

光线追踪原理

1.概念区分

“光线追踪”为生成3D图像的方式，其基础为追踪从摄像机到物体的光线路径。“路径追踪”为带有随机性质的光线追踪，是光线追踪的增强，在光线追踪的基础上进行随机采样。光线追踪进行基于物理的渲染，能够渲染出符合材质物理性质的较为真实的画面。

本文档将二者统称为光线追踪。标题或文字后的中括号中的名称表示此光线追踪实现中的标识符，可在项目中查找对应实现。

2.工作流程

2.1：空间和摄像机构建[Vec3, Point3, Camera]

光线追踪器通过三维向量（Vec3）类和三维点（Point3）类定义空间坐标系，并在此坐标系中定义空间物体、摄像机及其视口（viewport）。

视口作为一个有范围的空间平面（长方形），规定每一条光线发出的位置。通过定义视口平面的中心点、视口平面的大小和摄像机“看向”的点，构造以视口中心点为原点的视口坐标系，并以此得出光线发出的方向（所有光线均垂直于视口平面发出）。

2.2：发射光线[Camera::render]

通过视口尺寸（预先定义）和画面分辨率（渲染时设定），将连续的视口平面离散为多个点，这些点从视口左上角开始，像右下方均匀地分布在视口平面内，作为光线的起点。光线的初始方向在摄像机构建时确定。

在每一个像素上，构造光线 $P(t) = O + t\vec{D}$ ，其中O为光线起点，D为方向向量，t为空间参数，通过t定位光线这条射线在参数为t时的空间位置（一个点），从而与物体进行交互。

2.3：光线和物体相交[AbstractHittable::hit]

光线从视口中发出，在场景中因碰撞到物体而反弹，并在每一次反弹中接受来自物体的影响，改变其所携带的颜色，并将最后一次反弹结束（碰到背景，达到最大追踪深度等）时，将其最终颜色写入到其对应的像素上，从而在屏幕中显示。

因前面的物体会遮挡后面的物体，导致光路上只有最前方（最靠近发射点）的物体有效，这表现为t值最小。因此对于每一条光线的每一次追踪，都要在一个t值范围内和物体进行求交测试。hit函数作为判断光线是否和物体相交的函数，承担光线追踪的核心功能。在hit函数中，对超出t值范围（checkRange）的结果进行舍去。同时如果发生了有效碰撞，将碰撞信息（碰撞点材质，uv坐标，顶点法线）等信息记录（HitRecord），供渲染函数使用。

进行求交测试时，根据物体的几何特性，使用向量运算法则和物体的几何属性，导出对应的求交条件。如球体的求交测试实质为解一元二次方程，平行四边形的求交测试实质为判断点是否在平面范围内。求交测试需要进行较多浮点数运算，为主要的性能开销来源。

2.3.1：光线和球体相交方程的推导

设球体的球心坐标 (C_x, C_y, C_z) ，半径为 r ，光线方程为 $P(t) = Q + t\vec{D}$ ，将方程中的起点 Q 视为向量。则球体的方程为 $(C_x - x)^2 + (C_y - y)^2 + (C_z - z)^2 = r^2$

设 $P(x, y, z)$ ，则向量 $\vec{C} - \vec{P}$ 为从任一点指向球心的向量，且

$$(\vec{C} - \vec{P}) \cdot (\vec{C} - \vec{P}) = (C_x - x)^2 + (C_y - y)^2 + (C_z - z)^2 = r^2$$

则球体方程简化为

$$(\vec{C} - \vec{P}) \cdot (\vec{C} - \vec{P}) = r^2$$

将光线方程代入 P

$$\begin{aligned} (\vec{C} - (\vec{Q} + t\vec{D})) \cdot (\vec{C} - (\vec{Q} + t\vec{D})) &= r^2 \\ (-t\vec{D} + (\vec{C} - \vec{Q})) \cdot (-t\vec{D} + (\vec{C} - \vec{Q})) &= r^2 \end{aligned}$$

将此式视为一个完全平方式，按照完全平方的规则展开， t 为实数

$$t^2(\vec{D} \cdot \vec{D}) - 2t\vec{D} \cdot (\vec{C} - \vec{Q}) + (\vec{C} - \vec{Q}) \cdot (\vec{C} - \vec{Q}) - r^2 = 0$$

这是一个关于 t 的一元二次方程，则有

$$\begin{aligned} a &= \vec{D} \cdot \vec{D} \\ b &= -2\vec{D} \cdot (\vec{C} - \vec{Q}) \\ c &= (\vec{C} - \vec{Q}) \cdot (\vec{C} - \vec{Q}) - r^2 \end{aligned}$$

根据求根公式，可以解出 t

$$t = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

具体的交点情况需要根据判别式的符号和解出的 t 值确定

1. 判别式为负，光线和球体不相交；
2. 判别式为0，光线和球体相切；
3. 判别式为正，光线穿过球体。由于遮挡效果，只有最近的 t 值，即较小的根有效。
但光线方程中的 t 必须为正值，负 t 表示交点在视口平面之后，不可见，需要舍去。满足以上所有条件，能够找出一个 t 值，或无解。hit 函数通过这个 t 值，计算交点和交点法线等信息并记录。

2.3.2：光线和平行四边形相交方程的推导

设平面方程 $Ax + By + Cz = D$ ，平行四边形由起点 q 和边向量 \vec{u}, \vec{v} 确定；光线方程 $R(t) = P + t\vec{d}$ ，将方程中的起点 P 视为向量。

平面方程可写为两个向量的点积： $\vec{n} \cdot \vec{v} = D$ ，其中 $\vec{n} = (A, B, C)$ 为平面法向量。将光线方程代入 \vec{v} 中

$$\vec{n} \cdot (\vec{P} + t\vec{d}) = D$$

解出 t

$$t = \frac{D - \vec{n} \cdot \vec{P}}{\vec{n} \cdot \vec{d}}$$

当分母 $\vec{n} \cdot \vec{d}$ 为0时，光线和平面法向量垂直，此时光线和平面平行或在平面内，或解出的 t 值不在检查范围内，光线和平面不相交。

