行于分界面的方向传播, 称此时的 θ_i 为临界角度, 记为 θ_c 。
 - 当 $\theta_i > \theta_c$ 时, 发生全反射, 透射与反射光合二为一。
- 如何产生带有折射的透明效果呢?

光透射模型的研究

- 早期简单透射现象的模拟
- 1980年, Whitted光透射模型, 首次考虑了光线的折射现象
- 1983年, 在Whitted的基础上, Hall光透射模型, 考虑了漫透射和规则透射光

整体光照明模型

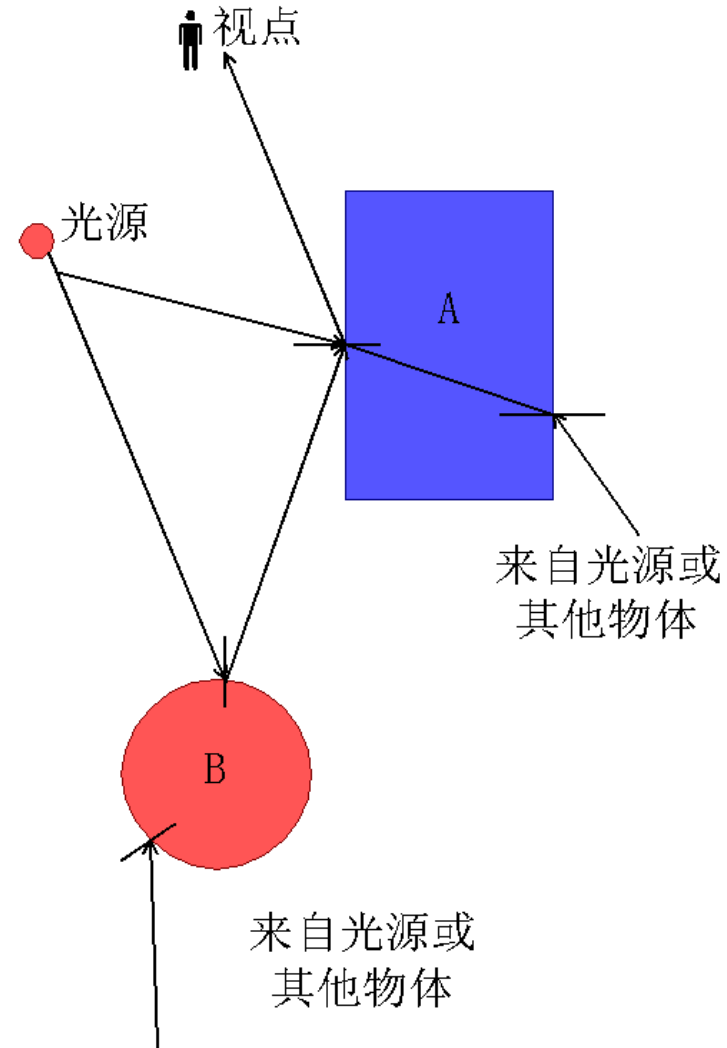
- 简单光照模型（亦称局部光照模型）不考虑周围环境对当前景物表面的光照明影响，忽略了光在环境景物之间的传递，很难表现自然界复杂场景的高质量真实感图形。
- 为了增加图形的真实感，必须考虑环境的漫射、镜面反射和规则投射对景物表面产生的整体照明效果。

整体光照明模型

- 物体表面入射光的构成
 - ① 光源直接照射
 - ② 其它物体的反射光
 - ③ 透射光
- 局部光照明模型仅考虑了 (1)

整体光照明模型

- 例如：从视点观察到的物体A表面的亮度来源于三方面的贡献：
 - 光源直接照射到A的表面，然后被反射到人眼中的光产生的。
 - 光源或其它物体的光经A物体折射到人眼中的光产生的。
 - 物体B的表面将光反射到物体A的表面，再经物体A的表面反射到人眼中产生的。
- 局部光照明模型仅考虑了 (1)



Whitted光照模型

Whitted光照模型基于如下假设：物体表面向视点方向V辐射的光亮度 I_λ 由三部分组成：

- ① 光源直接照射引起的反射光亮度 I_r 。
- ② 来自V的镜面反射方向R的其它物体反射或折射来的光的亮度 $I_{s\lambda}$ 。
- ③ 来自V的透射方向T的其它物体反射或折射来的光的亮度 $I_{t\lambda}$

Witted光照模型

*Witted*光照模型: $I_\lambda = I_\lambda + K_s I_{s\lambda} + K_t I_{t\lambda}$

或

- $I_{s\lambda}$ 为镜面反射方向的入射光强度; K_s 为镜面反射系数, 为0~1之间的一个常数
- $I_{t\lambda}$ 为折射方向光强, K_t 为透射系数, 是0 ~ 1之间的常数
- I_λ 的计算可采用Phong模型

因此, 关键是 I_s 和 I_t 的计算。如何计算呢?

