概述

在游戏和其他实时应用中,环境光遮挡普遍存在,以近似全局照明效果.但是,没有针对任意场景的环境光遮挡积分的解析解,并且使用通用数值积分算法的速度太慢,因此,即使它们不能准确解决AO积分,也常常凭经验使实践中使用的近似看起来令人满意.在这项工作中，我们介绍了一种新的环境光遮挡方式,即GTAO,它能够在半秒内匹配当前控制台硬件上的地面真实情况参考.这是通过使用环境遮挡方程的替代公式以及使用时空滤波分布计算的有效实现方式来完成的.然后,我们使用一种新技术扩展GTAO,该技术考虑了近场全局照明,当单独使用环境光遮挡时会丢失.最后,我们介绍一种与环境光对称的镜面光遮挡技术GTSO,它可以从基于探针的照明中计算出真实的镜面反射.我们的技术高效,可提供接近射线追踪地面实况的结果,并且已集成到最新的AAA主机游戏中.

1 介绍

全局照明是重要的视觉特征,是真实感渲染的基础,因为感知的场景照明的很大一部分来自间接反射.不幸的是,它通常计算起来非常昂贵,并且如果不进行大量简化,当前就不能包含在实时应用程序中.根据这些近似值，环境光遮挡(AO)是最流行的一种，因为它可以改善对物体形状(对比度)的感知，并可以捕获全局照明中的一些最重要的效果，尤其是由于近距离照明而产生的柔和阴影通过封堵器。环境光遮挡还可以与其他全局照明算法结合使用，甚至在使用预先计算（烘焙）的辐照度时也是如此，因为通常需要以相对较低的空间分辨率来计算(或存储)这些效果，因此计算每个像素的环境光遮挡可以增强整体间接照明的外观。不幸的是，在某些情况下(例如60fps的1080p渲染),解决环境光遮挡积分仍然很昂贵，因此过去已经开发了近似方法以实现足够快的性能.

我们介绍了一种用于环境光遮挡的新屏幕空间技术，称为地面真实环境光遮挡（GTAO）。 该技术的主要目标是匹配地面真实环境的遮挡，同时又要足够快以被包括在现代游戏机之类的高要求应用中。我们的技术基于基于地平线的方法，但是使用了问题的替代表达。该公式使我们可以显着降低效果的成本，并且在假设我们的场景表示为高度场（深度缓冲区）的假设下，仍可以用于精确求解环境光遮挡积分。我们通过使用时间重新投影和空间滤波来有效地实施我们的技术，以每帧仅0.5 ms的速度计算无噪声的环境光遮挡解决方案（在Sony Playstation 4上以1080p运行的游戏）.

基于此公式，我们扩展了环境光遮挡解决方案，以模拟一组单独使用环境光遮挡时通常会忽略的照明效果.一方面，我们引入了一种近似技术,可以非常快地计算校正因子来解决近场全局照明问题。该技术基于以下观察结果：局部表面反照率和环境光遮挡项与近场照明的多次反射之间的关系.根据这一观察，我们开发了一种高效，简单且局部的技术来解决仅计算环境光遮挡时丢失的局部照明.

最后,我们提出了一种与AO对称但又可用于任意镜面材质的新技术,称为地面真镜面光遮挡（GTSO），我们开发了它的公式,并在近似可见性的基础上提出了一种有效的计算方法 取决于该点的弯曲法线和环境光遮挡。考虑到表面的遮挡，GTSO可以有效地计算来自基于探针的照明的镜面反射。

2 背景和相关工作

从法线为的点到方向的反射辐射度可以建模为:

其中是以为中心并以为轴的半球,是从方向抵达的入射辐射,是处的BRDF.这是一个递归运算符，取决于所有场景中反射(和发射)辐射. 尽管许多工作都致力于解决这一问题,但在游戏等高要求的场景中解决它仍然太昂贵.在这里,我们关注环境光遮挡技术,并参考Ritschel等人的调查.[RDGK12]可以对该领域进行更广泛的概述.

