

快速高精度的可见面选择*

魏峰^{1,2+}, 王文成¹, 吴恩华^{1,3}

¹(计算机科学重点实验室(中国科学院 软件研究所),北京 100080)

²(中国科学院 研究生院,北京 100049)

³(澳门大学 科学技术学院 电脑与资讯科学系,澳门)

Fast Selecting Visible Surfaces in High Precision

WEI Feng^{1,2+}, WANG Wen-Cheng¹, WU En-Hua^{1,3}

¹(Key Laboratory of Computer Science (Institute of Software, The Chinese Academy of Sciences), Beijing 100080, China)

²(Graduate School, The Chinese Academy of Sciences, Beijing 100049, China)

³(Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, Macao, China)

+ Corresponding author: Phn: +86-10-62562028, Fax: +86-10-62522796, E-mail: weif@ios.ac.cn, <http://www.ios.ac.cn>

Wei F, Wang WC, Wu EH. Fast selecting visible surfaces in high precision. *Journal of Software*, 2006,17(10): 2199–2210. <http://www.jos.org.cn/1000-9825/17/2199.htm>

Abstract: It is very important to efficiently select visible surfaces in rendering complex scenes. As for this, a method is presented in this paper by combining the strategies for image-based selection and object-based selection. First, it classifies the surfaces by their normals, and manages every class of surfaces in hierarchical indexing structures by their positions in 3D space. Then, the visible surfaces are pixel-driven searched from near to far. As the index structures can manage the surfaces orderly, this search can be calculated quickly and all visible surfaces can be obtained. Meanwhile, the selected visible surfaces can be fused automatically to act as the occluders to implicitly cull the occluded surfaces in large quantities. Compared with existing methods for selecting visible surfaces, the new method is fast and in high precision, and can conveniently handle the scenes with dynamic objects.

Key words: visibility; high precision; fast; complex scenes; dynamic objects

摘要: 高效地选取可见面,对于复杂场景的快速绘制是非常重要的.提出一种可见面的选择方法,能将基于图像空间或基于物体空间的可见面选择进行高效的结合,实现可见面的快速高精度的选取.首先,它对场景中的面片进行基于法向的分类,并根据面片的空间位置,为每一类面片分别建立一种层次形式的索引结构进行管理;然后在绘制过程中,基于像素驱动对可见面片进行由近及远的选取.由于索引结构对面片进行了有序的管理,这种选取计算很快,并能将可见面都选取出来,而已绘制的可见面的集合又能自动地成为遮挡者,隐空地剔除了大量的不可见面.与目前国际上关于可见性处理的方法相比,新方法对可见面的选取具有很高的精度,速度很快,并且

* Supported by the National Natural Science Foundation of China under Grant No.60373051 (国家自然科学基金); the National Grand Fundamental Research 973 Program of China under Grant No.2002CB312102 (国家重点基础研究规划(973)); the Research Grant of University of Macau (澳门大学资助项目)

Received 2004-07-26; Accepted 2006-03-31

能够方便地处理带有动态物体的场景.

关键词: 可见性;高精度;快速;复杂场景;动态物体

中图法分类号: TP391 文献标识码: A

随着虚拟现实技术及其应用的发展,所处理的场景越来越复杂.虽然图形生成已经得到了硬件上的很大支持^[1-3],但仍不能很好地满足大规模复杂场景实时绘制的要求.因此,还需要研究各种加速方法.其中,提高可见面选取的效率,是一种非常重要的加速技术,因为这可以很好地降低场景绘制的复杂度.

国际上关于可见性处理的研究很多,文献[4,5]对此有较好的综述.总的来说,一个面片是可见的,它必须满足下列条件:它在成像的视域内,它的法向不是背离视点的,而且它没有被离视点更近的面片所遮挡.目前的可见性处理方法大致可以分为两类:一类是对场景中的面片进行一定的组织,为各个局部子空间建立可见面集合,以便每次成像时可以处理较少的面片;另一类是利用成像过程中的遮挡关系,很快地排除不可见面片.在第1类方法中,文献[6]对场景空间加上视线方位的5维空间进行局部子空间的可见面分类表达;文献[7]等提出基于门窗为建筑物中的各个子空间建立可见面集合;文献[8,9]提出的方法为场景中的各个子空间建立很紧凑的可见面集合.这类方法在每次成像时只需处理各个局部的可见面集合以加速成像,但其预处理一般比较复杂、费时,其各个可见面集合中的面片一般都多于实际成像所需的面片,并且保留这些可见面集合需要很大的空间.在第2类方法中,文献[10]等提出了一种基于包围盒测试和法向检测的可见性处理方法;文献[4]先从场景中抽取一些面片构成遮挡集,然后在成像时先处理遮挡集中的面片,并在成像面上生成可见性度量图,由此来快速处理后续面片的可见性;随后,文献[11]又提出用树结构来高效处理遮挡集;文献[12,13]提出一种基于可见性误差估计的层次化迭加处理的方法;文献[14]提出一种基于面片法向编码的方法,文献[15]提出了对可见性进行编码的方法.这类方法大都要对场景空间和成像面进行多分辨率的层次组织,以提高不可见面片的排除效率,比如文献[16]提出将 z-buffer 技术与四叉树、八叉树技术相结合的方法.文献[17]中的方法利用了多边形面片的 LOD (level of details)组织并进行动态控制来进行遮挡剔除.同时,该方法还利用漫游时相邻帧的连贯性(temporal coherence),以前一帧绘制的物体作为遮挡体来提高遮挡剔除的效率.一般地,它们在成像时要考察较多的面片,并在追求成像速度时舍弃一些可见面片.而且总的来说,虽然以上的方法对静态场景中的可见面选取都能进行比较好的处理,但是处理动态场景时效果并不好.

