

# 采用 GPU 加速的三维实体模型绘制

袁友伟

(杭州电子科技大学计算机与软件学院,浙江杭州 310018)

**摘 要:** 利用 GPU 的强大浮点数计算能力和并行处理能力,提出一种完全基于 GPU 的具有真实感三维实体模型快速绘制方法.本文利用现代图形加速卡中 GPU 的可编程管线,实现了快速的网格生成及简化.在保证不改变网格的拓扑结构的前提下,调整网格,使能量方程的数值尽量降低,从而大大降低线性曲面中三角形的数量.实验结果表明,该方法能够实现实时的三维实体模型快速绘制,具有重要的应用价值.

**关键词:** 基于 GPU 的通用计算;基于硬件的绘制;能量方程;网格优化

**中图分类号:** TP391 **文献标识码:** A **文章编号:** 0372-2112 (2008) 12A-144-03

## 3D Solid Models Rendering Based on GPU Acceleration

YUAN You-wei

(School of Computer Science and Software, Hangzhou Dianzi University, Hangzhou, Zhejiang 310018, China)

**Abstract:** This paper proposes a novel method of 3D solid models rendering by taking advantage of the parallelism and programmability of GPU (graphics processing unit). Rapid mesh generation and simplification is implemented by a programmable pipeline of a modern GPU. Then, under the condition that the topology structure of the mesh is not changed, the mesh is modified to decrease the energy functions value. Experiments show that the method is very valuable for the applications.

**Key words:** general purpose computation on graphics processing unit (GPU); hardware-based rendering; energy function; mesh optimization

### 1 引言

在计算机图形学中,三维实体建模是整个图形学的基础,要生成高逼真度的图像,计算机硬件的图形处理能力也常以每单位时间能处理的三角网格数量作为评估标准.复杂的模型常常需要数以百万计的三角形面片来表示,庞大的数据量给存储、传输和显示带来了很大的麻烦,如何加速生成具有真实感的三维实体模型及快速传输,在电影、游戏以及虚拟现实等领域均具有重要的研究价值.

GPU 英文全称 Graphic Processing Unit,中文翻译为“图形处理器”.GPU 可以有效地执行多种运算,从线性代数和信号处理到数值仿真.越来越多的研究者利用 GPU 的并行性和可编程性来求解偏微分方程 (partial differential equations)<sup>[1,2]</sup>和一些实际应用问题<sup>[3~5]</sup>.

文中提出了一种基于 GPU 的具有真实感三维实体模型快速绘制方法<sup>[6,7]</sup>,充分利用了 GPU 的强大浮点数计算能力和并行处理能力以及自适应细分策略,并使用能量最小化算法对网格顶点位置进行优化,针对优化模

型的特征,网格也更加逼近实际曲面,从而获得更具真实感的三维实体模型,且计算效率更高.

### 2 基于 GPU 加速的网格生成及优化系统结构

基于 GPU 加速的网格生成及优化系统结构如图 1 所示,图像处理通道分为四个不同步骤:物理实体、几何转换、三角形设定以及像素渲染. GPU 绘制管道的可编程并行处理能力分布在两部分:顶点处理器 (Vertex Processor) 和像素处理器 (Fragment Processor).

#### 2.1 顶点处理器和像素处理器

图形处理中流水线的第一阶段主要是把顶点由三维的物体坐标转为屏幕坐标.对顶点实施几何变换及属性处理,从 CPU 交由顶点处理器处理.随后通过光栅化的处理得到很多的像素片断从而传入像素处理器并得到最终的图像输出结果.整个计算映射到 GPU 上,从而利用 GPU 的并行性获得充分的加速,使得整个计算和绘制达到实时效果,而通过偏移纹理来获取正确的方位信息,并行方式进行多重操作,从而使三维实体模型得以快速地绘制.

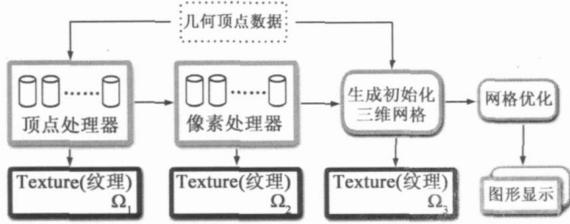


图1 GPU加速的网格生成及优化结构图

### 2.2 纹理映射

通过纹理空间与物体空间之间的坐标变换,可以把纹理映射到物体表面,并可以根据边界条件对边界上的变量进行修正。通过设定相应的剪切面,确定显示包围盒。将整个计算映射到 GPU 上来完成。当经过重建和简化后的模型具有较大三角片、超过一幅纹理图像的覆盖区域时,则无法实现纹理映射,可采用重构模型,即根据纹理图像的边界对模型进行分割,从而使同一三角片的每个顶点投影后都位于同一纹理图像中。避免在像素程序中重新对每一个像素点进行定位计算,提高了整个模拟的运算效率<sup>[8]</sup>。

