

生成真实感图形的四个步骤

（1）场景造型和几何建模

在计算机上进行场景造型。也就是采用数学的方法建立所需要的三维场景的几何描述，并将其输入到计算机中。

（2）透视变换

将三维几何描述转换为二维透视图，即进行场景变换和透视变换。

（3）可见面判定

确定场景中的所有可见面，也就是消隐处理，将视域之外或被其他景物遮挡的不可见面消除。

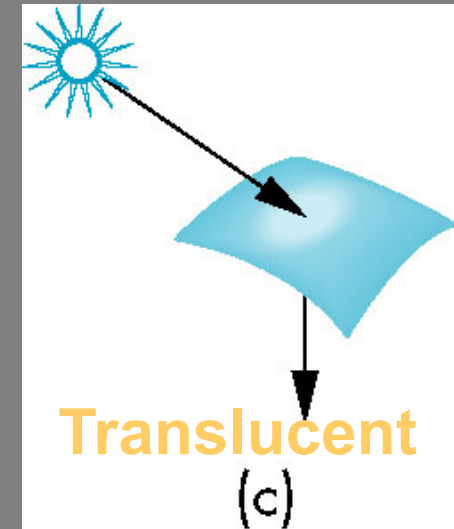
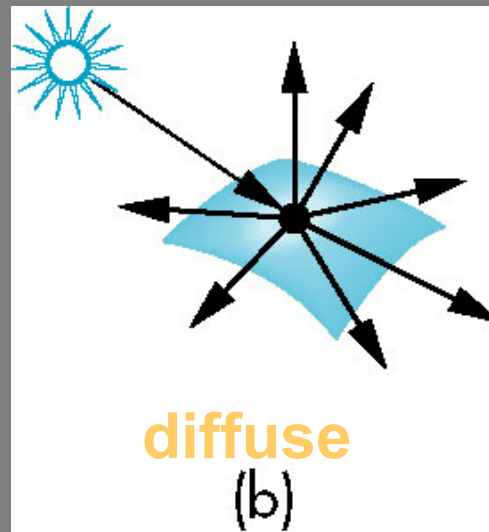
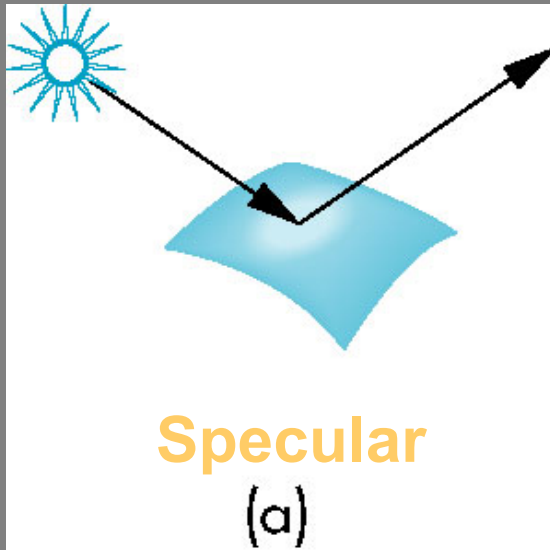
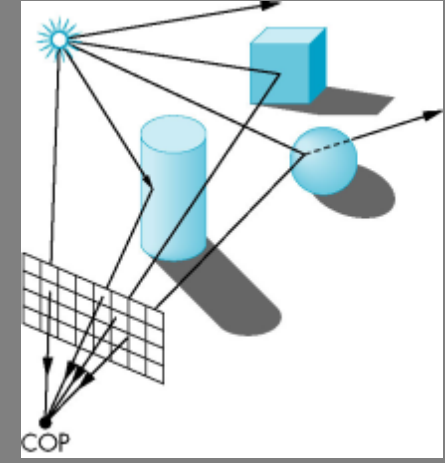
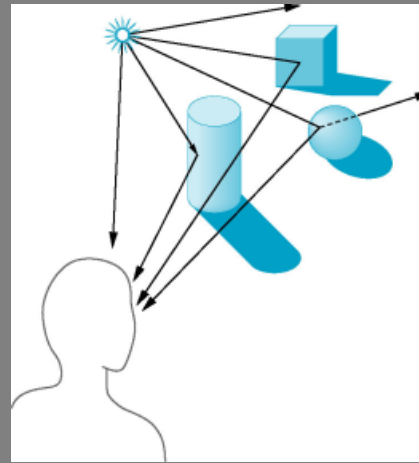
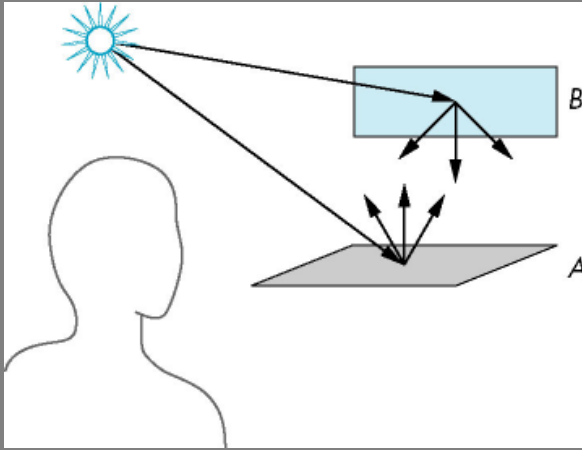
（4）光照及颜色处理

计算场景中可见的颜色，也就是根据假定的光照条件和景物外观因素，依据一定的光照模型，计算可见面投射到观察者眼中的光亮度和颜色分量，并把它们转换为适合图形设备处理的颜色值。从而确定投影画面上每一个像素的颜色，最终生成图形，达到视觉上的最佳效果。

