

译

解析近似区域光实时着色

专业班级：	软件工程 3 班
姓 名：	廖 添
指导教师：	李 志 淮

信息科学技术学院

摘 要

我们引入解析近似,以便对由非封闭区域光源照亮的表面进行准确的实时渲染。我们的解决方案利用了由 Arvo 开发的用于多边形光源照射的冯氏光照表面着色的辐照度张量。利用一维边界积分的重新构造,我们设计并开发了一个通用型结构框架,用于在一定常数时间内通过若干简单峰形函数对积分进行近似求解和评估。为突破冯氏光照的限制,我们提出一个低成本边缘分割策略,引入了半矢量参数化方法解释球经。由于这一新颖的扩展,我们准确地近似求解了常见的若干微面元双向反射分布函数,提供了一种实际方法来保证能与实际数据匹配的实时镜面延伸。最后,使用该解析近似框架,通过支持非均匀缩放操作和水平面裁剪的基于原创的多边形纺丝的方法,我们引入对球面以及圆盘面区域光源的运算支持。在实际的图形处理器上,我们提出的方法成功实现了无需任何对区域光源形状和表面粗糙度进行预先假设的实时渲染功能。

关键词: 区域光源, 着色, 解析近似, 微面元, 轴向弯矩, 实时

1 引言

考虑到区域光源的照明问题, 准确地进行镜面表面的实时渲染是一件具有挑战性的任务。其难度在于二维镜面辐照积分的估算没有实际的公式解, 除非使用高成本的基于蒙特卡洛算法的采样技术。多数引人注目的解决方案都采用计算最具代表性点(MRP)方法^{[1],[2]}, 降低着色积分的成本至廉价的点光照计算。然而, 这些方法都在保留次表面双反射分布函数(BRDF)的高光形状问题上失败, 并且难以处理部分水平面的可视性。Arvo^[3]提供了一个分析解决方案, 用于对被非封闭多边形光源照亮的光滑表面进行着色。不过它的实现依赖于高成本的边缘积分方法, 同时也受限于冯氏表面。最近一个并发方法使用线性余弦分布(LTC)^[4]来解决这个问题。然而, 该解决方案要求为每一个 BRDF 提供使用高成本的最小化技术构建的预计算表。

寻找一个结合了精确性、灵活性和实时性能的解决方案是一个具有挑战性的问题, 其包含了许多对高品质渲染需求应用程序(如预先照明工具、游戏引擎或工业渲染器)的期望。

在本论文当中, 我们通过利用 Arvo 开发的辐照度张量进行准确的解析近似来解决这些缺点(图 1)。同时, 进一步将该方法扩展至处理多轴取向余弦波问题, 突破冯氏光照的限制, 支持微面元 BRDF。最终, 我们将介绍一种原创的多边形旋转方法, 以支持通过相同的数学框架对球形或圆盘形区域光照亮的表面进行着色。

我们所作的工作有:

- 一个泛型框架, 使用简单的可积分峰形函数, 在 $O(1)$ 时间而不是 $O(n)$ 时间近似和评估边缘积分。
- 轴取向余弦波多重乘积的解析近似, 使得能够在更复杂的球面多边形上进行 BRDFs 的积分。
- 一种低成本的边缘分割策略, 用于处理通过半矢量参数化引入造成的扭曲失真, 从而实现微面元 BRDF 的支持。
- 一个原创的旋转算法, 利用我们的近似方法, 实现球形和圆盘形区域光的照明, 支持不均匀缩放操作。

2 相关工作

在过去几十年中, 区域光源的直接照明问题已经以各种方式解决了。在本节我们回顾我们认为与我们的方法最相关的工作, 重点是解决实时渲染约束下积分的镜面反射项。

蒙特卡洛积分。蒙特卡洛积分技术是基于概率抽样策略对复杂积分进行数值计算的常用方法。对于直接区域光照问题, 绘制采样需要考虑到有区域光形状所

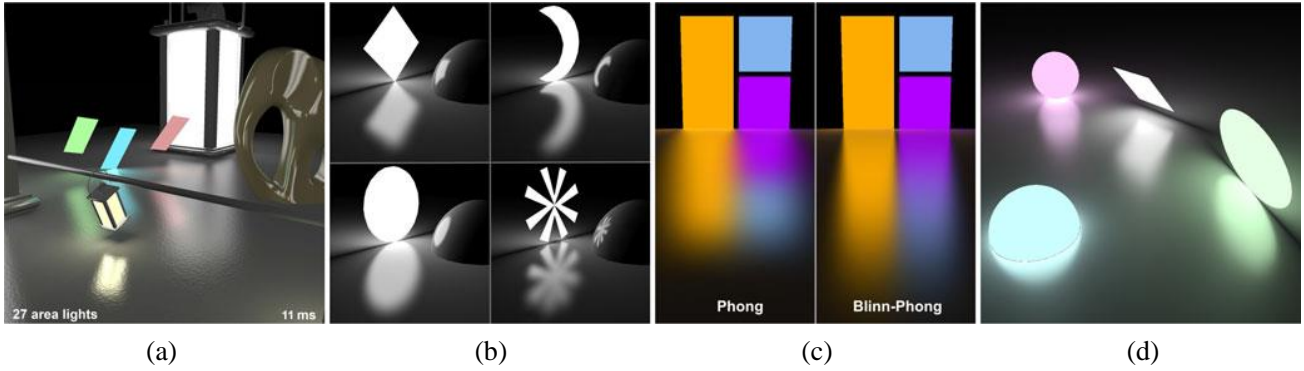


图 1：我们的解析近似为接近真实数据的质量提供了镜面区域光照的实时性能（a，在 NVIDIA GTX 980Ti GPU 上测量的时间）。我们支持任何类型的多边形光源形状（甚至是动画类型）（b），和由冯氏光照或 BRDFs 描述的表面（c）。我们提出的方法也容易扩展到支持非均匀缩放的球形和圆盘形区域光源（d，这里由微面元分布演示）。

