



上节要点

- 光照明模型及其物理基础
- 简单光照明模型

- **Phong模型** $I = I_a K_a + I_p K_d (L \cdot N) + I_p K_s (H \cdot N)^n$

- 各变量意义
- 实现与简化
- 特点

- **增量式光照明模型(Gourand, Phong)**

- 特点



上节要点

- 较为完善的局部光照明模型
 - 菲涅尔公式
 - 微平面理论
 - **Cook模型**

$$I_r = I_a K_a + I_i (N \cdot L) d\omega (K_d R_d + K_s R_s)$$

$$R_s = \frac{DG\rho(\theta, \lambda)}{\pi(N \cdot L)(N \cdot V)}$$



光照明模型(1)

- 简单的局部光照明模型(简单光照明模型)
- 较为完善的局部光照明模型
- 光透射模型



光透射模型的研究

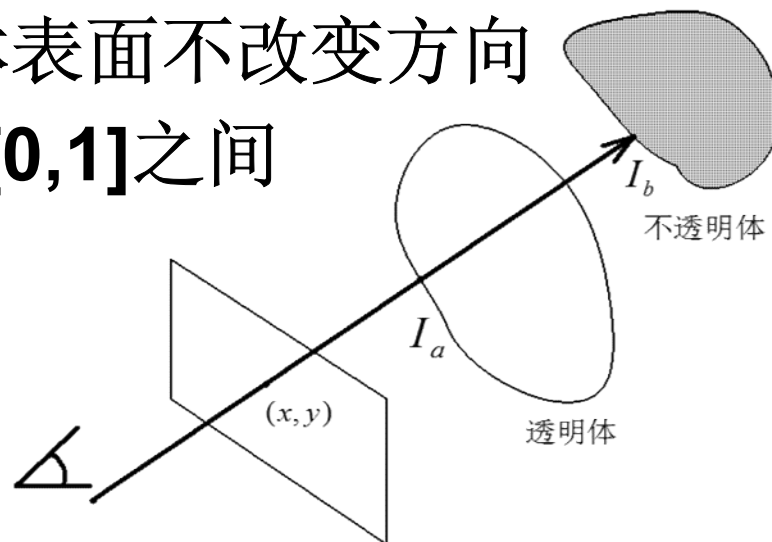
- 早期简单透射现象的模拟
 - 颜色调和法
 - 基于表面法向的颜色调和
- **1980年，Whitted**光透射模型，首次考虑了光线的折射现象
- **1983年，Hall**光透射模型，考虑了漫透射和规则透射光



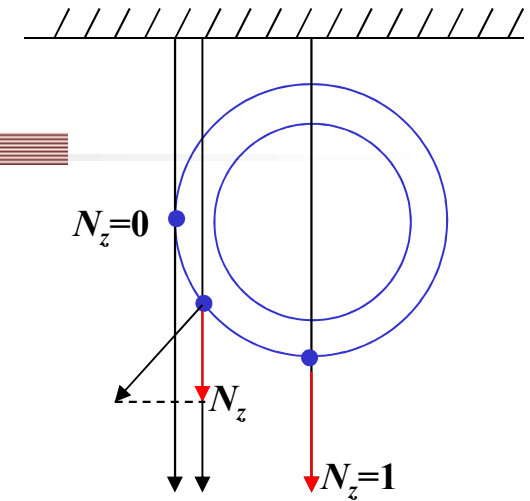
■ 颜色调和法

- 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
- t 是物体的透明度, 在 $[0,1]$ 之间
- 光强计算:

$$I = tI_b + (1-t)I_a$$



- 前后两物体光强可由简单光照明模型计算
- 可用隐藏面消除算法实现
- 模拟平面玻璃的透明效果



- 基于表面法向的颜色调和
 - 对于玻璃杯一类的透明物体
 - 将透明度 t 与透明体的表面法向量建立一种联系
 - 用表面法向量的 z 分量来决定 t 的大小:

$$t = t_{min} + (t_{max} - t_{min})N_z$$

- t_{min} 为透明体的最小透明度
- t_{max} 为透明体的最大透明度
- N_z 为透明体表面法向的 z 分量, $0 \leq N_z \leq 1$

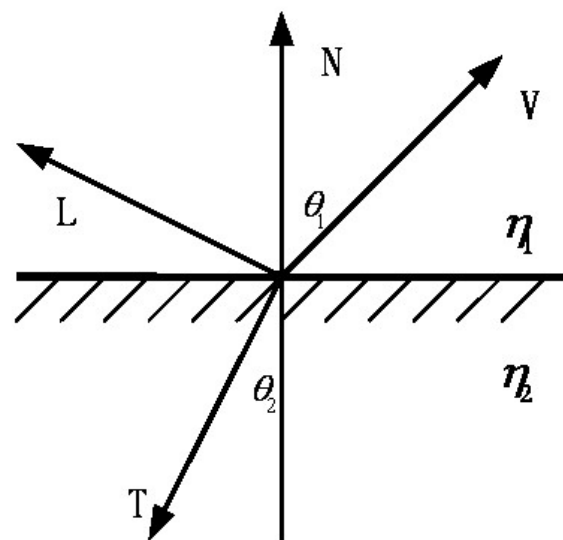


Whitted光透射模型

- 在简单光照明模型的基础上加上透射一项，得到**Whitted**光透射模型

$$I = I_a K_a + I_p K_d (L \cdot N) + I_p K_s (H \cdot N)^n + I_t K_t'$$

- I_t 为折射方向 T 的入射光强
- K_t' 为透射系数，
是0~1之间的常数

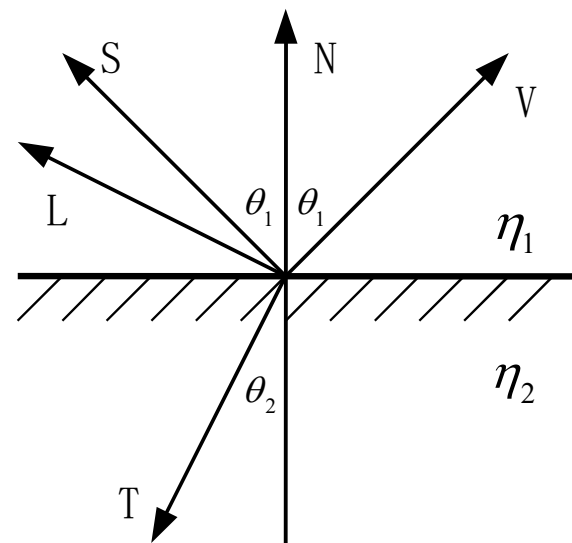




- 如果该透明体又是一个镜面反射体，加上反射光一项，得到较完整的**Whitted**光透射模型：

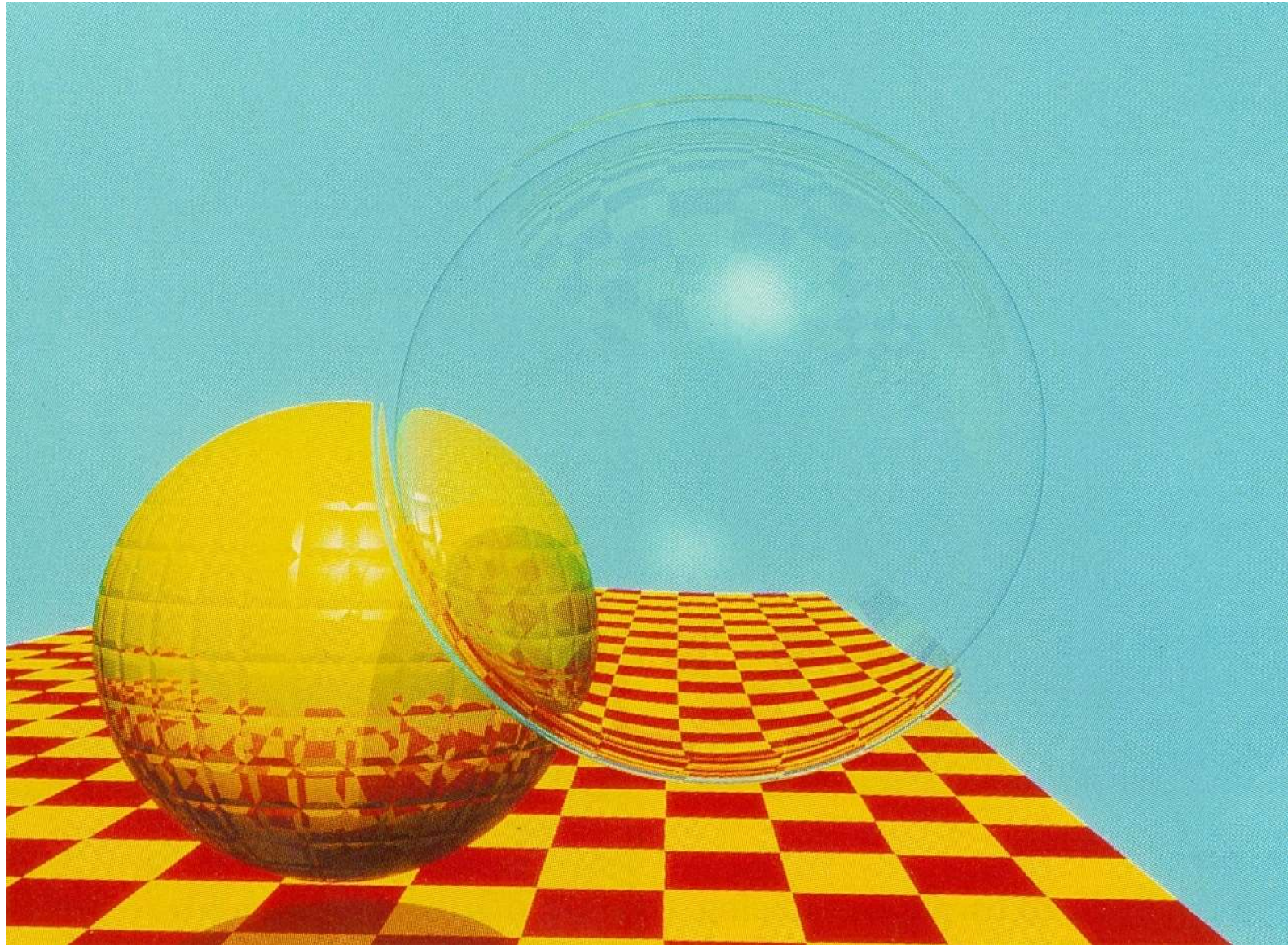
$$I = I_a K_a + I_p K_d (L \cdot N) + I_p K_s (H \cdot N)^n + I_t K_t' + I_s K_s'$$

