

Fast Many-Lights Rendering with Temporal Shadow Maps

Boning Zhang^{*}, Nian Li, Lili Wang

School of Computer Science and Engineering, Beihang University, Beijing, China

Email address:

Zhangboning234@126.com (Boning Zhang), linian0701@163.com (Nian Li), wanglily@buaa.edu.cn (Lili Wang)

^{*}Corresponding author

To cite this article:

Boning Zhang, Nian Li, Lili Wang. Fast Many-Lights Rendering with Temporal Shadow Maps. *Science Discovery*. Vol. 4, No. 1, 2016, pp. 45-51. doi: 10.11648/j.sd.20160401.18

Received: March 31, 2016; Accepted: April 6, 2016; Published: April 18, 2016

Abstract: We present an efficient method for computing shadows of many light sources (e.g. 1,024). Our work is based on the observation that conventional shadow mapping becomes redundant as the number of lights increases. First, we sample the scene with a constant number of depth images (e.g. 10), which we call temporal shadow maps. Then the shadow map for each light is approximated through rendering triangles reconstructed from all the temporal shadow maps. The cost of rendering the triangles reconstructed from the temporal shadow maps is much smaller than rendering the entire scene when the geometry is complex. The algorithm supports fully dynamic scenes. The experiment results show that our method can produce comparable soft shadows with much higher speed than ray tracing and shadow mapping methods.

Keywords: Many Lights, Visibility, Shadow Mapping

基于临时阴影图的多光源场景绘制方法

张博宁^{*}, 李念, 王莉莉

计算机学院, 北京航空航天大学, 北京, 中国

邮箱

Zhangboning234@126.com (张博宁), linian0701@163.com (李念), wanglily@buaa.edu.cn (王莉莉)

中文摘要: 本文提出了一种计算多光源(例如1,024)阴影的有效方法。在传统的阴影映射中随着光源数量的增加将产生巨大的可见性信息冗余,我们的工作就是在这发现下进行的。我们首先用恒定数量(例如10)的深度图来对场景进行采样,并将这些深度图称为临时阴影图;之后再这些临时阴影图重构出场景的三角网格,并在每个光源处绘制这些三角网格以生成近似的阴影图。绘制这些重构的三角网格的消耗远远小于绘制复杂的原始场景的消耗(参见4.2.小节)。我们的算法支持完全动态场景。实验结果表明,我们的算法能够绘制出可以与传统阴影映射和光线跟踪媲美的高质量阴影图,同时具有很高的帧率。

关键词: 多光源, 可见性, 阴影映射

1. 引言

如何高质量渲染具有大量光源的复杂场景同时保持应用的可交互性仍然是一个有待研究的开放性问题。怎样高效地确定场景中的物体和光源之间的可见性是其中

的关键。尤其在包含数百万三角面片以及数百甚至数千点光源模拟光照的场景中,确定物体可光源间的可见性将是一个巨大的消耗。在交互式图形应用中,渲染阴影的常规方法是阴影映射(shadow mapping),但它并不

适用于复杂的场景, 因为阴影映射需要在每个光源处都渲染一次场景[1]。

在本文中, 我们提出了一种计算多光源(例如1024)阴影的有效方法。在传统的阴影映射中随着光源数量的增加将产生巨大的可见性信息冗余, 我们的工作就是下这一发现下进行的。若 SM_i 为从光源 L_i 处绘制得到的阴影图, SM_i 中包含有其他不同光源 L_j 需要的有效可见性信息。如果存在一个由 k 个临时阴影图组成的集合, 这个集合中包含几乎所有的剩余光源需要的可见性信息, 那么渲染剩余光源处的阴影图将会是冗余的。相反的, 我们的方法从这 k 个临时阴影图中近似地计算每个光源处的阴影图。

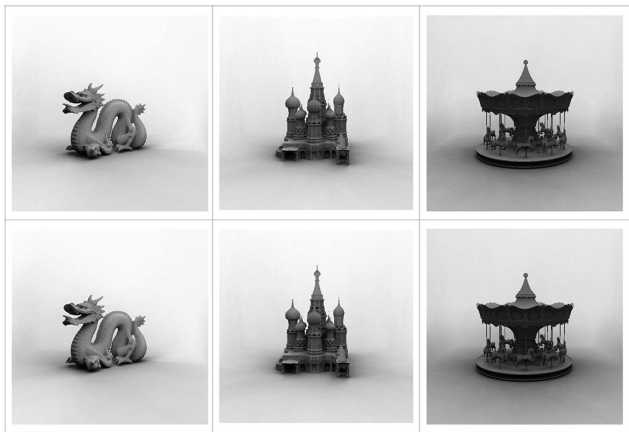


图1 由包含1,024个光源组成的场景分别由我们的方法(第1行)和光线追踪方法(第2行)渲染。与光线跟踪相比, 我们的方法每个像素的平均强度误差(average intensity error)分别为0.1%, 0.7%和1.1%, 同时, 我们的方法相对于光线跟踪的方法的加速比为12x, 15x和17x; 相对于传统的阴影映射的加速比为5x, 8x和10x。