动态场景的可见性计算,目前依然是一个困难的问题,因为物体的运动会引起场景中不同区域的可见性状态变化,对运动场景不便进行预处理,这使得动态场景的可见性计算开销很大.目前,这方面的工作大都采用基于图像处理的方法.文献[18]提出一种方法,以快速地处理物体分布比较密集的动态场景.它将场景空间均匀地剖分成一些网格,并根据网格中当时所包含的物体情况进行遮挡剔除效率的计算,以非透明度值来表示.在进行可见性计算时,它借用了文献[8]的遮挡融合的思路,并利用连续帧的相关性选择一些遮挡体、利用剖分网格的规整性加快光线的行走等.由于在物体密集的场景中,可见性计算的深度一般不大,因此,该方法在处理物体密集的运动场景时有较快的速度.

本文中,我们提出一种新的可见面选取方法,一方面对场景的面片在三维空间的位置进行有序的组织,以提高剔除不可见面的效率,同时加速可见面的搜索速度;另一方面,在绘制时,基于成像面上的像素进行可见面的选取,并使已绘制的可见面自动成为遮挡者以隐含地剔除大量被遮挡的面.其具体步骤如下:首先根据法向对面片进行分类,并对每类面片依其空间分布构造层次状的索引结构;然后,在成像过程中,用屏幕管理技术对成像面进行监控,根据成像面上像素的可见性由前往后顺序地操作面片的索引结构去选择可见面.这样,既能避免对大量不可见面片的处理,又能快速地选取可见面进行绘制.新方法可以处理各种复杂场景,并且它对面片进行分类及索引构造的预处理也很简便.同时,该方法也能高效地处理动态场景.本文的主要贡献有:

- 新方法根据法向对模型中的面片进行分类,并对每类面片进行基于空间剖分的有序索引化组织.这样,可以将背向面成群地剔除,比常用的一个一个背向面的剔除具有更高的剔除效率;
- 新方法根据由近及远的方式搜索可见面,并只对那些还没有被其相关的可见面所覆盖的像素进行操作

作,这样,已绘制的可见面就自动地成为遮挡者,使得大量被遮挡的面片被隐含地剔除了,从而可以节省大量时间;

- 由于索引结构对面片的空间分布能够进行很好的有序组织,新方法在搜索可见面时能够利用该有序的组织达到很高的速度;
- 新方法能够方便地处理各种动态场景,并且绘制时不需要大的算法改动.

本文第 1 节介绍新方法的主要思路.在第 2 节和第 3 节分别讨论面片的分类及其索引构造的预处理和基于像素的可见性选取可见面的方法.第 4 节讨论新方法对动态场景的处理.第 5 节进行实验结果的分析讨论.第 6 节对新方法进行总结和讨论.

1 主要思路

新方法由预处理和成像时选取可见面两个部分组成.预处理时,新方法首先根据面片的法向进行分类,使法向相近的面片属于一类,以便成像时更多、更快地排除法向背离视点的不可见面片;然后,对每一类面片分别形成垂直于 X 轴、 Y 轴和 Z 轴的 3 组层次状索引结构,每一组这样的结构都是根据面片的空间位置形成它们基于坐标的有序排列.由此,选取可见面时则能利用二分查找的技术提高搜索速度.

成像时,根据视点相对于场景的方位,选取一组索引结构.然后根据成像面上的可见像素对此结构中的面片进行由近及远的处理来选取可见面(可见像素,就是那些还没有被真正离它最近的面片所覆盖的像素,即在绘制工作完成之前,像素的深度信息还可能发生变化的像素).当然,根据可见像素上的成像视线选取可见面,不会处理那些法向背离视点的面片.

随着绘制过程的继续进行,可见像素越来越少,但它们的分布却不一定规则.为此,我们引用动态屏幕管理技术来管理这些可见像素,以使成像过程中选取可见面的操作能够高效地进行.由于面片是基于空间剖分进行索引化组织的,新方法可以很方便地将可见像素与其相应的空间区域快速关联.由此,在根据可见像素选取可见面时,那些与可见像素没有关联的区域中的面片就被隐含地剔除了.

2 面片的分类及索引组织

一个面片是否可见是由它的法向和空间位置决定的.因此,先根据法向对面片进行分类,并将同一类的面片根据其空间位置进行集束化的存储,由此,可以在成像时根据法向的情况成群地避免处理那些法向背离视点的不可见面.这样处理借鉴了文献[14]的基于法向编码的可见性处理方法,但将该方法中逐个面片的检测操作转化为一束面片的检测,以提高排除不可见面的效率.在此,根据视点指向各个可见像素区域的各个单束光线,对各自相关的背向面进行检测.具体的检测计算采用文献[14]中的方法.

由于面片的空间位置也是决定面片是否可见的重要因素,因此,面片的集束化组织是根据其空间分布进行索引方式来组织的.具体地说,我们为每一类的面片分别建立 3 组索引结构,而每一组是相对于 X 轴、 Y 轴、 Z 轴分别建立的.在这种索引结构中,场景中的面片被一种有序的结构管理了,能够便于采用高效的查找技术来检索面片,比如二分查找算法.

2.1 基于法向的面片分类

如图 1 所示,这是一个半径为 1 的方向球.因为三维空间中的任意一个方向都可以由从球心到球面的一个点的有向来表示,即球面上的任一点都表示一个方向,由此,每一个面片的法向都可以映射到球面上的一个点.根据球面上这些点的分布,可以分成一些类,每一类的点在球面上是相互间比较靠近的.如图 2 所示,即是由聚类的方法产生的分类结果:黑点表示对应面片法向的点,圆圈表示分类结果.

从绘制的效率考虑,分类越细致,则在绘制时越能更早期地把更多的不可见面片排除掉.当然,这里也有一个效率平衡的问题.也即分类太多,对它们的组织和管理就需要更多的资源和时间,因而阻碍了绘制效率的提高.