- I_s 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





- **Whitted**光透射模型的折射方向和镜面反射方向都是相对于视线而言的，是使视线成为折射方向和反射方向的入射光的方向，而且方向与光传播的方向相反。
- $I_s K_s$ 为其它物体在所观察物体表面上的镜面影像；
- $I_t K_t$ 为其它物体在所观察物体表面上的透射影像；





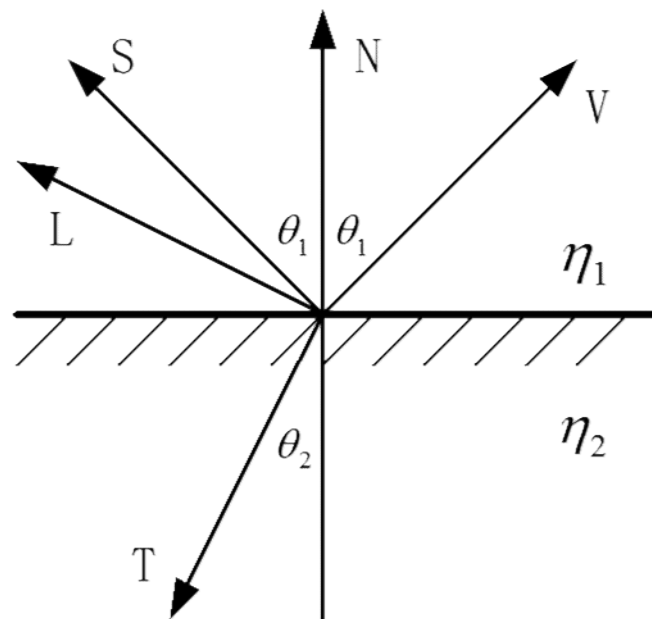
$$I = I_a K_a + I_p K_d (L \cdot N) + I_p K_s (H \cdot N)^n + I_t K'_t + I_s K'_s$$

反射、折射方向计算

- 已知视线方向 V ，求其反射方向 S 与折射方向 T
- 光学几何原理
- V, N, T 均为单位向量
- 视线 V 的反射方向 S

$$S = 2N(N \cdot V) - V$$

- N 是表面的法向





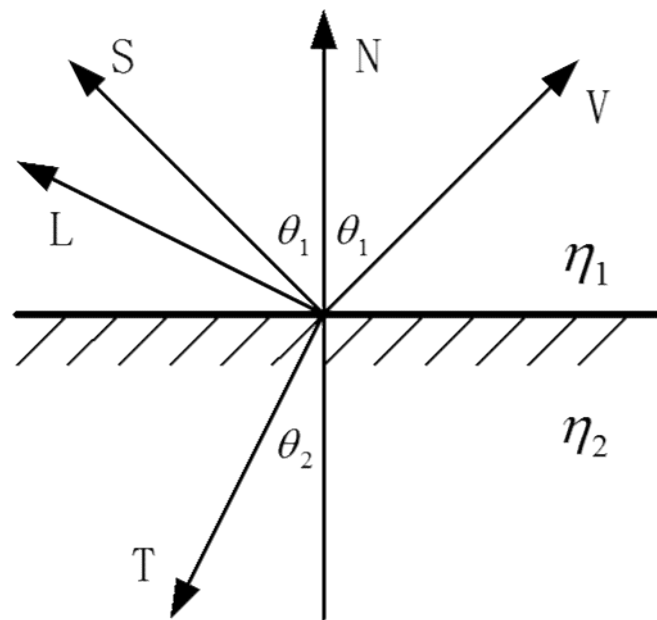
- η_1 是视点所在空间的介质折射率,
- η_2 为物体的折射率
- 折射定律:

$$\frac{\sin \theta_1}{\sin \theta_2} = \frac{\eta_2}{\eta_1} = \eta$$

- Whitted 中折射方向 T

$$T = k_f (N - V') - N$$

- 其中 $k_f = 1 / \sqrt{\eta^2 |V'|^2 - |N - V'|^2}$ $V' = \frac{V}{N \cdot V}$





- **Heckbert**的折射方向计算方法更简单

$$T = -\frac{1}{\eta}V - \left(\cos \theta_2 - \frac{1}{\eta} \cos \theta_1 \right) N$$

- 其中

$$\cos \theta_1 = N \cdot V$$

$$\cos \theta_2 = \sqrt{1 - \frac{1}{\eta^2} (1 - \cos^2 \theta_1)}$$

- 计算出的结果***T***为单位向量



■ Heckbert公式推导

设 T 为单位向量

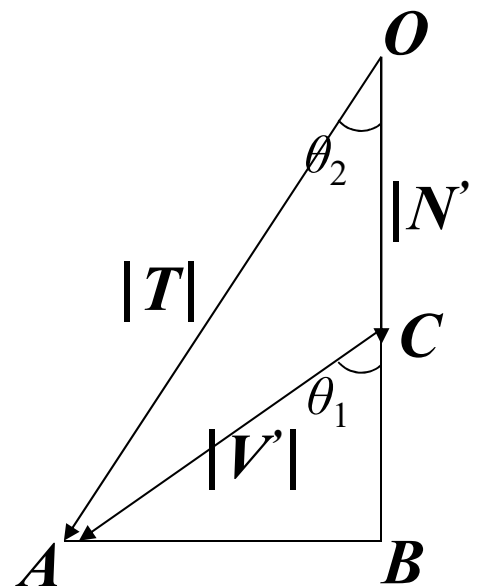
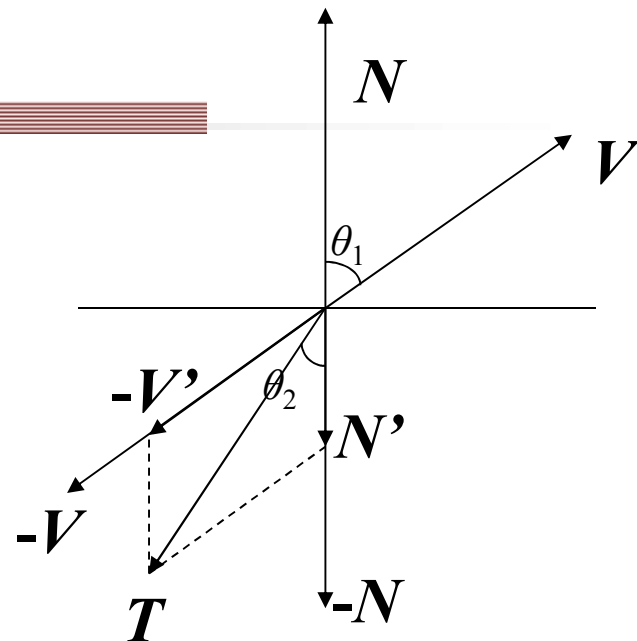
将 T 在 $-V$ 和 $-N$ 方向上分解:

$$T = -V' - N' = -V \cdot |V'| - N \cdot |N'|$$

$$|V'| = \frac{AB}{\sin \theta_1} = \frac{|T| \sin \theta_2}{\sin \theta_1} = \frac{\sin \theta_2}{\sin \theta_1} = \frac{\eta_1}{\eta_2} = \frac{1}{\eta}$$

$$|N'| = OB - CB = |T| \cos \theta_2 - |V'| \cos \theta_1$$

$$= \cos \theta_2 - \frac{1}{\eta} \cos \theta_1$$





Hall光透射模型

- 在**Whitted**光透射模型基础上推广而来
- 可以处理规则透射高光，以及理想的漫透射
- 理想漫透射
 - 透明体的粗糙表面对透射光的作用表现为漫透射，透射光的光强在各个方向均相等