由平行四边形所在平面的法向量 $\vec{n} = \vec{u} \times \vec{v}$ ，平面方程参数 $D = \vec{n} \cdot \vec{q}$ （平面方程可表示为平面法向量和平面上任意一点的点积），可通过确定 \vec{u} 和 \vec{v} 确定平面方程。

以上条件决定光线是否和平行四边形所在平面相交，当相交时，确定交点是否在平行四边形范围内。

平行四边形内一点 P 可表示为 $P = q + \alpha\vec{u} + \beta\vec{v}$ ，则目标变为解出 α 和 β 。

记向量 $\vec{p} = P - q = \alpha\vec{u} + \beta\vec{v}$ ，则

$$\begin{aligned}\vec{u} \times \vec{p} &= \vec{u} \times (\alpha\vec{u} + \beta\vec{v}) \\ &= \vec{u} \times \alpha\vec{u} + \vec{u} \times \beta\vec{v} \\ &= \alpha(\vec{u} \times \vec{u}) + \beta(\vec{u} \times \vec{v})\end{aligned}$$

同理， $\vec{v} \times \vec{p} = \alpha(\vec{v} \times \vec{u}) + \beta(\vec{v} \times \vec{v})$ 。

由向量叉乘自身为0，化简后可解出 α, β

$$\begin{aligned}\vec{v} \times \vec{p} &= \alpha(\vec{v} \times \vec{u}) \\ \vec{u} \times \vec{p} &= \beta(\vec{u} \times \vec{v}) \\ \vec{n} \cdot (\vec{v} \times \vec{p}) &= \vec{n} \cdot \alpha(\vec{v} \times \vec{u}) \\ \vec{n} \cdot (\vec{u} \times \vec{p}) &= \vec{n} \cdot \beta(\vec{u} \times \vec{v})\end{aligned}$$

解得

$$\begin{cases} \alpha = \frac{\vec{n} \cdot (\vec{v} \times \vec{p})}{\vec{n} \cdot (\vec{v} \times \vec{u})} = \frac{\vec{n} \cdot (\vec{p} \times \vec{v})}{\vec{n} \cdot (\vec{u} \times \vec{v})} \\ \beta = \frac{\vec{n} \cdot \vec{u} \times \vec{p}}{\vec{n} \cdot \vec{u} \times \vec{v}} \end{cases}$$

其中， $\vec{u} \times \vec{v} = \vec{n}$ 。

要判断交点 P 是否在平行四边形内，只需要判断

$$\begin{cases} 0 \leq \alpha \leq 1 \\ 0 \leq \beta \leq 1 \end{cases}$$

是否成立即可。

2.4：材质的物理行为[AbstractMaterial::scatter, AbstractLight::emitted]

当光线和物体发生碰撞（hit 函数返回 true）时，将根据碰撞信息和物体的材质更新光线的颜色。不同物体具有不同材质，不同材质对应的物理属性不同，和入射到物体表面的光线的交互行为也不同。非发光材质的 scatter 函数描述了材质与光线的相互作用。scatter 函数需要根据入射光计算出射光的方向和颜色变化（对于非发光材质为衰减）。发光材质不使用 scatter，而是使用 emitted 函数，emitted 函数返回光源的光颜色。光源作为只产生颜色贡献而不传递颜色的发光体，光线的散射逻辑将被跳过。（尽管屏幕上的颜色数值范围为[0, 1]，但在追踪计算时不限制颜色范围，允许光源颜色超出 1.0，只有在将像素的最终颜色写入屏幕时才对颜色数值进行裁剪）。

粗糙材质对入射光进行漫反射，即在入射表面的半球范围内，随机选择一个方向反射（或遵守重要性采样的余弦概率密度，将在后面说明），同时返回颜色衰减因子，对光线的颜色进行衰减，以模拟光反射时的能量衰减；金属材质为镜面反射，出射光将根据反射定律被精确计算。对于金属材质，引入 fuzz 毛绒属性，fuzz 值越大（最大为1），金属表面的毛绒感越明显，0为完全的镜面反射；电介质材质有自己的折射率属性，通过入射角和折射率决定是否发生全反射。光线将在 hit 函数的检测下决定是否进行

散射，在 `scatter` 散射函数的作用下生成散射方向，作为下一次光线追踪的初始光线，如此递归直达到结束条件，得出最终颜色。

2.4.1：金属材质的散射方程

向量从空气入射到金属表面，遵守镜面反射规律。设入射向量为 \vec{v} ，平面法向量为 \vec{n} 。

对于镜面反射，将入射向量分为垂直于法向量的 \vec{v}_\perp 和平行于法向量的 \vec{v}_\parallel 。反射后， \vec{v}_\parallel 取反。
反射前

$$\vec{v}_\parallel = (\vec{v} \cdot \vec{n}) \cdot \vec{n}$$

其中 $\vec{v} \cdot \vec{n}$ 表示入射向量在法向量方向上的投影长度

$$\vec{v}_\perp = \vec{v} - \vec{v}_\parallel$$

反射后

$$\begin{aligned}\vec{r} &= \vec{v}_\parallel' + \vec{v}_\perp = -\vec{v}_\parallel + \vec{v}_\perp \\ &= -(\vec{v} \cdot \vec{n}) \cdot \vec{n} + \vec{v} - (\vec{v} \cdot \vec{n}) \cdot \vec{n} \\ &= \vec{v} - 2(\vec{v} \cdot \vec{n}) \cdot \vec{n}\end{aligned}$$