我们的方法分为两步。首先, 我们用常数个(例如10)深度图采样场景, 我们称之为临时阴影图(TSMs); 之后, 我们根据这些临时阴影图重构出三角网格, 并在每个光源 L 处绘制所有临时阴影图中的重建三角网格来近似得到光源 L 处的阴影图。因为临时阴影图的数量是恒定的, 所以渲染从中间阴影图中重建的三角网格的时间消耗也是恒定的, 而且, 在复杂场景中, 这个时间消耗相对于渲染整个场景来说是非常小的。我们在一些复杂场景中测试了我们的方法, 并且获得了较高的阴影质量和很好的工作性能(如图1所示)。

2. 前期工作

阴影映射和光线追踪是阴影计算的传统方法。然而, 在多光源复杂场景中, 这两种方法都过于缓慢[1, 2]。为了加速阴影渲染中的可见性计算, 科研人员提出了两种近似的方法: 场景几何近似(Scene geometry approximation)和光源聚类(clustering Light sources)。场景几何近似用少量的三角面片来近似原始场景, 从而减少计算消耗; 光源聚类则是将光源的聚类来减少光源数量。除了将要介绍的相关工作, 我们还将分享一些最近的多光源技术(many-lights techniques)。

2.1. 基于光线追踪的方法

场景几何近似被广泛用于光线追踪这一基础方法之上。被称为微渲染[3](micro-rendering)的方法使用了点层次树来近似场景的几何结构。每个叶子节点的精确能见度在计算完切断树后由光线投射确定。另一个方法则基于增量体素空间的能见度计算[4], 该方法利用屏幕空间体素化来离散场景的几何形状, 并提出一种高效的增量查询方法来确定场景的可见性。

一些基于光线追踪的方法通过简化光源得到了更多时间上的优势, 但降低了生成的阴影质量。一种利用八叉树对光源进行分簇的方法[5]有效地近似计算了场景的可见性。另一种方法[6]则将场景划分为不同的区域, 再将每个区域内的光源集合为一组, 最后生成了非结构化灯光云。光源切割[7](Lightcuts)是面向多光源场景的一个高效绘制方法[8]。它渐进地对光源进行聚类组织, 并以二叉树进行管理, 由此在绘制时可用一些光源聚类来代表光源(统称为‘光源割’)进行光照计算, 以减少需要计算的光源数量。许多学者在原有的光源切割基础上进行研究以提高可见性的计算精度[9-11]。

还有一些基于光线追踪的方法同时使用光源聚类和场景几何近似以获得更快的绘制速度。例如可见性聚类法[12](Visibility Clusters), 该方法构造了具有较高能见度一致性并利用矩阵的稀疏结构只在集群间投射很少的光线来估计平均能见度。

2.2. 基于阴影映射的方法

与基于光线追踪的方法相比, 基于阴影映射的方法具有更快的速度, 但仍然不足以应对交互式情景下多光源复杂场景。研究员们大多通过场景的几何近似来对阴影映射进行加速。

Ritschel 等人提出一种名为 Coherent Shadow Maps[13](CSM)的方法。该方法利用预计算和压缩深度图, 能够检测移动的物体和大量光源之间的可见性。但这种方法不适用于放置在物体表面上的虚拟点光源, 仍需要计算全局光照中的间接光照。为解决这一问题, 研究员们提出了 Coherent Surface Shadow Maps[13](CSSMs)。该方法使用本地立方体贴图(Cube Maps), 对可见性有了更精确的近似。名为 Virtual Area Lights[14](VALs)的方法利用 CSSMs 直接计算由少数虚拟面光源(VALs)生成的软阴影, 相对于对每个点光源计算阴影这种方法有更好的性能。Imperfect Shadow Maps[15](ISM)是一种在完全动态场景中更加常见的多光源可见性判断方法。该方法由点来近似场景, 并以此生成低分辨率的阴影图。ManyLoDs[16]使用包围盒来层次近似场景的几何构成。虚拟阴影图, [17, 18]法对场景进行分簇, 并创建影响每个簇的光源列表, 以此来近似阴影。

