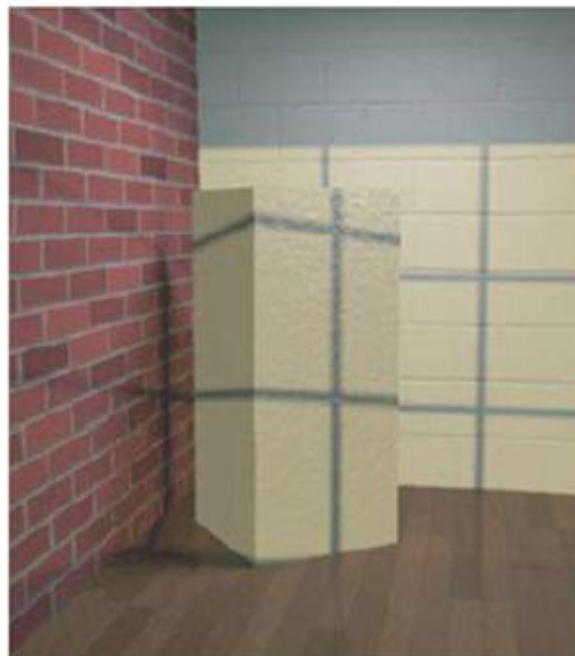


纹理映射

一、 什么是纹理和纹理映射？



没有纹理



有纹理

二、 纹理有什么用？

- n 表面可以用纹理来代替，不用痛苦地构造模型和材质细节，节省时间和资源，让用户做其他更重要的东西。
- n 可以用一个粗糙的多边形和纹理来代替详细的几何构造模型，节省时间和资源。

二、 纹理作用



+



=



三、 纹理分类

颜色纹理：颜色或明暗度变化体现出来的表面细节，如刨光木材表面上的木纹。



几何纹理：由不规则的细小凹凸体现出来的表面细节，如桔子皮表面的皱纹。

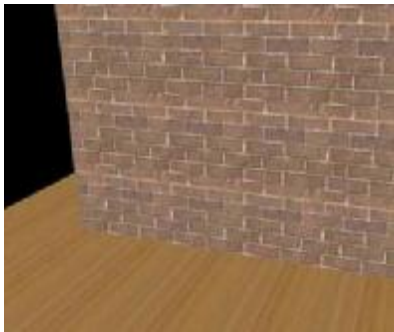


四、图形学中纹理定义

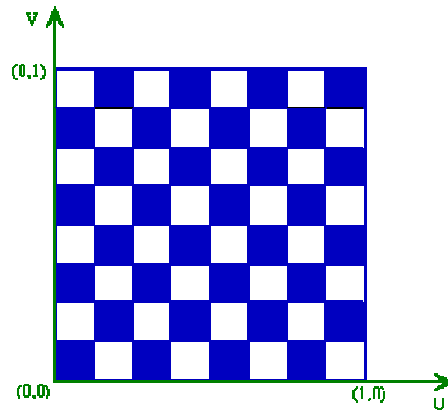
在真实感图形学中，可以用下列两种方法来定义纹理：

(1) 图象纹理：将二维纹理图案映射到三维物体表面，绘制物体表面上一点时，采用相应的纹理图案中相应点的颜色值。

(2) 函数纹理：用数学函数定义简单的二维纹理图案，如方格地毯；或用数学函数定义随机高度场，生成表面粗糙纹理即几何纹理。



图像纹理



函数纹理

$$g(u,v) = \begin{cases} 0 & [u \times 8] + [v \times 8] \text{ 为奇数} \\ 1 & [u \times 8] + [v \times 8] \text{ 为偶数} \end{cases}$$

其中： $\lfloor x \rfloor$ 表示小于x的最大整数

五、 纹理映射

纹理映射 (Texture Mapping) : 通过将数字化的纹理图像覆盖或投射到物体表面，而为物体表面增加表面细节的过程。

1974年Catmull首次提出了纹理映射的概念，其主要思想是通过寻找一种从纹理空间 (u, v) 到三维曲面 (s, t) 之间的映射关系，将点 (u, v) 对应的彩色参数值映射到相应的三维曲面 (s, t) 上，使三维曲面表面得到彩色图案。