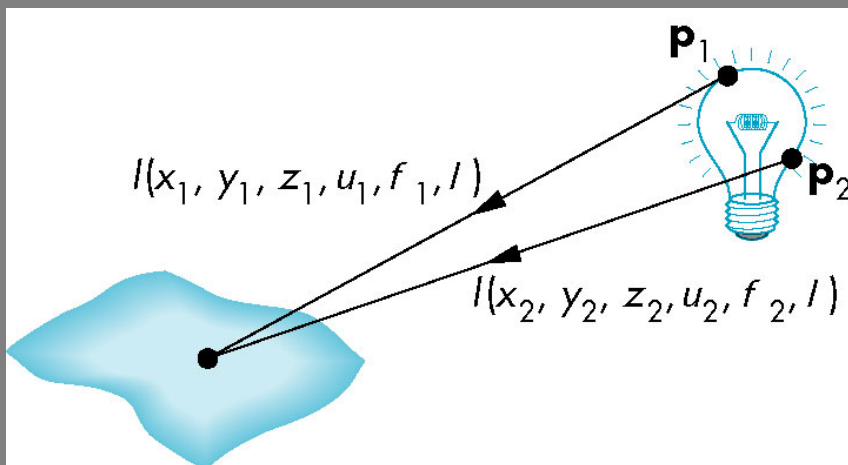
光线与物质 light and Matter

When light strike a surface, some of its absorbed, and some of it is reflected. If the surface is opaque, reflection and absorption account for all the light striking the surface. If the surface is translucent, some of the light is transmitted through the material and emerges to interact with other object. This interaction depend on wavelength. An object illuminated by white light appears red because it absorb most of the incident light but reflects light in the red range of frequencies. A shiny object appears so because its surface is smooth. Conversely, a dull object has a rough surface.

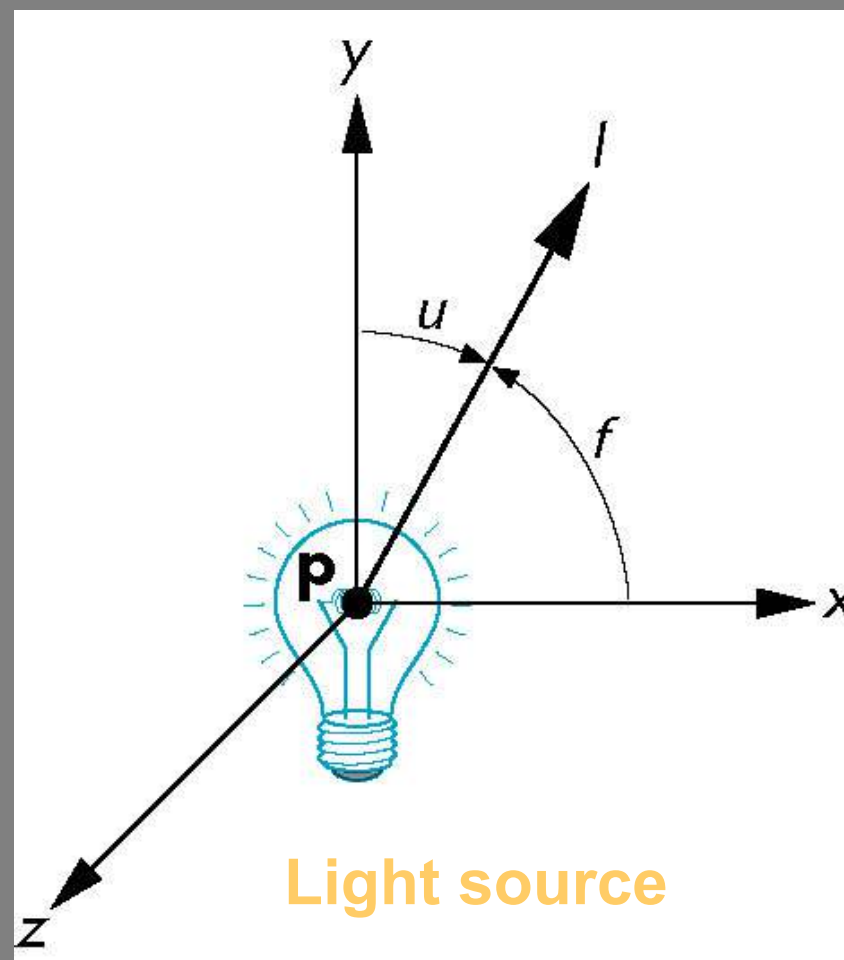
Light and Matter



光源 Light Sources

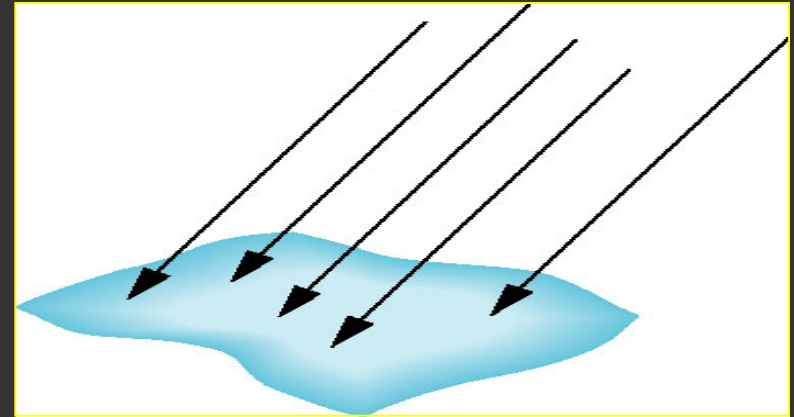
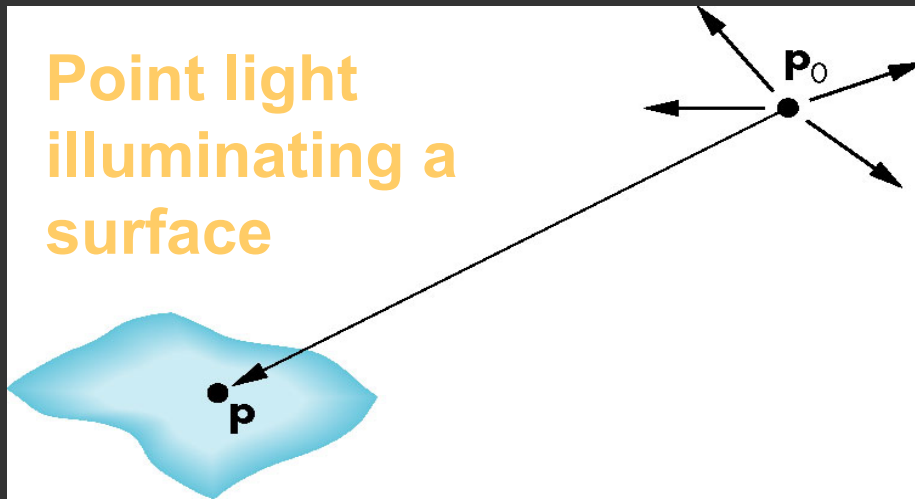


