

基于 CUDA 架构的改进 Marching Cubes 算法

周筠¹ 蒋富²

(湖南财政经济学院信息技术与管理学院 长沙 410205)¹(中南大学信息科学与工程学院 长沙 410075)²

摘要 Marching Cubes 是医学体数据可视化的经典算法,但生产的网格质量差、算法执行速度慢成为阻碍其用于数值分析的两个主要缺点。文中提出一种基于硬件加速的 Marching Cubes 改进算法。该算法采用统一设备架构(CUDA)充分发挥 Marching Cubes 算法分而治之的优点,利用 CUDA 的可编程性并行分类体数据,加快了活跃体素和活跃边的提取;同时,该改进算法将得到的活跃边按照中点投影方式进行偏移,从而达到了改善网格质量的目的。最后通过实验表明,该算法可以保证在阈值未知的情况下,进行交互式的高质量网格建模。

关键词 医学体数据, Marching Cubes, CUDA, 中点投影

中图分类号 TP391 **文献标识码** A

Improved Marching Cubes Based on CUDA

ZHOU Yun¹ JIANG Fu²

(Department of Information Management, Hunan University of Finance and Economics, Changsha 410205, China)¹

(School of Information Science and Engineering, Central South University, Changsha 410075, China)²

Abstract Marching Cubes (MC) is one of the classical algorithms for medical volume data. But poor mesh quality and slow execution speed have affected the further development such as finite element analysis. In this paper, an improved MC algorithm was presented based on the CUDA. Three kinds of parallel computing were proposed to extract active volumes and edges in the CUDA. Simultaneously, point projection was used in the algorithm to move the endpoints of the active edges and improve the mesh quality. Finally, experimental results show that the presented method can realize the interactive modeling.

Keywords Medical volume data, Marching Cubes, CUDA, Point projection

1 引言

由医学体数据进行等值面的三维重建一直是当今医学可视化领域研究的热点,其中 Lorensen 等提出的 Marching Cubes(MC)算法^[1]为等值面重建的经典代表,该算法是一种分而治之的方法,简单高效是其最大的优点,已广泛应用于医学可视化的各个方面^[2-4]。

由 MC 算法得到的等值面的基本单元为三角形,由于在抽取的过程中不考虑质量问题,算法生成的劣质三角形以及其本身的二义性问题阻碍其进一步发展与应用^[5]。此外,MC 算法的执行速度较慢,大部分时间用于空体素的判断,近几年已有部分基于硬件的改进措施^[6-7],但没有同时考虑网格质量。本文以改善这两个缺点为目的,提出一种 CUDA 架构下的高质量 MC 改进算法。

2 基于 CUDA 架构的高质量 Marching Cubes 算法

2.1 算法流程

MC 算法是一种分而治之的方法,分别处理体数据中的每一个体素,与等值面相交的体素称为活跃体素,与等值面不相交的体素称为空体素。活跃体素中与等值面相交的边称为

活跃边。MC 算法依次连接活跃体素中活跃边与等值面的交点,构成三角面片,所有活跃体素的三角面片合成等值面模型。

本文算法的流程如图 1 所示,该算法将等值面的提取分成 3 个主要部分:GPU 并行处理原体素数据,获得活跃体素以及活跃边;CPU 中引入中点投影方法,计算等值面与活跃边的交点;GPU 以三角形为基本单位,并行计算交点坐标和向量。

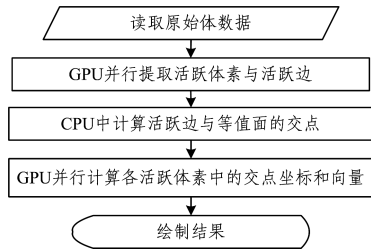


图 1 算法流程图

2.2 体素和活跃边的并行提取

在 CUDA 架构下,核函数的设计是实现 GPU 计算的核心^[9-10]。CUDA 利用线程管理机制为核函数实现并行计算,一系列数量的线程封装成块(block),相同线程维度的线程块组成线程格(grid)。设计核函数时,需要根据待处理的数据