$$I_{dt} = I_p \cdot K_{dt} \cdot (-N \cdot L)$$

- K_{dt} 为物体的漫透射系数， $0 \leq K_{dt} \leq 1$

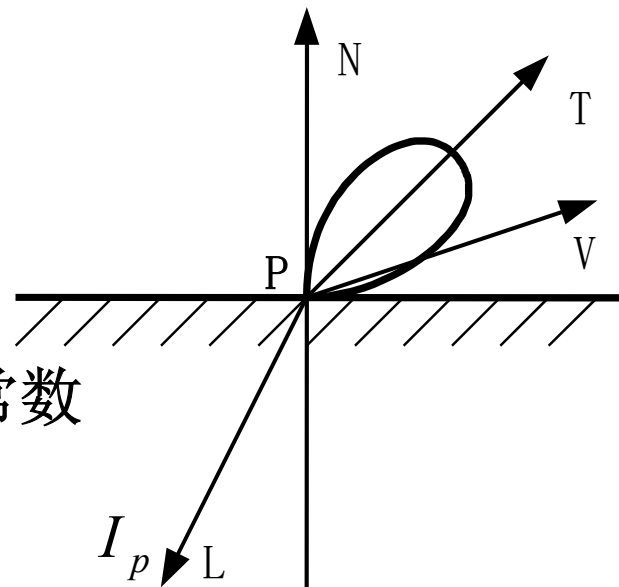


■ 规则透射高光

- 半透明的物体，视点在折射方向附近也能见到部分折射光，在**折射方向**周围形成高光域
- 高光光强计算

$$I_t = I_p \cdot K_t \cdot (T \cdot V)^n$$

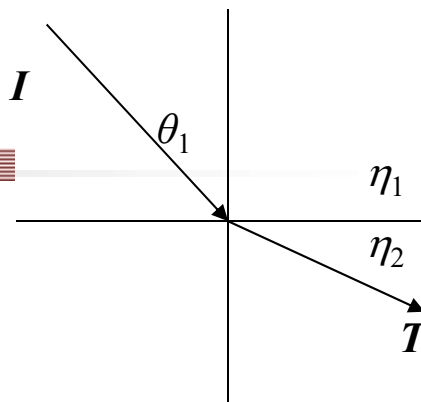
- K_t 为物体的透明系数
- n 为反映物体表面光泽的常数
- 透射方向 T





■ 减少计算量的假设

- 光源在无穷远处， L 方向为常量
- 视点在无穷远处， V 方向为常量
- 用 $(H_i \cdot N)$ 代替 $(T \cdot V)$
 - H_i 为虚拟理想透射面法向，该法向使视线为光线的折射方向



■ 使用Hall模型注意点

- 视点与光源在透明体两侧，才能透过透明体看到透射高光
- 不考虑光线射入透明体时的折射
- 折射的临界角现象：
 - 光线从高密介质(η_1)射向低密介质(η_2)、入射角大于临界角时，不再发生折射，只有内部反射
 - 临界角：

$$\theta_c = \arcsin \frac{\eta_2}{\eta_1} = \arcsin \eta$$

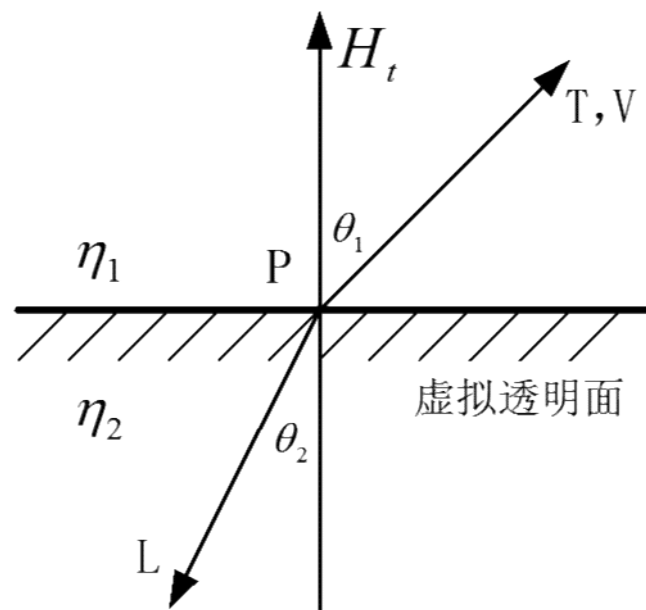


H_t 的求解

H_t 为当视线为光线折射方向时的面法向

$$L = -\frac{1}{\eta}V - (\cos \theta_2 - \frac{1}{\eta} \cos \theta_1)H_t$$

$$H_t = \frac{L + \frac{1}{\eta}V}{\cos \theta_2 - \frac{1}{\eta} \cos \theta_1}$$





- 简化、单位化：

$$H_t = \text{sign}(\eta_1 - \eta_2) \frac{\eta_2 L + \eta_1 V}{|\eta_2 L + \eta_1 V|}$$

- 符号由折射率确定： $\eta_1 > \eta_2$ 时，取正号，否则取负号



简单光反射透射模型

- 综合简单光照明模型，**Whitted**光透射模型和**Hall**光透射模型：

$$I = I_a K_a + \sum_i I_{p,i} \left[K_{ds} (L_i \cdot N) + K_s (H_{s,i} \cdot N)^{n_s} \right] \\ + \sum_j I_{p,j} \left[K_{dt} (-N \cdot L_j) + K_t (N \cdot H_{t,j})^{n_t} \right] + I_t K_t' + I_s K_s'$$

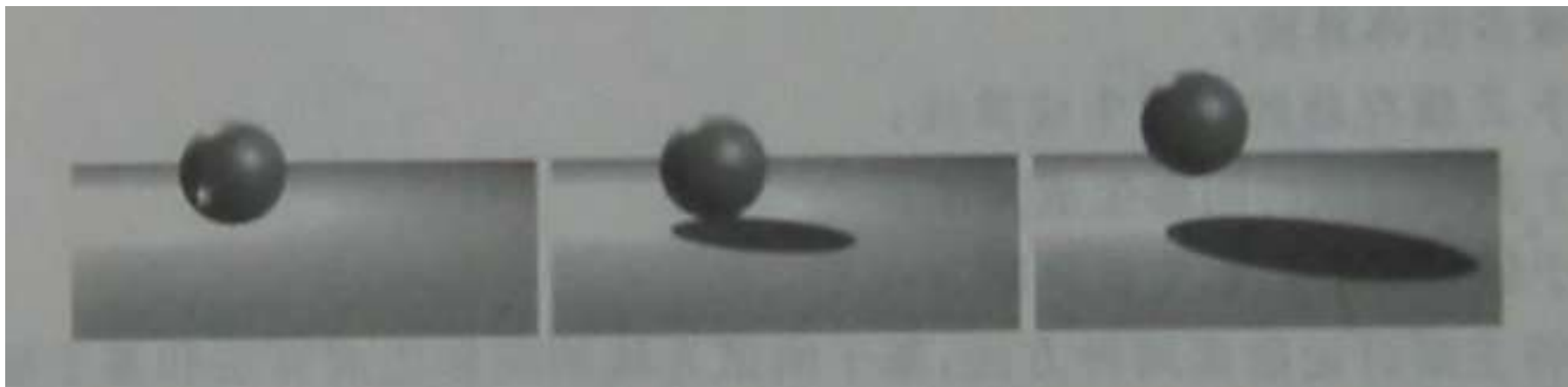
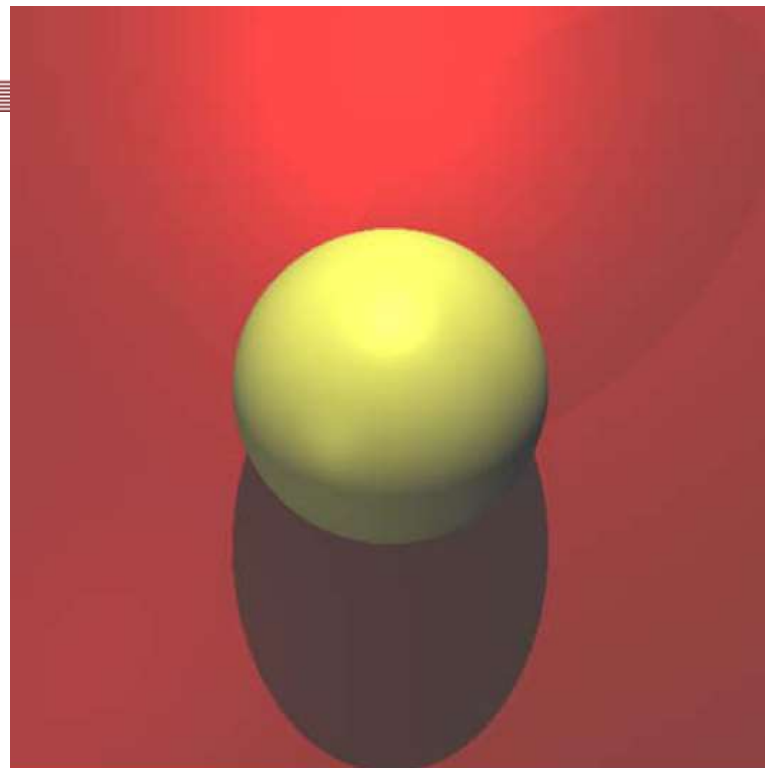