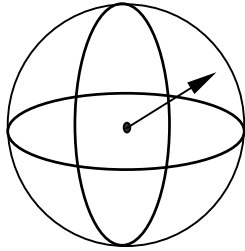


Fig.1 The vector sphere
图1 方向球

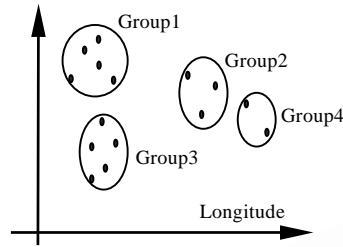


Fig.2 Normal vectors are classified by their distribution
图2 按法向分布的聚集性分类

2.2 面片的索引构造

为了能将物体的三维空间与图像空间快速地关联,新方法对每一类面片分别构造相对于 X 轴、Y 轴和 Z 轴的 3 组索引结构.在此,我们只讨论相对于 Z 轴的索引结构,而相对于 X 轴和 Y 轴的结构可以类似地建立.

首先,将场景的包围盒划分成一些与各个坐标轴都垂直的均匀网格,并在每个网格中记录哪些面片全部或部分地在此网格中;然后,基于这些网格建立索引结构.不失一般性,将这些网格建立如下的层次结构:所有网格都按照垂直于 Z 轴的方式一个切片一个切片地排列;每个切片中的网格,都按照垂直于 Y 轴的方式一条一条地排列;而在每一条中的网格,按照其 X 坐标顺序地排列.

一个类的面片,在有些网格中存在,而在有些网格中不存在.它们所存在的网格,称为实网格;否则,称为虚网格.显然,虚网格对成像是没有贡献的.于是,我们只对实网格进行组织.先考察哪些切片中有实网格,并把这些切片相关的 Z 坐标值记录并建立索引;对于每一个有实网格的切片,考察哪些条中有实网格,并把这些条相关的 Y 坐标值记录并建立索引;最后,对每一条中的实网格,为它们建立索引并按照 X 坐标顺序排列.图 3 是这种索引结构的一个示意图.

由于索引结构反映了面片的空间排列顺序,因此,在检索面片时,可以采用高效的查找算法(如二分查找算法)来操作各个索引序列.为了进一步减少二分查找算法处理的对象数目,我们先用一些垂直于 X 轴的面将包围盒空间分割成一些块,然后再在这些块中分别建立如图 3 所示的索引结构.如图 4 中二维示意图所示,包围盒空间被分割成 4 块,然后在每块中分别建立索引结构.当然,若一个块中没有实网格,它将不会有索引结构,如图 4 中的块 2.每个块记录了它的最大、最小 X 坐标和相关的索引结构.因此,整个场景中的面片可以认为是按照以下的 3 层结构组织的:类→块→层次索引结构.

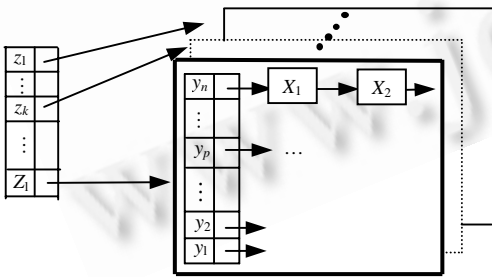


Fig.3 The hierarchical index structure corresponding to the Z axis

图3 相对于 Z 轴建立的索引结构示意图

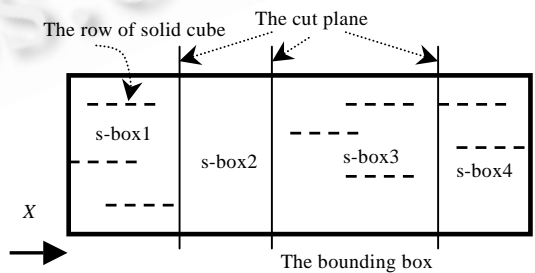


Fig.4 The bounding box is divided by the cut planes

图4 包围盒空间被垂直于 X 轴的面分割成一些块

3 基于可见像素选择可见面

根据视点相对于场景包围盒的方位,选择某组索引结构进行可见性面片的选取.如图 5 中的二维示意图所示,如果视点位于 1 区和 2 区,则选切片垂直于 Y 轴的索引结构;如果视点位于 3 区和 4 区,则选切片垂直于 X 轴

的索引结构;视点位于 5,6,7,8 区,则可任选一组索引结构.对于三维的情况可以类推.选择不同的索引结构的目的是,为了成像时便于由近及远地处理面片,以便更好地利用前面面片对后面面片的遮挡性,避免对不可见面片的处理.

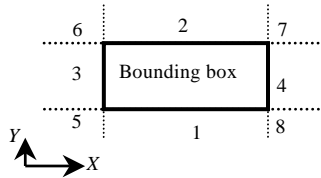


Fig.5 A suitable set of the index structure is selected by the location of the viewer

图 5 根据视点相对于场景包围盒的方位选择合适的索引结构

在选择了索引结构后,我们就可以根据成像面上的可见像素来选择可见面.不失一般性,设所选择的索引结构的切片是垂直于 Z 轴的,而视点的 Z 坐标是小于这些切片的 Z 坐标的.由此,对这些切片进行由近及远的逐个处理,并在处理每一个切片时,根据成像面的可见像素在此切片的实网格中挑选出可见面片.当可见像素被真正离它最近的面片填充后,它就变成了填充像素,不再参与选择可见面的操作.

下面,我们分别介绍成像面上可见像素的管理方法和基于可见像素选择可见面的方法.

3.1 可见像素的管理

开始成像时,成像面上的像素都是可见像素.随着面片的绘制,有些像素变成了可填充的.但在选择可见面时,只需基于可见像素处理,因为只有它们还需要继续寻找相关的可见面片.由于可见像素的分布具有随机性,我们运用动态屏幕管理技术^[19]与成像面的分区化组织相结合的方法进行管理,其操作如下:

首先,将成像面均匀地划分成许多长方形区域;然后,把每个区域当作一个点,用动态屏幕管理技术进行管理.这样,在判断一个区域是否还有可见像素时,可以方便地采用一些系统命令进行快速的计算,具体操作见下一节;而当一个区域中没有可见像素时,动态屏幕管理技术将剔除对它的管理,节省许多关于是否有可见像素的检测.

当然,成像面较大时,为提高可见像素的管理效率,上述的管理方式也可迭代地进行,即:对每个划分的区域再进行进一步的均匀划分形成一些子区域,而这些子区域也用动态屏幕管理技术进行管理;对这些子区域,也进行类似的迭代操作,直至满足需求为止.