通过引入一组假设,环境光遮挡[ZIK98]近似于等式(1):i）所有光都来自无限均匀的环境光，它可能被周围的几何形状所遮挡；ii）周围的所有表面都是纯吸收性的(即不反射任何光),并且iii）x处的表面是漫反射的.这将等式(1)转换为

其中是点处的环境遮挡项,是反照率的漫反射BRDF,而是在方向上的可见性项,如果在方向上有一个遮挡体比给定距离更近,则返回0,在其他位置返回1.请注意,以前的论文[ZIK98，Mit07，BSD08]将此可见性项建模为相对于到遮挡物的距离的衰减函数,称为*遮蔽[obscurance]*.该衰减函数用于创建临时解决方案,以避免由于忽略近场交叉反射而产生的典型AO过暗现象.相反,我们引入了一种新的公式来添加这种损失的光(第5节),同时保持辐射度正确的环境光吸收项.值得一提的是,还有一个关于环境光遮挡的替代定义,其中包括缩短距离:在本文的其余部分中,我们遵循辐射校正正确的余弦加权公式,在附录A中,我们以这种替代形式描述我们的技术.

**屏幕空间环境光遮挡** 环境遮挡项受场景中所有几何形状的影响，并且通常通过光线跟踪[ZIK98]计算，尽管存在更适合交互式渲染的基于点的方法[Bun05，REG + 09].但是,对于实时应用而言,这仍然太昂贵了.为避免进行昂贵的三维全场景可见性计算,Mittring [Mit07]建议将所有计算移至屏幕空间,假设只有摄像机可见的几何体充当遮挡物.这是通过在周围的点球体中采样GPU的场景深度图,并评估深度图中的点是否在几何图形的后面（在几何图形的后面）来实现的.此后,对屏幕空间采样的概念进行了一些改进,改进了采样策略[LS10，SKUT + 10，HSEE15],并通过过滤[MML12]降低了噪声.

**基于水平的环境光遮蔽** Bavoil等人[BSD08]提出了基于最大角度获得光线的方法来计算非遮挡区域.他们将积分域转换为一组参数，该一组参数由与表面相切的参数化,然后在每个方向上计算总的非闭合立体角,将等式(2)转换为:

其中项用于归一化为1(即).请注意,这里我们区分了实际的环境光遮挡和近似的屏幕空间项.

后来, Alchemy环境模糊[MOBH11，MML12]提高了屏幕空间方法的鲁棒性,并提高了所用采样程序的效率.尽管HBAO相对有效,但仍很昂贵,因为在找到最大水平线时需要从每个像素收集深度图的许多样本.Timonen [Tim13a]通过在所有图像上执行线扫来改善了这一点,从而允许他通过沿图像中的许多像素摊销采样,在恒定时间内找到给定方向的最大水平角. 与我们的工作密切相关,同一位作者[Tim13b]提出了一种新的环境光遮挡估算器，该估算器可以通过线扫描和过滤深度图，以较低的成本匹配地面真值解，从而甚至可以计算环境光遮挡 适用于非常大的聚集半径，覆盖整个屏幕.

我们的工作通过提出一种有效的环境光遮挡方式来改进这些工作，而无需临时衰减函数，通过允许非常高效的分析集成来节省计算时间.避免临时衰减函数的核心是我们的有效近似值，其中包括来自近场遮挡物的间接照明。此外，所有这些工作都假设是漫反射表面:相反,我们将环境光遮挡的概念推广到非朗伯曲面,从而引入了镜面光遮挡技术.

3 总览

在这项工作中,我们有两个主要目标:一是我们想要一种与地面真实结果相匹配的环境光遮挡技术,同时还要足够高的效率以用于要求苛刻的实时应用中;二是我们要扩展可以有效近似的全局照明效果的数量.第一个目标在输入数据,渲染次数和指令数量方面施加了严格的约束.受限于这些约束,我们开发了一种在屏幕空间工作的技术，仅将深度贴图和表面法线(可以通过微分从中得出或可以单独提供)作为输入,并且可以共存和增强其他来源整体照明(特别是烘烤的辐照度).为了实现第二个目标,我们放宽了对传统环境光遮挡所做的一些假设.尤其是，在我们假设白色(或单色)圆顶照明的同时,我们放宽了纯朗伯曲面的假设，并包括近场遮挡物的漫反射.