Adding the contribution
from a source

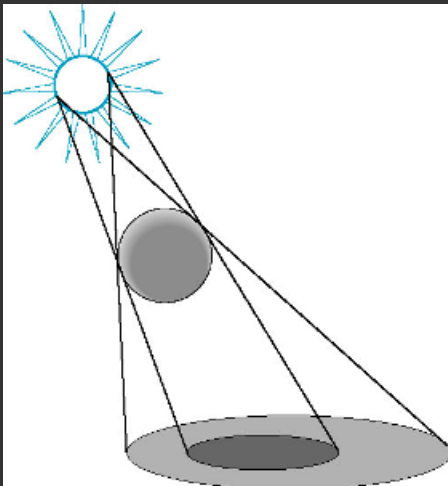


Light Sources

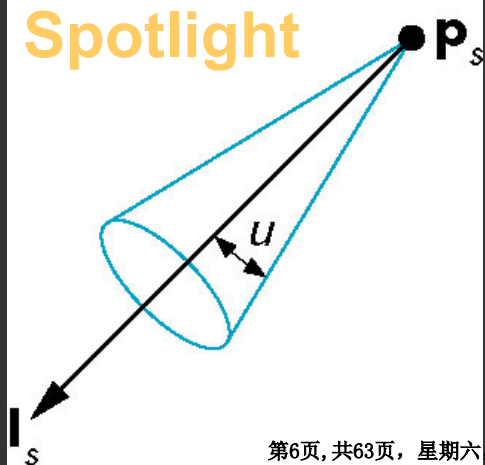
Point light
illuminating a
surface



Parallel light source



Shadows created
by finite-size light
source



光照模型 Lighting Model

简单光照模型

只考虑光源照射在物体表面的发射光，假定物体表面是光滑的，而且是由理想材料构成。

复杂光照模型

又叫整体光照模型，这种模型不仅要考虑发射光，还需要考虑周围环境的光对物体表面的影响、物体的透明度、阴影的处理、光源的位置和个数等。

光照模型 Lighting Model

· 环境光 $E_e = I_a K_a$

E_e ——环境光强； I_a ——泛光光强； K_a ——漫放射系数[0-1]

· 镜面反射光 $E_s = I_s K_s \cos^n \beta$

E_s ——反射光在观察方向的光强； I_s ——点光源的光强；

K_s ——物体表面反射率[0—100%]；

· 漫射光 $E_d = I_d K_d \cos \theta$

· Phong模型（环境光+镜面反射光+漫射光）

环境光 Ambient Reflection

环境光：光线在空间中经过复杂的传播之后形成的弥漫于整个空间，无固定的光源、方向的光。环境光可以认为是周围景物如墙壁、天花板、桌面等物体发出的通过物体表面而反射出来的光。