维持的立体角^{[5],[6]}、表面 BRDF 的重要性采样，或者使用两者的组合来减少镜面反射计算的方差。尽管采用这种抽样方法，但是这些方法需要大量样本才能收敛到无噪声的结果，几乎无法兼容实时渲染的约束。

另外一个常用于解析近似区域光源的方法是使用虚拟点光源(VPLs)^[7]集合，将积分当中的镜面反射项的计算降低至对于多个点光源的计算。分类归并聚集方法^[8]已被提出以进一步降低 VPLs 的算法复杂度，并成功地在实时渲染中得到了应用^{[9],[10]}。然而，这些方案往往限于解决低频光照问题，例如漫反射或弱光滑表面等，以限制采样量并保持良好的实时性能。使用这些方法进行高频光照仍然是一个具有挑战性的问题，仅使用大量采样或昂贵的集成技术只能远离实时渲染的考量。

最具代表性点。MRP 方法通过识别最有助于光照的区域光上的代表点来降低采样技术的昂贵计算成本。该方法将着色积分问题减少到单点照明计算，为实时渲染提供了实用的解决方案。该方法的早期工作可以在[11]的冯氏区域光照部分中找到。这里的 MRP 被定义为距离观察反射方向最近的点。相反，Drobot^[1]考虑了区域光和具有由表面粗糙度参数化的孔径的锥体之间的交叉区域中的一个点。Karis^[2]解决了能量守恒问题，并且通过改进镜面分布来更好地匹配镜面微面元模型的强烈高光。然而，这些方法都有缺点。冯氏 BRDF 的高光形状能够很好地近似，但在使用微面元 BRDF 时则变得不准确。水平面处理同时是另外一个问题。MRP 在近似简单的几何形状光源时效果良好，但当光源在水平面处被裁剪时计算将变得更加复杂。

解析方法。另有方法尝试推导出着色积分的近似解，或者至少是一个能够让人接受的近似。Bao 和 Peng^[12]使用低度泰勒级数展开的二维多项式的双积分进行近似，将它们的方法限制在低指数冯氏表面。田中和高桥^[13]扩展了 Poulin 的线性区域光法^[14]，并将立体角分解为沿着大圆的一维有符号积分。然后使用切

比雪夫多项式近似来评估每个一维积分，将该方法限制到低频冯氏表面。Arvo^[3]开发的辐照度张量方法为被多边形光源点照亮的光滑表面的直接照明提供了准确的分析解决方案。使用张量理论和斯托克斯边缘积分，着色积分沿着多边形光的球形边界边缘被分解为一维有符号积分的和。然后使用绑定到冯氏光滑度 n 的线性时间算法来解析每个边缘积分。实时图形的实际实现，包括水平裁剪，可以在[15]中找到。尽管它拥有很好的准确性，但该方法只适用于冯氏表面，因为它 $O(n)$ 的时间复杂度，导致其在实时渲染应用程序中的使用仅限于弱光滑面。

球形高斯函数(SGs)。SG 是应用于许多环境照明问题的球面函数，例如环境照明或全局照明，随后用于实时区域光照明。Wang 等人^[16]利用 SG 为 SG 近似 BRDF 的积分产物提供了闭合形式表达式来近似球面光。为了处理微面元 BRDF，通过半矢量变换引入的球形翘曲使用单个各向同性的 SG 进行近似。然而，这种方法无法表现出掠射角的细长镜面延伸。Xu 等人^[17]使用各向异性球形高斯(ASG)近似球面翘曲。球面光源照明的实际实现可以在[18]中找到。这些方法有两个主要的局限性。首先，球面翘曲近似假设的光源是各向同性的。第二，由于区域光近似作为 SG，高光滑度表面倾向于显示出高斯形状。与我们方法相似的是，Wang 等人^[19]使用 SG 的分段线性近似来进行多边形可见度评估，将一维边缘积分减少到解析表达式。相反，Xu 等人^[20]在平行平面上使用边缘参数化来导出使用分段线性逼近评估的一维边界积分表达式。两种方法都需要分段解析积分域。此外，它们仅限于各向同性 SG，并且不能处理微面元分布的各向异性。