随着绘制过程的继续进行,可见像素越来越少.当成像面上没有可见像素,或最后一个切片上的实网格都被处理的时候,这就标志着绘制所需的可见面片都被选出了.

3.2 由可见像素选择可见面

根据可见像素选择可见面,先根据可见像素相关的视线方向决定哪些类的面片可能与此视线相交,然后再考察这些类的索引结构,以便基于面片的有序组织快速地找到可见面.

3.2.1 根据可见像素的视线方向选择相关类的面

一个面片如果与一条视线相交,那么,该面片的法向与视线方向向量的点积一定是负数.在方向球上的反映如图 6 所示,vector 表示视线的方向矢量,则与它能够相交的面片的法向在球面上的对应点一定是在图中的阴影区域,即与 vector 相反的半球面上(此半球面由过球心并与 vector 垂直的平面分割).此阴影区域不包括半球最大圆上的方向,因为它们与 vector 是垂直的.

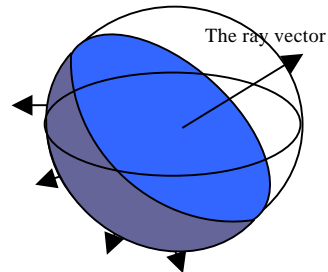


Fig.6 Normal intersected with the ray vector

图 6 与视线方向相交的面的法向一定在阴影区

在透视投影成像时,视域是以视点为顶点的锥形区域,视线的矢量分布在方向球上的一个连续的区域.那么,

与该区域中任一方向的视线有可能相交的面片的法向在方向球上的分布很容易得到.据此,那些由于法向的原因而与这些方向的视线不可能相交的面片将不会被处理.为便于程序实现及提高法向选择的效率,我们将单位球的球面按照经纬线分成一些较均匀的区域,并为每个区域记录下面的内容:若视线矢量在该区域中,则依据法向判断可能与这些视线相交的面片类.

3.2.2 根据索引结构选择可见面

上节中,在根据法向选择选出来可能与视线相交的面片类中,有些面片由于空间位置的关系也不会与任何

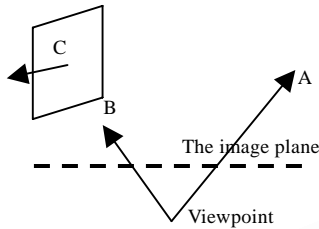


Fig.7 Visibility with position

图 7 空间位置对可见性的影响

视线相交.如图 7 所示,根据法向选择,法向为 C 的面片可能与视线 A 相交,但由于空间位置的关系,它不会与视线 A 相交;另一方面,根据法向选择,这一面片也不可能与实现 B 相交.为此,我们将上节选出来的各类面片的索引结构中的块分别往成像面上投影,根据投影区域的像素的相关视线矢量可以判断出各个块中的面片有无可能与这些视线相交.由此,可进一步选出那些有效块,即该块中的面片很可能与某些视线相交.

对这些有效块在成像面上的覆盖区域,我们也进行记录,并按照像素行的方式进行管理,即为覆盖区域中的每条像素行记录此行像素中

哪些像素被该块所覆盖.这样,在根据可见像素去挑选可见面时,可以很快地知道在哪些块中寻找.

为每一个有效块分配一个切片层次深度记录器,以记录该块中的面片已处理到哪一个切片层次了.在根据一个可见像素选取可见面时,对于每一个覆盖该像素的块,考察其切片层次深度记录器,以选择离成像面最近的切片层次进行处理.当一个切片层次上的可见面处理完以后,它所在块的记录器就移到下一个切片层次.如此循环操作,直至绘制完成.

当根据可见像素的视线在一个切片层次中寻找可见面时,先求该视线在此切片中的线段.根据此线段的 X, Y 轴坐标值范围,在此切片的索引结构中查找有无实网格与此线段相交(这种查找,可以运用二分查找算法提高检索速度).如图 8 所示,根据视域范围内的可见像素,确定可见视域为 AVB 和 CVD,而当前层次中在此视域中的实网格只有 1,3,4,因此,只需在这 3 个实网格检测是否有可见面即可.由于一个像素可能被不同类的块所覆盖,相同坐标的实网格可能在多个块的索引结构中出现.此时,要把这些实网格中的面片放到一起进行考察.

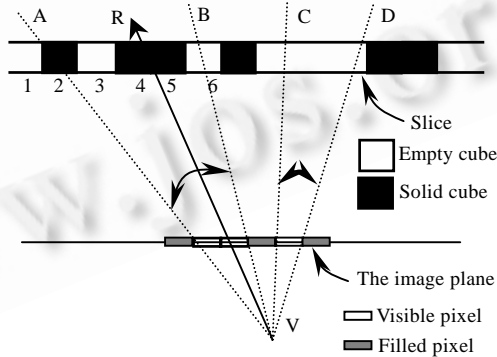


Fig.8 Check whether the slice have visible polygons for visible pixels

图 8 在切片层次中选择可见面

可见,像素在一个实网格中寻找可见面的操作如下:先将此网格中记录的面片进行考察,排除那些法向与该像素视线的矢量方向的点积不为负数的面片,即那些法向背离视点的面片;然后,将其余面片投影到成像面上,检测是否有面片覆盖了此像素;若有,则离此像素最近的面片为所选择的可见面;否则,就表明此实网格中没有该像素的可见面.