环境光强 $E_e = I_a K_a$

I_a ——漫射光强度（泛光光强）；

K_a ——漫反射系数 $[0 \leq K_a \leq 1]$

不同的物体有不同的 K_e 值

镜面反射光 Specular Reflection

镜面反射光：在光源照射下，物体表面可以产生高亮度或是亮点，即所谓的高光效应，这种现象称为镜面反射。

$$E_s = I_s K_s \cos^n \beta$$

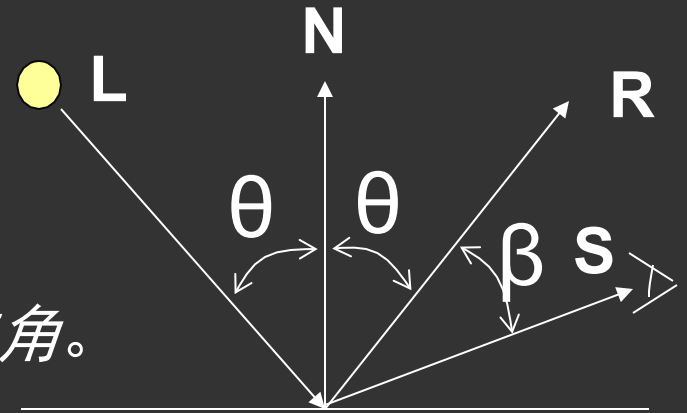
E_s —反射光在观察方向的光强；

I_s —点光源的光强；

K_s —物体表面反射率[0—100%]；

β —视点方向 S 与镜面反射方向的夹角。

n —物体表面光滑度系数（0—2000）



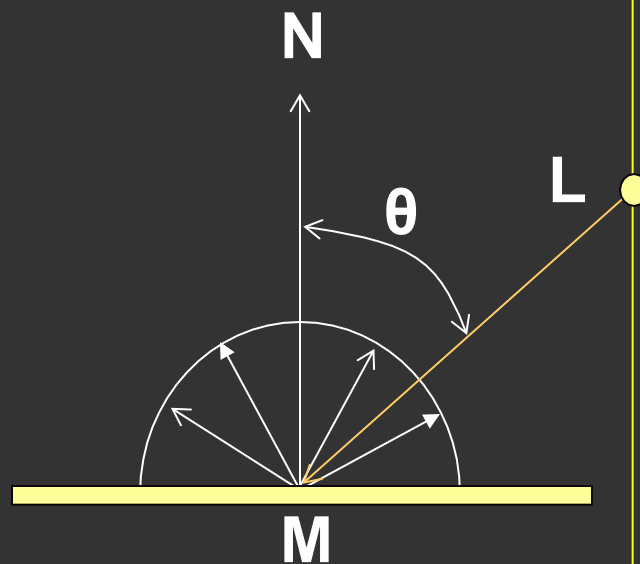
漫反射光 Diffuse Reflection

- 漫反射光：是由特定光源在物体表面反射光中那些向空间各个方向均匀反射出去的光。这种光的反射强度与观察点的位置无关。

- $$E_d = I_d K_d \cos\theta$$

I_d —点光源发出的入射光的强度

K_d —入射光的漫反射系数 (0-1之间)



Phong模型 Phong Reflection Model

Phong模型（环境光+镜面反射光+漫射光）

单个光源

$$E = E_e + E_d + E_s = I_a K_a + I_d K_d \cos \theta + I_s K_s \cos^n \beta$$

多个光源 (n 个)

$$E = I_a K_a + \sum_{i=1}^n I_{(d,i)} K_d \cos \theta_i + \sum_{i=1}^n I_{(s,i)} K_s \cos^n \beta_i$$

Phong Reflection Model

! 可分解为RGB三种颜色:

$$E_R = I_{aR}K_{aR} + \sum_{i=1}^n I_{(dR,i)}K_{dR} \cos \theta_i + \sum_{i=1}^n I_{(sR,i)}K_{sR} \cos^n \beta_i$$

$$E_G = I_{aG}K_{aG} + \sum_{i=1}^n I_{(dG,i)}K_{dG} \cos \theta_i + \sum_{i=1}^n I_{(sG,i)}K_{sG} \cos^n \beta_i$$

$$E_B = I_{aB}K_{aB} + \sum_{i=1}^n I_{(dB,i)}K_{dB} \cos \theta_i + \sum_{i=1}^n I_{(sB,i)}K_{sB} \cos^n \beta_i$$

光的强度随距离的衰减公式

$$E = I_a K_a + \sum_{i=1}^n f(d_i) I_{(d,i)} K_d \cos \theta_i + \sum_{i=1}^n f(d_i) I_{(s,i)} K_s \cos^n \beta_i$$

$$f(d) = \frac{1}{a + bd + cd^2}$$

d—从点光源发出的光线到某一观察点的所经过的距离
a、b、c——系数，调整其大小可以得到不同光照效果。

明暗处理 shading

- 在计算机三维图形真实感绘制过程中，得到了物体在光照条件下的表面个点的强度值后，还需要将这些强度值转换为计算机图形系统所能支持的明暗模式，从而进行浓淡处理。
- 对曲面，可通过计算曲面上每点的亮度来获得光照效果
- 对平面，由于其上每点的法向量相等，可以使用同一亮度来获得光照效果。
- 不同的多边形具有不同的亮度，在进行明暗处理以后，可以使所生成的物体图形具有层次感

恒定光强的明暗处理

也叫Flat明暗处理法

采用一种颜色来对多边形进行绘制，在多边形上任意取一点，利用简单光照模型计算出该点的颜色，把这点的颜色作为多边形的颜色。适用于平面体的真实感图形的处理。

当：

- (1) 光源无穷远；（ $\cos\theta$ and 衰减函数=const.）
- (2) 观察点距物体足够远；（ $\cos\beta$ =const.）
- (3) 多边形中不含有曲线面。

时，可以得到一个比较真实的场景。

由于相邻多边形的法向量不同，因而计算出的颜色也不同。存在在相邻边界处颜色过渡不光滑的问题。

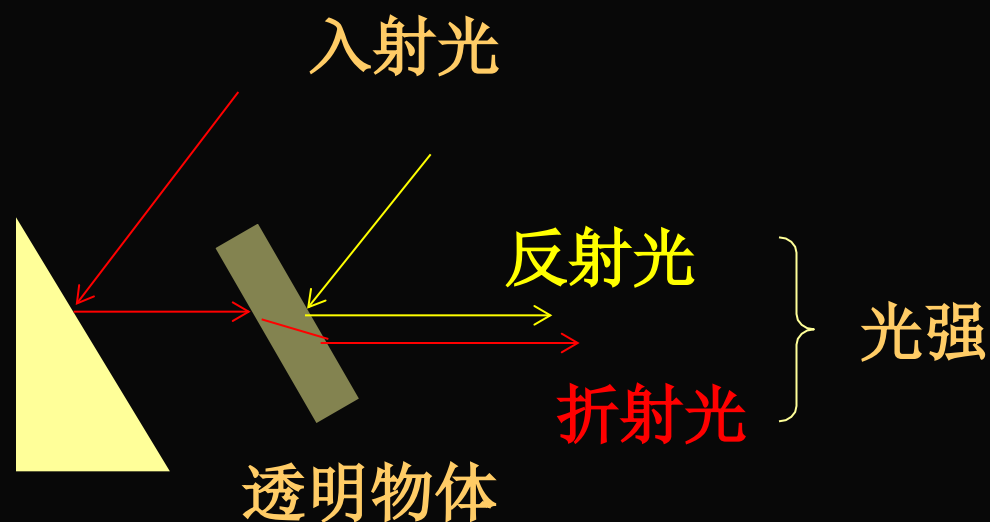
Gouraud明暗处理

- 也叫亮度插值明暗处理，它是通过在面片上将光强度值进行线性插值来对该多边形面片进行绘制的。步骤如下：
 - 确定每个多边形在顶点处的平均单位法向量；
 - 根据光照模型，计算每个顶点的光强度；
 - 用顶点强度对多边形表面上的光强进行线性插值。具体算法见书P256页。
- 不足之处：会出现过亮或过暗的马赫条纹；只是考虑了漫射光；对于具有层次分别且相互平行的多边形面，有时会难于表现它们之间层次感。