五、 纹理映射

颜色纹理坐标转换通常使用下列两种方法：

(1) 在绘制一个三角形时，为每个顶点指定纹理坐标，三角形内部点的纹理坐标由纹理三角形的对应点确定。

即指定：

$$(x_0, y_0, z_0) \rightarrow (u_0, v_0)$$

$$(x_1, y_1, z_1) \rightarrow (u_1, v_1)$$

$$(x_2, y_2, z_2) \rightarrow (u_2, v_2)$$

(2) 指定映射关系：

$$u = a_0x + a_1y + a_2z + a_3$$

$$v = b_0x + b_1y + b_2z + b_3$$

五、 纹理映射

几何纹理使用一个称为扰动函数的数学函数进行定义。

扰动函数通过对景物表面各采样点的位置作微小扰动来改变表面的微观几何形状。

设景物表面由下述参数方程定义： $Q = Q(u, v)$

则表面任一点 (u, v) 处的法线为： $N = N(u, v) = \frac{Q_u(u, v) \times Q_v(u, v)}{|Q_u(u, v) \times Q_v(u, v)|}$

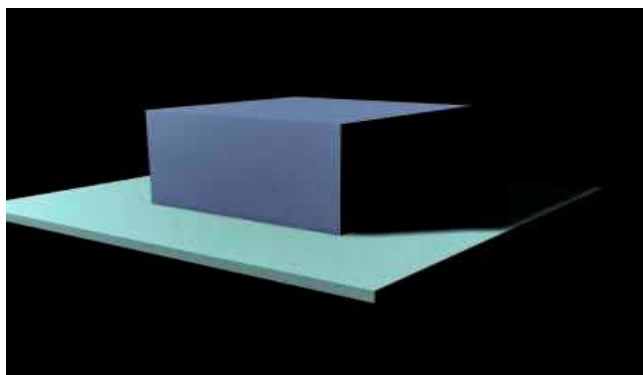
设扰动函数为： $P(u, v)$

扰动后的表面为： $Q' = Q(u, v) + P(u, v)N$

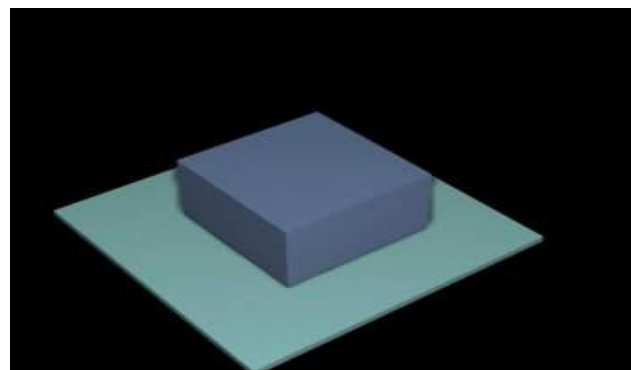
阴影处理

一、 什么是阴影？