消除以前的限制,我们可以将公式(2)转换为:

其中是菲涅耳反射率项,是我们的环境光遮蔽项(第4节),它与地面真相的结果相匹配,我们称之为地面真实环境光遮蔽(GTAO,是根据环境光遮挡项引入了漫反射近场间接照明（第5节）的函数,而是镜面光遮挡项(第6节),乘以与BRDF预卷积的乘积.接下来,我们将描述该公式中每一项.

4 GTAO:地面真实的环境光遮蔽

我们对环境光遮挡的描述遵循了Bavoil等人[BSD08]的基于水平的方法.但提出了一组关键差异,可以在不牺牲质量的情况下进行有效的计算.首先,我们重新制定水平计算的参考框架,并因此重新定义积分域:我们遵循Timonen [Tim13a]，并计算相对于视角向量的水平角度(见图2).**这意味着将在周围的整个球体中搜索水平线,并将球面积分轴设置为.**在实践中，这使我们可以简化表述，并且正如我们稍后将看到的那样，可以减少所需的超越功能.

第二个主要区别是,与Bavoil的工作不同,我们的能见度项只是一个二元函数，而不是作为遮挡物距离(环境遮蔽度)连续衰减的函数.用这种方式表示AO可以使我们简单地将公式(3)的内部积分计算为方向的两个最大水平角和之间的弧的积分.围绕表示积分,并使用二元可见性项将等式(3)转换为:

其中是法线与视角向量之间的角度,.该公式非常重要,因为允许在解析地计算内部积分的同时匹配地面真实环境遮挡.这意味着,仅需通过具有随机的蒙特卡洛积分就可以对最外面的积分进行数值计算.在下文中,我们将详细介绍如何计算水平角和内部积分.

**计算最大水平角** 方程(5)的解决方案的核心是找到图像平面中以旋转角为参数的方向上最大水平角和.为此,我们在屏幕空间方向和分别搜索(点的投影像素)邻近的像素以获得每一个角,并得出相对于视角向量的最大水平角为:

其中,是像素在图像平面的世界坐标系的投影.角的计算方式与类似,但是在相反的图像平面上.请注意,邻域的大小取决于与相机的距离:这是使视角独立的必要条件,并且将其固定为最大半径(以像素为单位),以避免在非常近的物体上聚集半径过大,这会不必要地破坏GPU缓存.

鉴于我们只对辐射立体角感兴趣,因此我们只需要跟踪最大角度即可,无需像以前的工作一样跟踪其它量(例如最大距离).这允许在AMD GCN硬件[AMD12]（我们的目标平台）上仅使用四分之一速度指令（rsqrt）计算搜索循环.使用此公式,着色器将完全成为内存限制.

**求解内部积分** Timonen[Tim13a]通过在运行时访问的查找表中预先计算结果来解决此积分(包括衰减函数)的问题.但是,我们公式的一个关键特性是，给定我们计算出的水平角和，我们可以解析地求解方程(5)的内部积分为

要注意,该公式要求法线位于由和定义的平面P中,通常不成立.按照Timonen [Tim13a],我们将角度计算为归一化投影法线与之间的角度,因此.然后,我们乘以的模来校正点积的变化,从而使等式(5):

我们发现我们的分析解决方案非常快,特别是使用快速acos和sqrt指令[Dro14].在先验函数方面,经过优化后，我们得到的代码只有2个cos和1个sin，再加上三个附加的acos函数来设置集成域.这使我们的着色器内存有限，因此所需的ALU操作在性能方面几乎没有区别.

4.1 实现

我们的技术受内存限制，因此对内存的访问次数决定了目标平台的最终性能。鉴于我们的性能目标是将技术整合到每秒60帧的游戏中，因此我们只有大约半毫秒的时间来进行计算，这使得实现优化成为必须。一方面，我们以半分辨率计算环境光遮挡，然后将其上采样为全分辨率。而且，为了在不损害性能的情况下尽可能多地计算样本，我们在空间和时间上分布遮挡积分：我们在每个像素仅一个方向采样地平线，但使用在4×4邻域中收集的信息使用双边滤波器进行重建。此外，我们通过在6个不同的旋转之间交替并使用指数累积缓冲区重新投影结果来积极利用时间相干性。所有这些，每个像素总共提供4×4×6 = 96个有效采样方向.图3显示了空间和时间聚集对最终重建的影响。

与环境模糊技术相反,在我们的公式中,我们不考虑任何衰减函数,因为衰减函数可能会导致计算出的遮挡突变,特别是因为我们的聚集半径无法覆盖整个屏幕.为了最小化伪影,我们采用保守的衰减策略.这样做的目的是确保地面真相近场遮挡,而对于远场遮挡则将其衰减为零,因为在我们的用例中，通常将远场遮挡与间接照明一起烘焙.我们的衰减函数是从给定的足够大的距离到最大搜索半径从1到0的线性混合.