线性变换余弦。Heitz^[4]等人最近提出的并发的方法 使用夹余弦分布（以下称为 LTC）的线性变换来近似各向同性 BRDF，包括微面元。作为结果，通过对多边形应用逆变换，将着色操作减少到分析形状因子计算。LTC 的确比我们的方案更准确，更简单。然而，匹配 BRDF 需要对入射方向集合和粗糙度值的变换矩阵进行预计算和存储。存储和存储器访问的需求并不总是必要的，特别是在低端 GPU 上。我们的解决方案是完全接卸的，不需要任何预计算步骤或存储。此外，我们的方法可以集成在半矢量空间中，具有更高的自由度。在这个意义上，这两种方法是互补的。

3 我们的方法

我们的方法建立在辐照度张量和 Arvo^[3]开发的边缘积分方法的基础上，我们将在第 4 节简要回顾一下。这种方法代表了几个挑战。

第一个挑战是摆脱 $O(n)$ 时间瓶颈的实时渲染效率。我们通过以更简洁的方式（第 5 节）重写一维积分来解决这个问题，使用合理的峰形积分框架（第 6 节）来确定准确的 $O(1)$ 时间近似。与切比雪夫和傅里叶近似不同，我们的方法仅限于 1 或 2 个有理函数，并且不会受到任何边缘振荡效应的影响。

其次，组合几个轴向余弦波的球形多边形上更复杂的分布的积分是另一个挑战。通过从球面高斯借用算子，我们得到简单的解析表达式（第 7 节），准确地逼近余弦波的多个乘积。最后，最后一个挑战就是克服冯氏光照的 BRDF 的限制，并支持更可信的 BRDF。在微面元理论中发现的半矢量参数化引入了使用非各向同性多边形光源难以预测的球面变形。基于大圆形畸变的观察，我们可以使用多边形方法（第 8 节）以比以前的方法更灵活的方式真实地近似该球形翘曲。

4 辐照度张量和边缘积分

Arvo^[21]开发的辐照度张量为球面 S^2 上多项式的解析积分提供了有用的框架。这些多项式对应于由轴取向余弦波分布描述的 n 阶单项式。该表达式在球面区域 $\Omega_A \subset S^2$ 上的积分产生了关于 \mathbf{r} 轴的 n 阶轴向弯矩的定义：

$$M^n(\Omega_A, \mathbf{r}) = \int_{\Omega_A} (\mathbf{u} \cdot \mathbf{r})^n d\mathbf{u} \quad (1)$$

使用张量积和斯托克斯定理，Arvo 将轴向弯矩表达式引申为投影面积光边

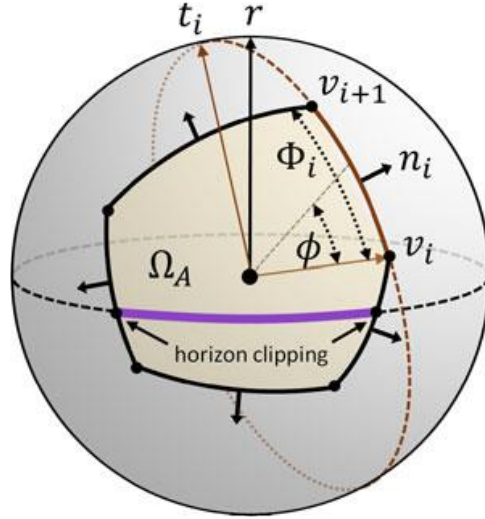


图 2：在球面区域 Ω_A 上朝向 \mathbf{r} 的余弦分布轴的积分在 Ω_A 的边界上减小到一维轮廓积分。给定多边形光源的封闭形式表达式，使用局部基底 $\mathbf{v}_i, \mathbf{t}_i$ 中第 i 个球形边缘的参数化（红色虚线表示此处支持边缘 $(\mathbf{v}_i, \mathbf{v}_{i+1})$ 的大圆。

界 Ω_A 处的一维边缘积分。考虑到多边形光源，可以在沿着大圆的球形边缘的参数化之后获得一维积分的闭合形式表达式（参见图 2）。让我们考虑一个 m 边界的多边形。遵循图 2 所示的符号，闭合表达式如下：

$$(n+1)M^n(\Omega_A, \mathbf{r}) = z_n \Omega_A - \sum_{i=0}^m (\mathbf{n}_i \cdot \mathbf{r}) F(\Phi_i, c_i, \delta_i, n-1) \quad (2)$$

其中：

$$c_i = \sqrt{a_i^2 + b_i^2}; \quad \delta_i = \tan^{-1} \left(\frac{b_i}{a_i} \right)$$

$$a_i = \mathbf{v}_i \cdot \mathbf{r}; \quad b_i = \mathbf{t}_i \cdot \mathbf{r},$$

且

$$F(\Phi, c, \delta, n) = \sum_{k=0}^{\frac{n-1+z_n}{2}} c^{2k+1-z_n} \int_{-\delta}^{\Phi_i-\delta} (\cos\varphi)^{2k+1-z_n} d\varphi \quad (3)$$