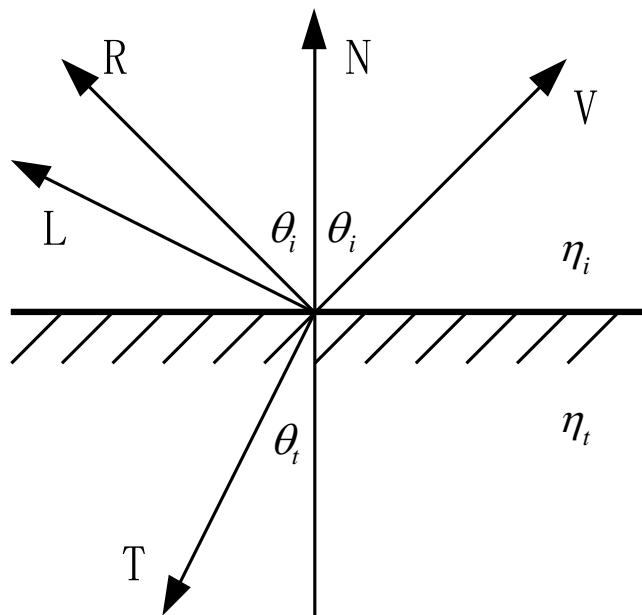
Witted光照模型-反射、折射方向计算



- 已知视线方向 V ，求其反射方向 R 与折射方向 T (N 是表面的法向方向)
- 视线 V 的反射方向 R (P.25)
$$R = 2N(N \bullet V) - V$$
- 折射方向 T (P.69)

$$\text{令 } \eta = \frac{\eta_i}{\eta_t}$$

$$T = (\eta \cos \theta_i - \cos \theta_t)N - \eta I$$



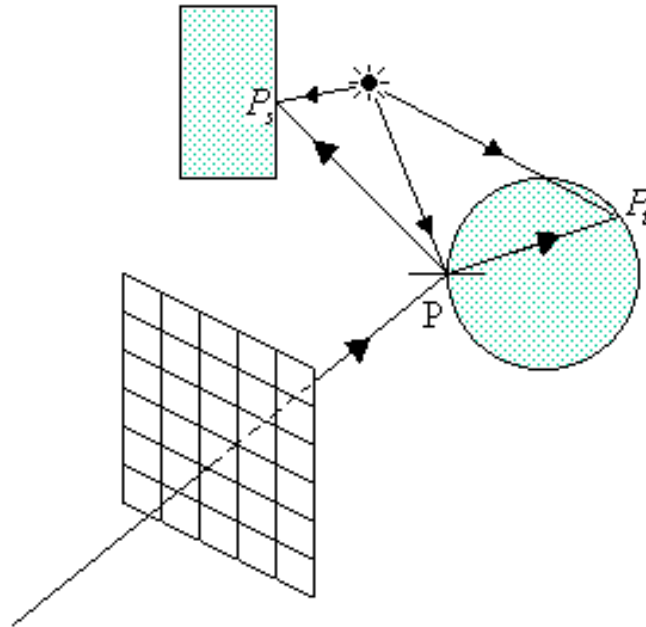
光线跟踪算法的基本原理

- 自然界中光线的传播过程

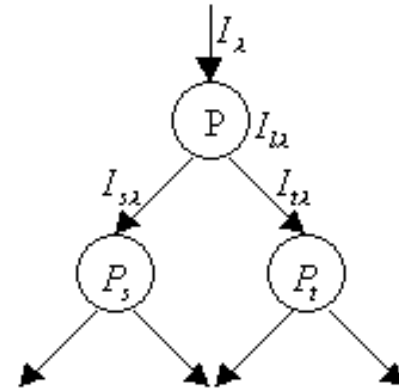


- 光线跟踪过程----光线传播的逆过程

光线跟踪算法的基本原理



(a)