最后，由于我们无法从深度缓冲区推断出厚度，因此较薄的特征往往会产生过多的遮挡，从而难以实现.虽然这可以通过例如深度剥离,在我们的案例中是不切实际的.取而代之的是，我们引入一种保守的启发式方法，该假设是基于对象的厚度与其屏幕空间大小相似得出的.这种启发式方法引入了对水平搜索的修改(等式(6)),因此对于每次迭代的搜索中,我们将范围更新为:

其中blend是基于指数移动平均值的混合算子,且.此启发式方法还具有不偏斜简单角(例如墙)的遮挡结果的属性,这在我们的应用中很常见.图4显示了这种启发式的效果.

5 近似基于遮挡的间接照明

环境光遮挡的主要假设之一是,到达着色点的唯一光直接来自均匀的照明环境.这意味着由于多次表面反射而入射的光完全消失了.这种能量损失转化为边缘和角落的过度暗化:这些区域是环境光遮挡影响最大的区域,但是这些区域也是近场交叉反射更为主要的区域. 先前有关环境遮蔽的工作（例如[MML12，Tim13b]）使用临时衰减功能来使部分光返回.但是,这些是临时解决方案,没有潜在的物理意义.

为了解决这个问题,在保持效率的同时避免全局照明解决方案的昂贵计算,我们做出了关键的观察,即恒定反照率区域中的点的近场间接照明与周围环境遮挡值表现出很强的关系.通常对于变化的反照率并非如此,但我们最关心附近的将近距离间接照明反射到的遮挡物，这使得类似反照率的假设更有可能发生.此外，由于我们不想在每个遮挡物表面采样反照率,以保持在运行时性能要求的范围内，因此在环境光遮挡邻域中的反照率假设几乎恒定.

基于此关键观察,并假设周围所有点的反照率为,我们希望设计处的反照率和环境光遮挡之间的映射以及处的全局反射照明.为了构建此函数,我们计算了七个具有不同反照率的模拟()在一组场景中显示了各种不同类型的遮挡条件(请参见图5).我们同时计算环境光遮挡和多次反射间接照明(在我们的示例中,最多三个反射).图6显示了每个反照率以及图5中每个场景的和之间的映射.通过结合所有点,我们使用三次多项式对每个反照率拟合该映射(图7(左)),为每个场景反照率生成一组多项式系数.然后我们观察到，所述系数通过线性拟合作为输入反照率的函数很好地近似(图7).最后的观察结果使我们能够在反照率和环境光遮挡之间建立二维映射:

这种基于拟合的近似满足了一些要求:一方面,它可以无缝集成在任何环境光遮挡框架中,从而可以重新整合由于全局照明而导致的缺失能量.另一方面,它非常有效,因为它基于已经计算出的信息,并且不需要昂贵的光传输模拟,同时给出了看起来合理的结果.这使其非常适合我们的目标实时应用程序.

6 GTSO:镜面遮挡

在这里,我们介绍镜面光遮挡的解决方案,这是基于Lambertian的环境光遮挡的光泽对应项.因此，我们希望开发一种照明模型,其中近场光遮挡在支持任意BRDF的同时调制基于探测器的照明.此外,对于恒定探头照明的特定情况，我们希望有一个模型可以提供地面真实结果,类似于AO.