### 2.3 使用能量函数进行网格优化

三角初始网格模型不可避免地存在各种噪声和扰动,为了提高网格质量,便于后续处理,因而有必要保持网格模型的拓扑信息和几何特征不变前提下,对模型进行优化。

用于三角网格模型光顺的主流算法有两类:能量法和拉普拉斯光顺法。文中使用能量最小化算法对网格顶点位置进行优化,针对优化模型的特征,优先计算网格中的关键点,并把优化模型转化为线性方程组求解<sup>[9]</sup>。

能量函数如下所示:

$$E(M) = E_{\text{dist}}(M) + E_{\text{spring}}(M) + E_{\text{scalar}}(M) + E_{\text{disc}}(M) \quad (1)$$

式中:  $E_{\text{dist}}(M)$  为  $M$  的距离能量;  $E_{\text{spring}}(M)$  为  $M$  的弹性能量;  $E_{\text{scalar}}(M)$  为  $M$  度量  $M$  标量属性精度的能量;  $E_{\text{disc}}$  为  $M$  度量  $M$  上视觉不连续的特征线(如边界线、侧影轮廓线等)几何精度的能量<sup>[10]</sup>。

$E_{\text{dist}}(M)$  定义为点集  $P = \{p_1, p_2, \dots, p_m\}$  到网格  $M$  距离的平方之和,如式(2)所示

$$E_{\text{dist}}(M) = \sum_{i=1}^n d^2(p_i, \Phi_V(|K|)) \quad (2)$$

$$d^2(x_i, \Phi_V(|K|)) = \min_{b_j \in |K|} \|x_i - \Phi_V(|b_j|)\|^2 \quad (3)$$

其中,  $M = \Phi_V(|K|)$ ,  $b_j \in |K| \subset R^m$ ,  $K$  为边的弹性系数。 $v_1, v_2$  为边的两个顶点,曲面片  $P^k$  的控制顶点记作  $\{V_{i,j}^k\}$  中,  $i = 0, n - 1; j = 0, \dots, 2m\}$ 。将中间点记为  $V_0$ 。

使用能量函数进行网格优化,由以下公式得到:

$$E(K, V, B) = \sum_{i=1}^n \|x_i - \Phi_V(|b_i|)\|^2 + \sum_{(j,k) \in K} k \|v_j - v_k\|^2 \quad (4)$$

这样,可以得到使用能量函数优化网格后所有顶点的新位置<sup>[10]</sup>。

### 2.4 三维实体模型显示

要生成高逼真度的图像,首先要生成高质量的三维实体模型。绘制三维网格必须对每个顶点进行三维几何变换,而变换要通过  $4 \times 4$  矩阵乘法来完成,对每个顶点进行计算和操作,然后进行光栅化形成图形片元;对于像素数据,像素操作结果被存储在纹理组装用的内存中,就象几何顶点操作一样光栅化形成图形片元。

为了提高绘制速度,我们对原网格进行简化,生成了一个近似的简化网格,最后对该简化网格进行绘制,越是平滑的部分,越可以用简单的拟合模型,这样到达提高了三维实体模型的显示速度。隐式迭代以获得绝对稳定的格式,从而满足大的时间步长。由于整个计算映射到 GPU 上,从而利用 GPU 的并行性获得充分的加速,使得整个计算和绘制达到实时效果<sup>[11]</sup>。

### 3 实验结果与讨论

本文所采用的实验平台为 Intel Pentium 2.8GHz,主内存为 2G,显卡采用的是 GeForce FX5950 Ultra,显卡内存为 256M,显卡的核心频率为 375MHz,驱动的版本为 54.16,操作系统为 Windows XP。图 2 显示了我们方法的有效性和高效性,在图 2 中,(a)图表示输入的初始网格,顶点数为 9968,三角形个数为 19972;(b)(c)(d)分别表示进行网格优化后的结果。(e)图表示绘制的实体模型

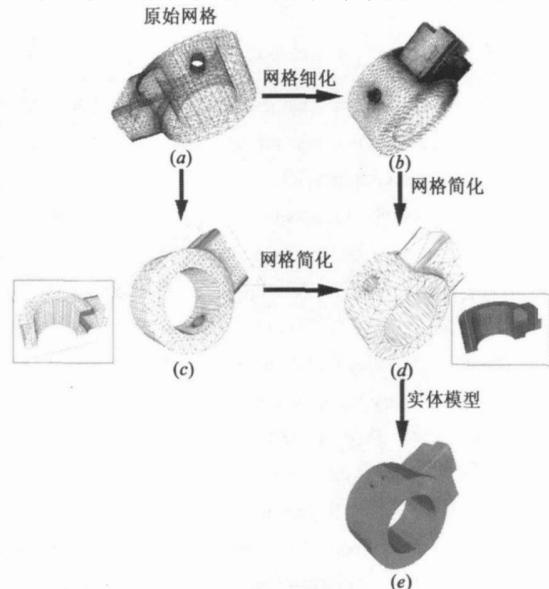


图2 采用本文GPU加速的三维实体模型绘制过程。(a)图表示输入的初始网格;(b)(c)(d)分别表示进行网格优化后的结果;(e)图表示绘制的实体模型

模型。(c)图中的红色矩形表示网格模型的范围轮廓,即(Bounding Box)。(c)图的左下角图形表示(c)图的截面图,(d)图的右下角表示(d)图的截面图,其绿色的阴影表示其范围盒的大小尺寸。从图中可以清楚地看到:越是平滑的部分,越用简单的拟合模型,其三角形的个数越少,从而加快了绘制速度。

