

11.1 Snell 定律与 Fresnel 定律

11.1.1 折射率与 Snell 定律

光在真空中的速度 **c** 与在透明介质中的速度 **v** 之比，称之为该介质的绝对折射率，简称折射率。光在真空中的折射率等于 1，通常我们认为光在空气中的折射率也近视为 1。 如下公式所示：

$$n = \frac{c}{v}$$

折射率较大介质的称为光密介质，折射率较小的介质称为光疏介质。下面给出一些常用介质的折射率，如表所示：

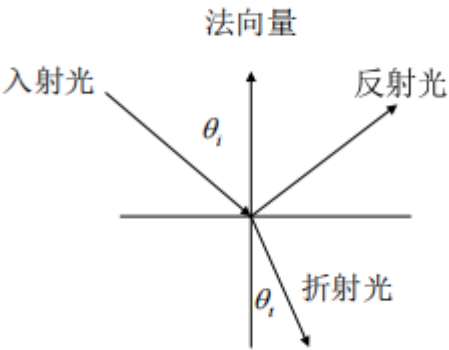
| 材质 | 折射率值 |
|-------|-------------|
| 真空\空气 | 1.0\ 1.0003 |
| 水 | 1.333 |
| 玻璃 | 1.5-1.7 |
| 钻石 | 2.417 |
| 冰 | 1.309 |

常用介质折射率表

Snell 定律描述光线从一个介质传播到另一个介质时，入射角、折射角和介质折射率的关系。如下图所示，假设光线从空气射入水面，入射角度为 θ_i ，空气对光线的折射率为 n_i ，折射率角度为 θ_t ，水对光线的折射率为 n_t ，则存在：

$$\sin \theta_i * n_i = \sin \theta_t * n_t$$

通过 **snell** 定律，我们可以根据入射光的方向向量求取折射光的方向向量。折射率本质上反映的是光在介质中的运行速度，以及折射方向；



11.1.2 色散

色散分为正常色散和反常色散，通常我们所说的色散都是指反常色散，即，对光波透明的介质，其折射率随着波长的增加而减小。1672 年，牛顿利用三棱镜将太阳光分解为光谱色带（红橙黄绿蓝靛紫），这是人类首次所做的色散试验。天空中的彩虹也是由于光与水蒸气交互形成的色散现象（图 29 所示）。法国数学家柯西在 1936 年首先给出了正常色散的经验公式，称为柯西公式。

计算机中的颜色是三原色，即只有红、绿、蓝三种颜色，给出光的红、绿、蓝分量的不同折射系数，可以近似模拟色散效果。

11.1.3 Fresnel 定律

光线照射到透明物体上时，一部分发生反射，一部分进入物体内部并在介质交界处发生折射，被反射和折射的光通量存在一定的比率关系，这个比率关系可以通过 Fresnel 定律进行计算。根据 Fresnel 定律计算得出的数据称为 Fresnel 系数。严格而言，fresnel 系数分为 fresnel 反射系数和 fresnel 折射系数，通常我们所说的 fresnel 系数指“反射系数”。

schlick 给出了 Fresnel 反射系数的一个近似，精度在 1% 范围内：

$$F = f_0 + (1 - f_0)(1 - V \cdot H)^5$$

- f_0 为入射角度接近 0（入射方向靠近法向量）时的 Fresnel 反射系数；
- V 是指向视点的观察方向；
- H 为半角向量；

观察公式，可以得出一个结论：随着入射角趋近 90，反射系数趋近 1，即擦地入射时，所有入射光都被反射。

在 fresnel equations-Wikipedia 中列举了当入射角度接近 0 时的 fresnel 反射系数的计算方法：

$$f_0 = \frac{(n_i - n_t)^2}{(n_i + n_t)^2}$$

求出 fresnel 反射系数后，用 1 减去该系数，就得到了折射系数，所以当入射角度接近 0 时的 fresnel 折射系数的计算方法为：

$$f_{t0} = 1 - f_0 = \frac{4n_i n_t}{(n_i + n_t)^2}$$

综合，fresnel 反射系数的计算公式为：

$$F \approx \frac{(n_i - n_t)^2}{(n_i + n_t)^2} + \frac{4n_i n_t}{(n_i + n_t)^2} * (1 - V \bullet H)^5$$

上述计算公式比较消耗时间，所以通常在程序中使用入射角接近 0 时的fresnel 系数。下面的公式计算精度不高，但是胜在计算速度快，利于硬件实现，所以如果想动态的计算 fresnel 系数，而又不希望消耗太多的时间，可以采用公式：

$$F \approx (1 - V \bullet H)^4$$

普通玻璃的反射系数大约是 4%。

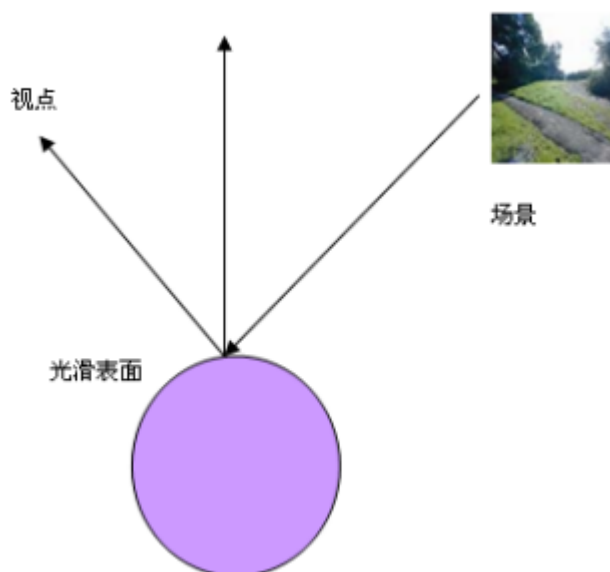
11.2 环境贴图

环境贴图（**Environment Mapping, EM**）也称为反射贴图（**Reflection Mapping**），用于模拟光滑表面对周围场景的映射效果。