第十章 阴影



阴影(Shadow)

- **阴影**: 场景中那些没有被光源直接照射到的区域。



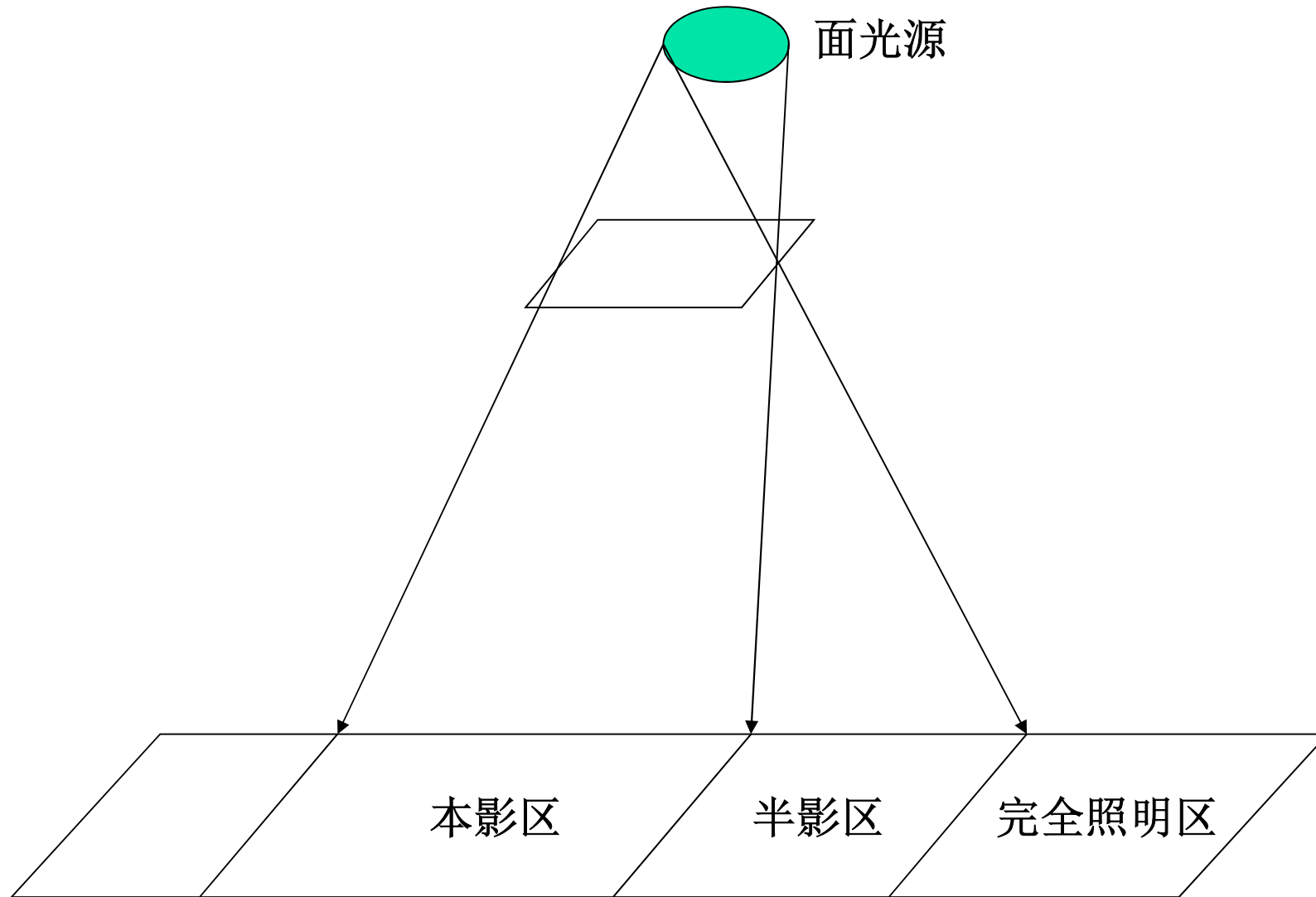


■ 自身阴影和投影阴影

- 自身阴影：表面背离光源照射方向时，在该表面上形成的阴影
- 投影阴影：场景中因不透明景物对光源入射光线的遮挡而在位于其后的景物表面上形成的阴影

■ 本影和半影

- 本影：景物表面上没有被光源直接照射的部分
- 半影：景物表面上被部分光源直接照射的部分





- 单个点光源只能产生本影，多个点光源或线/面光源才能产生半影；
- 阴影的生成技术需要解决的问题：
 - 计算阴影区域的形状
 - 计算阴影区域的光亮度
 - 近似

$$I = I_a K_a + \sum_i S_i f_i I_{pi} (K_d (L_i \cdot N) + K_s (R_i \cdot V)^n)$$

$$S_i = \begin{cases} 0, & \text{该点位于第} i \text{个光源的阴影内} \\ 1, & \text{该点位于第} i \text{个光源的阴影外} \end{cases}$$



- 生成阴影的算法:
 - 阴影扫描线算法
 - 阴影细节多边形算法
 - 阴影**Z-Buffer**算法(**Shadow Mapping**)
 - 影域多面体算法(**Shadow Volume**)

- 光线跟踪
- 辐射度



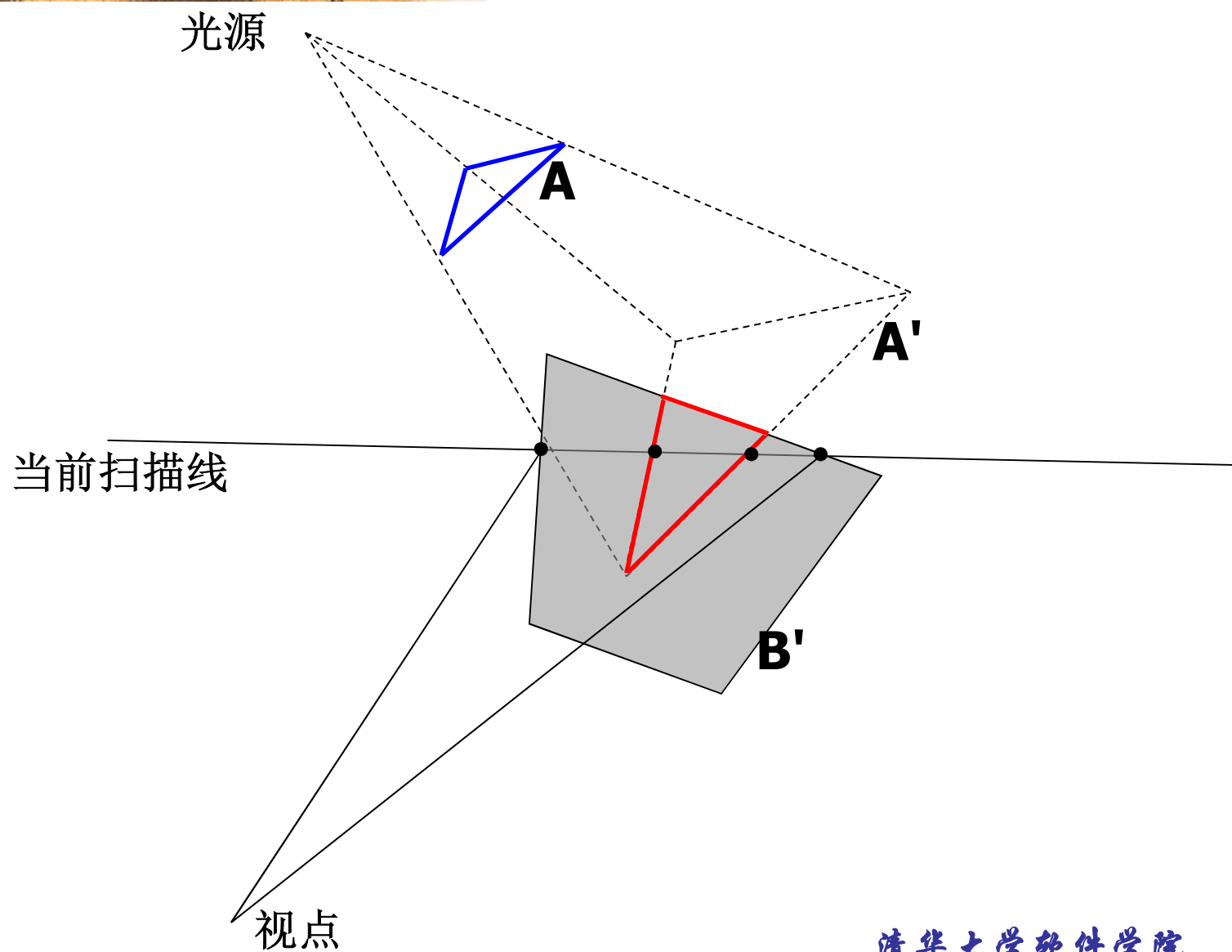
阴影扫描线算法

- 最早的阴影生成方法：将扫描线算法进行改进，交错进行阴影和可见面处理。
- 预处理：
 - 阴影多边形：能在其它多边形上形成阴影的多边形。
 - 对每个多边形，建一个数据结构，指向其所有阴影多边形
 - 具体过程：把多边形投影到球心在点光源位置的球表面，对重叠的多边形进行排序