表1表示了图2中的各项参数以及生成同样的网格GPU与CPU的时间之比。实验结果表明,GPU的时间仅仅为CPU的大概十五分之一。

表1 本文GPU的方法与基于CPU方法的绘制速度比较

图形	Vertices (顶点数)	Triangles (三角形个数)	Resulting energies		GPU:CPU (sec)
			E <sub>dist</sub> (距离能量)	E(M) (总能量)	
(a)	9968	19972	$6.56 \times 10^{-4}$	$4.44 \times 10^{-3}$	1:14.09
(b)	12771	25546	$9.13 \times 10^{-4}$	$8.69 \times 10^{-3}$	1:16.14
(c)	6768	13540	$3.45 \times 10^{-4}$	$2.17 \times 10^{-3}$	1:13.28
(d)	918	1840	$9.13 \times 10^{-5}$	$9.98 \times 10^{-3}$	1:18.26

#### 4 结论

充分利用了GPU并行能力以加速计算,提出一种新的基于GPU加速的三维实体模型的绘制方法。GPU具有的片元着色器的高速并行计算能力和完全基于GPU的细分框架,在表面绘制精度保持不变的情况下,有效地降低了细分浮点数运算使得GPU可以充分地发挥其具有的高性能。整个计算映射到GPU上,从而利用GPU的并行性获得充分的加速,使得整个计算和绘制达到实时效果。本文算法可以应用在三维动画造型或CAD中的原型设计等领域中。

#### 参考文献:

- [1] Bolz J, Farmer I, Grinspun E, Schröder P. Sparse matrix solvers on the GPU: Conjugate gradients and multigrid [J]. ACM Trans On Graphics, 2003, 22(3): 917 - 924.
- [2] Harris M J, Coombe G, Scheuermann T, Lastra A. Simulation of cloud dynamics on graphics hardware [A]. Proc of the Graphics Hardware [C]. Aire-la-Ville: Eurographics Association, 2003. 92 - 101.
- [3] Goodnight N, Woolley C, Luebke D, Humphreys G. A multigrid solver for boundary value problems using programmable graphics hardware [A]. Proc of the Graphics Hardware [C]. Aire-la-Ville: Eurographics Association, 2003. 102 - 111.
- [4] Krüger J, Westermann R. Linear algebra operators for GPU implementation of numerical algorithms [J]. ACM Trans on Graphics, 2003, 22(3): 908 - 916.

- [5] 刘保权,刘学慧,吴恩华.基于GPU的实时深度图像前向映射绘制算法[J].软件学报,2007,18(6):1531 - 1542.  
Liu Bao-quan, Liu Xue-hui, Wu En-hua. Real-time rendering depth images on GPU by forward warping [J]. Journal of Software, 2007, 18(6): 1531 - 1542. (in Chinese)
- [6] 柳有权,刘学慧,吴恩华.基于GPU带有复杂边界的三维实时流体模拟[J].软件学报,2006,3,17(3):568 - 576.  
Liu You-quan, Liu Xue-hui, Wu En-hua. Real-time 3D fluid simulation on GPU with complex obstacles [J]. Journal of Software, 2006. 3, 17(3): 568 - 576. (in Chinese)
- [7] Y W Yuan, L M Yan, Q P Guo. Efficient surface mesh reconstruction from unorganized points using neural networks [J]. Chinese Journal of electronics, 2005, 14(1): 26 - 29.
- [8] Lefohn A E, Kniss J M, Hansen C D, Whitaker R T. Interactive deformation and visualization of level set surfaces using graphics hardware [A]. Seattle: IEEE Visualization [C]. IEEE Computer Society, 2003. 75 - 82.
- [9] 王世东,张佑生,偶春生,谢颖.基于能量最小化的网格优化算法[J].计算机工程与设计,2007,28(2):267 - 269.  
Wang Shi-dong, Zhang You-sheng, Ou Chun-sheng, Xie Ying. Algorithm for mesh optimization based on energy minimization [J]. Engineering and Design, 2007, 28(2): 267 - 269. (in Chinese)
- [10] Y W Yuan, Q P Guo. A Generic strategy for dynamic load balancing of dynamic parallel distributed mesh generation [A]. DCABES 2006 PROCEEDING [C]. Shanghai: Shanghai University Press, 2006. 199 - 202.
- [11] 杨正龙,金林,李蔚清.基于GPU的图形电磁计算加速算法[J].电子学报,2007,35(6):6501 - 0601.  
Yang Zheng-long, Jin Lin, Li Wei-qing. Accelerated GRECO based on GPU [J]. Acta Electronica Sinic, 2007, 35(6): 6501 - 0601. (in Chinese)

#### 作者简介:



袁友伟 男,1966年生于湖北荆州,杭州电子科技大学博士,教授,研究兴趣为计算机图形图像处理。

E-mail: y.yw@163.com.