根据成像面上管理像素的方式,为可见像素检索可见面的操作如下:

- (1) 对于动态屏幕管理技术所管理的每一个区域,检索它在当前索引切片中有没有对应的包含可见面片的实网格.为此,我们将该索引切片作为一个位图并且根据投影方向对其进行偏移;然后,使用 GPU 的硬件扩展函数 `GL_ARB_multitexture`,对成像面位图和索引切片的偏移位图进行‘与或’计算,则结果位图对应包含可见面的实网格;
- (2) 运用上面讨论的方法在这些实网格中搜索可见面,并对找到的可见面进行绘制.此时,启动 z-buffer 功能,填充所覆盖的可见像素并修改相关的深度值;
- (3) 对于动态屏幕管理技术所管理的任意一个区域,当它对应的该索引切片的实网格中的面片都被处理后,检测该区域中是否还有可见像素.此时,使用硬件扩展函数 `GL_NV_occlusion_query` 检索该区域中可见像素的个数.如果为 0,则表明该区域中没有可见像素,该区域不再被动态屏幕管理技术所管理;
- (4) 对于成像面上有可见像素的所有区域,当它们在一个索引切片中对应的实网格都处理完后,就转入下一个索引切片进行处理.如此迭代进行,直至所有索引切片都被处理了,或没有可见像素了.

新方法的算法流程如下所示:

根据当前视点的方位选取可见面片类及其更适于逐个切片层次处理的索引结构

根据面片的空间位置选取有效块,即该块中的面片很可能与一些视线相交

计算有效块在成像面上的覆盖区域

对有效块的索引结构中的切片进行由近及远的处理

while ((屏幕上还有可见像素)&&(覆盖可见像素的有效块还有切片层次))

```
{
    确定最近的一个还没有处理的切片为当前切片
    for(本层所有可见像素)
    {
        for(覆盖一个可见像素的各个有效块的当前切片层次)
        {
            if (该切片层次中有实网格与可见像素的视线相交)
            {
                检测该实网格中的面片是否有与视线相交的
                if (有)
                    绘制该面片
            }
        }
    }
    一些可见像素变为了填充像素,将它们从可见像素的管理表中去掉处理下一个切片层次
}
```

4 动态物体的处理

新方法能够很好地处理带有刚性运动物体的场景.首先,预处理时我们为每一个动态物体按其局部坐标系建立层次索引结构;在绘制时,可以先对动态场景和背景的层次索引结构进行合并;然后按照相同的方法绘制所有的面片.但是,合并索引结构的操作需要很大的开销.

为了方便地处理动态物体,并与静态场景进行协调的绘制,我们采用以下的方式:确定视点和成像面以后,首先分别绘制所有的动态物体;在绘制这些物体时,所用的坐标系是每个物体的层次索引结构所在的局部坐标系,不同的是,需要将成像面上像素的深度坐标从物体所在的局部坐标系转换到场景对应的全局坐标系下,这样就可以根据转换后的深度坐标进行比较来选择动态物体中的可见面进行绘制;绘制完动态物体后,再对场景进

行绘制.这时,一些成像面上像素的深度坐标已经改变,而且选择了一些临时的可见面.但是,我们把所有这些像素仍然当作可见像素,以进行对静态场景中可见面的选择.这样,由于像素上深度坐标的存在,动态物体上的可见面能得到保留,而不可见面会被自动地覆盖掉.

如图 9 中二维的例子所示:首先绘制动态物体,然后绘制场景.用这种方法可以很容易地得到视线 VR 所对应的动态物体中的可见面.

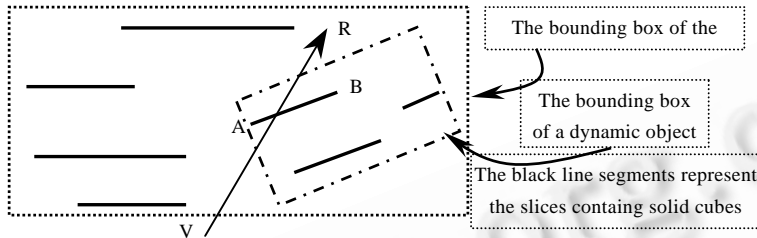


Fig.9 VR can find its visible polygon on the slice AB of the dynamic object

图 9 动态的可见面选择(AB 是视线 VR 所对应的动态物体中的可见面)

5 实验结果及分析

我们用 VC++6.0 在一台 Dell GX270 的微机上实现了新方法.该微机配置了 2.8GHz CPU,2G 内存和 GF FX5950 的图形卡(256 兆显存).其操作系统为 Windows 2000.作为比较,我们在同样的环境下也实现了文献[10]中的可见性处理算法,因为很多可见性处理方面的工作都将它作为一个比较对象.例如:在影响较大的文献[4]中,可见面处理的速度是文献[10]中的 6~7 倍;而文献[13]中的处理速度是其 4~5 倍.

首先,我们检测新方法选取可见面的精度,即它在找出所有可见面时共处理了多少个面片.由于检测精度与面片分类的粗细及场景空间所分割的网格大小有关,我们检测了面片分类及网格大小对检测精度的影响.实验中,我们对电厂模型在各个方位进行了实验,该模型有 12 748 510 个三角面片.为它们建立索引结构所需的空间见表 1,其中空间开销为预处理建立索引结构文件的开销.在此表中列出的帧速是各种情况下新方法对电厂模型进行漫游的平均成像速度.

Table 1 Time and space cost of the index structure

表 1 索引结构的时间和空间开销

	a: 50×50×50 cubes		b: 100×100×100 cubes		c: 200×200×200 cubes	
	Time (s/frame)	Storage (Mb)	Time (s/frame)	Storage (Mb)	Time (s/frame)	Storage (Mb)
A: 32 classes	0.068 7	424.9	0.077 9	864.6	0.094 2	1 220.3
B: 128 classes	0.067 2	748.7	0.076 3	1 055.9	0.108 5	1 425.8
C: 512 classes	0.065 3	913.6	0.079 3	1 272.3	0.119 0	1 854.5

图 10 给出了在各种分类及网格条件下新方法检测面片的数目、文献[10]中方法检测面片的数目和实际可见面的数目(RVP).从表中可知:新方法具有很高的可见面选取精度,当分类和网格剖分比较合理时(如 Ac 情况下),新方法处理的面片绝大部分是实际可见的.

我们对新方法的运行速度也进行了检测.按照设定的视点的几条运行路线,测试了基于新方法 with 文献[10]方法的成像速度.实验结果如图 11 所示:新方法能够很好地提高速度,当选择的网格大小和分类数量比较合理时(比如 Ba 情况下),速度能够提高 8~9 倍.