■ 修改的扫描线消隐算法

- 若某个多边形不存在阴影多边形，按正常的扫描线消隐算法进行处理；
- 若存在阴影多边形，分三种情况：
 - 阴影多边形在当前扫描线上没有重叠，按正常的扫描线消隐算法进行处理；
 - 阴影多边形与当前扫描线段完全重叠，扫描线段内各像素按阴影计算公式计算光强；
 - 阴影多边形部分覆盖当前扫描线段，需要将线段细分，并递归上述过程，直到出现上述两种情况为止。





阴影细节多边形算法

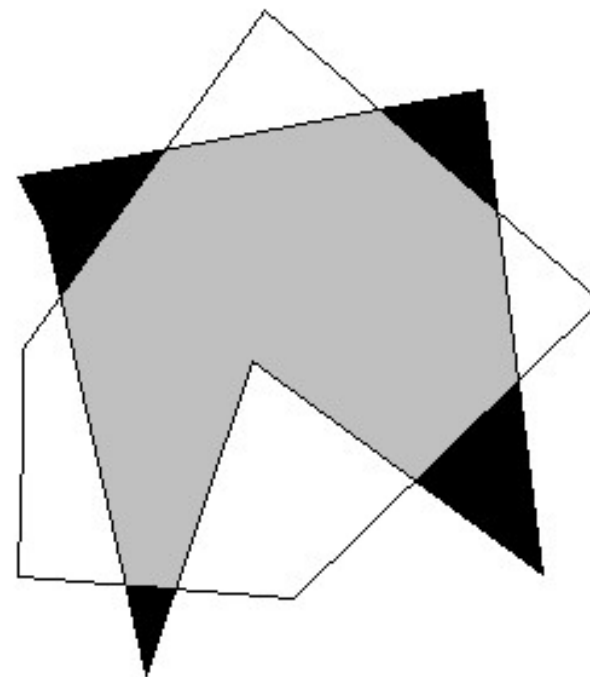
- 阴影生成过程实际上是以光源为视点对场景中的景物进行消隐的过程。
- 阴影区域仅与光源和景物的相对位置有关，而与视点位置无关。
- **1978年Atherton, Weiler, Greenberg**第一次将消隐算法用于阴影生成。



■ Weiler-Atherton消隐算法

■ 基于Weiler-Atherton裁剪算法

- 主多边形和裁剪多边形;
- 交点: 入点和出点;
- 裁剪结果区域由两个多边形的部分边界共同组成;
- 多边形的顶点表;
- 若交点为入点, 在主多边形顶点表内跟踪, 否则在裁剪多边形顶点表内跟踪;



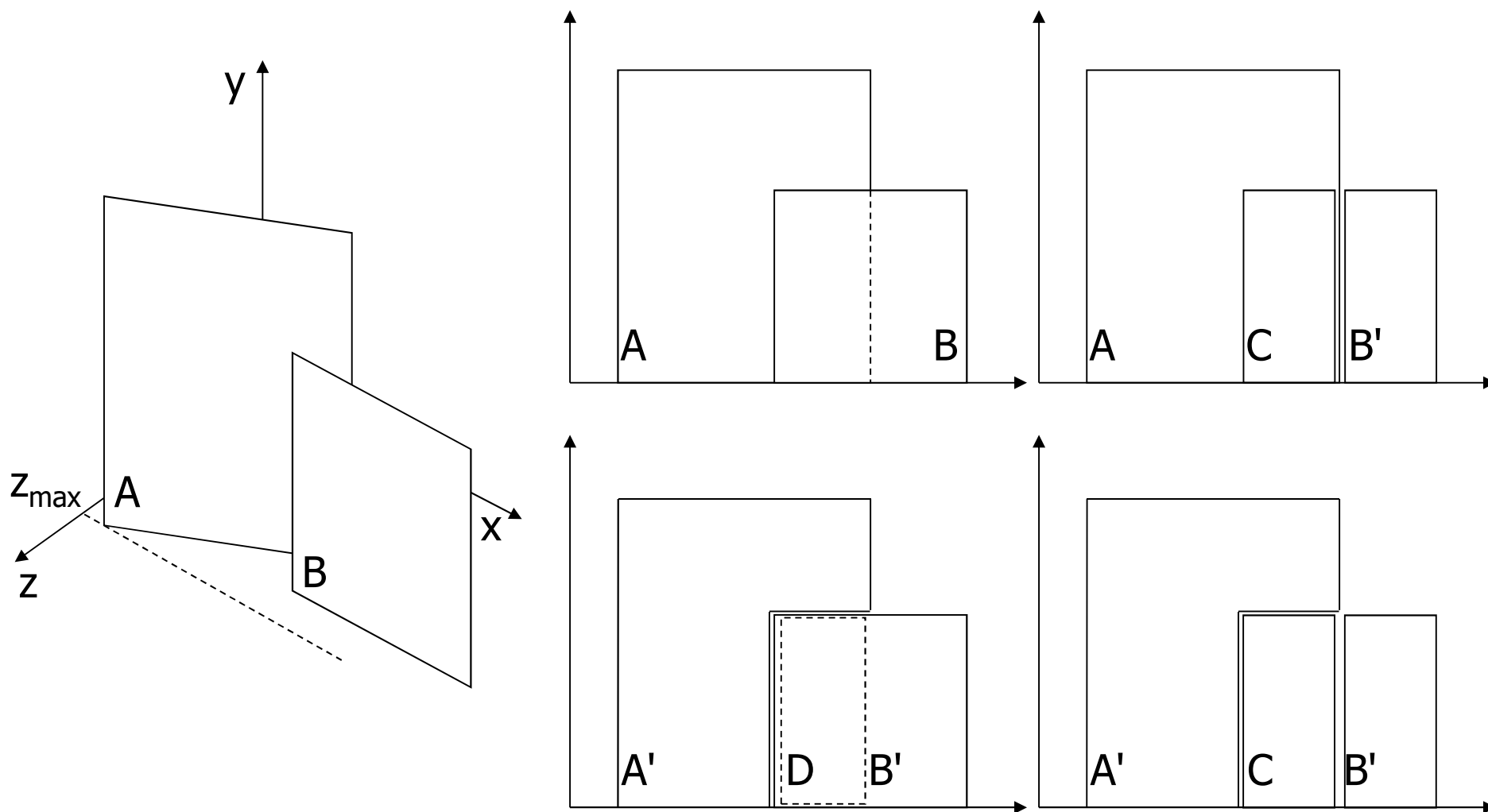


■ Weiler-Atherton消隐算法(物体空间)

- 场景中所有景物多边形按它们距离视点的远近排序，建立近似的深度优先级表；
- 取离视点最近的景物多边形作为裁剪多边形，其余多边形(包括第一个多边形)为主多边形，建内部表和外部表。用**Weiler-Atherton**算法进行裁剪，每个主多边形位于裁剪多边形内的部分被置入内部表中，位于裁剪多边形外的部分被置入外部表；