Phong 法向插值法

- ❖ 为了克服Gouraud方法对镜面反射区域所引起的畸变，Phong提出了用法向的插值来代替强度插值的方法。
- ❖ 具体计算步骤如下：
 - (1) 计算每一个多边形顶点处的平均单位法矢量；
 - (2) 用双线性插值方法对多边形内部各个点的法矢量进行求解；
 - (3) 根据光照模型沿着每条扫描线计算面上各个点所对应的投影像素的光强度。
- ❖ 特点：比Gouraud明暗处理的精度要高，但是计算量要大得多。

透明的处理 Transmitted Light



透明物体表面的光强=反射光+折射光

透明的处理 Transmitted Light

透明性的简单模型（不考虑漫射、折射）

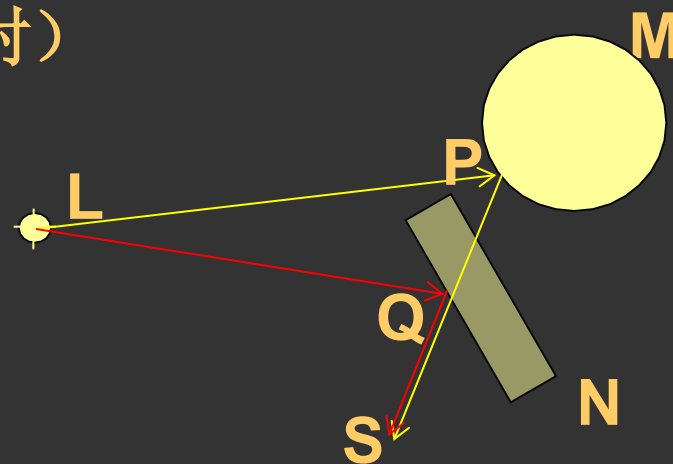
到达S点的光强：

$$E_s = (1-k)E_Q + kE_P \quad (0 \leq k \leq 1)$$

k 为透明物体的透射系数

$k=1$ 时，完全透明,观察着可以完全看到背景景物

$k=0$ 时，物体表面不透明



通常采用深度缓冲器的算法来实现透明效果：先处理不透明物体以决定可见不透明表面的深度，然后将透明物体的深度值与先前存在深度缓冲器中的值进行比较。这样就可以增加对透明表面深度及其他参数的储存，再将可见透明面的光强与其后面可见不透明面的光强综合考虑进行绘制。

透明的处理 Transmitted Light

Snell's Law

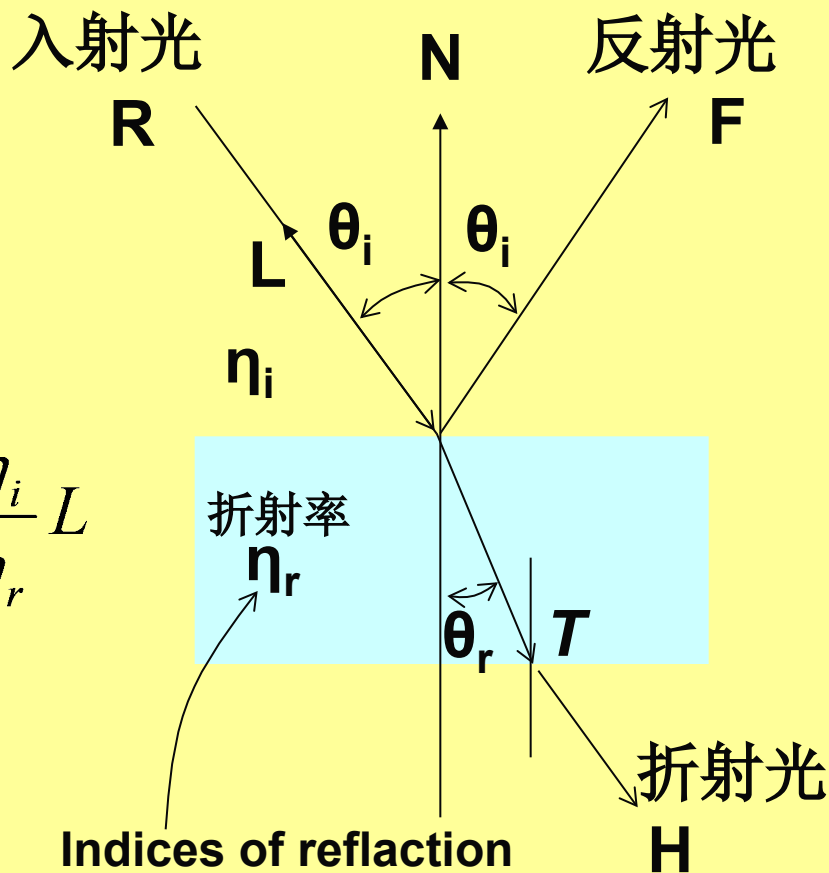
$$\sin \theta_r = \frac{\eta_i}{\eta_r} \sin \theta_i$$

折射方向的单位投射矢量:

$$T = \left(\frac{\eta_i}{\eta_r} \sin \theta_i - \cos \theta_r \right) N - \frac{\eta_i}{\eta_r} L$$

