Photon Mapping算法原理简介

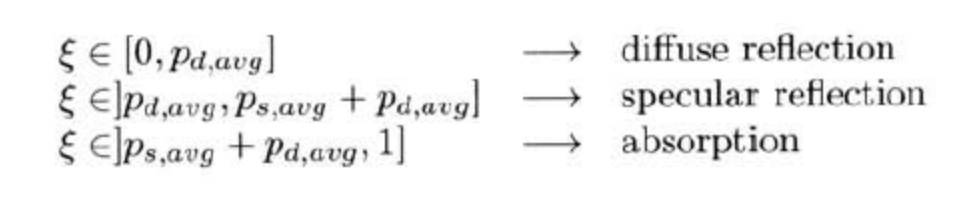
光子映射算法相较于传统路径追踪算法更容易模拟SDS路径传导（即光线经由镜面反射到漫反射表面，再由漫反射表面反射到镜面上）。光子映射算法分为两个部分：光子追踪和渲染。

1. 光子追踪
2. 投影图

由于光源投射光子的是向很多方向的（比如点光源是投向四面八方的），但有些方向是没有景物的，所以往那些方向投射光子也没有什么意义。这时引入投影图，即将光源最开始要投的所有方向集合成一张投影图，再把这张投影图细分成一个一个的cell，如果这个cell里有景物，那么这个cell的状态就是”on”，否则就是”off”状态。而光源实际上仅向”on”状态的cell方向投射光子即可。

1. 俄罗斯转盘

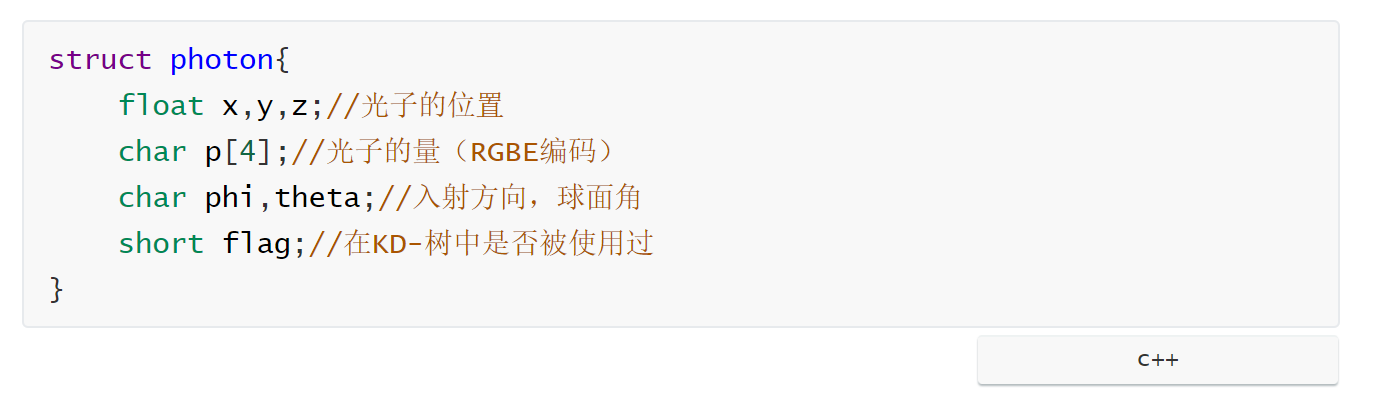
真实情况下光子碰撞到景物表面可能一部分被漫反射，一部分被镜面反射，一部分被吸收，那么光子所具有的能量也会被分成若干部分。而PM算法为了简化，使用俄罗斯转盘的算法，即使得一个光子不会分裂为几部分，而是一个光子看成一个整体在一定的概率下仅有一个结局（比如这个光子有多大概率被漫反射，多大概率被吸收等等）。而俄罗斯转盘的具体实现如下:



1. KD-树

光子在被漫反射之后需要存储到光子映射中。而渲染这一步需要快速地查找到某一点的辐照度而据此进行渲染，因此使用KD-树进行对光子进行存储。而KD-树则用数组的方法进行存储，下标0表示root,2k表示左子树，2k+1表示右子树。