在矩阵行列采样法[19](Matrix Row-Column Sampling)中, 研究员用矩阵的列表示被单独光源照亮的每个像素采样; 用矩阵的行则表示被所有灯光照亮的一个单独采样。该方法首先使用传统的阴影映射来计算一些有代表性的行和列的集合。矩阵中行通过在該行的采样点对应的视点处渲染得到; 矩阵的列通以該列对应的光为视点

出渲染场景得到。之后其他的元素由这些具有代表性的元素插值得到。可见性聚类法[20] (Visibility Clustering method) 首先将灯光聚类, 再渲染一个具有代表性的阴影图来近似表示每个类的可见性, 并将把个近似的可见性利用每个灯光的精确遮挡结合起来。相比于传统的矩阵行列采样法可见性聚类法只需渲染很少的阴影图。矩阵行列采样法还被用于渲染拥有海量几何体和复杂光源的大规模场景[21]。另一项对于该方法的扩展使用了新的矩阵采样和法线策略来聚集光照效果, 使得只需要采样少量灯光和表面顶点的可见性就能够达到较好的阴影效果。

我们的方法是基于阴影映射的。类似于可见性聚类法, 我们的方法首先渲染了一系列临时阴影图, 然后将这些临时阴影图反投影到每个光源视点下。相比较于可见性聚类法, 我们的方法计算得出的能见度更为精确。

3. 基于临时阴影图的多光源场景绘制方法

我们的方法避免了渲染大量阴影图时产生的冗余。我们用极小数量的临时阴影图来近似每个光源出的阴影图。在研究中我们发现了多种方法来有效利用包含在临时阴影图中的可见性信息。

其中一个方法利用到了对极几何 (Epipolar Geometry)。对于输出图像的一个采样点 S_{uv} , 由光源 L_i 发出的光线 $L_i S_{uv}$ 和临时阴影图 TSM_j 的交点能够由向 TSM_j 投射 $L_i S_{uv}$, 并追踪投影得到。该方法在反转三维图像的翘曲[22]中被提到, 并且随后被用在了浮雕纹理贴图 (relief texture mapping)[23] 和镜面反射渲染 (specular reflection rendering)[24]中。该方法的优点是减少了光线和深度图求交时的计算消耗。

然而不同于以上几种情景中每个输出图像的像素只跟一条光线关联, 在软影效果绘制中每个像素都跟 n 条光线相关, 其中 n 为光源数量, 尤其在大规模场景中, n 可能成百上千。幸运的是, 这些大量的光线集合在阴影渲染中是有相关性的。这使得我们有了另一个更为有效的方法来利用临时阴影图 TSM_j 中可见性信息来近似得到光源 L_i 处的阴影图。在本文中我们主要介绍这一方法。在该方法中我们利用每个阴影图 TSM_j 重建出了三角形网格, 并在每个光源 L_i 处渲染这些三角网格。考虑到 GPU 强大的并行计算能力, 我们的方法主要在 GPU 上运行。

3.1. 算法概述

算法1描述了我们算法的主要步骤。

算法 1 基于临时阴影图的多光源场景绘制方法

输入: 包含 N 个三角面片和 n 个光源 L_i 的场景 S , k 个参考视点 V_j 和注视点 V 。

输出: 在主视点 V 处绘制的包含阴影的场景 S 的效果图 I

1: for $j=1$ to k do

2: 在参考视点 V_j 处渲染临时阴影图 TSM_j

3: 利用 TSM_j 重构三角网格 TM_j

4: 在主视点 V 处渲染不带阴影的效果图 I

5: for $i=1$ to n do

6: 将立方体阴影图 SM_i 初始化为空

7: for $j=1$ to k do

8: 在光源 L_i 处渲染 TM_j 生成 SM_i 每个面的阴影图

9: 利用 SM_i 将相应的阴影添加到效果图 I 中

10: Return效果图 I

在步骤1-3中, 我们用传统的阴影映射方法从参考视点 V_j 处渲染场景得到临时阴影图, 以此来对场景进行一致且完整的采样。我们将生成临时阴影图的10个参考视点放在了场景AABB包围盒的八条棱的中点以及左右面的中心 (图2)。

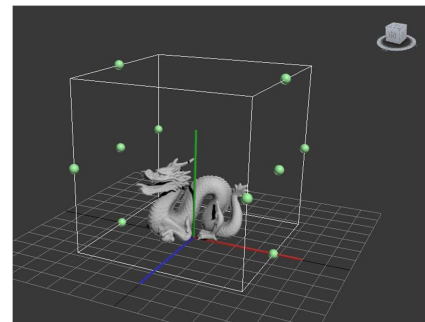


图2 绘制临时阴影图的参考视点放置。

之后在临时阴影图 TSM_j 中每个 2×2 邻域定义两个三角形, 以此将 TSM_j 转换为三角网格 TM_j 。如果相邻像素之间的深度不连续将不能生成三角形。深度不连续性是通过深度图的二阶差分结果设置阈值过滤得到。二次差分具有表面方向无关特性, 例如: 对于任何方向的平面, 其上的二次差分总是为0。步骤3将在章节3.2. 中详细描述。