\vec{r} 即为反射光线的方向向量。

2.4.2：电介质材质的散射方程

电介质材质拥有折射率属性，可能发生折射或全反射。设入射介质的折射率为 n ，折射介质的折射率为 n' ，则 $n \cdot \sin(\theta) = n' \cdot \sin(\theta')$ 。入射光线 \vec{R} ，折射光线 $\vec{R}' = \vec{R}'_\perp + \vec{R}'_\parallel$ ，折射点表面法向量 \vec{n} 。
现推导折射方程。根据折射定律

$$\begin{aligned}\vec{R}'_\perp &= \vec{R}' \sin(\theta') \\ &= \vec{R}' \cdot \frac{n}{n'} \cdot \sin(\theta) \\ &= \frac{n}{n'} \cdot \vec{n} |\vec{R}| \cos(\theta) + \vec{R} \\ &= \frac{n}{n'} (\vec{R} + |\vec{R}| \cos(\theta) \cdot \vec{n}) \\ &= \frac{n}{n'} (\vec{R} + (-\vec{R} \cdot \vec{n}) \cdot \vec{n}) \\ &\quad (\text{负号表示 } (\vec{R} \cdot \vec{n}) \cdot \vec{n} \text{ 方向与 } |\vec{R}| \cos(\theta) \cdot \vec{n} \text{ 相反}) \\ \vec{R}'_\parallel &= -\sqrt{1 - |\vec{R}'_\perp|^2} \cdot \vec{n} \\ &\quad (\text{负号表示与法向量反向})\end{aligned}$$

反射时，方程和金属材质相同。判断是否发生全反射通过折射定律和Schlick近似。当 $\frac{n}{n'} \cdot \sin(\theta) > 1$ 或满足Schlick近似的全反射条件时，发生全反射。

3.抗锯齿

在最简单的实现中，每一个像素发射一条光线，此光线的最终颜色直接作为像素的最终颜色。这会导致物体的边缘产生锯齿，原因为相邻两个采样点间距过远，导致像素本身的形状凸显。但每个光线发射点对应一个像素，因此需要进行亚像素采样。在一个像素的范围内，随机选取多个点并发射光线，将每条光线的结果取平均值，达到平滑物体边缘的效果。此方法为超采样抗锯齿（SSAA）。在这种方法

下，提升每像素采样数，将极大提升渲染所需时间。在实时渲染领域，常用的抗锯齿方式有FSAA、MSAA和基于时间的TAA、基于机器学习的NVIDIA DLAA等。

4.噪点（光线追踪的核心问题）

在以上内容所包含的光线追踪实现中，包含了随机的部分，如材质的散射和超采样抗锯齿，这些随机性带来了渲染结果的不确定性。由于光线弹射方向的不确定，在场景中有光源时（实际上为必定，没有任何光源将得到完全黑色），一些光线将不会和任何光源（灯材质或背景）相交，且达到追踪终止条件，将返回黑色，或处于物体边缘的一些光线由于随机出发点而和完全不同的物体相交，从而在一片区域内，相邻像素得到的颜色相差极大，显然不符合物理规律。这些黑点，白点或颜色相差极大的点为噪点。为了解决这些噪点，必须保证即使有部分光线得到了错误的颜色值，整体的颜色仍然正确，从而必须提升每像素采样数，这样最终的平均值将不会受到部分错误光线的影响（不会完全正确，但不可察觉），极大的每像素采样数是数学上完全解决噪点的唯一方法。事实上，光线追踪（包括以上提及的渲染函数），其实现都是对基于物理推导出的渲染方程

{

$$L_0(p, \omega_0) = L_e(p, \omega_0) + \int_{\Omega} f_r(p, w_i, \omega_0) L_i(p, w_i) (w_i \cdot \vec{n}) dw_i$$

其中

$L_0(p, \omega_0)$ ：从 p （碰撞点）沿 ω_0 方向出射的光的辐射度，在当前追踪器中表现为计算得出的结果，即光的颜色。

$L_e(p, \omega_0)$ ：点 p 自身发出的光，非发光物体此项为0。

$\int_{\Omega} \cdots dw_i$ ：碰撞点和碰撞方向所在半球面上所有方向入射光颜色的和，由于光入射方向的连续性，以积分形式表示。

$f_r(p, w_i, \omega_0)$ ：双向反射分布函数（BRDF）：描述光线沿 w_i 方向入射到 p 点时，有多少光会反射到 ω_0 方向，由物体的材质特性决定，scatter 函数指定具体行为。

$L_i(p, w_i)$ ：从 w_i 方向入射到 p 点的光的辐射度（颜色）。

$(w_i \cdot \vec{n})$ ：入射光方向向量和碰撞点法线的点积，由于二者均为单位向量，点积等同于二者夹角余弦值，即入射角的余弦值。此项又被称为朗伯余弦项，为粗糙表面的散射PDF。

}

的近似。由于渲染方程中的积分项 $L_i(p, w_i)$ 需要来自其他方向的光线颜色，而其他方向的光线颜色必须在全部计算完成后才能确定，但光线追踪器必须在计算完当前位置的颜色后，才能计算其他方向的颜色，这本身就是矛盾的，且这个积分没有初等解。因此尝试解出这个方程，从而达到完全精确的计算，在离散且有限的计算机上是不可行的。但实际上图形学并不需要完全正确的颜色，只要不能被察觉出错误即可。因此，光线追踪器采用提升采样质量（使相同采样数得到的噪点数量更少）和降噪器以在有限的计算资源中得到质量较好的渲染结果。

提升采样质量方面的内容将在后面介绍。在降噪器方面，首先降低每像素采样数（如果需要实时交互，需要每像素最多采样1次），并在渲染完成后，使用由机器学习模型驱动的降噪器对充满噪点的图像进行降噪（通常需要结合法线和基础材质颜色等辅助信息）以得到平滑的图像。目前业界的常见降噪器，分使用的处理器，有

$$\begin{cases} \text{CPU端: Intel OIDN等} \\ \text{GPU端: NVIDIA OptiX Denoiser, NVIDIA NRD等} \end{cases}$$

GPU端的降噪器将在GPU版本的渲染器中得到应用。此外，UE等大型3D引擎都实现了自己的降噪器，供用户直接调用。

5.BVH加速结构[BVHTree, AxisAlignedBoundingBox]

BVH加速结构为光线追踪器的重要组成部分，能够极大提升渲染速度。其核心思想为构造每个几何体的包围盒，包围盒为能够完全包裹住物体的最小简单几何体。在此渲染器中，使用了轴对齐包围盒（AABB，经典的包围盒设计，在实际生产和游戏中被广泛使用）。之后，将一个或多个几何体（“图元”）按照排序规则合并，取其包围盒的并集，构造BVH树叶子节点的包围盒。此实现中，使用了随机选择排序轴或选定最长轴（性能略好于前者）的排序策略。再将两个叶子节点的包围盒合并，构造上一层节点的包围盒。如此重复，直到有一个节点的包围盒包括了所有子包围盒，也就包括了所有图元，此节点作为BVH二叉树的根节点。