如第4节所述,假定所有光都来自无限远的照明环境(光探测器),以将等式(1)表示为:

通过数值积分计算该积分对于实时应用而言过于昂贵,并且当使用Cook-Torrance [CT82]微面BRDF时,当前的最新技术是采用假定恒定完美可见性的公式(),并使用分裂积分近似[Laz13，Kar13]作为:

其中是第一个积分所需的标准化因子,以确保其始终在, 当时,是表面[TS67]的法线分布函数,而是半矢量.直观地,第二个积分是在均匀白色照明环境下像素处的整个微面BRDF,并且可以存储在预先计算的查找表中,而第一个积分是实际照明环境与近似于Cook-Torrance BRDF分布函数的圆对称波瓣之间的卷积.当我们将照明环境表示为辐照度立方体贴图时,可以通过将立方体贴图与来自不同表面粗糙度的波瓣预卷积来计算此第一个积分，这使它非常有效地渲染有光泽的材质,尽管大多数近似方法都忽略了遮挡或使用启发式方法对其进行了近似.

为了计算镜面封闭光照,我们选择一种类似于分裂积分近似的方法,并将可见性项与第一个积分分离为常数。本质上，该想法是计算一个遮挡项，该项会根据环境遮挡的精神来调节达到x的照明量。这使我们可以将公式（12）转换为三个积分的乘积:

其中建模可见性是我们的镜面遮挡项,计算公式为:

其中归一化项确保镜面遮挡的范围为.如我们所见,镜面反射遮挡的定义由BRDF加权,因此与方向相关.在以下小节中,我们将详细说明镜面反射光遮挡的计算.

有趣的是,归一化因子与后者的积分相同,因此在等式(13)中被抵消,而使等式为:

这种最终形式具有以下特性:对于恒定的波瓣照明,它与地面真相完全匹配.此外，如果将其与原始的拆分积分近似值(公式(12))进行比较,可以看到主要区别在于可见性项已移至BRDF积分,并且不再假定为常数.

6.1 计算镜面反射遮挡

我们有效地计算镜面反射遮挡的关键思想是为可见性和BRDF波瓣建模一个近似值,然后计算这两者之间的交点作为镜面反射遮挡.考虑到这一点,问题就减少到如何紧凑地表示可见性和BRDF以及如何计算两者之间的交点的问题.

对于可见性,我们假设可以将其近似为圆锥,它是根据弯曲法线[Lan02]和环境光遮挡项计算得出的.可以动态计算这两个值(请参见第4节),也可以预先计算这两个值(例如存储为纹理或顶点数据).我们之所以选择这种表示法,是因为它允许我们重用第4节中计算的高质量屏幕空间环境光遮挡的数据.弯曲法线**b**充当圆锥的方向.要计算圆锥幅度,我们依赖于环境光遮挡:假设在弯曲法线周围可见性是均匀的,则是所有方向的最大水平角(请参见公式(5)).基于这些假设,环境光遮挡可以解析地表示为:

取其逆得到锥角关于的函数:

与可见性相似,我们可以将镜面波瓣建模为以反射方向为中心的圆锥.这提出了一些假设,包括将BRDF的波瓣约束为在上旋转对称(对于微面BRDF而言并非如此,但与cubemap预卷积相同,这是近似的),并将实际的BRDF近似为单个常数.但是,这允许将镜面反射遮挡计算为

可见角和镜面锥与镜面锥的立体角的交点的立体角之比(见图8).尽管可以大致近似地反映镜面反射率，但是该比率可以通过分析来计算，并且给出良好的结果。 图8（左）显示了使用视锥近似可见性和BRDF的示例（有关详细信息，请参阅附录B）。

但是,在作为目标的实时应用程序中,这些计算可能仍然很昂贵,并且最终在预先计算的三维查找表中进行烘焙,该表由弯曲法线和反射向量之间的角度以及两个锥体和的振幅参数化而成.考虑到这一点,我们选择了一种更精确的,基于预计算的近似方法,在该方法中,将镜面反射遮挡计算为可见性锥与实际BRDF F的乘积（图8，右）:

其中是一个二进制函数,如果则返回1,而在其他位置返回0.假定具有以粗糙度值参数化的GGX NDF [WMLT07]的基于各向同性的微面BRDF,我们将反射方向建模为相对于法线的角度.基于这些假设,并且为了清楚起见,省略空间依赖性,我们可以将表示为一个四维函数:

该函数可以紧凑地烘焙为四维表.此外,通过假定法线为弯曲法线**b**,则,这将以较小的误差为代价将工作台的尺寸减小为三个为维度.假设函数相对平滑,我们可以将其编码为多维（对于3D近似为）BC4 8位查找表,可以在运行时高效地对其进行访问.