其中： $z_n = 1 - (n \bmod 2)$

有关辐照度张量的完整说明以及如何达成这一表达方式，读者应参考[21]和[3]。需要注意的是，与 Arvo 相比，我们使用略微不同的符号来缓解下一节进一步开发的数学推导。在函数 F 上的一维积分的总和使用复杂度 $O(n)$ 每边缘时间的复现算法以闭合形式进行评估， n 为冯氏指数。在 GPU 上实现，该方法对于弱光滑表面 ($n < 40$) 有效，但是随着冯氏指数的增加，性能下降，并且对于高光滑表面 ($n > 1000$) 而言变得不切实际。为了降低高冯氏指数的评估成本，Arvo 建议一旦达到所需的相对精度，提早终止迭代循环。但是，根据我们的经验，我们观察到严重的性能下降，特别是在表面的掠射视角，是因为达到所需精度而需的大量迭代次数。预测性能的难度使其成为实时渲染考虑的不可行解决方案。在实际的 GPU 实现中，无论冯氏指数如何，边缘积分 F 应理想地在 $O(1)$ 中进行评估。

表 1
本论文中使用的符号

符号	描述
$L(x, \mathbf{v})$	以点 x 为原点向量 \mathbf{v} 为方向散射的辐射
Ω_A	由区域光源 A 维持的区域 \parallel 的立体角
\mathbf{n}	面法线
\mathbf{v}	视线的单位向量
\mathbf{r}	视线折射线的单位向量
\mathbf{h}	由 $(\mathbf{i} + \mathbf{v})/ \mathbf{i} + \mathbf{v} $ 得出的单位半角向量
m	多边形光源的边的数量
\mathbf{v}_i	多边形光源的第 i 个顶点的球面投影
n	余弦指数
\mathbf{n}_i	由 $(\mathbf{v}_i \times \mathbf{v}_{i+1})/ \mathbf{v}_i \times \mathbf{v}_{i+1} $ 得出的边的向外法线
\mathbf{t}_i	由 $\mathbf{t}_i = \mathbf{v}_i \times \mathbf{n}_i$ 得出的边的切线向量
Φ_i	边弧长度
F	球面边缘积分
\tilde{F}	近似球面边缘积分

5 边缘积分的再推导

我们建议通过低成本和准确的解析近似来代替方程 (3) 的昂贵的边缘积分评估，其允许在任何冯氏光滑度 n 下都能够进行常数时间的评估计算。设置精确的近似要求至少有知道内在特征的被积函数。通常的方法是探测边缘被积函数来提取这些特征。然而，以目前的形式，这需要评估被积函数的总和。我们建议以不同的形式重写边缘积分，以获得更简单和更紧凑的表达。为此，我们首先引入

一个函数 f 和一个临时项 q :

$$f(\varphi, c, n) = \begin{cases} c \cos \varphi & \\ 1 & \end{cases} \quad q = \begin{cases} (n-1)/2 & \text{if } n \text{ is odd} \\ n/2 & \text{if } n \text{ is even} \end{cases}$$

通过切换和与积分算子，并通过使用上面引入的术语，等式 (3) 中定义的边积分 F 可以重新写成如下:

$$F(\Phi, c, \delta, n) = \int_{-\delta}^{\Phi-\delta} f(\varphi, c, n) \sum_{k=0}^q (c \cos \varphi)^{2k} d\varphi \quad (4)$$

其和表现为具有通用公式的 x^{2k} 形式的几何级数:

$$\sum_{k=0}^q x^{2k} = (x^{2(q+1)} - 1) / (x^2 - 1)$$

这允许我们在替换临时变量 q 之后消去总和并获得单个函数进行积分:

$$F(\Phi, c, \delta, n) = \int_{-\delta}^{\Phi-\delta} \frac{(c \cos \varphi)^{n+2} - f(\varphi, c, n)}{(c \cos \varphi)^2 - 1} d\varphi \quad (5)$$

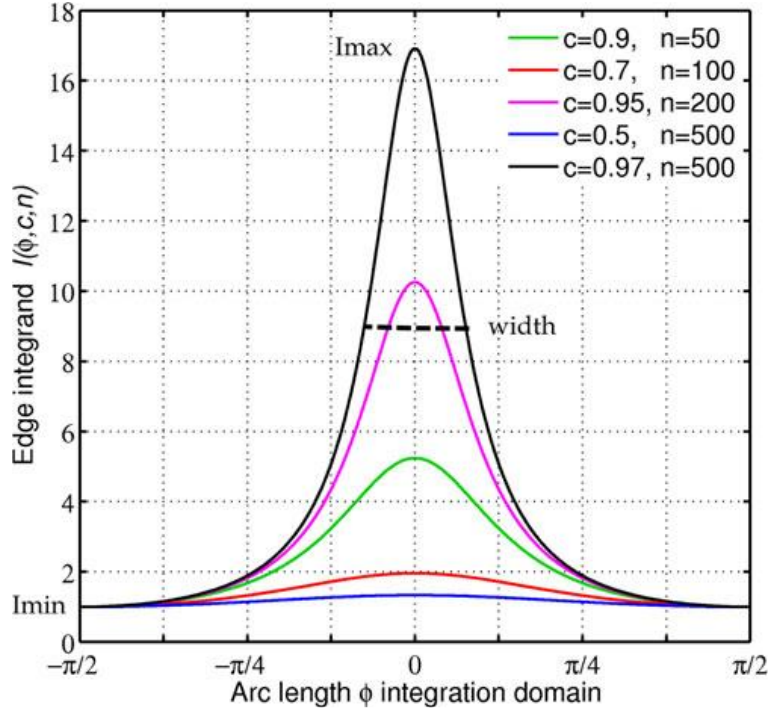


图 3: 边缘被积函数 I 展现各种高度和宽度的峰形函数。核心思想是使用具有相同特征的简单且可积分的峰形函数去近似 I 。

这种重新推导允许在常数时间内评估被积函数。另一个优点是能够平滑地表示非整数值的 n ，特别是对于低与 n 的指数。虽然不存在无限积分，但可以从我们的重新推导中获得准确的分析近似。