使用BVH加速结构后，每一条光线的每一次追踪由遍历场景中所有图元，依次进行相交测试，改为先遍历BVH二叉树，和包围盒进行相交测试，仅在光线和最小包围盒相交的情况下，才和物体进行相交测试。光线遍历将从BVH树的根节点开始逐层向下，和每一层节点的包围盒进行相交测试，逐渐缩小检索范围，直到和图元进行精确测试。在BVH树遍历的任何一层，如果光线没有和包围盒相交，就可以直接判定此光线没有和任何物体相交。由于光线和包围盒的相交测试开销极小，渲染器中光线总量极大，因此BVH结构能够带来显著性能提升，特别是在渲染没有物体或物体数量较少的区域。

6. 离焦模糊

在相机摄影中，“焦距”指焦点到成像平面的距离，离焦距大于一定距离（离成像平面过远或过近）的物体将表现为模糊，因此实际应用时，需要根据拍摄对象的不同选择不同的焦距，突出特定内容，以达到特定的艺术效果。“景深”表示在焦距附近的一段距离范围，在此范围（离成像平面距离适中）的物体，可以被判定为清晰，景深越大，清晰显示的物体范围越大。在光线追踪器中，“焦距”被定义为视口（成像平面）到“相机看向位置”这一点的距离，而“景深”由 Camera 的构造参数 `focusDiskRadius` 指定。这一参数为0时，场景中所有物体都不受景深的影响，清晰渲染。实际上“景深”的实现原理为当物体偏离焦点位置时，物体的成像将不在成像平面上聚集成一个点，而是一个面，由于镜头为圆形，这个面是圆。当这个面的面积达到一定值时，物体不再清晰。而参数 `focusDiskRadius` 指定的就是这个圆的半径：光线追踪是从成像平面向场景中发射光线，和实际上摄影的光线方向正好相反。当 `focusDiskRadius` 不为0时，主渲染函数在每次采样时，都将在以选定的采样点为圆心，`focusDiskRadius` 参数为半径的圆上随机选点发射光线，以模拟景深效果，这一模拟在图形学中通常称为离焦模糊。如果需要启用景深，可尝试将 `focusDiskRadius` 参数设置为0.5。有关摄影中焦距和景深的具体概念和图形化演示，可参考相应科普资料。

7. 运动模糊

如果要渲染运动物体，首先赋予物体以速度。在此渲染器中，为 Sphere 类添加了运动支持，可通过调用构造方法，指定运动的起始点来构造运动球体。光线追踪中实现运动模糊的方式为在时间维度上进行超采样。

1. 定义相机快门：Camera 类的 `shutterRange` 参数定义了相机快门的开启时间，当前帧的所有光线都将在快门开启时间内被发射，通常是 `Range(0.0, 1.0)`。
2. 光线带有时间参数：光线 Ray 将从简单的 $P(t) = O + t\vec{D}$ 升级为增加了 `time` 属性，表示光线的发射时间。在渲染过程中，忽略光的传播速度，因此在主渲染循环中，不会对光线的 `time` 属性作出任何更改。
3. 计算有时效性的光线-物体相交：物体不再是静止的，其几何中心（或代表点）不再是一个定值，而是以时间为自变量的方程。将光线的时间参数传递给物体，得到对应时刻物体的位置，计算此位置光线和物体的相交测试。运动物体的包围盒需要包括在快门时间内其整个连续的运动路径中经过的所有位置。
4. 多条光线求平均：在主渲染函数中构造带有时间参数的光线，将时间参数设定为快门开启时间内的随机值，实现时间维度的随机采样。由于抗锯齿环节已经发射了多条光线，时间维度超采样可不需要额

外进行循环。将最终的结果求平均（直接使用抗锯齿超采样的实现），即可得到带有运动模糊的帧。

8.纹理[AbstractTexture]

物体的材质定义了物体表面散射光线的行为（`scatter` 函数），但不能决定物体随空间变化的外观。如果要物体具有图案，需要使用纹理。在此渲染器中，纹理被材质类 `Rough` 包装。

1. 定义纹理：将纹理作为物体的成员，允许在构造物体时传入纹理对象的智能指针。
2. 记录UV坐标：在 `hit` 函数中，根据碰撞点，通过对应的映射函数，将碰撞点映射到UV坐标。这样可以通过UV坐标找到该位置的纹理颜色。
3. 通过材质使用纹理：`Rough` 材质类的 `scatter` 函数使用对应位置纹理的颜色值定义颜色衰减因子，从而使用的对应位置的纹理颜色。当前渲染器实现了棋盘纹理（两种颜色相间）、图像纹理（可以把世界地图贴到球体上，见 `Example::test03()`）和柏林噪声纹理（支持多种柏林噪声生成器）。

9.变换[Transform, Matrix]

如果物体需要变换（旋转、平移、缩放），就不能直接渲染物体。光线追踪中，对于要变换的物体，不是直接变换物体本身，而是

1. 对光线进行逆变换：将光线的起点 O 和方向向量 \vec{D} 进行物体所使用变换的逆变换
2. 使用逆变换后的光线和物体进行求交测试：过程同没有变换的物体
3. 对碰撞信息进行正变换：光线不再使用，只需要记录正确的碰撞信息，则对碰撞点和顶点法线进行变换。其中，对碰撞点进行正变换，对顶点法线进行逆变换的转置（推导过程附在此节末尾）。其他参数如 t 值和UV坐标不需要变换。

此方式的效率高于直接变换物体本身，且实现难度较低。如果变换物体本身，需要对物体的所有顶点进行变换，但物体的顶点数对于碰撞函数或变换类不可知，会带来不必要的复杂度。

当前变换实现使用图形学中常用的4x4矩阵统一表示三种变换（尽管三维空间中的线性变换只需要3x3矩阵，但平移变换需要4阶矩阵，因此统一使用4阶矩阵表示），`Transform` 的构造方法允许同时进行三种变换，最终变换矩阵 M 为三个变换矩阵的乘积。