步骤4渲染了不包含阴影的效果图 I , 这是对主视点下可见场景的完整采样。步骤5-9对效果图 I 添加了每个光源 L_i 处产生的阴影。我们在光源 L_i 处渲染由所有临时阴影图重构的三角网格 TM_j 来得到近似的立方体阴影贴图 SM_i 。最后 SM_i 被用来计算光源 L_i 处产生的阴影并被加到效果图 I 中。

3.2. 利用临时阴影图重构三角网格

我们使用GPU将临时阴影图重构为三角网格。通过对临时阴影图中每个 2×2 并行处理来减少运行时间, 具体的描述见算法2。

算法 2 利用临时阴影图重构三角网格

输入: 临时阴影图 TSM_j

输出: 由 TSM_j 重构出的三角网格 TM_j

1: for each TSM_j 采样点 (u, v) do

2: if IsConnected((u, v) , $(u, v+1)$, $(u+1, v+1)$) then

3: $TM_j += [(u, v), (u, v+1), (u+1, v+1)]$

4: if IsConnected((u, v) , $(u, v+1)$, $(u+1, v+1)$) then

5: $TM_j += [(u, v), (u, v+1), (u+1, v+1)]$

6: Return TM_j

采样点 (u, v) 为每个 2×2 邻域的左上采样点。每个邻域的4个采样点由两个三角形连接。每个通过连续性测试的三角形都将被保留。三角形连通性测验函数 $IsConnected(a, b, c)$ 将检测三角形每条边 (a, b) , (b, c) , (a, c) 的连通性。举例来说, 如果边 $((u, v), (u, v+1))$ 要通过连通性检验必须同时满足以下条件:

$$|z(u, v-1) + z(u, v+1) - 2z(u, v)| < \varepsilon \quad (1)$$

$$|z(u, v) + z(u, v+2) - 2z(u, v+1)| < \varepsilon \quad (2)$$

其中, $z(u, v)$ 表示采样点 (u, v) 处的深度值, ε 则是跟场景相关的深度阈值。

图3表示了一定情景下算法保留和丢弃的三角形, 其中包括了无效的三角形和空白的采样点(例如背景采样点)。

4. 结果和讨论

在本章中, 我们讨论了使用我们方法渲染出的阴影质量, 同时呈现了我们测试的帧率, 最后我们还讨论了方法的局限性。

我们用以下几个场景测试了我们的方法: 龙模型(871k三角面片), 教堂模型(1,868k三角面片), 旋转木马(1,336k三角面片), 城市场景(1,117k三角面片), 飞机矩阵(1,000k三角面片)和草模型(1,198k三角面片)。本文中所有的性能测试都在一下平台上展开: 配有3.8GHz Intel(R) Core(TM) i7-2600K CPU, 12 GB RAM 内存和 NVIDIA GeForce GTX 660 显卡的计算机。我们使用了 NVIDIA 公司的 Optix 光线跟踪器来实现光线追踪产生阴影。

4.1. 质量测试

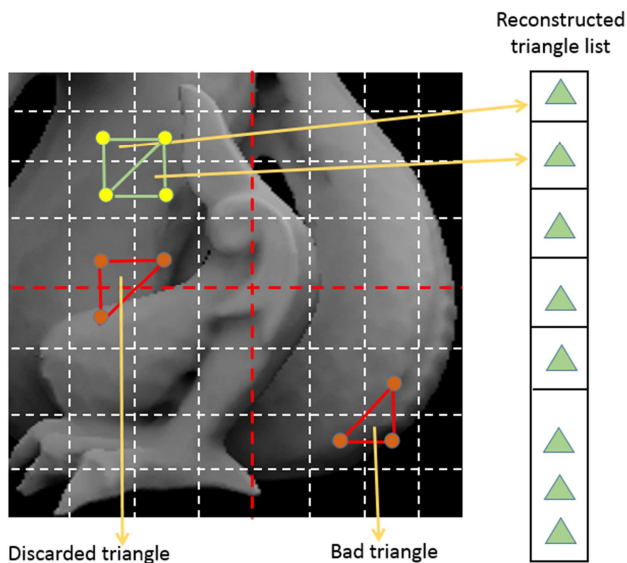


图3 从临时阴影图中重建三角网格。

我们将使用光线跟踪法渲染的效果图视为真实值, 并把使用我们的方法绘制出的图片和这些真实值做了对比。我们使用每个像素的平均阴影强度误差作为我们的方法跟真实值间的误差度量。尽管在可见度计算中, 这是一种较为宽松的误差度量方法, 因为不同灯光的能见度误差可能会产生抵消(例如一个在光源下产生了正的误差, 而在另一光源下则产生了负的误差), 但平均强度误差却是和效果图的最终质量息息相关的。