我们对有动态物体的场景的可见面选择速度也进行了测试.实验中,一个 20 616 个面片生成的飞船根据固定的路线在由 231 820 个面片组成的树林上空运动.在此,我们也对不同分类和网格划分情况下的新方法成像速度进行了检测,并与文献[10]中方法进行了对比.实验结果在图 12 中给出.

由表 1、图 10、图 11 中的实验结果可知:新方法 with 文献[10]中的方法相比,在绘制速度大为提高的同时,检测精度也提高了很多;而且网格越细,检测精度越高.因为网格越细,每个网格中所含的面片越少,因而对面片遮

挡关系的计算就能越精确.当然,网格越细,所需的检测时间也会相应增加.

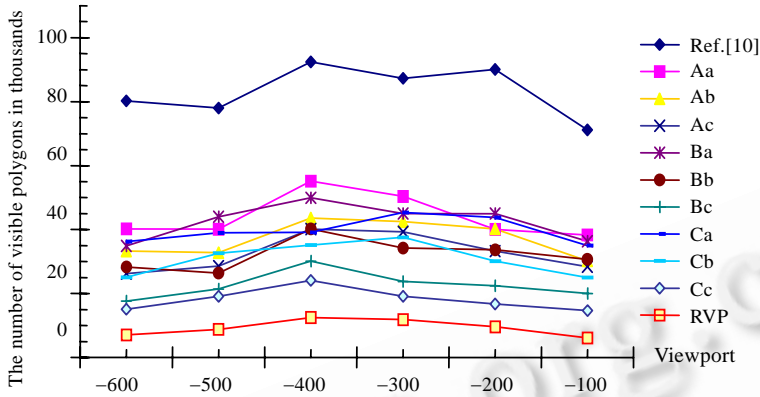


Fig. 10 Number of visible polygons of power plant by the new method, in various classification of cubes

图 10 在各种网格分类情况下,新方法处理电厂模型时的可见面数

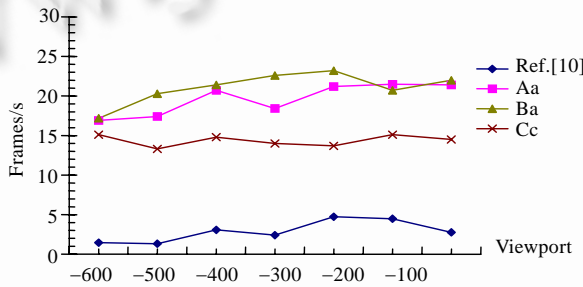


Fig. 11 Rendering speed of power plant by the new method and the method of Ref.[10]

图 11 电厂模型使用参考文献[10]的方法和新技术的绘制速度比较

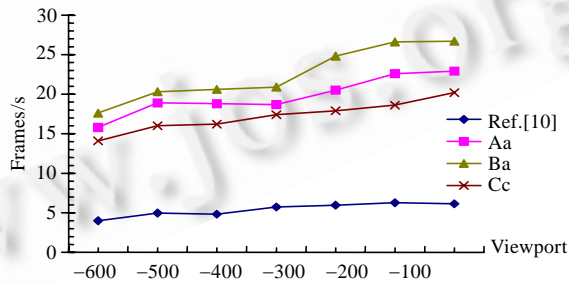


Fig. 12 Rendering speed for the dynamic scene by the new method and the method of Ref.[10]

图 12 文献[10]中方法和新技术处理动态模型的绘制速度的比较

同样,增加法向分类时,可以增加预处理时背向面的选择精度.但是,最终对可见面的选择精度影响并不是很大.因为没有被基于法向选择所淘汰的背向面,在基于像素驱动的选择过程中可以被淘汰.只要每个类中的法向变化幅度不大,就可以保证可见面的判断有很高的精度.在一定程度内,随着法向分类的增多,在像素驱动的可见面选择过程中,调入的类和块会大为减少.这时,分类越多,绘制速度就越快.但是,如果继续细分,随着分类管理开销的增大,速度会逐渐降低.

当然,网格及面片类的划分与具体模型有关,但各种模型在选择适当的分类和网格时,都能取得很好的可见

面选择精度和绘制速度.

另外,空间块的划分对速度的影响也与具体的模型关系较大.当面片在空间和法线方向分布的聚合性较好(如院落、城市模型等)以及空间划分比较细时,则没有实网格的“空”的区域比较多,绘制速度会相应地加快;但当模型面片在空间和法线方向分布的聚合性较差时(如树林模型),空间块的细分对速度的提高影响不大.

总的来说,新方法与近期国际上先进的可见面选择方法相比,能快速高精度地选取可见面.例如:与文献[20,21]中的方法相比,在处理相同的电厂模型时,新方法的速度与它们的速度相似,但新方法只使用了一个CPU,而它们使用了多台机器进行并行绘制,且这些机器的配置性能与我们所用机器的配置很相近;与文献[17,22]中的方法相比,新方法与这两种方法都能交互地绘制大规模的场景,但文献[17,22]中的方法是部分地牺牲了绘制质量来提高速度,而新方法对绘制质量没有影响;另外,新方法的预处理很简单,选择可见面的操作也较为简捷,具有很好的并行性,这些都有利于该方法用硬件实现;同时,新方法也可以很好地处理各种动态场景,而已有的方法大多不便于处理动态场景,少数几种能处理动态场景的方法也只便于处理一些特殊的情况,如文献[18]中的方法只能较好地处理物体密集的场景(因为此时的可见性计算的深度不大,有时序共性可利用).

图 13 是关于这几个模型的成像例子.