6 精确解析近似

让我们考量等式 (5) 中边缘积分的积分项：

$$I(\Phi, c, n) = \frac{(c \cos \varphi)^{n+2} - f(\varphi, c, n)}{(c \cos \varphi)^2 - 1} \quad (6)$$

根据图 3，我们观察到 I 的形状对应于各种高度和宽度的对称峰形函数，这取决于参数 c 和 n ，有最小值为 $\varphi = \pm\pi/2$ ，最大值为 $\varphi = 0$ 。我们的方法的核心思想是使用具有已知解析积分的简单有理表达式描述的峰形函数来近似 I 。近似依赖于将峰形函数映射到被积函数 I 的特征（如最小值，最大值和宽度）的简单拟合过程。

根据等式 (6)，给出最小值和最大值的闭合公式：

$$I_{\min}(c, n) = \begin{cases} 0 \\ 1 \end{cases} \quad I_{\max}(c, n) = \begin{cases} \frac{c^{n+2} - c}{c^2 - 1} & n \text{ odd} \\ \frac{c^{n+2} - 1}{c^2 - 1} & n \text{ even} \end{cases}$$

半宽度估计。宽度被定义为对应于横坐标 x_w 的半最大值上的半宽度 (HWHM)，例如：

$$I(x_w, c, n) = \frac{(I_{\max} - I_{\min})}{2} - I_{\min}$$

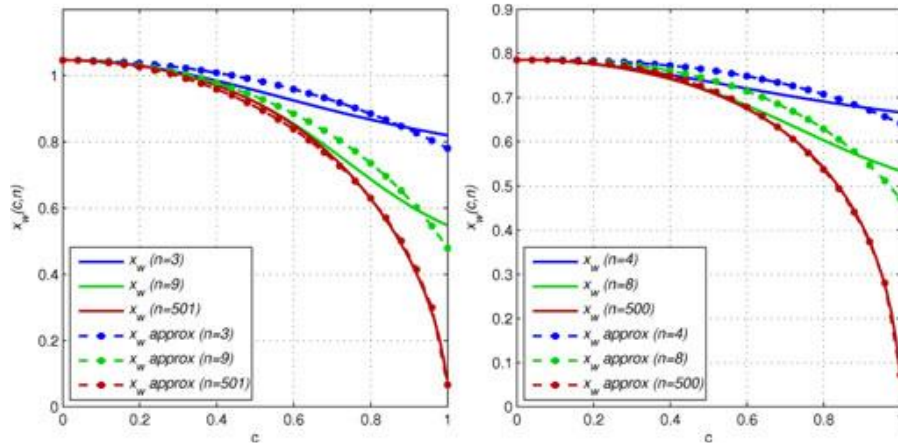


图 4：对于 n 为奇（左）和 n 为偶（右）的各种值，被积函数 I 的半宽度 x_w 近似。

然而，为 x_w 找到一个封闭形式的表达式有点困难。一种方法是将 x_w 的预计算值存储在离散条目 (c, n) 的二维表中。另一种方法是进行解析近似。在实验测量研究之后，我们发现 x_w 可以经验近似如下：

$$x_w(c, n) \approx \begin{cases} \frac{\pi}{3} \sqrt{1 - \left(c - \frac{c}{n}\right)^2} & n \text{ odd} \\ \left(\frac{\pi}{4} \left(1 - \left(c - \frac{c}{n-1}\right)^{2.5}\right)\right)^{0.45} & n \text{ even} \end{cases} \quad (7)$$

即使这是一个粗略估计（图 4），下一节描述的拟合处理将保证我们的近似值将通过点 $(x_w, I(x_w, c, n))$ 。

6.1 泛型积分框架

我们通过一般的峰形函数得出一个用于近似和评估方程（3）的一般框架。为此，我们首先考虑由最小 P_{\min} ，最大 P_{\max} 和宽度 P_w 定义的通用峰形函数 P 。通过调整 P 与 I 的相同特性可以获得 I 的精确近似。拟合过程包括如下定义的缩放，偏移和宽度调整：

$$\tilde{I}(\varphi, c, n) = \frac{I_{\max} - I_{\min}}{P_{\max} - P_{\min}} (P(\varphi, x_w) - P_{\min}) + I_{\min}$$

需要注意的是，为简洁起见，省略了功能参数。我们可以通过将所有常数项合并在一起进一步减少此表达式：

$$\tilde{I}(\varphi, c, n) = s P(\varphi, x_w) + t \quad (8)$$

其中， $s = \frac{I_{\max} - I_{\min}}{P_{\max} - P_{\min}}$ ， $t = I_{\min} - s P_{\min}$ 。

从而得到对于方程（3）的泛型积分：

$$F(\Phi, c, \delta, n) = s \int_{-\delta}^{\Phi - \delta} P(\varphi, x_w) d\varphi + t\Phi \quad (9)$$