尽管是和光线跟踪作对比, 我们的方法在包含1,024个光源的复杂场景中仍然取得了很好的渲染效果(见图1和图4)。在所有的实验中, 临时阴影图的默认分辨率为 128×128 , 在光源处渲染的阴影图的默认分辨率为 512×512 , 默认的临时阴影图数量是10, 默认的光源数量是1,024。我们在图5中对图1和图4中的6个场景误差进行了可视化。

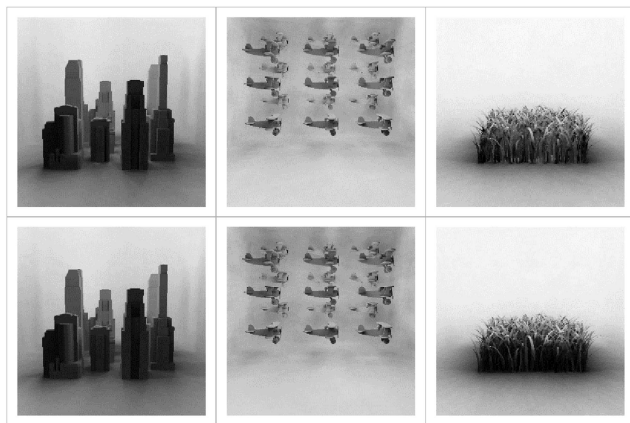


图4 包含1,024个光源的另外几个场景, 分别使用我们的方法(第1行)和光线跟踪方法(第2行)绘制, 与光线跟踪方法相比, 我们的方法产生的相对误差分别为0.8%, 3.4%和3.9%, 加速比为15×, 21×和24×。

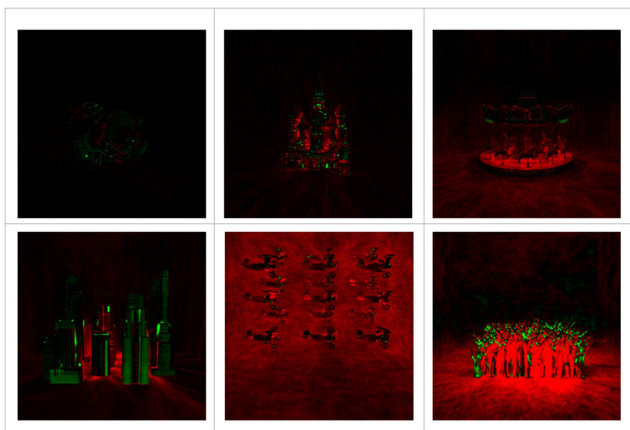


图5 我们的方法绘制的图1和图4中场景的误差可视化。为了使观察更为简单我们对误差进行了10倍放大。红色的像素点表示我们的方法产生了更亮的结果(丢失阴影), 绿色的像素点表示我们的方法产生了更暗的结果(多余阴影)。

我们的方法对场景的遮挡有一定的低估, 因为我们在渲染每个光源处的阴影图时只是使用了由临时阴影图重构出的三角网格, 这和原有场景相比, 表面会有不可避免的侵蚀, 从而导致更多的光线穿过场景。因此, 这很小的误

差导致了一般情况下我们的绘制效果比真实效果稍微偏亮。偶然出现的比真实效果偏暗的情况是由于深度连续性检验中出现偏差,导致重构出的三角网格有了多余的遮挡。在6个实验场景中,飞机矩阵场景和草模型是对我们的方法来说最为极端的场景,因为这两个场景中都包含了很高的深度复杂度。尤其是我们使用极少数的临时阴影图重构三角网格展现场景中及其细微的细节和表面遮挡。然而尽管是在这些极端场景中,我们的方法也只是产生了很小的误差。

表1,表2,和表3分别说明了在龙模型中临时阴影图的分辨率,近似阴影图的分辨率和临时阴影图的数量对渲染结果误差的影响。和预料中的一样,渲染误差随着这些参数的增大而减小。在这个场景中,这三个参数的默认值分别是128×128,512×512和10。