为了支持基于矩阵的变换，当前渲染器包含一个基础的矩阵类 `Matrix`，包含矩阵的基础运算（加减、数乘、乘法）和基于高斯消元的矩阵运算函数（求逆矩阵、行列式、正交化）。具体实现可直接查看 `src/util/Matrix.cpp`。

注意：此矩阵实现直接迁移了线性代数教科书中的数学算法，数学上正确，但算法效率没有经过优化，且数值稳定性较差（特别是正交化函数），只用于学习。在图形学应用中，应直接使用成熟矩阵库（Eigen3、cuBLAS），且对于旋转变换，使用四元数而不是旋转矩阵，能够提升计算效率和数值稳定性。

为了确保变换后的法线依然垂直于变换后的物体表面，而正变换矩阵会在变换为非统一缩放或错切变换时破坏垂直关系，因此不能直接使用正变换矩阵。现推导碰撞点法线的变换矩阵。设碰撞点表面的切线为 T ，表面法线为 N ，则有

$$N \cdot T = 0$$

将此式写成矩阵乘法形式：设 T 为列向量，将 N 转置得到行向量

$$N^T \cdot T = 0$$

设物体的变换矩阵为 M ，则物体上任意一点 P ，表面切线 T 经过变换后为

$$\begin{aligned} P' &= MP \\ T' &= MT \end{aligned}$$

设所求矩阵为 G ，则

$$\begin{aligned} N' &= GN \\ (N')^T T' &= 0 \end{aligned}$$

将 $T' = MT, N' = GN$ 代入 $(N')^T T' = 0$

$$(GN)^T (MT) = 0$$

由矩阵转置的性质 $(AB)^T = B^T A^T$ 和矩阵乘法的结合律

$$\begin{aligned} N^T G^T (MT) &= 0 \\ N^T (G^T M) T &= 0 \end{aligned}$$

对比 $N^T (G^T M) T = 0$ 和 $N^T \cdot T = 0$ ，得

$$G^T M = I$$

等式两边同时乘以 M^{-1} 并取转置

$$\begin{aligned} G^T M M^{-1} &= I M^{-1} \\ G &= (M^{-1})^T \end{aligned}$$

即表面法线的变换矩阵为逆矩阵的转置。

10. 体积雾[ConstantMedium, Isotropic]

此渲染器实现了凸多面体的体积雾效果。ConstantMedium 类为包裹物体的体积雾外壳，Isotropic（各向同性）为一种材质，被 ConstantMedium 类所使用。

为了模拟出柔和的雾气效果，需要将以上两个类配合使用。ConstantMedium 类包含一个指向物体的指针，体积雾效果将围绕此物体表面生成。ConstantMedium 的 hit 函数通过两次求交，找到光线进出体积雾作用范围的t值范围，并通过物理定律（比尔-朗伯定律）结合随机数判断光线是否受体积雾散射效果影响。对体积雾的模拟充满随机性，光线有可能直接穿过体积雾范围，也可能受其影响而散射，此时将调用 Isotropic 的 scatter 函数，此函数将光线作用于能量衰减后，随机选择一个方向为散射方向。在多次的随机散射下，形成柔和的雾气效果。

比尔-朗伯定律在光线追踪中应用的公式：

$$hitDistance = -\frac{1}{density} \cdot \ln(randomNumber)$$

其中，hitDistance 为光线在体积雾范围内的穿行长度，当长度超过体积雾的范围时，光线不受影响。density 参数为体积雾的密度，决定了光散射的概率。randomNumber 为一个 (0, 1) 之间的随机数。

11. 采样优化

以下内容包含光线追踪的数学原理与采样优化技术。

11.1：蒙特卡洛积分法

虽然从数学上精确求解渲染方程不可行，但数学提供了离散求和的方法近似计算积分，并控制近似的精度。此方法称为蒙特卡洛积分法，可以使用计算机的强大离散运算能力。其核心思想是：一个函数在某个区间上的定积分，可以被看作是该函数在该区间内随机取值的期望。前者是无解析解的连续积分，而后者是可以离散计算的有限求和。现对一维定积分的蒙特卡洛法则作出证明。

命题：对于一个一维定积分

$$I = \int_a^b g(x) dx$$

有

$$\lim_{N \rightarrow \infty} \hat{I}_N = \lim_{N \rightarrow \infty} (b-a) \frac{1}{N} \sum_{i=1}^N g(x_i) = I$$

其中

X_1, X_2, \dots, X_N 为从区间 $[a, b]$ 上的均匀分布 ($X \sim U(a, b)$) 中独立随机抽取的 N 个样本点。

证：

由 $X \sim U(a, b)$ ， X 的概率密度函数为：

$$f_X(x) = \begin{cases} \frac{1}{b-a}, & \text{if } a \leq x \leq b \\ 0, & \text{otherwise} \end{cases}$$

则函数 $h(x) = (b-a)g(x)$ 关于 X 的期望 $E[h(x)]$ ，根据连续随机变量期望的定义：

$$\begin{aligned} E[h(x)] &= \int_{-\infty}^{\infty} h(x) f_X(x) dx \\ &= \int_a^b (b-a)g(x) \left(\frac{1}{b-a}\right) dx \\ &= \int_a^b g(x) dx \\ &= I \end{aligned}$$

则求解积分的问题转化为求解随机变量 $Y = h(X) = (b-a)g(X)$ 的期望，将 $h(x_i)$ 带入定义式中：

$$\begin{aligned} \hat{I}_N &= \frac{1}{N} \sum_{i=1}^N h(x_i) \\ &= \frac{1}{N} \sum_{i=1}^N (b-a)g(x_i) \\ &= \frac{b-a}{N} \sum_{i=1}^N g(x_i) \end{aligned}$$

根据大数定律，一系列独立同分布的随机变量，其样本均值随样本数量 N 的增大依概率收敛于期望值 $E(X)$ ，即

$$\lim_{N \rightarrow \infty} \bar{Y}_N = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{i=1}^N Y_i \xrightarrow{P} E(Y)$$