N —物体表面的单位法矢量

L —光源方向的单位矢量



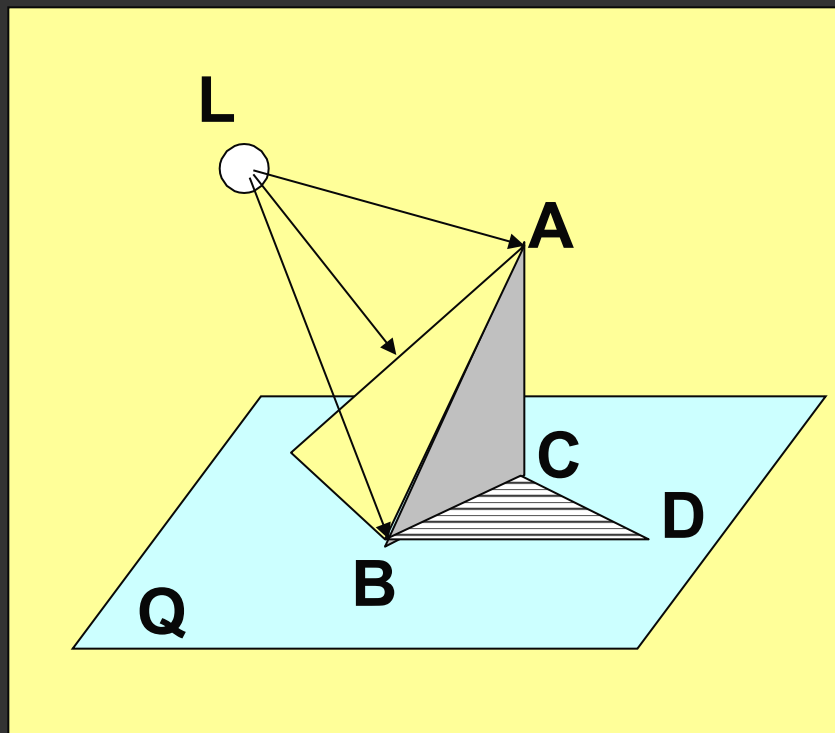
阴影产生技术 Shading Technology

理论上讲，从视点以及光源看过去都是可见的面是位于阴影之外的，只有当从视点看过去，而从光源看过去不可见的面，才会落在阴影之内（下图）

产生具有阴影的图形算法步骤：

（1）将视点移到光源所在位置，用多边形区域进行排序消隐算法，将多边形分成向光和背光多边形；

（2）将视点移到原来观察位置，对向光多边形和背光多边形进行消隐，并采用某种光照模型计算多边形的亮度。



整体光照模型

Whitted 光照模型(1980): 在Phong模型中增加了环境镜面反射光和环境规则透射光亮度。它不仅考虑了光源直接照射时所引发的反射光到达观察者的光亮度,还考虑了从场景中其他物体镜面反射或透射过来的光。