表1 绘制误差随临时阴影图(TSM)分辨率的变化。

TSM的分辨率	64×64	128×128	256×256	512×512
平均相对误差	0.56%	0.12%	0.09%	0.08%

表2 绘制误差随每个独立光源的近似阴影图(ILSM)分辨率的变化。

ILSM的分辨率	256×256	512×512	1024×1024	2048×2048
平均相对误差	0.18%	0.12%	0.10%	0.09%

表3 绘制误差随临时阴影图(TSM)数量的变化。

ILSM的数量	6	8	10	12
平均相对误差	0.30%	0.16%	0.12%	0.11%

4.2. 性能测试

我们分别和NVIDIA的Optix光线跟踪器以及传统的阴影映射进行了性能上的对比。表4提供了每个场景使用三种方法时平均每一帧的渲染时间。我们的方法比光线跟踪快11.8–24.9倍,比传统的阴影映射快5.0–17.5倍。速度的增加主要来自在计算每个光源的阴影图时我们用临时阴影图重建的三角网格替换了场景原始的几何结构。在这些场景中我们重构出的三角网格的三角面片数量分别是119k, 43k, 135k, 69k, 112k, 和94k, 这与原始场景的三角面片规模来比是相当小的。例如要渲染含有 N 个三角面片和 n 个光源的场景,我们使用了 k 个分辨率为 $w \times w$ 的临时阴影图,用我们的方法渲染这个场景所需绘制的三角面片数量至多为 $kN + 2nkw^2$ 。而传统的阴影映射则需要绘制 nN 个三角面片。因此我们的方法在多光源复杂场景中有更好的表现。

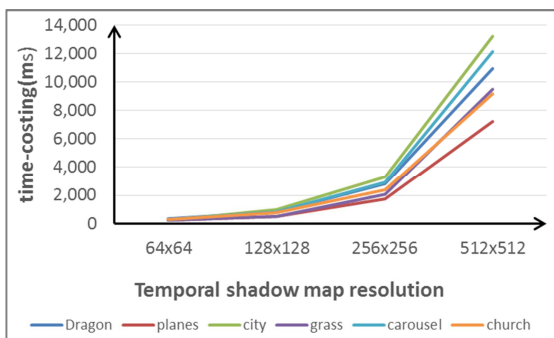


图6 每一帧的平均渲染时间随临时阴影图分辨率的变化。

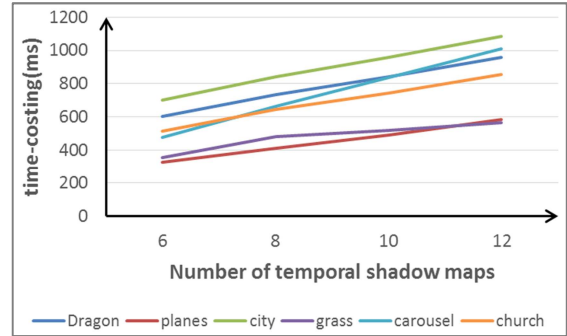


图7 每一帧的平均渲染时间随临时阴影数量的变化。

图6,图7和图8的曲线图表示我们的渲染性能和临时阴影图的分辨率二次相关,和临时阴影图的数量和灯光数量线性相关。

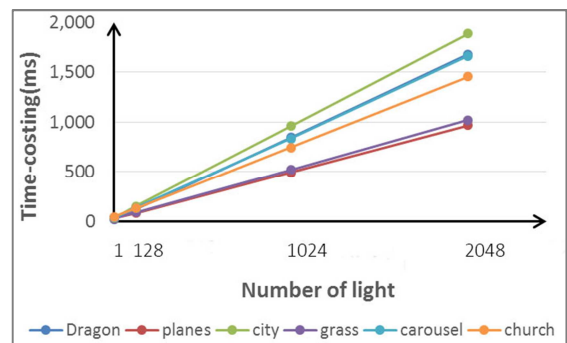


图8 每一帧的平均渲染时间随场景中光源数量的变化。

图9的曲线图表示每个独立光源均需计算的近似阴影图的分辨率对性能并没有太大的影响,这也表明独立光源近似阴影图渲染并不是帧率的限制所在。

4.3. 方法限制

正如上文提到过的,我们的方法使用临时阴影图近似了场景的几何表面,这在一定程度上导致了当对场景的采样不完整时更多的光线穿过场景。尤其当场景含有复杂的深度信息时,采样的效率将大大降低,例如含有较多层次结构,高复杂度表面和具有大量微小细节的场景。我们采用了一种直接的方法来面对这些问题:提高临时阴影图的数量和分辨率。我们根据场景的复杂度和实际应用对这两个参数选取合适的值。

