

分布式并行绘制系统中带宽降低方法的研究

高欣¹, 贾庆轩², 孙汉旭², 宋荆洲¹

(1. 北京航空航天大学机械工程及自动化学院, 北京 100083; 2. 北京邮电大学自动化学院, 北京 100876)

摘要: 把静态与视点无关网格简化技术和动态与视点相关的网格简化技术结合起来, 根据应用领域的要求, 采用考虑了离散曲率边折叠算法构造的多分辨率模型, 把得到的简化模型提交给与视点相关的网格简化技术处理, 在基于联网 PC 的分布式并行绘制系统中实现模型的渐进传输并进行实时绘制; 同时采用几何索引等方式来降低立即模式分布式系统的网络传输数据量, 进一步缓解网络带宽压力, 增强数据的实时动态交互性。

关键词: 分布式并行绘制; 多分辨率建模; 离散曲率; 背面剔除; 几何索引

Research on Bandwidth Reduction Method in Distributed Parallel Rendering System

GAO Xin¹, JIA Qing-xuan², SUN Han-xu², SONG Jing-zhou¹

(1. School of Mechanical Engineering and Automation, Beijing University of Aeronautics and Astronautics, Beijing 100083;

2. School of Automation, Beijing University of Posts and Telecommunications, Beijing 100876)

【Abstract】 This paper uses edge collapse method with discrete curvature error metric to construct multi-resolution models, simplified models are provided to view-dependent simplification method to deal with, and implement it with a conventional distributed parallel rendering system; While this paper uses the algorithm called geometry tracking that avoids the redundant geometry transmission by indexing geometry data, alleviating network bandwidth bottleneck.

【Key words】 distributed parallel rendering; multiresolution modeling; discrete curvature; backface culling; geometry index

目前, 在采用了客户端-服务器模式的PC集群分布式并行绘制系统应用中, 由于通过网络的几何指令数与几何模型的规模正相关, 而并行绘制系统处理的一般都是大规模的场景, 网络带宽总是容易成为系统性能的瓶颈^[1], 影响系统的性能和可扩展性。因此, 如何减少数据的传输量、缓解网络带宽压力, 是分布式并行绘制系统研究的一个重要问题。

1 三维模型的多分辨率构建

文献[2]提出了一种计算网格顶点 P 处平均曲率的精确、健壮的离散逼近算法, 根据局部的微分几何学理论, 通过把定义在多边形表面上的一阶和二阶的离散局部近似度量方法和通常的距离误差度量相接合, 定义了一个新的误差度量函数, 和现有的算法相比, 该误差度量方法的优点在于经过一次大的简化过程后, 能够尽量保持原始模型的拓扑结构、边界和尖锐棱边之类的形状特征, 并具有产生的几何误差小、计算速度快等特点, 因此本文采用该方法来构建三维模型的多分辨率表示。该简化算法对网格中的顶点 p 所采用的误差度量函数如下:

$$DDEM(v) = f(v) + f'(v) + f''(v) \quad (1)$$

其中, $f(v) = Q^v(p)$ 是距离误差度量函数; $f'(v) = T^v(p)$ 是切向误差度量函数; $f''(v) = C^v(p)$ 是离散曲率误差度量函数。

在 $DDEM(v)$ 中, 采用点到平面距离的平方作为误差度量的标准, 原始网格的每个顶点 v 的二次误差函数是用和该顶点相邻的平面的加权的二次误差矩阵的和来表示的。

$$Q^v(p) = \sum_{f \ni v} area(f) \cdot Q^f(p) \quad (2)$$

对于每次边折叠 $(v_1, v_2) \rightarrow v$ 所产生的新点 v 来说, 二次距

离误差度量函数为顶点 v_1, v_2 所对应的二次距离误差度量函数之和:

$$Q^v = \sum_{f \ni v_1} area(f) \cdot Q^f + \sum_{f \ni v_2} area(f) \cdot Q^f \quad (3)$$

折叠后产生的新点 v 的切向误差度量函数 $T^v(p)$ 可以表示为

$$T^v(p) = \sqrt{2(\sqrt{1-\vec{n}_1^T \vec{n}_v} + \sqrt{1-\vec{n}_2^T \vec{n}_v})} \quad (4)$$

其中, \vec{n}_v 表示顶点 (v_1, v_2) 折叠后新点 v 的切平面所对应的法向量; \vec{n}_1 和 \vec{n}_2 分别表示顶点 v_1, v_2 的切平面所对应的法向量。