则

$$\lim_{N \rightarrow \infty} \hat{I}_N = \lim_{N \rightarrow \infty} \left(\frac{b-a}{N} \sum_{i=1}^N g(x_i) \right) = E(Y)$$

$$\lim_{N \rightarrow \infty} \hat{I}_N = E(Y) = I$$

将 $\hat{I}_N = \frac{b-a}{N} \sum_{i=1}^N g(x_i)$ 式两边取极限，命题得证。

从而我们有了一种近似计算无初等解的积分的方法，此结论可扩展到渲染方程中的积分。从而，用有限求和求解无限积分是可行的，只是因为求和项数有限，不能精确求解，但是对于渲染出正确的图像来说，这一有限近似是可以接受的。

11.2：分层采样

以上介绍的的采样方法是在每一个像素周围的亚像素范围内随机选点采样，此方式实现简单，但由于随机点的范围分布在整个亚像素长方形内，在一些情况下可能出现采样点分布不均匀，从而出现大块而分散的噪点，视觉上表现为明显的噪点。一种简单的提升采样质量的方式是将亚像素长方形分为规则正方形网格，网格的边长为每像素采样数的平方根，并在每个网格内随机选点采样。此方式可以保证每个网格有且仅有一个采样点。这种方式使得采样点分布更加稳定且平均，减小了采样结果的方差，视觉表现为噪点更小，分布更均匀，更加不易被察觉。现对分层采样方差减小作出证明。

首先将问题形式化。根据简单采样方式的原理，要计算的值是采样区域内所有光线颜色值的和（再除以采样数求平均），由于采样区域是一个平面区域，因此实际上求解的是一个二重积分。为方便起见，将采样区域 A 归一化为单位正方形 $[0, 1] \times [0, 1]$ ，其面积为1。

要计算的积分：

$$I = \int_0^1 \int_0^1 f(x, y) dx dy$$

根据蒙特卡洛积分法，在采样区域内随机选择 N 个点 P_1, P_2, \dots, P_N ，将积分值转化为估计量 \hat{I}_N

$$\hat{I}_N = \frac{1}{N} \sum_{i=1}^N f(P_i)$$

证明分层采样得到的估计量的方差 $Var(\hat{I}_{ss})$ 小于纯随机采样得到的估计量的方差 $Var(\hat{I}_{rs})$

证：

纯随机采样的方差 $Var(\hat{I}_{rs})$ ：

在采样区域 A 内均匀随机选点 P_1, P_2, \dots, P_N ，这些采样点独立同分布。令随机变量 $Y_i = f(P_i)$ ，则 Y_i

也独立同分布。

则所求

$$Var(\hat{I}_{srs}) = Var\left(\frac{1}{N} \sum_{i=1}^N Y_i\right) = \frac{1}{N^2} \sum_{i=1}^N Var(Y_i)$$

现求单样本方差 $Var(Y_i)$

单样本期望

$$E(Y_i) = \int_0^1 \int_0^1 f(x, y) \cdot pdf(x, y) dx dy$$

由于均匀分布，

$$pdf(x, y) = \begin{cases} \frac{1}{A}, & 0 \leq x, y \leq 1 \\ 0, & otherwise \end{cases}$$

其中A为区域的面积，在当前简化的积分区域 $[0, 1] \times [0, 1]$ 中， $A = 1$

则

$$E(Y_i) = I$$

$$Var(Y_i) = E[(Y_i - I)^2] = \int_0^1 \int_0^1 [f(x, y) - I]^2 dx dy = \delta^2$$

δ^2 为函数 $f(x, y)$ 在采样区域内的整体方差。

则纯随机采样方差 $Var(\hat{I}_{srs}) = \frac{1}{N^2} \sum_{i=1}^N N\delta^2 = \frac{\delta^2}{N}$

分层采样方差 $Var(\hat{I}_{ss})$:

将采样区域 A 分为 N 个子区域 A_1, A_2, \dots, A_N ，每个区域的面积都是 $\frac{1}{N}$ ，样本 P_j 从每个子区域中随机抽取，则 P_j 独立不同分布（函数 $f(x, y)$ 在不同子区域的函数值不同）。同上，求单样本的方差 $Var(Y_j)$ ，为子区域 A_j 内的方差。

子区域内均值

$$\mu_j = E(Y_j) = \int_0^1 \int_0^1 f(x, y) \cdot pdf(x, y) dx dy$$

此处积分区域面积为 $\frac{1}{N}$ ，则 $pdf(x, y) = N$ 。

$$\mu_j = N \int_0^1 \int_0^1 f(x, y) dx dy$$

子区域内方差 $\delta_j^2 = Var(Y_j)$

$$\begin{aligned} Var(Y_j) &= \int_0^1 \int_0^1 [f(x, y) - \mu_j]^2 \cdot pdf(x, y) dx dy \\ &= N \int_0^1 \int_0^1 [f(x, y) - \mu_j]^2 dx dy \end{aligned}$$

则分层采样方差 $Var(\hat{I}_{ss}) = \frac{1}{N^2} \sum_{j=1}^N \delta_j^2$

其中 δ_j^2 为函数在每个子区域内的方差。

为了比较 $Var(\hat{I}_{srs})$ 和 $Var(\hat{I}_{ss})$ ，使用全方差公式

$$Var(X) = E[Var(X|Y)] + Var[E(X|Y)]$$

展开全局方差 δ^2 。令 $X = Y = f(P_i)$ ， J 为随机变量，其值为样本 P_i 落入每个子区域的编号， $J \in \{1, 2, \dots, N\}$, $P(J = j) = \frac{1}{N}$ 。

则

$$\delta^2 = Var(Y) = E[Var(Y|J)] + Var[E(Y|J)]$$

等号右侧第一项：由 $Var(Y|J = j)$ 表示在样本 P 落在区域 j 的前提下， Y 的方差，即 δ_j^2 ，根据离散型随机变量期望定义

$$E[Var(Y|J)] = \sum_{j=1}^N P(J = j) \cdot Var(Y|J = j) = \frac{1}{N} \sum_{j=1}^N \delta_j^2$$