6.2 峰形函数的近似

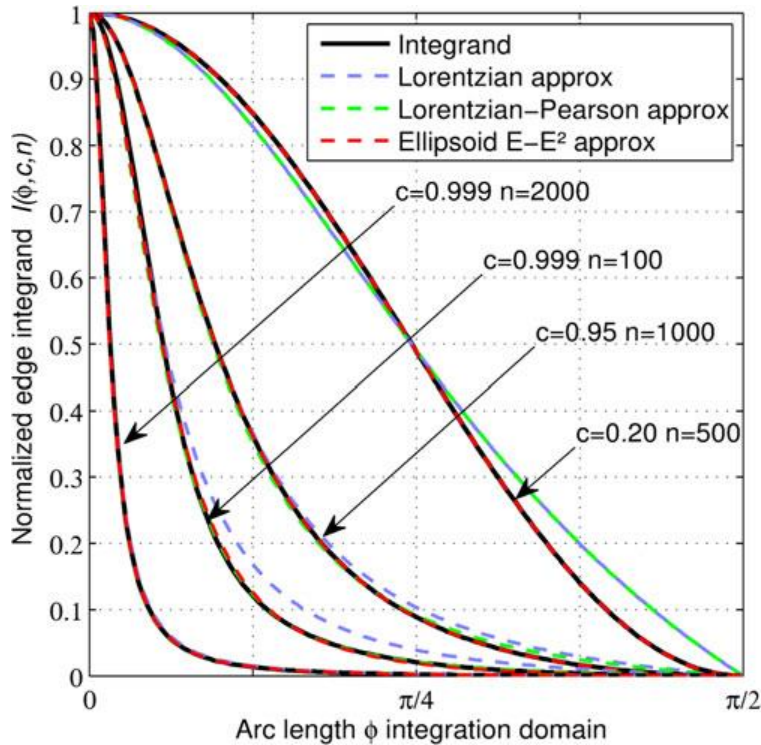


图 5：使用 c 和 n 的各种值的峰形函数近似边缘被积函数 I 。洛伦兹函数相当好地逼近边缘被积函数 I ，但是在尾部的精度不足。Lorentzian-Pearson 更接近于 I ，但是对于较大的宽度值，它仍然不准确。相比之下，无论函数的宽度如何，椭圆近似都能提供最佳的准确度。

筛选出具有不确定积分的几个峰形函数 P ，其简单到足以避免耗时的评估运

算，并提供对于方程（6）的准确估计。我们通过使用单轴弯矩评估来实现能量守恒的单轴冯氏模型，验证了我们近似值的准确性，并用真实数据比较：

$$L(x, \mathbf{v}) = \int_{\Omega_A} f_{\text{Phong}}(\mathbf{i}, \mathbf{v})(\mathbf{n}, \mathbf{i}) d\mathbf{i} = \rho_s \frac{n+1}{2\pi} M^n(\Omega_A, r) \quad (10)$$

水平裁剪。水平裁剪考虑到当地区光源有一部分在水平线以下时的能量损失。虽然[3]没有明确地解决裁剪过程，但在[15]中可以找到可行的实现。我们在我们的实现中采用相同的过程。

6.2.1 洛伦兹近似

最简单的近似可以通过洛伦兹峰形函数得到：

$$P(\varphi, c, n) = \frac{1}{1+a\varphi^2} \quad (11)$$

$$\text{其中，} \int P = \frac{1}{\sqrt{a}} \tan^{-1}(\sqrt{a}\varphi)。$$

我们使用公式（7）来计算大致对应于 I 的最大值的拟合点 $I(x_w, c, n)$ 。求解方程 $I(x_w, c, n) = \tilde{I}(x_w, c, n)$ 产生未知参数 a 的分辨率：

$$a = \frac{1 - y_w - \frac{4x_w^2}{\pi^2}}{y_w x_w^2}$$

$$\text{其中，} y_w = \frac{I(x_w) - I_{\min}}{I_{\max} - I_{\min}}。$$

通过将公式（9）中的积分项替换为等式（11）中定义的积分项，我们获得了在常数时间内评估的 F 的解析近似。

$$\tilde{F} = \frac{s}{\sqrt{a}} \left(\tan^{-1}(\sqrt{a}(\Phi - \delta)) - \tan^{-1}(-\delta\sqrt{a}) \right) + t\Phi$$

值得注意的是 $\tan^{-1}x - \tan^{-1}y = \tan^{-1}\left(\frac{x-y}{1+xy}\right) \bmod \pi$ 这个表达式可以进一步减少为单个反正切评估以节省 GPU 指令。

$$\tilde{F} = \frac{s}{\sqrt{a}} \tan^{-1}\left(\frac{\sqrt{a}\Phi}{1+a(-\delta)(\Phi-\delta)}\right) \bmod \pi + t\Phi \quad (12)$$

误差分析。图 6 表明，无论表面的粗糙度如何，洛伦兹近似值都相当准确，接近真实参照情况。然而，当增加整体亮度时，我们可以观察到突出形状周围的轻微漏光最为显着。仔细观察图 5 中的洛伦兹近似结果，可以得知误差是由于对函数 I 尾部的过高估计而产生的。