离散曲率误差度量是指折叠的边相邻顶点离散曲率差值的绝对值, 让顶点 $v_i, i=1, \dots, n$ 表示顶点 v 的相邻顶点, 顶点 v 的离散曲率误差度量函数 $C^v(p)$ 被定义为

$$C^v(p) = \sum_{i=1}^n |C_{v_i} - C_v| \quad (5)$$

选择绝对值离散平均曲率 $|H|$ 作为顶点 v_i 离散曲率度量函数 C_{v_i} , C_v 是指顶点 v_i 在一次边折叠后的离散曲率:

$$|H| = \frac{\sum_{i=1}^n \|\vec{e}_i\| \beta_i}{\frac{1}{3}A}$$

其中, A 是围绕顶点 v 的三角形面积之和; \vec{e}_i 是和该点相关

基金项目: 北京市自然科学基金资助项目(3062014); 教育部“新世纪优秀人才支持计划”基金资助项目(NCET-05-0115)

作者简介: 高欣(1974-), 男, 博士研究生, 主研方向: 虚拟现实, 人机交互, CAM; 贾庆轩, 教授; 孙汉旭, 教授、博士生导师; 宋荆洲, 博士研究生

收稿日期: 2006-09-20 **E-mail:** xlh74@sohu.com

的一条边； β_i 是指入射到该条边的相邻两平面向量的夹角。

在简化过程中，不断地执行下面的步骤，直至简化结果达到预设的要求。

(1)通过式(4)计算最初多边形模型的 DDEM，度量所有的顶点的简化误差，并且按照简化误差排序所有的顶点，建立顶点的优先级队列；

(2)对每次边折叠 $(v_1, v_2) \rightarrow v$ 计算新顶点的位置。考虑顶点 v 的离散曲率的误差度量作为该次边折叠的代价；

(3)把所有合法的边折叠放入堆栈，在顶部是折叠代价最小的边；

(4)从堆栈中迭代的去除最小折叠代价的一个边折叠，合并这条边，更新顶点合并后所有涉及到 v_1 和 v_2 的边折叠的代价，更新顶点的优先级队列。

具体的操作过程如下：对当前分辨率的顶点顺序进行顶点对扫描。如果顶点对连续，则判断对应边折叠的合法性。如果合法，执行上述操作，在新点的选择上，将两个顶点合并到它们的中间位置，产生一个新的顶点，为该新顶点赋予顶点对中第 1 个顶点在顶点顺序中的索引值，并根据该索引值修改网格的连通性。

如果顶点对不连续，或者连续的顶点对的边折叠会引起不合理的三角形退化，则不进行边折叠操作，该顶点对的两个顶点均保留它们在顶点顺序中的索引值。所有新产生的顶点和不进行合并保留下来的顶点根据它们的索引值由小到大排序形成下一分辨率的顶点顺序。

2 视点依赖简化及在分布式绘制系统中的实现

为了保证系统的交互速度，本文采用层次细节(level of detail, LOD)储存对象，先根据应用领域的要求，采用考虑了离散曲率边折叠算法简化模型，构造多分辨率模型，在实时绘制时，绘制程序根据一定的标准选择一个适当层次的模型来绘制景物，再把得到的特定分辨率的简化模型提交给与视点相关的网格简化技术处理，在基于联网 PC 的分布式并行绘制系统中实现模型的渐进传输并进行实时绘制，流程图见图 1。