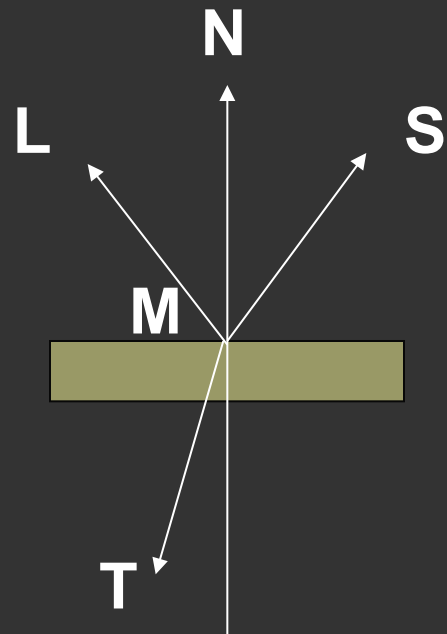
$$E = E_e + K_s I_s + K_t I_t$$

E_e —反射光亮度,按Phong模型计算。

K_s 、 K_t —反射系数、透射系数, $\{0,1\}$ 之间。

I_s —镜面反射方向上,其他物体向点M辐射的光亮度。

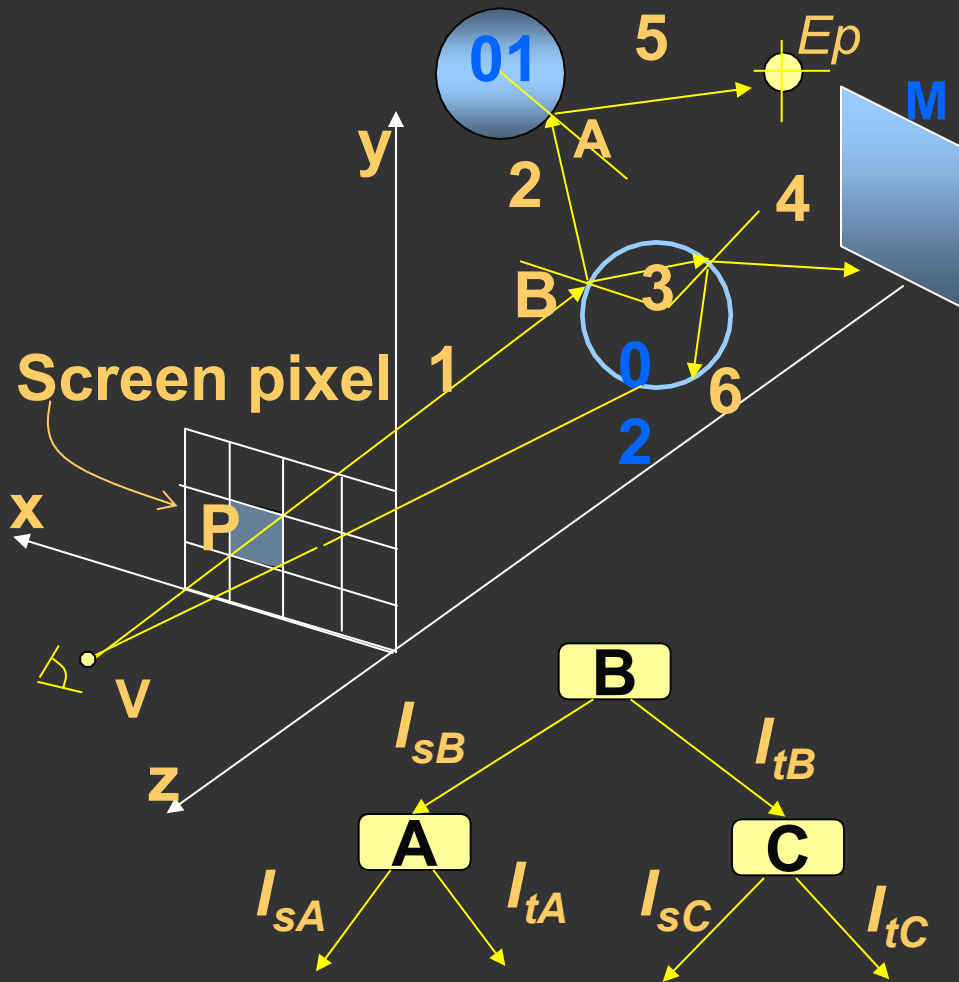
I_t —在折射方向上其他物体向点M辐射的光亮度。



光线跟踪 Light Trace

- 光线跟踪算法利用光线的可逆性，不是从光源出发，而是从视点出发，沿着视线进行追踪，模拟光的传播路径来确定反射、折射和阴影。
- 首先从观察点位置开始向光源反方向跟踪光线，单一光线投向物体表面再反射到视点，然后对每个像素点进行计算并建立每个像素点的亮度值，最终显示在画面上的像素点是物体表面上距离观察点的最近的点，从而产生高度真实的三维图形，因此计算量很大。但对于每个像素点都单独计算，故能较好的表现曲面细节。其产生的明暗效果也最真实。

Light Trace



光线跟踪二叉树算法

$$E_B = E_{eB} + K_{sB} I_{sB} + K_{tB} I_{tB}$$

E_{eB} 光源直接照射及环境光所产生的局部光强

$K_{sB} I_{sB}$ 反射光2来的光强对B点的作用

$K_{tB} I_{tB}$ 透射光3来的光强对B点的作用

$$I_{sB} = E_A, \quad I_{tB} = E_C$$

$$E_A = E_{eA} + K_{sA} I_{sA} + K_{tA} I_{tA}$$

E_{eA} 光源直接照射及环境光所产生的局部光强

$K_{sA} I_{sA}$ 反射光5来的光强对A点的作用

$$I_{sA} = E_p$$

$K_{tA} I_{tA} = 0$ (A为不透明体)

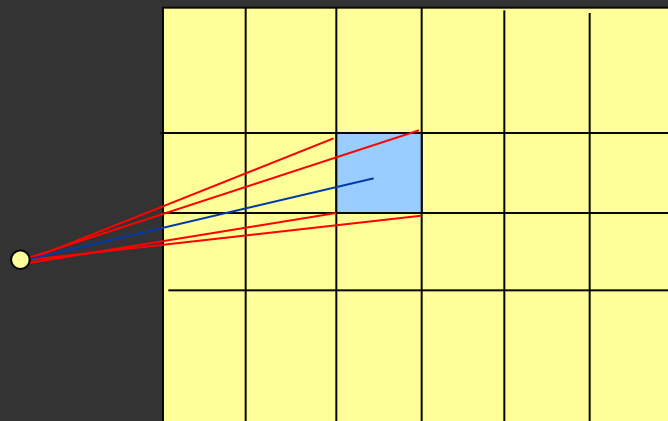
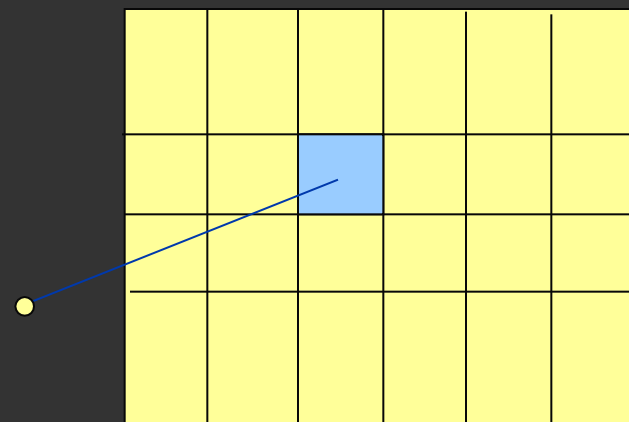
$$E_C = E_{ec} + K_{sc} I_{sc} + K_{tc} I_{tc}$$

I_{tc} = 背景光强, I_{sc} = 6点的内反射光强

光线跟踪的反走样技术

图形的走样现象：由于相邻的两个像素点之间有一定的距离是相互离散的，而且纯粹的光线跟踪算法又是一种典型的点采样技术，因此用像素点阵组合出的图形与真实的景物之间不可避免的会存在误差。以直线为例，计算机所画出的直线就有可能呈现出锯齿状，细小的物体在屏幕上也可能显示不出来。

像素细分反走样技术：将发生图形走样的像素点细分为4个子像素，再分别对子像素进行光线跟踪。子像素的位置就是走样像素的四个角点。通过这四个角点分别进行光线跟踪，最终像素的光亮度就是这四个角点光亮度的算术平均值。