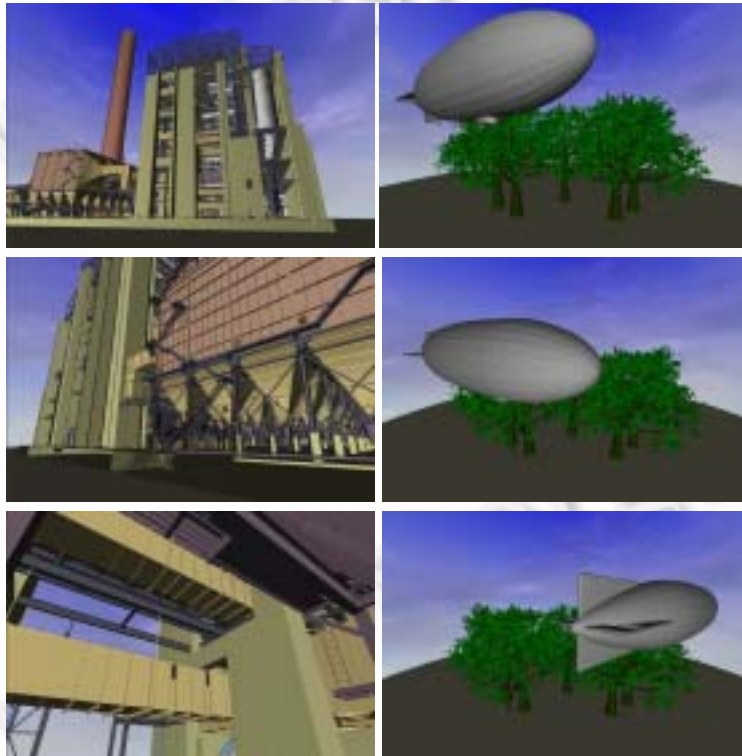


Fig.13 Rendering results of the models

图 13 几种模型的成像结果

6 总结与讨论

本文提出的可见面选取方法对面片进行基于法向的分类,并对每类面片进行基于空间位置的有序的层次化索引构造.这样,各条视线可以方便地将法向背离视点的不可见面片进行成群的剔除,视线在寻找可见面时可以利用高效的检索技术加快前行的步长,以提高检测速度,因为索引构造能够很好地、有序地反映面片的空间位置.此外,成像时,基于可见像素来选择可见面,能够很好地利用已绘制的可见面片隐含地剔除大量被遮挡的面片,节省了大量时间.由于新方法的预处理结果是可以相对于物体的局部空间来计算的,物体运动对预处理的

结果没什么改变,因此新方法能够方便地处理动态场景.总之,新方法对可见面的选取具有很高的精度,并且速度很快,能方便地处理动态场景.

为使新方法更好地发挥作用,对面片基于法向的分类和对场景包围盒空间的网格划分还需要更有效的处理.一般而言,面片的分类越细致,则根据视线矢量选取可见面时可以更多地排除不可见面片,提高检测精度和速度.但是,分类太多,又可能在绘制时要花费更多的开销来处理,并导致绘制效率的降低.而包围盒空间划分的网格越细,则每个实网格中包含的面片就越少,并对实际场景有更好的网格逼近表达,有利于可见面选取效率的提高.但网格越细,所需的存储空间可能更多,且视线前行的步长变短,也会影响可见面检测的速度.以上这些问题都是需要进一步研究的.

References:

- [1] Montrym J, Baum D, Dignam D, Migdal C. Infinite reality: A real-time graphics system. In: Proc. of the SIGGRAPH'97. Los Angeles, 1997. 293-303. <http://www.cs.utexas.edu/users/amenta/gmv/infinite.pdf>
- [2] McCormack J, McNamara R, Gianos C, Seiler L, Jouppi N, Correll K. Neon: A single-chip 3D workstation graphics accelerator. In: Proc. of the Eurographics/SIGGRAPH Workshop on Graphics Hardware. Lisboa, 1998. 123-132. <http://www.cs.virginia.edu/~gfx/Courses/2002/BigData/papers/Case%20Studies/Neon%20-%20A%20Single-Chip%203D%20Workstation%20Graphics%20Accelerator.pdf>
- [3] nVIDIA. GeForce 6 series technical briefs. 2004. http://www.nvidia.com/object/geforce_6_series_tech_briefs.html
- [4] Zhang H, Manocha D, Hudson T, Hoff III KE. Visibility culling using hierarchical occlusion maps. In: Proc. of the SIGGRAPH'97. Los Angeles, 1997. 77-88. <http://citeseer.ist.psu.edu/zhang97visibility.html>
- [5] Bartz D, Meißner M, Huttner T. OpenGL-Assisted occlusion culling of large polygonal models. Computers and Graphics—Special Issue on Visibility—Techniques and Applications, 1999,23(5):667-679.
- [6] Arvo J, Kirk D. Fast ray tracing by ray classification. Computer Graphics, 1987,21(4):55-64.
- [7] Teller ST, Sequin CH. Visibility preprocessing for interactive walkthroughs. Computer Graphics, 1991,25(4):61-69.
- [8] Schaufler G, Dorsey J, Decoret X, Sillion F. Conservative volumetric visibility with occluder fusion. In: Proc. of the SIGGRAPH 2000. New Orleans, 2000. 229-238. http://groups.csail.mit.edu/graphics/pubs/siggraph2000_consvol.pdf
- [9] Durand F, Drettakis G, Thollot T, Puech T. Conservative visibility preprocessing using extended projections. In: Proc. of the SIGGRAPH 2000. New Orleans, 2000. 239-248. <http://people.csail.mit.edu/fredo/sig2000/occluext.pdf>
- [10] Rohlf J, Helman J. IRIS performer: A high performance multiprocessing toolkit for real-time 3D graphics. In: Proc. of the SIGGRAPH'94. Chicago, 1994. 381-394. <http://www.cs.virginia.edu/~gfx/Courses/2002/BigData/papers/Interfaces%20and%20Software%20Systems/IRIS%20Performer.pdf>
- [11] Bittner J, Havran V, Slavík P. Hierarchical visibility culling with occlusion trees. In: Proc. of the Computer Graphics Int'l 1998 (CGI'98). Hannover, 1998. 207-219. <http://doi.ieeecomputersociety.org/10.1109/CGI.1998.69426>
- [12] Klosowski JT, Silva CT. The prioritized-layered projection algorithm for visibility set estimation. IEEE Trans. on Visualization and Computer Graphics, 2000,6(2):108-123.
- [13] Klosowski JT, Silva CT. Efficient conservative visibility culling using the prioritized-layered projection algorithm. IEEE Trans. on Visualization and Computer Graphics, 2001,7(4):365-379.
- [14] Zhang H, Hoff K. Fast backface culling using normal masks. In: Proc. of the ACM Symp. on Interactive 3D Graphics. Providence, 1997. 103-106. <http://www.cs.unc.edu/~zhangh/backface.pdf>
- [15] Meißner M, Bartz D, Gunther R, Straßer W. Visibility driven rasterization. Computer Graphics Forum, 2001,20(4):283-294.
- [16] Greene N, Kass M, Miller G. Hierarchical Z-buffer visibility. In: Proc. of the SIGGRAPH'93. Anaheim, 1993. 231-238. <http://www.cs.princeton.edu/courses/archive/fall02/cs526/papers/greene93.pdf>
- [17] Tsuji T, Zha H, Kurazume R. Interactive rendering with LOD control and occlusion culling based on polygon hierarchies. In: Proc. of the Computer Graphics Int'l 2004. Crete, 2004. 536-539. <http://doi.ieeecomputersociety.org/10.1109/CGI.2004.1309261>
- [18] Batagelo HC, Wu S. Dynamic scene occlusion culling using a regular grid. In: Proc. of the XV Brazilian Symp. on Computer Graphics and Image Processing (SIBGRAPI 2002). Fortaleza-CE, 2002. 43-51. <http://www.dca.fee.unicamp.br/~harlen/batagelo-wu-2002-3dvis-draft.pdf>
- [19] Reynolds RA, Gordon D, Chen LS. A dynamic screen technique for shaded graphics display of slice-represented objects. Computer Vision, Graphics and Image Processing, 1987,38(3):275-298.
- [20] Naga KG, Avneesh S, Yoon SE, Dinesh M. Interactive visibility culling for complex environments using occlusion-switches. In: Proc. of the 2003 Symp. on Interactive 3D Graphics. Monterey, 2003. 103-112. <http://gamma.cs.unc.edu/switch/Paper/i3d03.pdf>