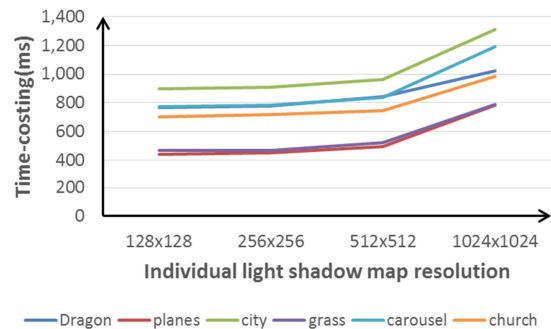


图9 每一帧的平均渲染时间随每个光源独立阴影图分辨率的变化。

另一个值得指出的限制是，为了使我们重构出的三角网格规模相对来说更小，需要有相当复杂的原始场景；同时为了能在渲染近似阴影图中抵消前期绘制临时阴影图和重构三角网格的时间消耗，需要有足够多的光源。我们的方法更适用于多光源高复杂度的场景。

表3 分别使用我们的方法 (TSM), 传统阴影映射 (SM) 和光线跟踪 (RT) 绘制场景时每一帧的平均渲染时间。

场景	TSM (ms)	SM (ms)	SM/TSM	RT	RT/TSM
龙模型	844	4,200	5.0×	10,000	11.8×
教堂	744	13,000	17.5×	18500	24.9×
旋转木马	825	8,300	9.9×	14300	17.1×
城市	960	7,300	7.6×	12500	15.2×
飞机矩阵	493	4,100	8.3×	10500	21.3×
草模型	680	9,400	13.8×	16000	23.5×

5. 结论和未来工作

我们针对多光源场景渲染提出了一种普遍且高效的算法。该算法对具有百万面片和数千光源的超动态复杂场景具有很高的鲁棒性，并能够渲染高质量的软阴影。我们只需绘制极少次原始场景以生成临时阴影图，再利用临时阴影图重建出的三角网格来近似得到每个光源的可见性从而减少了在多光源阴影绘制中传统阴影映射方法带来的冗余。临时阴影图中包含每个独立光源的足够多的可见性信息。我们将临时阴影图反投影到每个独立的光源视点下以获得这些信息。我们的方法没有利用插值来近似可见性，因此产生的阴影是明显非线性的。我们也没有对光源进行聚簇，而是对每个独立光源都单独进行了可见性计算。在实验中，我们场景中的光源由标准的均匀分布逐渐聚集到一起最后成为一个点光源，我们的方法绘制出了高质量的阴影，逐渐地由软阴影变为硬阴影，并没有发生跳变。

我们用多个不同的场景对我们的方法绘制的效果和以光线跟踪以及传统阴影映射绘制的效果进行了对比。结果显示我们以很小的误差为代价换来了极大的性能提升。

我们的方法使得可用于交互式图形学应用中光源有了极大的提高。今后工作的一个重要方向是提供一个算法以自动放置和校准各个独立的光源。我们还将把我们的方法拓展于全局光照领域作为另一个未来工作的重要方向。

参考文献

[1] WILLIAMS L. Casting curved shadows on curved surfaces[J]. Acm Siggraph Computer Graphics, 1978, 12(12):270 - 274.

[2] WARD G J. Adaptive Shadow Testing for Ray Tracing[M]. [S.l.]: Springer Berlin Heidelberg, 1991: 11 - 20.

[3] ITSCHEL T, ENGELHARDT T, GROSCH T, et al. Micro-rendering for scalable, parallel final gathering[J]. Acm Transactions on Graphics, 2009, 28(5):89 - 97.

[4] NICHOLS G, PENMATSA R, WYMAN C. Interactive, Multiresolution Image-Space Rendering for Dynamic Area Lighting[C]//Computer Graphics Forum.. [S.l.]: [s.n.], 2010, 29:1279 - 1288.

[5] PAQUETTE E, POULIN P, DRETTAKIS G. A Light Hierarchy for Fast Rendering of Scenes with Many Lights[J]. Computer Graphics Forum, 1998, 17(3):63 - 74.

[6] KRISTENSEN A W, AKENINE-MÖLLER T, JENSEN H W. Precomputed local radiance transfer for real-time lighting design[J]. Acm Transactions on Graphics, 2005, 24(3):1208 - 1215.

[7] WALTER B, FERNANDEZ S, ARBREE A, et al. Lightcuts: A Scalable Approach to Illumination[J]. Acm Transactions on Graphics, 2005, 24(3):págs. 1098 - 1107.

[8] 王光伟, 谢国富, 王文成. 基于空间聚类增强lightcuts的光照计算[C]//第十一届中国虚拟现实大会 (ICVRV2011). [S.l.]: [s.n.], 2011:2364 - 2370.

[9] KERLUND O, UNGER M, WANG R. Precomputed Visibility Cuts for Interactive Relighting with Dynamic BRDFs. [C]//Conference on Computer Graphics & Applications. [S.l.]: [s.n.], 2007:161 - 170.