- n 阴影是由于观察方向与光源方向不重合而造成的；
- n 阴影使人感到画面上景物的远近深浅，从而极大地增强画面的真实感。

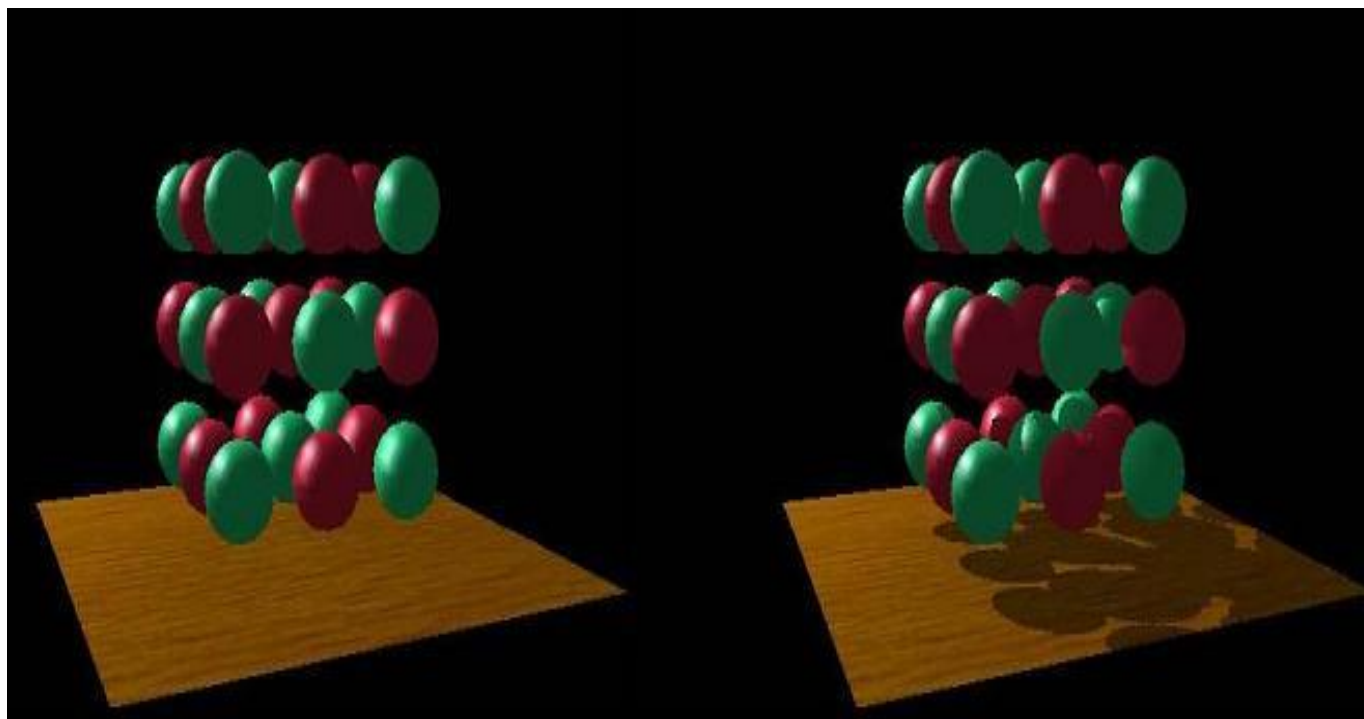


观察方向与光源方向不重合

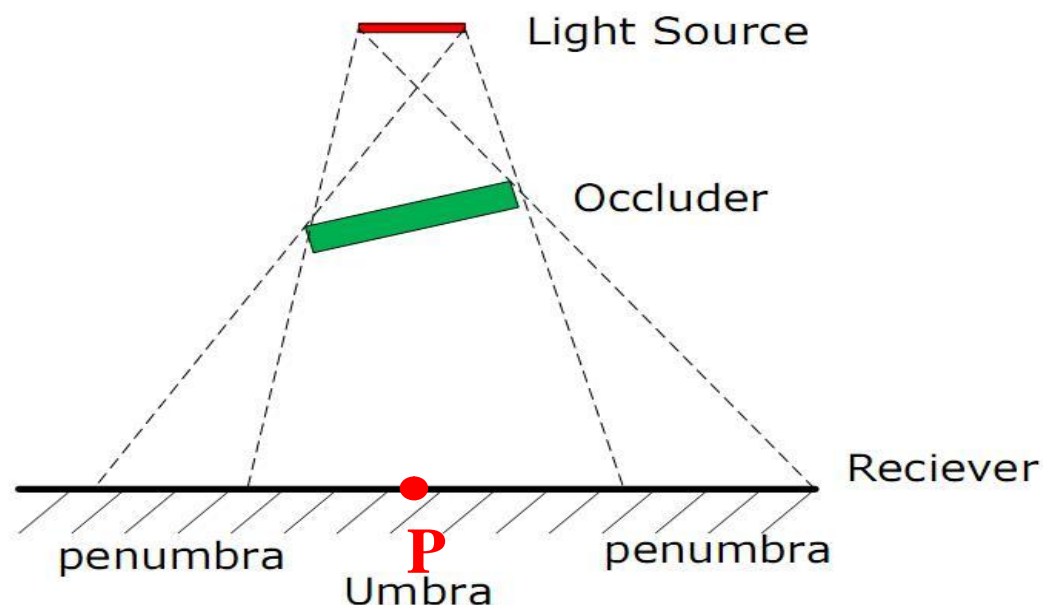


观察方向与光源方向重合

一、 阴影有什么用？

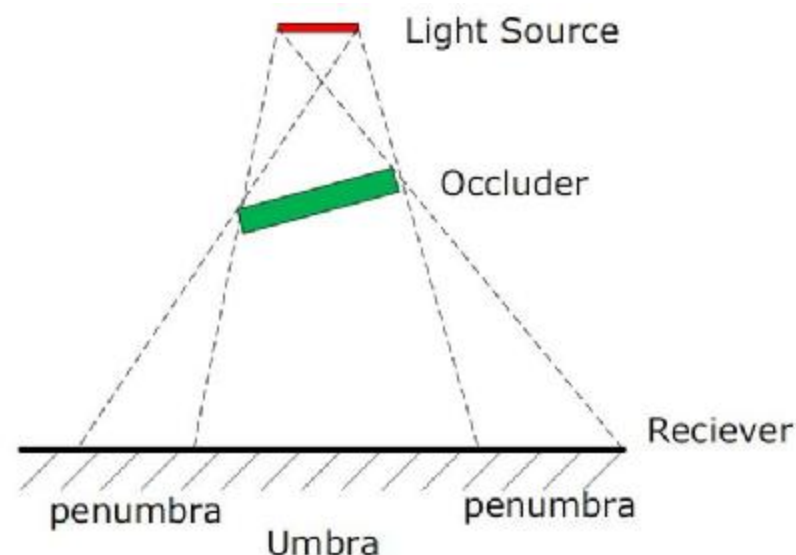


二、 什么是本影



umbra- 本影区-场景中的一个点P，如果它不被光源的任何一部分所照射到，就称为在本影区里。本影就是不被任何光源所照到的区域。

二、什么是半影



Occluder-遮挡物：阴影的生成是因为空间有遮挡物。这些遮挡物把光源挡住了，所以在receivers上有些部分就很阴暗。

阴影是本影和半影的组合。求出本影和半影的并集（union）来绘出阴影。

二、 阴影

- n **自身阴影**：由于物体自身的遮挡而使光线照射不到它上面的某些面；
- n **投射阴影**：由于物体遮挡光线，使场景中位于它后面的物体或区域受不到光照射而形成的。

三、 阴影算法（1）

阴影体法（ **Shadow Volume** ）

Frank Crow1977年提出来的，可以在任意的物体上生成阴影。

Crow, Franklin C: "**Shadow Algorithms for Computer Graphics**",
Computer Graphics (SIGGRAPH '77