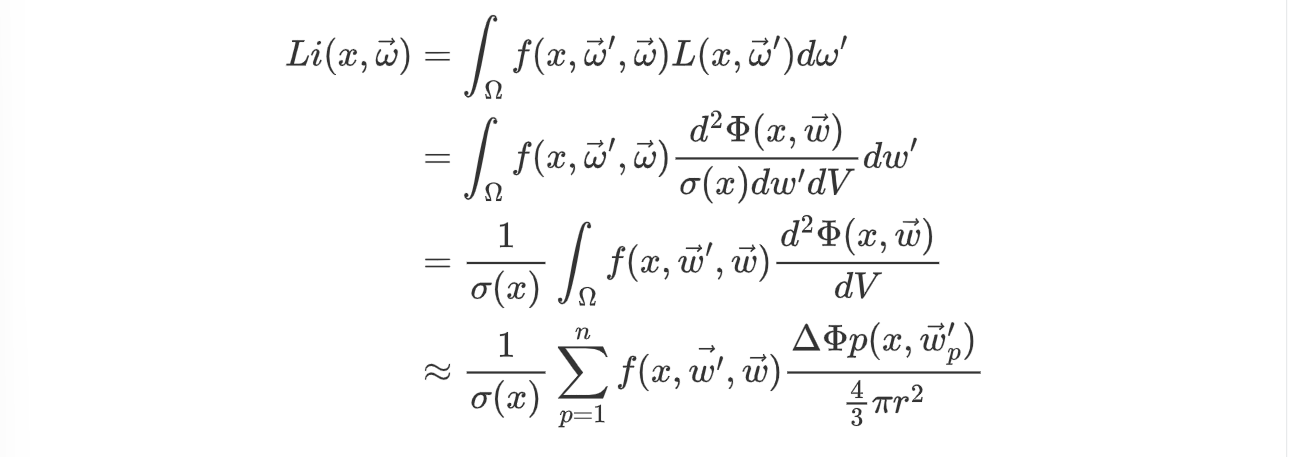
每个光子的结构如下：



1. 渲染

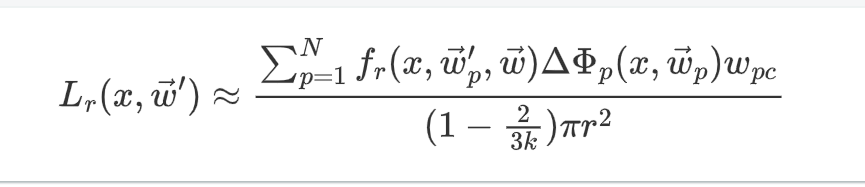
1.K-近邻估计法

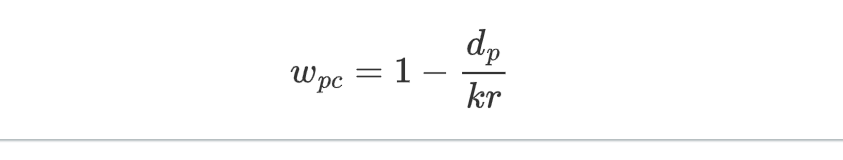
如果要根据某一点的辐照度进行渲染，那么可以使用K-近邻估计法，即在空间中距离此点小于r的距离中找到刚好k个最近的光子，用它们的亮度分布来估算目标点的亮度。其公式如下:



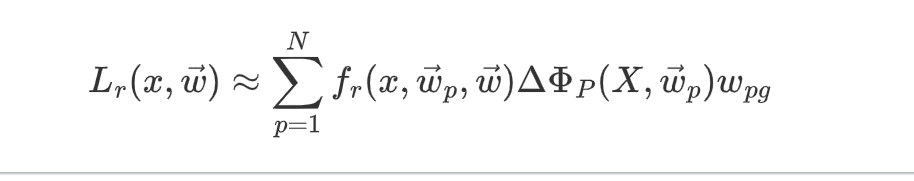
2.锥形滤波和高斯滤波

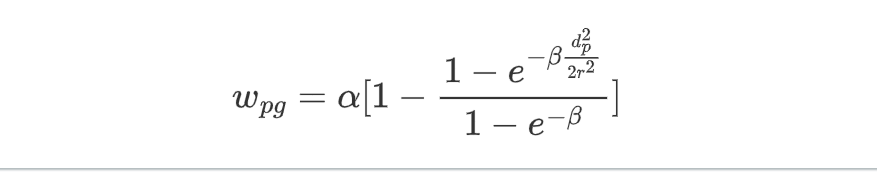
经过K-近邻估计法的处理，由于光子分布不均匀以及r取值不确定，所以光照图的边界会变得模糊，所以要对照度进行滤波