等号右侧第二项：由 $E(Y|J = j)$ 表示在样本落入子区域时函数 $f(x, y)$ 的期望，即 μ_j ，则此项表示 μ_j 本身的方差，表示函数在自取件内的波动情况，为 $Var(\mu_j)$ 。

比较二者

$$\begin{aligned} Var(\hat{I}_{srs}) &= \frac{1}{N} \left(\frac{1}{N} \sum_{i=1}^N \delta_i^2 + Var(\mu_j) \right) \\ &= \frac{1}{N^2} \sum_{i=1}^N \delta_i^2 + \frac{Var(\mu_j)}{N} \\ &\geq Var(\hat{I}_{ss}) = \frac{1}{N^2} \sum_{i=1}^N \delta_i^2 \end{aligned}$$

原命题得证。

无论是普通采样还是分层采样，都是用有限的采样近似区域内的连续积分，是蒙特卡洛积分法的具体实现。

12：重要性采样

12.1：不均匀随机数

在普通采样实现中，随机生成器（`std::uniform_real_distribution<>`）生成均匀分布的随机数，则依赖随机数的路径追踪采样也是均匀的，从而没有和光源相交的光线可能会呈现出错误的颜色，形成噪点。而提升采样质量是解决噪点问题的方法之一。通过重要性采样，能够让有限的采样光线发挥出更大的作用，从而做到在相同的SPP（每像素采样数）下，生成噪点更少的图像。

重要性采样为非均匀采样，需要根据给定的概率密度函数（PDF）生成非均匀随机数。因此需要一个方法，将随机数生成器生成的均匀随机数映射到非均匀范围内。使用逆变换采样法求出映射函数。事实

上，此映射函数就是分布函数的反函数，能将均匀分布的随机变量映射到服从概率密度函数 $p(x)$ 的随机变量。

命题：设 X 为连续型随机变量，其概率密度函数为 $p(x)$ ，分布函数为 $F_X(x)$ ，若随机变量 U 服从 $[0, 1]$ 上的均匀分布，则随机变量 $Y = F_X^{-1}(U)$ 和 X 同分布。

证明：要证 Y 和 X 同分布，即证二者的分布函数或概率密度函数相同。现求 Y 的分布函数 $F_Y(y)$ 。

$$\begin{aligned} F_Y(y) &= P(Y \leq y) \\ &= P(F_X^{-1}(U) \leq y) \\ &= P(F_X(F_X^{-1}(U)) \leq F_X(y)) \\ &= P(U \leq F_X(y)) \end{aligned}$$

由均匀分布的性质，对于任何 $u \in [0, 1]$ ，其分布函数 $F_U(u) = P(U \leq u) = u$ ，则

$$P(U \leq F_X(y)) = F_X(y) = F_Y(y)$$

即二者同分布，命题得证。

通过这个方法，要生成符合概率密度 $p(x)$ 的随机数，只要求其分布函数的反函数，就可以完成均匀随机数到非均匀随机数的转换。

12.2：权重修正

如果使用了非均匀的随机数，那么基于此随机数的采样也是非均匀的，这将导致采样结果偏向概率密度高的方向，造成偏差。为了平衡此偏差，需要对每个样本（蒙特卡洛积分法中求和的每一项）进行加权。由于每个样本的权重是 $p(x)$ ，那么应该乘以因子 $\frac{1}{p(x)}$ 进行修正，即 $\int f(x) = \sum \frac{f(x_i)}{p(x_i)}$ 。

12.3：材质采样概率密度

重要性采样的其中一部分为材质散射光线的概率密度。最经典的需要重要性采样的材质为粗糙材质，因为其散射光线的方向随机确定，而不是金属材质或电介质材质的根据反射定律精确计算。对于粗糙材质，离法线近（和法线夹角小）的光线能够产生更大的光照贡献。为了让光线采样的概率正比于该方向对最终颜色的贡献度，从而用更少的样本获得噪点更低的图像，并根据渲染方程，推导出应采用余弦加权采样。

渲染方程的反射部分

$$\int_{\Omega} f_r(p, w_i, w_0) \cdot L_i(p, w_i) \cdot \cos(\theta_i) dw_i$$

对于漫反射材质，其 f_r 为 $\frac{\text{albedo}}{\pi}$ ， L_i 项未知， $\cos(\theta_i)$ 为入射光方向向量和表面法线的夹角余弦，决定了表面接收到的有效能量。而在重要性采样中，希望采样的概率密度函数尽可能与被积函数的成正比。以达到对积分最准确的估计。从而设采样概率密度函数 $p(w_i) = C \cdot \cos(\theta_i)$ ，现求比例常数 C 。

由概率密度函数在随机变量取值范围（积分区域）上的积分为1，而渲染方程的积分区域为半球，则

$$C \cdot \int_0^{2\pi} \int_0^{\frac{\pi}{2}} \cos(\theta) \sin(\theta) d\theta d\phi = 1$$

解得 $C = \frac{1}{\pi}$ ，则漫反射表面的采样概率密度函数 $p(w_i) = \frac{\cos(\theta)}{\pi}$ 。

此余弦概率密度函数是一个基于“入射光能量贡献度”的重要性采样函数。在程序中，它被用来主动地“生成散射的出射方向”，目的是为了采样的分布与能量贡献的分布相匹配。正因为知道那个方向入射的光可能带有最大的能量，因此向这些方向进行更多的采样，能够更快地找到能量来源。

为了生成按余弦概率密度的散射光线，需要使用转换函数。在球坐标系中，当半径固定时，只需要两个变量即可确定球面上一点，因此需要将两个随机数转换为球坐标。

1. 方位角 $a(\phi) = \frac{1}{2\pi}$ ：“经线”，取各个“经度”的概率相等
2. 极角 $b(\theta) = 2\pi f(\theta)\sin(\theta)$ ：“纬线”，由于球体本身的性质，各纬度的“纬线”长度不等，因此需要进行伸缩变换。

$b(\theta)$ 的推导过程：

原始球面PDF： $p(\phi, \theta)$ ，为联合概率密度。由于PDF的值和方位角无关，则实际上等价于只有 θ 的边缘概率密度 $p(\theta)$ 。记 $f(\theta) = p(\phi, \theta)$ 。