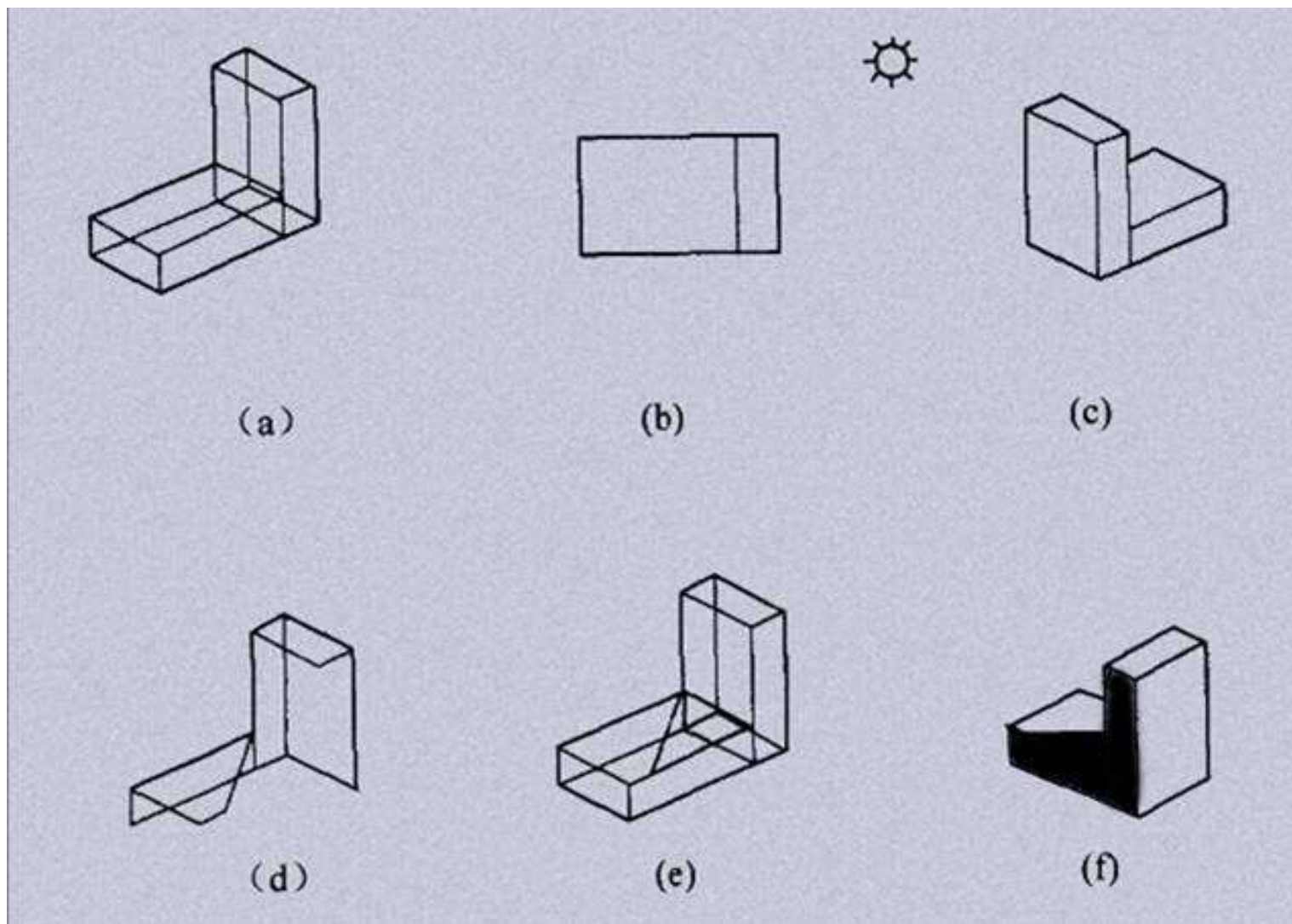


- 内部表中每个多边形深度与裁剪多边形的深度 z_{min} 进行比较
 - 如果内部表中主多边形上的所有顶点深度值都不大于 z_{min} ，则将其从内部表中删除，若内部表中剩余的多边形只有裁剪多边形，则显示，继续处理外部表；
 - 如果内部表中某个多边形顶点深度大于 z_{min} ，该主多边形至少部分位于裁剪多边形之前，用该多边形的原形作为新的裁剪多边形，原内部表作为主多边形表，进行裁剪；





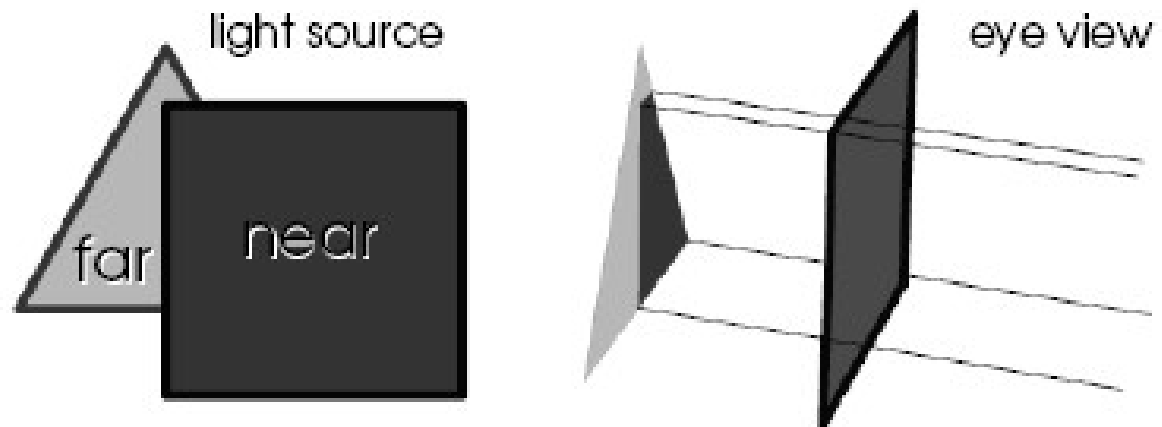
- 阴影细节多边形算法(两步法)步骤:
 - 取光源方向为视线方向对景物进行消隐：用 **Weiler-Atherton** 消隐算法确定对光源可见的景物多边形(称为细节多边形)，并通过标识数将这些多边形与它们覆盖的原始景物相联系；
 - 取视线方向对景物进行消隐：如果多边形某部分相对视点可见，但未覆盖细节多边形，则该部分光亮度按阴影计算。





阴影Z-Buffer算法(Shadow mapping)

- 阴影生成过程实际上是以光源为视点
对场景中的景物进行消隐的过程。
- **1978年Williams**
- 两步法：一遍从光源方向，一遍从视点方向





■ 算法

- 以光源作为视点，在光源坐标系用**Z-Buffer**算法对景物进行消隐，此时阴影缓冲器中存储的是那些离光源最近的景物的深度值 (**shadow map**);
- 用**Z-Buffer**算法按视线方向进行消隐，对每个像素计算离视点最近的景物表面的可见点，将该点变换到光源坐标系下，记为 \mathbf{z}' ，并取对应的阴影缓冲器中的深度值 \mathbf{z}_{map} 进行比较，若 $\mathbf{z}' > \mathbf{z}_{\text{map}}$ ，则按阴影区域光强显示，否则，正常计算该点的光强。