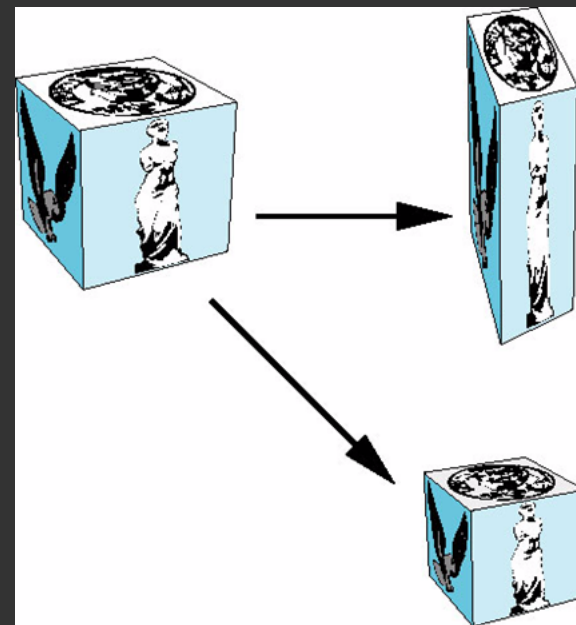


纹理处理 texture Mapping

- 现实世界中的物体表面往往是具有各种各样的纹理的，我们把这种纹理称为表面细节。物体表面细节可以分为表面的颜色纹理和几何纹理。
- 颜色纹理：木制的家具表面、建筑墙面的拼花图案等。颜色纹理可以用纹理映射来描述。
- 几何纹理：由不规则的细小凹凸组成的桔子表面的褶皱等。几何纹理可以采用一个扰动函数来描述。
- 纹理的两种定义方法：
 - （1）连续法：把纹理函数定义为一个二元函数，其函数的定义域就是纹理的空间。
 - （2）离散法：把纹理定义在一个二维数组中，用以代表纹理空间中的行间隔、列间隔固定的一组网络点上的纹理值

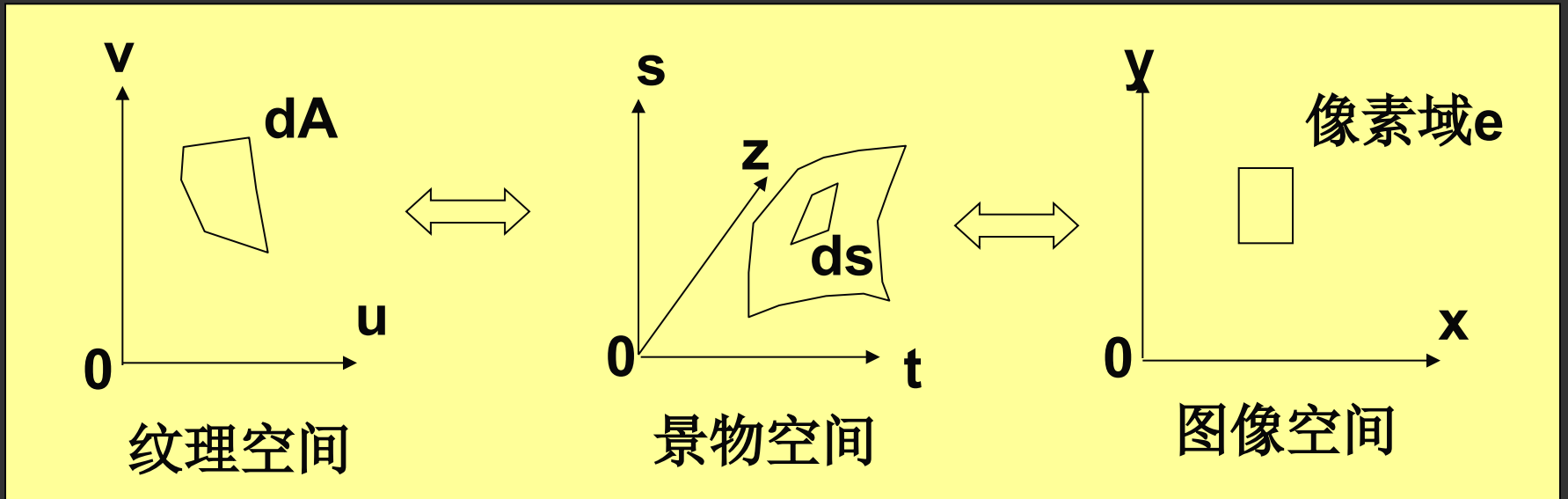
纹理映射 Texture Mapping

纹理映射：将一定的纹理函数映射到物体表面上，在对物体表面进行光亮度计算时可采用相应的纹理函数值作为物体表面的漫反射光亮度值代入光照模型进行计算。因此，可以采用纹理映射来对颜色纹理进行模拟。在光滑的物体表面上描绘花纹实际上就是将花纹在表面上进行一定的映射而成的，这个过程实际上就是将任意的平面图形或图像覆盖到物体表面上，使得物体表面形成真实的色彩花纹。



Texture Affine

- 纹理映射涉及到纹理空间、景物空间和图像空间（屏幕空间）三个空间的相互转换和映射。



Texture Mapping

- 一般说来，实现映射可以有两种办法：
 - (1) 将纹理模式映射到物体表面，然后由物体表面映射到（图像空间）投影屏幕上。
 - (2) 将像素区域（图像空间）映射到物体表面，然后再有物体表面映射到纹理空间

Texture Mapping

纹理空间 (u,v) 向物体空间 (s,t) 的映射(用线性函数):

$$s=f_s(u,v)=a_s u+b_s v+c_s$$

$$t=f_t(u,v)=a_t u+b_t v+c_t$$

物体空间 (s,t) 向投影屏幕空间 (x,y) 的映射:

$$x=x(s,t)$$

$$y=y(s,t)$$

$$z=z(s,t)$$

$$w=w(s,t)$$

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/387025002101010001>