[10] CHESLACK-POSTAVA E, WANG R, AKERLUND O, et al. Fast, realistic lighting and material design using nonlinear cut approximation. [J]. Acm Transactions on Graphics, 2008, 27(5):32 - 39.

[11] WALTER B, KHUNGURN P, BALA K. Bidirectional lightcuts[J]. Acm Transactions on Graphics, 2012, 31(4):13 - 15.

[12] WU Y T, CHUANG Y Y. VisibilityCluster: Average directional visibility for many-light rendering[J]. Visualization and Computer Graphics, IEEE Transactions on, 2013, 19(9):1566 - 1578.

[13] RITSCHEL T, GROSCH T, KAUTZ J, et al. Interactive global illumination based on coherent surface shadow maps[C]//Graphics Interface.. [S.l.]: [s.n.], 2008:87 - 96.

[14] DONG Z, GROSCH T, RITSCHEL T, et al. Real-time Indirect Illumination with Clustered Visibility. [C]//VMV.. [S.l.]: [s.n.], 2009:187 - 196.

[15] RITSCHEL T, GROSCH T, KIM M H, et al. Imperfect shadow maps for efficient computation of indirect illumination[J]. Acm Transactions on Graphics, 2008, 27(5):32 - 39.

[16] HOLLANDER M, RITSCHEL T, EISEMANN E, et al. ManyLoDs: Parallel Many-View Level-of-Detail Selection for Real-Time Global Illumination[C]//Computer Graphics Forum.. [S.l.]: [s.n.], 2011:1233 - 1240.

- [17] OLA O, MARKUS B, ERIK S, et al. More Efficient Virtual Shadow Maps for Many Lights. [C]//Meeting of the ACM SIGGRAPH Symposium on Interactive 3d Graphics and Games.. [S.l.]: [s.n.], 2014:701 – 713.
- [18] OLSSON O, SINTORNE E, KÄMPE V, et al. Efficient virtual shadow maps for many lights[C]//Proceedings of the 18th meeting of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games.. [S.l.]: [s.n.], 2014:87 – 96.
- [19] HAŠAN M, PELLACINI F, BALA K. Matrix row-column sampling for the many-light problem[C]//ACM Transactions on Graphics (TOG). [S.l.]: [s.n.], 2007, 26:26.
- [20] DAVIDOVIC T, KRIVÁNEK J, HAŠAN M, et al. Combining global and local virtual lights for detailed glossy illumination[C]//ACM Transactions on Graphics (TOG).. [S.l.]: [s.n.], 2010, 29:143.
- [21] WANG R, HUO Y, YUAN Y, et al. GPU-based Out-of-Core Many-Lights Rendering[J]. Acm Transactions on Graphics, 2013, 32(6):2504 – 2507.
- [22] YANG T, HUI-ZHONG W, FU X, et al. Inverse image warping without searching[C]//International Conference on Control, Automation, Robotics and Vision.. [S.l.]: [s.n.], 2004:386 – 390.
- [23] OLIVEIRA M M, BISHOP G, MCALLISTER D. Relief Texture Mapping[C]//Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques. 2000. NewYork, NY, USA:ACM Press/Addison-Wesley Publishing Co., SIGGRAPH ' 00, <http://dx.doi.org/10.1145/344779.344947>.
- [24] FERIS R, RASKAR R, TAN K H, et al. Specular Reflection Reduction with Multi-Flash Imaging[C]//Proceedings of the Computer Graphics and Image Processing, XVII Brazilian Symposium. 2004. Washington, DC, USA: IEEE Computer Society, SIBGRAPI' 04, <http://dl.acm.org/citation.cfm?id=1025131.1026268>.
- [25] Ola O, Markus B, Erik S, et al. More Efficient Virtual Shadow Maps for Many Lights. [J]. Proceedings of the Acm Siggraph Symposium on Interactived Graphics & Games, 2015, 21(6):701–13.
- [26] Bárbara G M. Fiscal state aid: lights and shadowx in the 2013 commission decision on teh Sapanish tax lease system[J]. European Taxation, 2015, 55:págs. 69–77.
- [27] Pagano E, Sobrero S, Cavallero C, et al. Economic Considerations on the Follow-Up Practice in Gynecologic Cancers: Few Lights and Many Shadows From a Literature Review. [J]. International Journal of Gynecological Cancer, 2015, 25.
- [28] Addison D M, Stewart B. Nighttime Lights Revisited: The Use of Nighttime Lights Data as a Proxy for Economic Variables[J]. Social Science Electronic Publishing, 2015, 53(1):86–87.