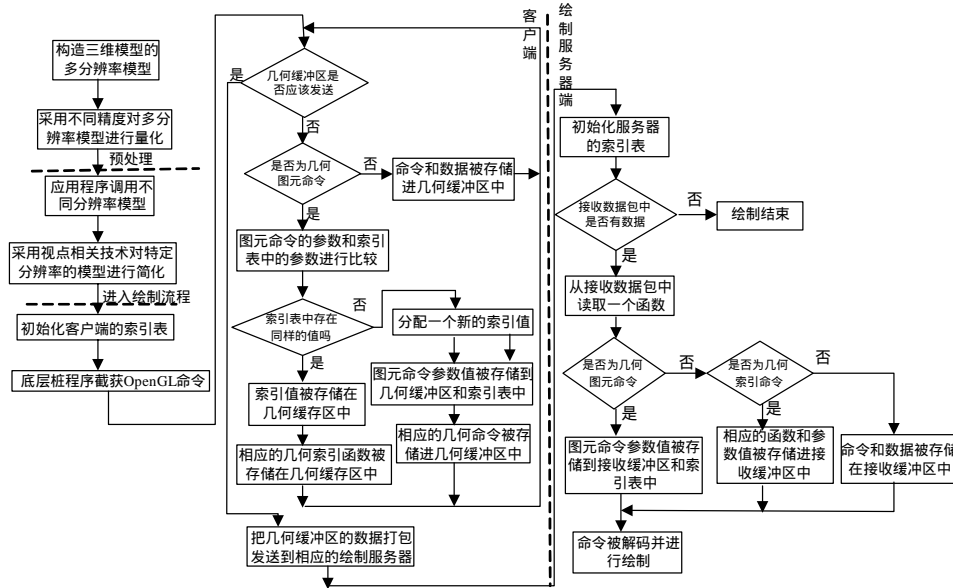


图 1 系统绘制流程

遮挡部分的统计包括后向剔除和前向遮挡。其中，后向剔除可以大量减少待处理数据。因此，对于所获取的基于当前视点的多细节层数据，笔者预先计算了顶点的法向

量，若记顶点 v 的相临三角形集合为 N_v ，那么这里顶点 v 法向量的计算公式为

$$n_v = \frac{1}{M} \sum_{f \in N_v} n_f \quad (6)$$

即取相临三角形的平均法向量。然后计算顶点到视点的向量和法向量的内积，若该值大于 0，就认为顶点背向视点，有待绘制元素集中剔除。

在笔者搭建的环幕演示系统中，采用了类似WireGL^[3]的方法，将整个系统划分为两类节点：客户节点和绘制服务器。客户节点即应用程序节点，它负责截获OpenGL命令，采用双逻辑缓冲区对OpenGL命令和参数进行打包，根据它们的包围盒将打包后OpenGL命令及其参数数据包发送到服务器节点，服务器节点负责OpenGL包的解码和绘制，并且将最终经过几何、颜色校正的图像发送到显示设备进行显示。

3 用几何索引的方法来降低传输的数据量

对于基于联网 PC 的分布式并行绘制系统而言，网络带宽和传输速率对系统的实时性能有着决定性的影响。在目前的网络环境下，如何减少传输数据对提高实时性能十分关键。WireGL 中将命令划分为 3 种类型：图元命令，状态命令，特殊命令。图元命令不影响图形上下文，状态命令必须跟踪并切换图元上下文。特殊命令，如 glClear, glFinish 需要采用广播的方式。

通信的目的在于传递信息，由概率论相关知识，事件的不确定程度，可以用其出现的概率来描述：

$$H = - \sum_{i=1}^n P_i \log P_i \quad (7)$$

基于这种认识，得到：消息中的信息量与消息发生的概率紧密相关，消息出现的概率愈小，则消息中包含的信息量就愈大，编码系统的有效性依赖于信息源的熵，有效的编码能使熵最大化，通过仅仅对改变的信息进行编码并发送到远程节点，客户端使冗余信息的产生量最小化，因此使应用节点的熵最大化，从而降低了发送 OpenGL 命令和参数所需要的网络带宽。

笔者采用了文献[4]提出的几何跟踪算法对每次由客户端发送到绘制服务器端的数据包中所有的几何数据执行索引和跟踪，如果相同自变量的值重复发生，顶点的几何数据不需要被重复传输，仅仅是它的索引值被传输。为了只传输冗余几何命令的索引值，同时也为了能在绘制服务器端检测到对应冗余几何命令，文献[4]定义了下面 3 种新的命令：glIndex_Vertex*，glIndex_Normal*，和 glIndex_TexCoord*。这些索引命令仅仅在 client-servers 内部使用，对于应用程序来讲它们是隐藏的。图 1 显示了几何跟踪算法的流程。

在客户端，当应用程序调用 OpenGL 的库函数时，WireGL 截获应用程序对图形硬件的调用，如果被调用的函数是如下几何命令之一(glVertex*, glNormal*, glTexCoord*)，应用程序

将在索引表中查找相应函数的自变量的值,如果这些值在索引表中已经存在,意味着同样的几何命令以前已经调用过,因此相应的索引值被存储在几何缓冲区中以替换被调用函数的自变量的值,然后相应的索引命令被编码和存储。

如果被调用函数的自变量的值没有存在于索引表中,意味着同样的几何命令以前还没有被调用过,被调用函数自变量的值被直接存储进几何缓冲区中,同时也存储在索引表中并被分配以新的索引值,这个过程反复进行,直到几何缓冲区被发送清空,然后几何数据和索引值被传输到相应的绘制服务器端。

在服务器端,被传送来的数据包中的操作码和数据被依次读出。如果操作码相对应的函数是几何命令之一,这意味着它不是一个冗余的操作码,因此该图元命令的参数值被存储进服务器的索引表中,同时也被直接存储进接收缓冲区中,如果操作码相对应的函数是几何索引命令之一($glIndex *$),使用相应的索引值在绘制服务器端的索引表中查找,然后和这个函数相对应的参数值从索引表中读出被存储在接收缓冲区中,同时原始的操作码也被存储在接收缓冲区中。最后,接收缓冲区中的操作码在相应的服务器上被解码和绘制。