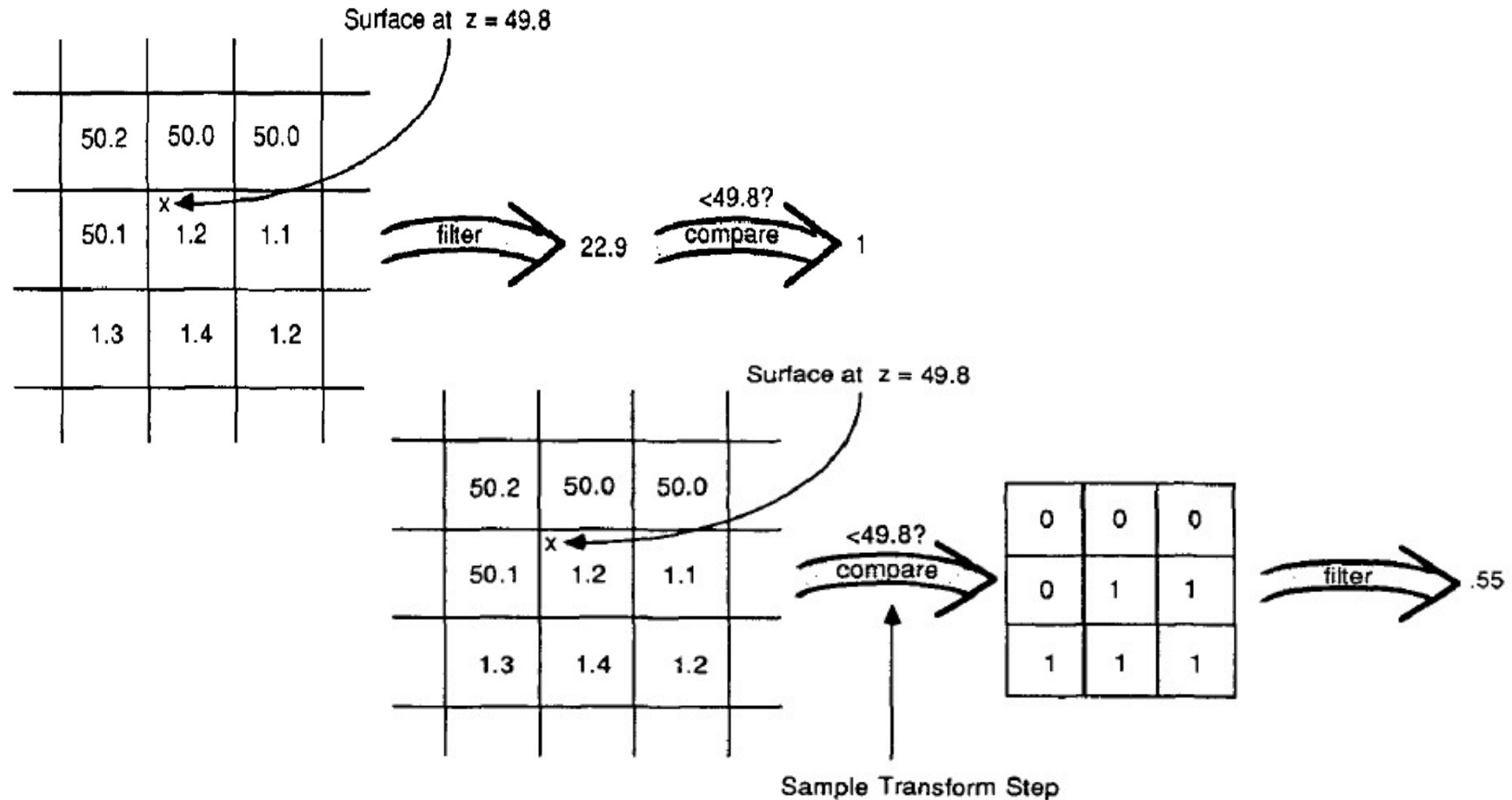
■ 特点

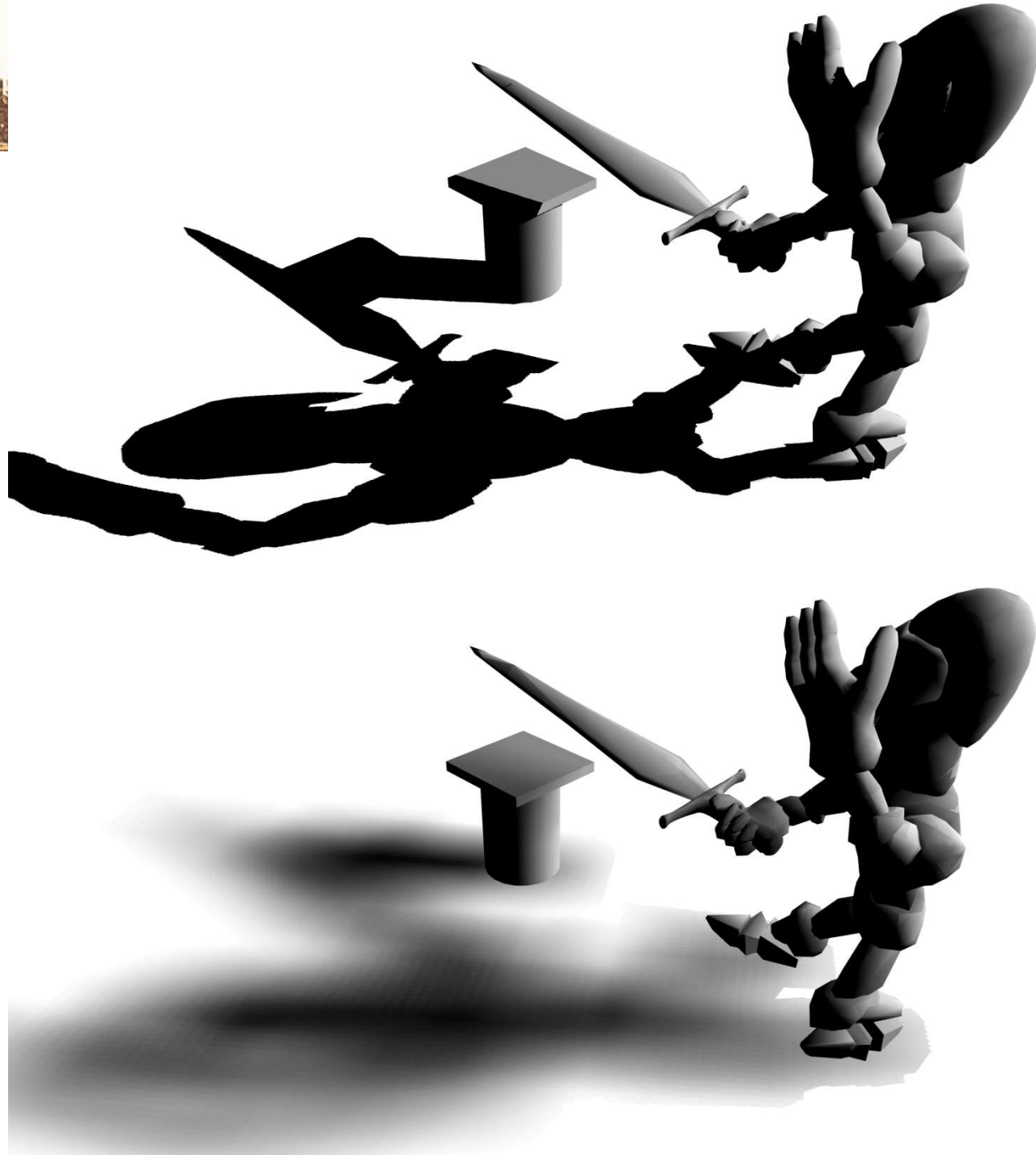
- 算法简单，而且能处理任意复杂的景物，计算其近似阴影；
- 只要光源与物体位置固定， **shadow map** 可重复利用；
- 对多个光源，需要一个一个计算；
- 当光源偏离视线方向较远时，在阴影区域附近会产生走样；



■ 软阴影生成(Percentage-Closer Filtering)

- 在一邻域内比较, 得到阴影权值





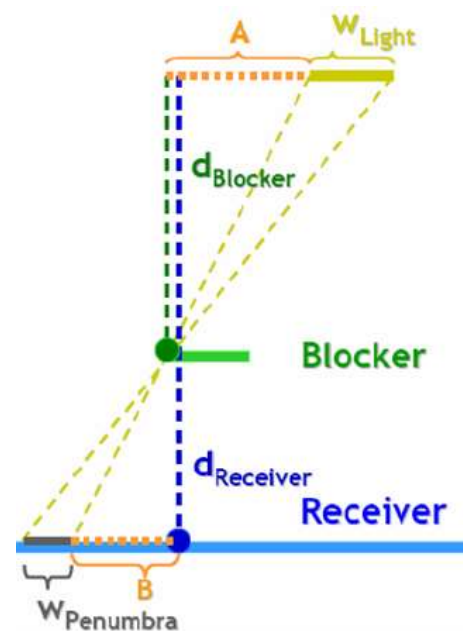
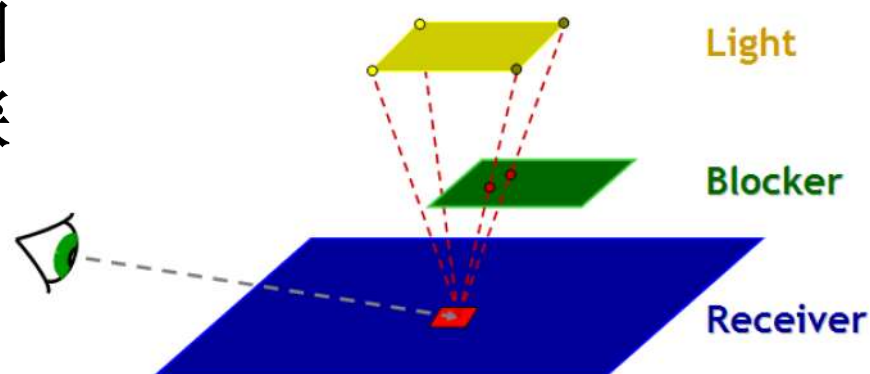


■ Percentage-Closer Soft Shadows

- 在阴影图中搜索与物体相关的遮挡区域，并计算深度平均值
- 光源尺寸
- 物体到光源的距离
- 估计半影大小

$$w_{Penumbra} = \frac{(d_{Receiver} - d_{Blocker}) \cdot w_{Light}}{d_{Blocker}}$$