1. 锥形滤波:  
   



2)高斯滤波





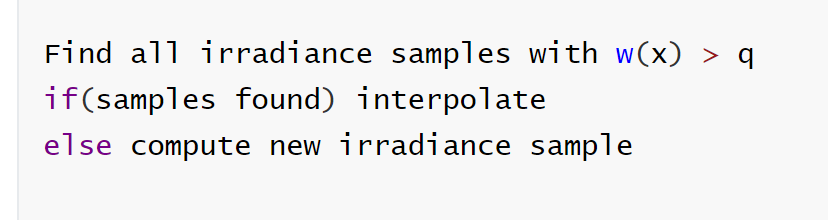
3.Final Gather

如果光子数量不够充分那么会产生大量低频噪声，那么FG在每个采样点的上半球区域向所有方向发射采样射线，利用这些射线与场景求交，而后获取交点处的照度叠加到采样点上求均值。

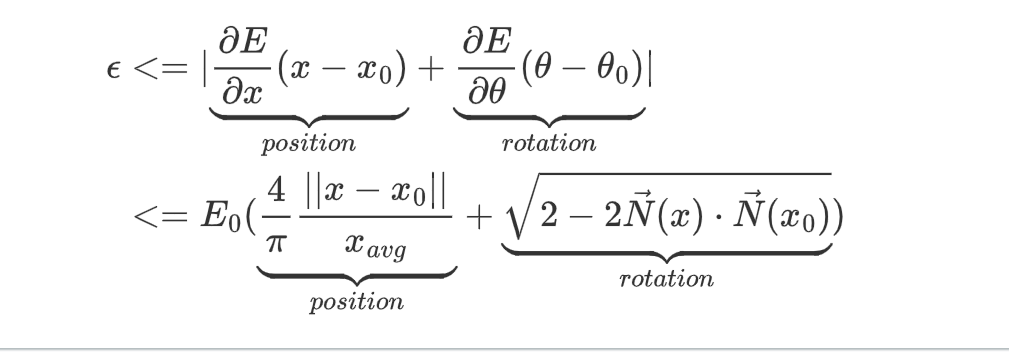
1. 照度缓存

假设照度的变换是连续且缓慢的，那么大部分照度值可以利用已有的照度值通过梯度插值计算出来。

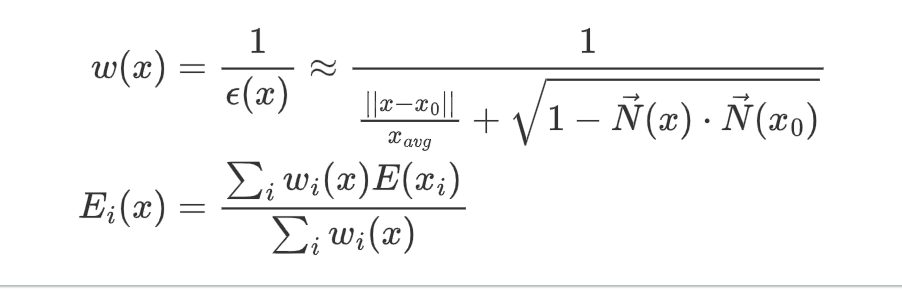
算法:



平移和旋转梯度:



插值方式;



1. 缺点:
2. 因为K近邻估计法的使用以及带有滤波的评估，导致近似有偏；使用球面估计时过高估计，使得边缘过亮，导致边界有偏；使用球面估计时使得物体表明照度大于实际照度，导致拓扑有偏。
3. 俄罗斯轮盘的使用会加大方差使得收敛变慢。