伸缩因子： $\sin(\theta)$ 因子在两极时为0，在赤道为1，描述了纬线长度随纬度的变化规律。或球坐标的伸缩比 $r^2\sin(\theta)$ ，在此情况下， $r = 1$ 。

已知联合概率密度，求边缘概率密度，则进行积分

$$\begin{aligned} b(\theta) &= \int_0^{2\pi} f(\theta)\sin(\theta) d\phi \\ &= 2\pi f(\theta)\sin(\theta) \end{aligned}$$

根据不均匀随机数生成法，需要求概率密度函数 $a(\phi)$ 和 $b(\theta)$ 的分布函数的反函数，以将两个随机数 r_1, r_2 映射到球坐标。

积分得

$$\phi = 2\pi r_1$$

当在整个球面上均匀采样时（无需求出 θ ，求出 $\cos(\theta)$ 即可）

$$\cos(\theta) = 1 - 2r_2$$

由球坐标和直角坐标的转换式

$$\begin{cases} x = \cos(\phi) \cdot \sin(\theta) \\ y = \sin(\phi) \cdot \sin(\theta) \\ z = \cos(\theta) \end{cases}$$

代入并化简得

$$\begin{cases} x = \cos(2\pi r_1) \cdot 2\sqrt{r_2(1-r_2)} \\ y = \sin(2\pi r_1) \cdot 2\sqrt{r_2(1-r_2)} \\ z = 1 - 2r_2 \end{cases}$$

同理，对于余弦加权采样，可得

$$\begin{cases} x = \cos(2\pi r_1) \cdot \sqrt{r_2} \\ y = \sin(2\pi r_1) \cdot \sqrt{r_2} \\ z = \sqrt{1-r_2} \end{cases}$$

这样，只需要生成两个 $[0, 1)$ 之间的随机数 r_1, r_2 ，即可生成对应概率密度的方向向量。

12.4：物体采样概率密度（直接光源采样）

场景中能量来源是光源。为了减少噪点，需要尽可能多的向光源采样。实现方式是直接的：直接在光源物体上选点，构造从碰撞点指向光源上随机点的光线，计算此光线的有效性及其颜色贡献。

为了向光源直接采样，需要构造球面PDF，使得生成的光线的方向全部集中在光源所对应的球面立体角，现推导平行四边形光源对应的球面的概率密度函数。

由立体角定义式的微分形式 $dw = \frac{ds}{r^2}$ ，立体角为从点 P 看向球面区域 A ，所对应的二维球心角。此处， A 为光源， r 为碰撞点到光源上一点的距离。由于定义式中要求面积微元 ds 在垂直于球心和球面上一点的连线的平面上，则需要对光源面积进行投影修正：乘以因子 $\cos(\theta)$ ，其中 θ 为光源平面 A 法向量与 P, Q (Q 为光源上任意一点) 连线的夹角。

代入立体角定义式，有

$$dw = \frac{dA \cdot \cos(\theta)}{\text{distance}^2(P, Q)}$$

由在光源上选到区域 dA 的概率和在球面上选到方向 dw 的概率相等，则

$$p(w) \cdot dw = p_q(q) \cdot dA$$

其中 $p_q(q)$ 为在光源上选点的概率密度，可设置为均匀采样， $p_q(q) = \frac{1}{A}$ ， A 为光源面积。

代入并化简得

$$p(w) = \frac{\text{distance}^2(P, Q)}{A \cdot \cos(\theta)}$$

以上算式适用于平行四边形物体。对物体的重要性采样可以不局限于光源，或当光源不是平行四边形，如球体时，需要使用特定的球面PDF。现推导对球体的重要性采样PDF。

从点 P 看向一个球体，则能够被采样的范围形成了以 P 为顶点，过球心的圆为底的圆锥，张角 θ_{max} 定义为过 P 的切线和过 P 和球心连线直接的夹角，采样方向 w 必须落在这个圆锥内。易知 w 关于球心连线旋转对称，则 $p(w) = f(\theta) = C$ ，其中 $0 \leq \theta \leq \theta_{max}$ 。

同材质PDF，需要建立直角坐标和球坐标的转换关系

$$\begin{aligned}\phi &= 2\pi r_1 \\ b(\theta) &= 2\pi f(\theta) \sin(\theta) \\ &= 2\pi \cdot C \cdot \sin(\theta)\end{aligned}$$

由概率密度函数的性质，在整个积分区域 $0 \leq \theta \leq \theta_{max}$ 内积分为1

$$\int_0^{\theta_{max}} 2\pi \cdot C \cdot \sin(\theta) d\theta = 1$$

解得

$$C = \frac{1}{2\pi(1 - \cos(\theta_{max}))}$$

现计算 $0 \leq \theta \leq \theta_{max}$ 范围对应的立体角 α

$$\begin{aligned}\alpha &= \int_0^{2\pi} \int_0^{\theta_{max}} \sin(\theta) d\theta d\phi \\ &= 2\pi(1 - \cos(\theta_{max}))\end{aligned}$$

对比 C 的计算公式，发现

$$C = \frac{1}{\alpha}$$

则得出结论：对于任何一个指向球体可见部分的方向 w ，满足 $0 \leq \theta \leq \theta_{max}$ ，其概率密度为常数，大小等于球体所张开的立体角的倒数。

12.5：混合采样（MIS）

以上内容包括材质PDF和物体PDF，通常需要将二者结合以实现较好的采样效果。首先明确二者的区别：材质PDF是局部的，它只根据当前入射光线，给出一个可能是主要贡献方向的散射方向；物体PDF是全局的，使用物体PDF需要光源的几何参数（位置、形状），通过将二者结合，可以实现多重重要性采样（MIS）。

为了将多个PDF结合，使得结果仍然是PDF，需要采用线性方式组合各个函数。

$$mix_pdf = w_0 p_0 + w_1 p_1 + \cdots + w_n p_n \quad (w_0 + w_1 + \cdots + w_n = 1)$$

组合后，PDF函数值按照以上组合式，合成多个PDF的贡献；但每次采样只能选取一个PDF，则采用随机选择策略：将所有PDF添加到列表，采样时从列表中等概率随机选择一个PDF计算散射方向。