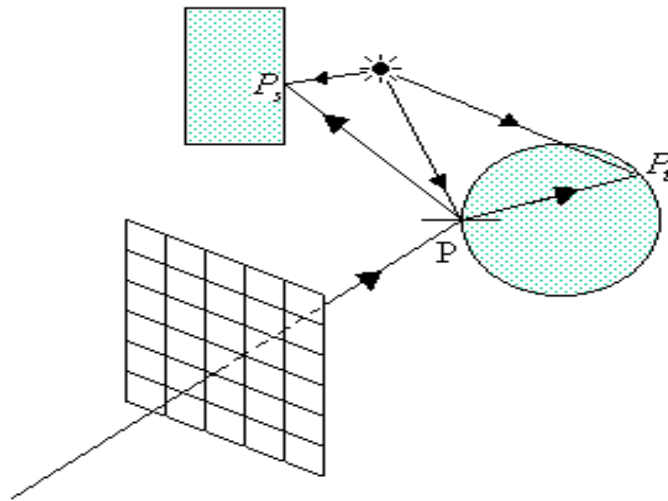
$$I_{\lambda} = I_{\lambda} + K_s C_{s,\lambda} I_{s,\lambda} + K_{t,\lambda} C_{t,\lambda} I_{t,\lambda}$$

(b)

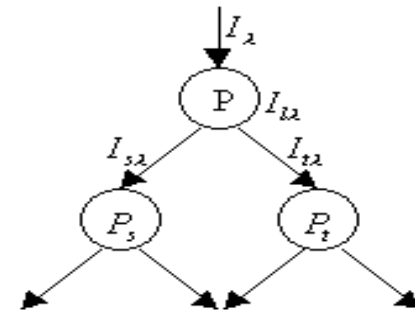
- 从视点向每个像素发出一条光线，它与场景中的一些物体表面相交，最近的交点即为可见点，记为P，像素的亮度即由P点的亮度确定。
- 由Whitted光照模型可知，P点的亮度由三部分组成：其中 I_{λ} 可以直接由局部光照模型计算得到。

光线跟踪算法的基本原理

- 为了求 $I_{s\lambda}$ 和 $I_{t\lambda}$ ，从P点发出反射光线和透射光线，它分别交场景中的物体表面于 P_s 和 P_t ， P_s 和 P_t 点的亮度即分别为 $I_{s\lambda}$ 和 $I_{t\lambda}$ ，将它们代入Whitted模型即可。
- 但 $I_{s\lambda}$ 和 $I_{t\lambda}$ 同样由Whitted模型确定，即Whitted模型是一个递归式，从而计算 $I_{s\lambda}$ 和 $I_{t\lambda}$ 需重复以上的计算过程：计算局部光亮度、发出反射光线与透射光线。可用一棵光线树来表示



(a)



$$I_{\lambda} = I_{\lambda 2} + K_s C_{s\lambda} I_{s\lambda} + K_{t\lambda} C_{t\lambda} I_{t\lambda}$$

(b)

光线跟踪算法的基本原理

递归终止条件:

- 光线不与场景中的任何物体相交
- 被跟踪的光线达到了给定的层次
- 由于 K_s 和 K_t 都小于1.0, 当光线经过反射和折射后, 其亮度会衰减。因此可以预先设置一个阈值, 在进行光线跟踪时, 若被跟踪光线对像素亮度的贡献小于这个阈值, 便停止跟踪。

光线跟踪算法-算法描述

设置视点，投影平面以及窗口的参数；

For （窗口内的每一条扫描线）

for （扫描线上的每一个像素）

{ 确定从视点指向像素中心的光线ray；

像素的颜色=RayTracing(ray,1)；

}

光线跟踪算法描述

```
Color RayTracing(Ray ray, int depth)
{求ray与物体表面最近的交点P;
  if (有交点)
  { 用局部光照明模型计算P点的Ic;
    color = Ic;
    if (depth<给定的最大跟踪层次)
    { 计算ray的反射光线;
      Is=RayTracing(反射光线, depth+1) ;
      if (物体是透明的)
      { 计算ray的透射光线;
        It=RayTracing(透射光线, depth+1);
      }
      color = Ic + Is + It ;
    }
  } else color = black;
  return color ;
}
```

光线跟踪算法

优点:

能够方便的产生阴影，模拟镜面反射与折射现象。

缺点:

计算量大，每一条光线都要与场景中的物体进行求交、计算光照模型等。