6.2.2 洛伦兹-皮尔森 VII 方法

通过将洛伦兹函数与具有较短尾部的第二峰值函数组合可以找到尾部周围的更好的近似。这个想法是通过这两个近似来包含尾部区域中的被积函数，并从尾部挑选的值中找到混合因子。第二峰函数由对应于提高到功率 m 的洛伦兹函

数的 Pearson VII 函数定义。我们选择 $m=2$ ，其具有不确定的积分，足以避免耗时的计算。

$$P = \left(\frac{1}{1 + b\varphi^2} \right)^2$$

且

$$\int P = \frac{\varphi}{2(b + \varphi^2)} + \frac{\tan^{-1}\left(\frac{\varphi}{\sqrt{b}}\right)}{2\sqrt{b}}$$

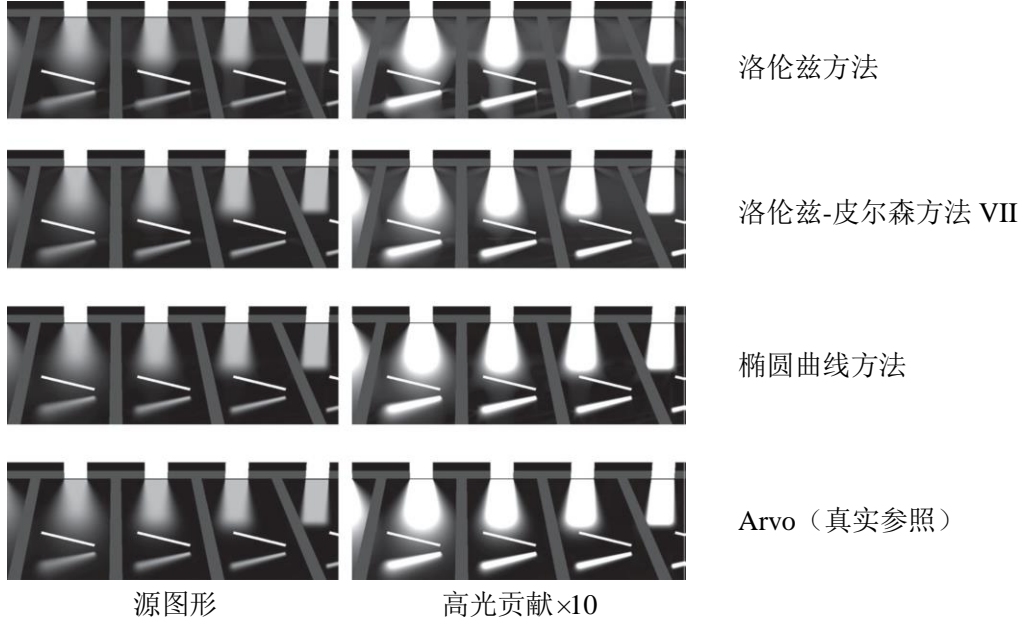


图 6: 我们的近似值与真实参考图像的比较。右列描绘了左侧图像的缩放版本，以使差异可见。虽然洛伦兹方法的泛光溢出清晰可见、洛伦兹皮尔森方法几乎不可见，但椭圆近似值的缩放版本的泛光溢出几乎不显眼，难以被察觉。

Pearson VII 功能的顶部部分完全类似于洛伦兹功能，而且底部部的分尾部较短。然而，找到一个 b 参数，如 $\tilde{I}(x_w) = I(x_w)$ ，需要分辨率为涉及复杂计算的四度多项式方程。幸运的是，通过重新使用为洛伦兹近似计算的参数，可以大大简化 b 的计算并节省 GPU 计算时间。在我们的实验结果当中，我们发现 $b \approx \frac{a}{2}$ 总是包含目标被积函数 I 。

线性调和。将 I_p 调整到与 I 相同的宽度给出了另一个近似值，其低估了尾部中的 I ，同时保留了其上的拟合。然后，最佳近似值位于两个函数之间，可以使用简单的线性混合操作找到：

$$\tilde{I}_{LP}(\varphi) = \alpha \tilde{I}_L(\varphi) + (1 - \alpha) \tilde{I}_p(\varphi) \quad (13)$$

其中:

$$\alpha = \frac{\tilde{I}_p(x_{tail}) - I(x_{tail})}{\tilde{I}_p(x_{tail}) - \tilde{I}_L(x_{tail})}$$

线性调和运算需要在位于函数尾部的位置 x_{tail} 处评估被积函数 I 。然而，为 I 找到闭合表达式表示与半宽度估计的困难相同。因此，我们再次使用经验方法进行近似:

$$x_{tail} \approx x_w + 0.3946 x_w(0) \left(1 - \left(1 - \frac{x_w}{x_w(0)}\right)^{12}\right) \quad (14)$$

进一步精确近似。洛伦兹-皮尔森近似极大地提高了镜面高光的总体精度，并抑制了大多数可观察到的伪影。尽管如图 6 所示，我们仍然在镜面高光区域以外的区域存在微弱的泛光溢出。当峰形 I 非常大时，即当值 c 小时，会发生这些溢出。仔细看看图中的细节，图 5 显示了在集成域边界处的近似值被高估了。特别是，当 $\varphi = \pm \frac{\pi}{2}$ 时，一阶导数为零，则洛伦兹-皮尔逊近似则为零。