4 试验结果与结论

220°五通道环幕演示系统硬件配置如下:

绘制服务器端 10 台演示机器配置为:CPU P4 2.4GHz,内存 512MB,图形加速卡为 Quadro4 750 XGL,显存 128MB,硬盘容量为 120GB;主控机器配置为:CPU P4 2.4GHz,内存 256MB,显卡为 GeForce3 的普通显卡,显存 64MB,硬盘容量为 80GB;PC 机之间通过 100Mb/s Ethernet 网连接,显示输出设备为 NEC MT1065+投影机 10 台。均分 5 幅画面,每幅画面由两台投影机叠加投射以形成被动立体,环幕演示系统能提供 5120×768 显示分辨率的立体图像。

根据本文所论述的原理和方法,通过使用 MultiGen Creator, VC++, OpenGL, Vtree 等软件在环幕演示系统中建立了北京邮电大学虚拟演示场景,并在已有的单机版自行车仿真健身器系统^[5]基础上,设计并实现了基于联网PC平台的自行车仿真健身器(漫游场景见图 2,实物见图 3)。该自行车漫游系统(VR-BWS)可实现对虚拟场景的交互漫游,帧速率达到 30 帧/s(单个绘制节点显示分辨率 1024×768 , 32 位真彩色),系统的功能和性能指标如下:

(1)当参与者骑在自行车上使用自行车运动时,可实现在虚拟场景中前行、拐弯、后退,可以刹车,画面流畅,人机交互和谐、自然。

(2)逼真的漫游场景画面,视场角 220° ,场景中三维模型的多边形面片数超过了 6 万个,景象有道路、草地、建筑、房屋、蓝天、白云等。



图 2 漫游场景图



图 3 自行车漫游实物图

实验表明,本文论述的方法能降低立即模式分布式系统的网络传输数据量,进一步缓解网络带宽压力,增强数据的实时动态交互性,对于提高三维虚拟场景在分布式并行绘制系统绘制服务器端的绘制速度来讲,是一个有益的尝试。

参考文献

- 1 Staat O G, Walker J, Nuber C, et al. A Survey and Performance Analysis of Software Platforms for Interactive Cluster-based Multi-screen Rendering[C]//Proc. of the 9th Eurographics Workshop on Virtual Environments. 2003.
- 2 Sun-Jeong Kim, Soo-Kyun Kim, Chang-Hun Kim. Discrete Differential Error Metric for Surface Simplification[C]//Proceedings of Pacific Graphics and Applications, Beijing, China. 2002: 276-283.
- 3 Humphreys G, Eldridge M, Buck I, et al. WireGL: A Scalable Graphics System for Clusters[C]//Proc. of Annual Conference Series on Graphics Proceedings. 2001: 129-140.
- 4 Won-Jong Lee, Hyung-Rae Kim, Woo-Chan Park, et al. A New Bandwidth Reduction Method for Distributed Rendering Systems[C]//Proc. of the 1st EurAsia Conference on Information and Communication Technology. 2002: 387-394
- 5 高 欣, 孙汉旭, 贾庆轩, 等. 自行车仿真健身器的设计与实现[J]. 系统仿真学报, 2005, 17(5): 1163-1167.
- 4 Tadepalli C, Ok D. Model-based Average Reward Reinforcement Learning[J]. Artificial Intelligence, 1998, 100(2): 177-224.
- 5 Konda T, Yamaguchi T. LC-learning: Phased Method for Average Reward Reinforcement Learning: Analysis of Optimal Criteria[C]//Proceedings of the 7th Pacific Rim International Conference on Artificial Intelligence. 2002.
- 6 Brafman I, Tennenholtz M. R-MAX: A General Polynomial Time Algorithm for Near-optimal Reinforcement Learning[J]. Journal of Machine Learning Research, 2002, 3: 213-231.
- 7 Stone P. Layered Learning in Multi-agent Systems: A Wining Approach to Robotic Soccer[M]. Cambridge, MA: MIT Press, 2000.

(上接第 19 页)

参考文献

- 1 Mahadevan S. Average Reward Reinforcement Learning: Foundations, Algorithms, and Empirical Results[J]. Machine Learning, 1996, 22(1): 159-196.
- 2 张 芳. 向多移动机器人系统的再励学习方法研究[D]. 上海: 上海交通大学, 2002.
- 3 Schwartz A. A Reinforcement Learning Method for Maximizing Undiscounted Rewards[C]//Proceedings of the 10th International Conference on Machine Learning. Amherst, MA: Morgan Kaufmann, 1993: 298-305.