- [21] Naga KG, Brandon L, Yoon SE, Avneesh S, Dinesh M. Interactive shadow generation in complex environments. ACM Trans. on Graphics, 2003,22(3):501-510.
- [22] Gobbetti E, Marton F. Far voxels: A multiresolution framework for interactive rendering of huge complex 3D models on commodity graphics platforms. ACM Trans. on Graphics, 2005,24(3):878-885.



魏峰(1978 -),男,山东济宁人,博士生,主要研究领域为可视化,计算机图形学.



吴恩华(1947 -),男,博士,研究员,博士生导师,CCF 高级会员,主要研究领域为真实感成像,虚拟现实,计算机图形学.



王文成(1967 -),男,博士,研究员,博士生导师,主要研究领域为可视化,虚拟现实,计算机图形学.

《软件学报》Internet 环境下基于构件的软件理论与方法专刊

征 文 通 知

Internet 的出现使计算机软件所面临的环境开始从静态封闭逐步走向动态开放。为了适应这样一种发展趋势,软件系统开始呈现出一种柔性、多目标、连续反应式的新系统形态。从技术的角度,以软件构件等技术支持的软件实体以开放、自主的方式存在于 Internet 的各个节点之上,任何一个软件实体可在开放的环境下通过某种方式加以发布,并以各种协同方式与其它软件实体进行跨网络的互连、互通、协作和联盟,从而形成一种类似于信息 Web 的 Software Web。这样的一种 Software Web 形成之后,它可感知外部网络环境的动态变化,并随着网络环境的变化按照功能指标、性能指标和可靠性指标等进行动态的演化以使系统具有尽可能高的用户满意度。近年来国内外学者面向 Internet 环境下这样一种基于构件的软件新形态,在相关理论与方法方面开展了大量研究工作。《软件学报》2006 年 11 期将推出“Internet 环境下基于构件的软件理论与方法”专刊,总结国内外相关领域取得的研究成果,展望进一步的发展方向,探讨 Internet 环境下基于构件的软件理论与方法在信息产业中的应用前景。

专刊题目: Internet 环境下基于构件的软件理论与方法

特约编辑: 何积丰(华东师范大学) 李宣东(南京大学)

一、征文范围

关注在 Internet 动态开放环境下采用构件形式、自底向上构造软件的相关理论、方法与技术,拟收录如下研究方向的论文:

1. 基于构件的软件数学模型和精华理论。
2. 基于 Agent 的软件模型、方法与安全机制。
3. 面向构件的软件开发方法、质量保障机制。
4. 基于知识的构件需求建模理论与方法。
5. 基于构件的中间件平台模型与框架。
6. 面向领域的相关软件示范工程。

征文范围不仅限于上述内容,所有与 Internet 环境下基于构件的软件理论与方法相关的内容均可投稿。

二、投稿要求

1. 投稿方式: 采用“软件学报在线投稿系统”(http://www.jos.org.cn:8080/jos/jsp/index.jsp)投稿。注意事项: 请在投稿时,在投稿类型中选择“专刊”类型,“软件理论与方法”字样。
2. 稿件格式: 参照《软件学报》论文格式(学报网站上提供了论文模版 http://www.jos.org.cn/download.htm,可下载)。
3. 投稿文章未在正式出版物上发表过,也不在其他刊物或会议的审稿过程中,不存在一稿多投现象;保证投稿文章的合法性(无抄袭、剽窃、侵权等不良行为)。
4. 其他投稿须知请参阅《软件学报》投稿指南 http://www.jos.org.cn/directory.htm
5. 专刊投稿文章不收审理费;通过初审的投稿作者需提交投稿声明;录用刊发文章将收取软件学报标准版面费;发表之后,将按软件学报标准支付稿酬,并赠送样刊及单行本。

三、重要时间

截稿日期: 2007 年 6 月 10 日

录用通知发出时间: 2007 年 8 月 10 日

录用修改稿提交日期: 2007 年 9 月 10 日

出版日期: 2007 年第 11 期