6.2.3 椭球近似

使用基于椭球的峰形函数可以获得更好的精确度，特别是在边界上。这些函数就像洛伦兹方法具有令人感兴趣的特性，但在 $\varphi = \pm \frac{\pi}{2}$ 时存在一个零一阶导数。

$$\text{椭球形式: } P_E = \frac{a}{1+(a-1)\cos^2(\varphi)} \quad (15)$$

$$\text{不定积分: } \int P_E = \sqrt{a} \tan^{-1} \left(\frac{\tan \varphi}{\sqrt{a}} \right). \quad (16)$$

椭球平方函数式:

$$P_{E^2} = \left(\frac{b}{1+(b-1)\cos^2(\varphi)} \right)^2$$

$$\int P_{E^2} = \frac{\sqrt{b}}{2} (b+1) \tan^{-1} \left(\frac{\tan \varphi}{\sqrt{b}} \right) - \frac{b}{2c} (b-1) \sin(2\varphi)$$

我们遵循与 6.2.1 节和 6.2.2 节中描述的完全相同的步骤，以将 I_E 和 I_{E^2} 拟合为 I ，并使用线性调和找到最佳近似值。第一个近似项 I_E 的参数 a 对应于:

$$a = \frac{y_w(1-\cos^2(\varphi))}{\cos^2(\varphi)(1-y_w)} \quad (18)$$

对于 I_{E^2} ，参数 b 大致如下:

$$b \approx a \left(2.1 + 1.28 \frac{x_w}{x_w(0)} \right)$$

近似精确度。如图 6 所示，无论的不明显伪像的功能的宽度如何，椭球近似

都能提供最佳的精度。

表 2

在 NVIDIA GTX 580 和 GTX 980Ti 上，每个边缘的渲染时间以毫秒为单位，具有三峰形状近似的呈现精度

方法	指数	时间/边 (ms)		RMSE
Arvo (参照)	$n=100$	13.6	0.41	n/a
	$n=500$	49	1.57	n/a
	$n=5000$	476	9.8	n/a
洛伦兹方法	$n=100$	0.25	0.12	0.004354
	$n=500$	0.25	0.12	0.005506
	$n=5000$	0.25	0.12	0.004128
洛伦兹-皮尔森方法	$n=100$	0.40	0.125	0.003641
	$n=500$	0.40	0.125	0.003094
	$n=5000$	0.40	0.125	0.002551
椭球方法	$n=100$	0.47	0.127	0.001500
	$n=500$	0.47	0.127	0.001652
	$n=5000$	0.47	0.127	0.001014

6.3 性能与精确度分析

我们在 GPU NVIDIA GTX 580 上实现和测试了我们的近似结果。表 2 提供了每个边缘的渲染时间（以毫秒为单位）以及使用归一化 RMSE 的渲染精度测量。考虑到以 720p 分辨率为代表最的情况处理，对所有屏幕像素的渲染进行了测量。需要注意的是，时间也包括了 \mathbf{n} 周围和 \mathbf{r} 周围的双水平裁剪的处理。

正如所预期的那样，使用 Arvo 解决方案获得的渲染时间随着指数 n 而增加，而我们的近似值则保持不变。洛伦兹逼近实现了最佳性能，而椭球方法是最准确的，与真实参考数据无明显差异，并且 GPU 耗时切线评估引入了较小的计算开销。大多数情况的洛伦兹近似可以满足诸如游戏等高性能要求的应用。对于高质量要求苛刻的应用，如生产渲染前景照明，洛伦兹-皮尔森或椭球近似是最好的选择。

7 多轴评估

辐照度张量允许使用张量产物的分解来评估多个轴。Arvo^[3]提出了由 \mathbf{n} 和 \mathbf{l} 的两个余弦波的乘积描述的双轴力矩的闭合形式表达式。然而，组合几个轴的任意次序的泛化是相当困难的，涉及复杂的数学表达式并且需要每轴时刻评估。为了减少问题的数学复杂性，我们证明了余弦波的乘积可以通过单独的余弦波精确近似，为多轴力矩评估提供了可行而有效的方法。

$$\int_{\Omega_A} (\mathbf{u} \cdot \mathbf{r}_1)^{n_1} (\mathbf{u} \cdot \mathbf{r}_2)^{n_2} \dots (\mathbf{u} \cdot \mathbf{r}_i)^{n_i} \approx \int_{\Omega_A} \mu(\mathbf{u} \cdot \mathbf{r})^n \quad (19)$$

Meunier 等人^[22]首先提出了这样的数学降解近似方法。他们的方法使用耗时的 L_2 最小化技术来确定逼近两个余弦波的乘积的参数 (μ, r, n) 。将结果预先计算并存储在表 $(n_1, n_2, \angle(\mathbf{r}_1, \mathbf{r}_2))$ 样本的大集合中。在内存占用和缓存效率方面, 处理表并不总是令人满意的, 特别是在低端 GPU 移动设备上。为了避免这个问题, 我们提出一种快速的分析方法, 不需要任何预计算步骤或存储器。