- 用**PCF**生成阴影，邻域大小正比于半影大小

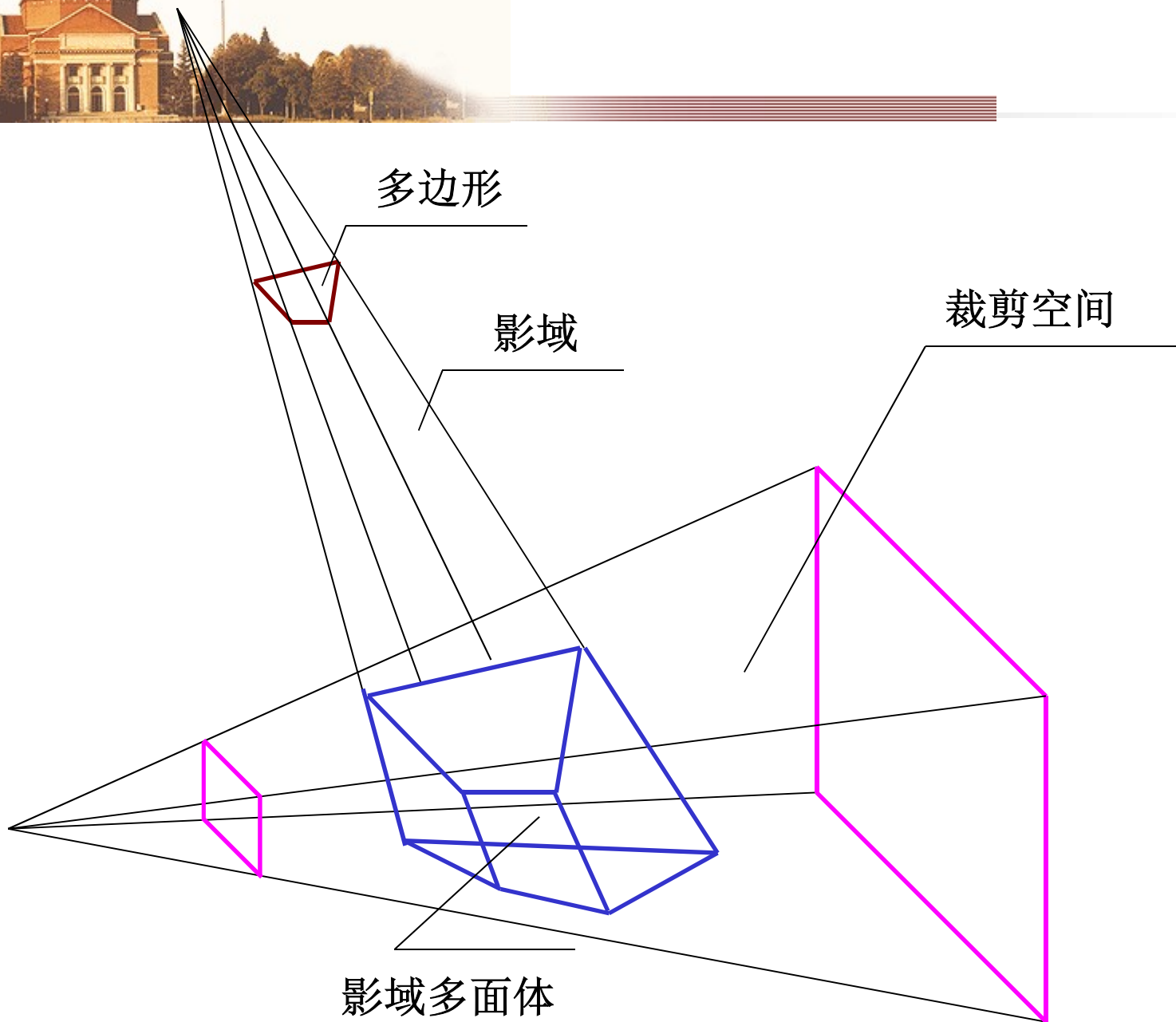
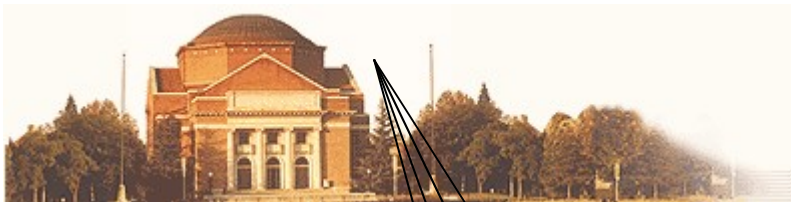






影域多面体算法(Shadow Volume)

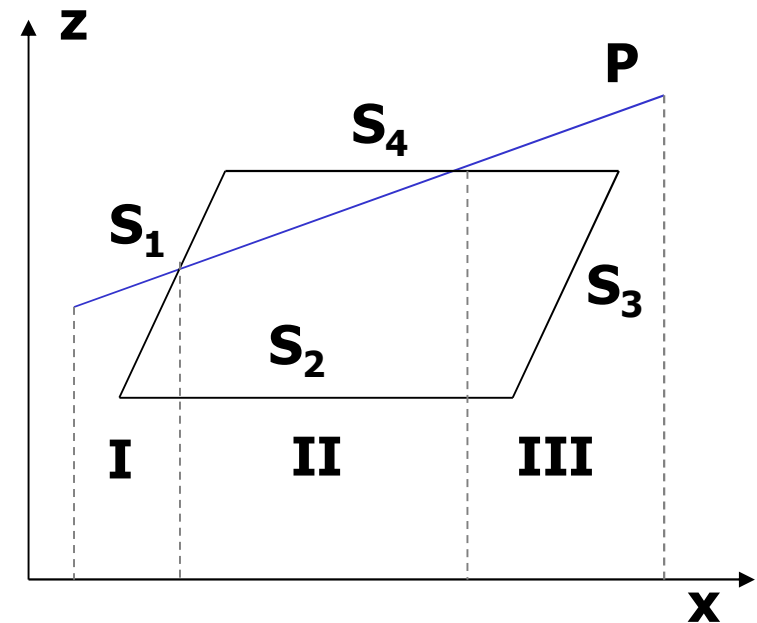
- **影域：**光源直接照射到不透明表面，在该表面后面会形成的三维的阴影区域。
- **影域多面体：**用裁剪空间对三维影域进行裁剪得到的封闭多面体。
- **影域多边形：**影域多面体上各侧面多边形，非实际存在的景物表面。





■ 1977年Crow基于扫描线消隐算法提出

- 将影域多边形与景物多边形一起参加扫描和排序；
- 每个扫描平面与各多边形相交，形成一系列截交线段
 - 对每一个景物多边形与影域多面体在当前扫描平面上的截交线段(图中 P 与 S_1, S_2, S_3, S_4)，根据其相交情况，判断线段 P 中各部分是否处于阴影内



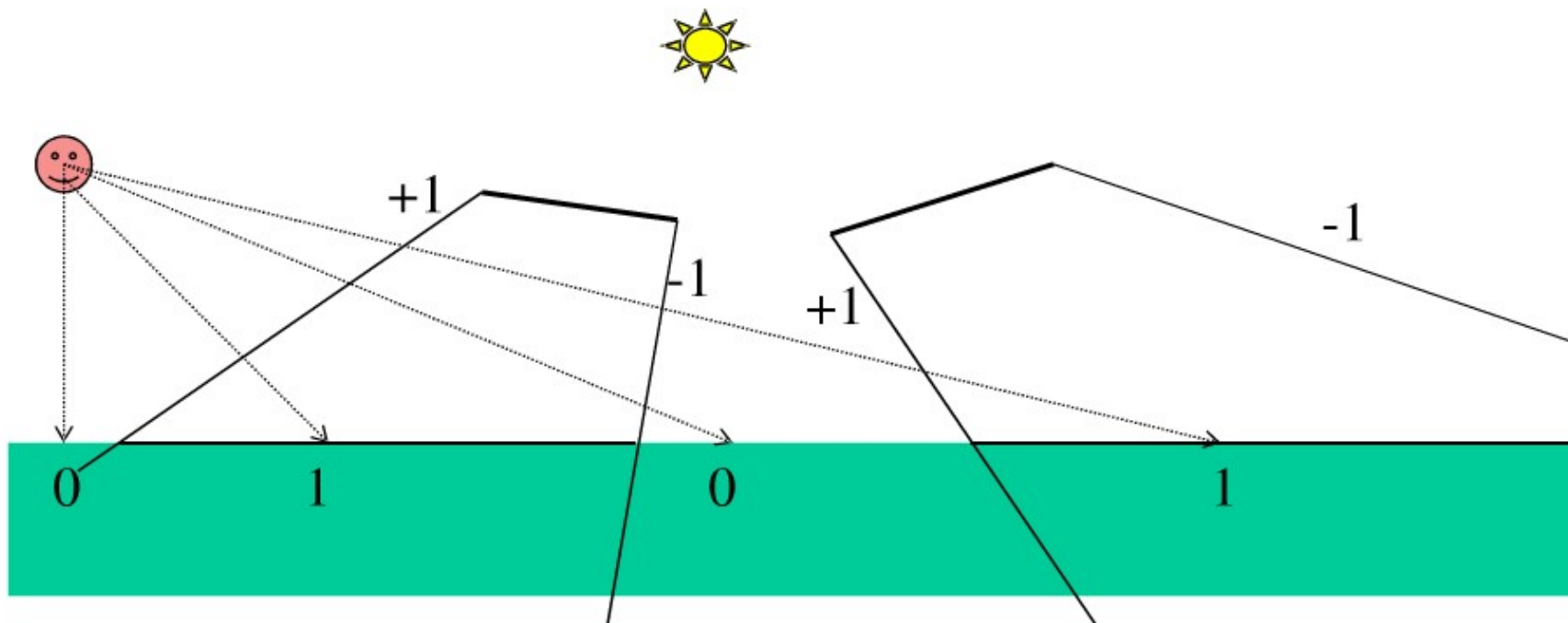


Z-Pass

{

- 用**Z-Buffer**算法绘制场景，**ZB**中记录各像素对应场景深度；
- 处理所有**shadow volume**
 - { 对前向影域面覆盖的像素，如果影域面深度大于**ZB**对应值，则**stencil+1**，否则**stencil**不变；
对后向影域面覆盖的像素，如果影域面深度大于**ZB**对应值，则**stencil-1**，否则**stencil**不变； }
- 根据像素**stencil**判断是否位于阴影内，若**stencil>0**，则计算阴影光强；

}



- **shadow volume**的生成
- 在传统扫描线算法的基础上稍加扩充
- 由于存在大量影域多边形，增加了场景复杂性，影响了算法效率



大作业选题报告

- 汇报时间
 - **10月22日(第7周周二)**
 - 每组陈述**7分钟**
- 汇报内容
 - 题目、要实现的功能、拟采用的方法、团队分工等
- 修改完善后的**PPT**提交时间
 - **10月27日(周日)晚24:00**