Proceedings), vol. 11, no. 2, 242-
248.

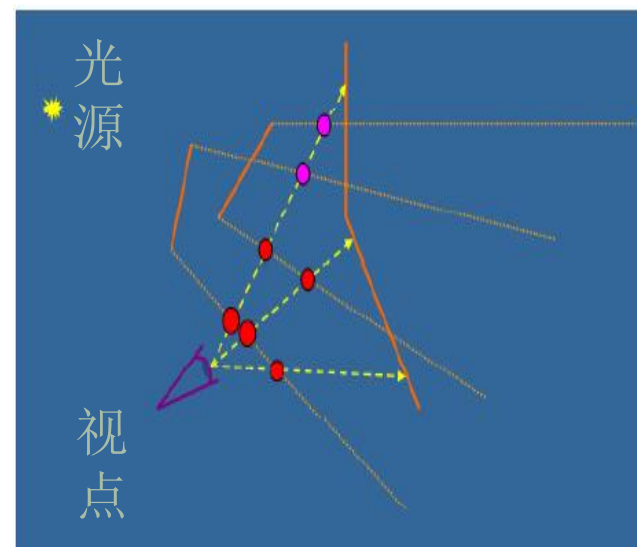


三、 阴影算法（1）

阴影体法（ **Shadow Volume** ）

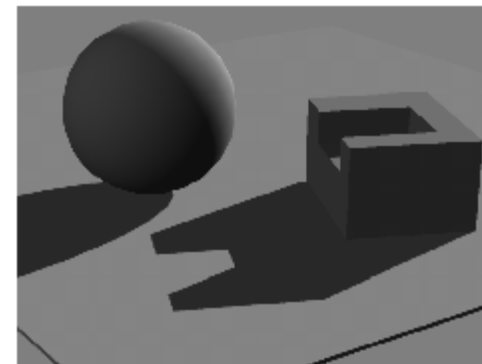
由一个点光源和一个三角形可以生成一个无限大的阴影体。落在这个阴影体中的物体，就处于阴影中。

在对光线进行跟踪的过程中，如果这条射线穿过了阴影体的一个正面（朝向视点的一个面），则计数器加1。如果这条射线穿过了阴影体的一个背面（背向视点的一个面），则计数器减1。如果最终计数器的数值大于0，则说明这个像素处于阴影中，否则处于阴影之外。



三、 阴影算法（2）

阴影图法（ **Shadow Mapping** ）



- n 这种方法的主要思想是使用Z缓冲器算法，从投射阴影的光源位置对整个场景进行绘制。
- n 这时，对于Z缓冲器的每一个像素，它的z深度值包括了
这个像素到距离光源最近点的物体的距离。一般将Z缓冲器中的整个内容称为阴影图（Shadow Map），有时候也称为阴影深度图。

三、 阴影算法（2）

阴影图法（ **Shadow Map** ）

- n 为了使用阴影图，需要对场景进行二次绘制，不过这次是从视点的角度来进行的。
- n 在对每个图元进行绘制的时候，将它们的位置与阴影图进行比较，如果绘制点距离光源比阴影图中的数值还要远，那么这个点就在阴影中，否则就不在阴影中。