环境贴图，顾名思义，在一副图片上展现周围的环境。环境贴图假设进行反射的光源和物体都位于很远的位置，同时反射体不会反射自身。最常被用到的环境贴图是立方体环境贴图。该方法是将相机放置在环境的最中央，然后从上、下、左、右、前、后，6 个方向拍摄周围环境，最后将这些信息投影到立方体的 6 个面上，所得到的纹理称之为立方体环境贴图。

环境贴图的文件表现形式是一种纹理类型，例如立方体环境贴图在 DirectX中可以保存为.dds 格式，也很多开源图形引擎中使用 6 个方向上的二维纹理动态组成环境贴图。除了立方体环境贴图，还有球面贴图、抛物面贴图。不过这两种贴图类型，与立方体环境贴图相比没有明显的优势。

使用环境贴图，是为了模拟光滑表面对周围场景的映射效果。光滑表面对周围场景的映射，是由从场景出发的光线投射到光滑表面上然后被反射到人眼所形成的视觉效果，如图所示：



我们将这个过程倒退回去，从视点发射一束射线到反射体上的一个点，然后这束射线以这个点为基准进行反射，并根据反射光线的方向向量检索环境图像的颜色。这就是环境贴图算法的基本思想。

环境贴图算法的步骤如下：

1. 首先根据视线方向和法向量计算反射向量；
2. 然后使用反射向量检索环境贴图上的纹理信息；
3. 最后将该纹理信息融合到当像素颜色中。

环境贴图在 Cg 语言中属于 `samplerCUBE` 类型变量，标准函数库中的 `texCUBE` 函数提供依据反射方向向量检索环境贴图的功能。



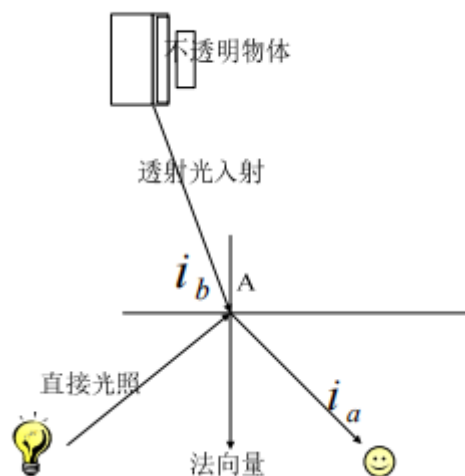
环境映射实现效果

11.3 简单透明光照模型

简单透明光照模型不考虑透明物体对光的第二次折射、次表面散射，以及光在穿越透明物体时的强度衰减，只是简单的使用颜色调和的方法，即我们最终所看到的颜色，是物体表面的颜色和背景颜色的叠加。

如图所示，透明物体位于视点与另一个不透明物体之间，透明物体的不透明度为 t ，点 A 为透明物体上的一点，点光源直接照射到 A 点上产生的反射光强为 i_a ，视线穿过透明体与另一个物体相交处的光强为 i_b ，则点 A 处的最终可观察的光强为：

$$i = (1 - t) * i_b + t * i_a$$



简单透明光照模型示意图

i_a 和 i_b 都可以用前面所讲的光照模型进行计算。通过透射光方向计算透射光强，首先需要进行光线和空间物体的求交运算，以确定透射光的来源，这是非常消耗时间的，如果真的这样做，其实就是演变为光线跟踪算法了。为了保证实时性，在实际使用中，通常是根据入射光方向向量和法向量求取折射光方向，然后根据折射光方向检索环境纹理上的颜色值作为 i_b 。简单透明光照模型渲染效果如图所示：



简单透明光照模型渲染效果

11.4 复杂透明光照模型与次表面散射

光射入透明物体时会发生一次反射和折射，光从透明物体内射出时，又会发生一次反射和折射。透明光照的简单模型实际上只是通过计算了第一次反射和折射，近似的模拟光透效果。2005 年，Wyman 在

ACM SIGGRAPH 大会上提出了在 GPU 中用近似的方法实现两次折射的透明物体绘制算法（Interactive image-space refraction of nearby geometry）。

次表面散射是光射入半透明物体后再内部发生散射，最后射出物体并进入视野中产生的现象。次表面散射材质是高质量渲染中最复杂的材质之一，一个重要原因在于此表面散射物体内部的任何一点的光照度取决于体内其他点的光照度和材质本身的透光率。抛开材质本身的性质不说，这一特性使得次表面散射的光照方程变成一个复杂的微分方程，求出此方程的准确解是十分困难的，另一方面，材质本身可能具有复杂的各向异性和不均匀密度等性质，因此计算这样的积分变得非常困难。