本文受湖南省教育厅科学研究基金项目:虚拟手术仿真系统中的交互式建模技术研究(13C095)资助。

周筠(1984-),女,博士,讲师,主要研究方向为虚拟现实技术、有限元网格生成方法等,E-mail:jzyf0403@163.com(通信作者);蒋富(1983-),男,博士,副教授,主要研究方向为虚拟现实、医学数据可视化等。

量自定义线程块和线程数量,这样的并行模式是 CUDA 最重要的创新点之一。

2.2.1 体素的并行提取

体素的并行提取主要涉及到 3 个核函数,功能分别为体素分类和活跃体素的提取。体素分类采用线程-体素一对一的方式进行计算,假设原始数据中体素总数为 N ,按照每个线程块包含 128 个线程的容量,依据以下代码定义三维线程网格 $grid$ 。

```
//grid的定义
dim3 grid(N/128,1,1)
```

在体素分类的并行处理中,每个线程 T_i 的计算步骤如下:

Step1 T_i 计算第 i 个体素的顶点状态;

Step2 根据 S 得到该体素生产三角形需要的顶点数目 n ,当 $n=0$ 时,表示体素为空体素;

Step3 依据 n 的状态标记活跃体素;

完成上述步骤后,得到两个数组 $Avolume_edge[i]$ 和 $Avolume[i]$,前者用于记录体素状态 S ,后者用于记录活跃体素,以便后续活跃边的提取。

活跃体素的提取部分主要用于计算活跃体素的数目 N_{active} 并进行标记体素标号。该部分的线程分配和体素分类部分一致,每个线程 T_i 的计算步骤如下:

Step1 根据数组 $Avolume[i]$,判断体素 i 是否活跃,若是活跃体素则进行下一步,否则不做任何计算;

Step2 对于活跃体素,依据辅助数组找到该体素的原始数据中的标号,计入数组 $CAvolume[i]$;

完成上述步骤后,得到仅存储活跃体素标号 F_{active} 的数组 $CAvolume[N_{active}]$ 。

2.2.2 活跃边的并行提取

根据已提取的活跃体素,进一步提取其中的活跃边。由于这部分仅针对活跃体素,因此线程数的分配可大大减少,每个线程块仅包含 32 个线程。

```
//grid的定义
dim3 grid(N_active/32,1,1)
```

由于该部分需要处理活跃体素 12 条边的信息,因此应定义相应的数据结构,存放边的端点坐标以及每个体素中 12 条边的坐标信息。

每个线程 T_i 计算第 i 个活跃体素的步骤如下:

Step1 根据前面的计算结果获得活跃体素标号 F_{active} 以及该体素的状态 S ;

Step2 由 F_{active} 计算该体素中 0 号顶点的坐标 P_0 ;

Step3 依据 MC 三角化表,计算体素中活跃边的局部编号 $Eege[12]_0$;

Step4 计算活跃边端点坐标并记录该体素中活跃边的数目 $N_{actedge}$ 。

通过上述 4 个步骤,所有活跃体素的活跃边信息被保存,并用于 CPU 中进行投影计算。

2.3 活跃边的中点投影算法

劣质的三角形主要是由与活跃边平行的等值面形成的,本文引入中点投影方法^[11-12],适当移动活跃边,从而改善网格质量。

中点投影法的基本思想是计算活跃边中点到等值面切平

面的投影。该方法在程序执行中需要分解成两个部分:1)活跃边端点的偏移;2)活跃边的交点计算。

活跃边端点的偏移是投影的核心,由于涉及切平面的计算、端点的移动以及多种物理求值,因此只能在 CPU 内进行,步骤如下。

设待移动的活跃边为 l_{ab} ,两个端点分别记为 A 和 B ,实现这一部分的具体步骤如下:

Step1 计算 l_{ab} 的中点 mid ,获得点 A 和 B , mid 的物理值 $Value(A)$, $Value(B)$, $Value(mid)$,根据物理值,重新定义 l_{ab} 的始末端点 A_{org} , B_{dest} ;

Step2 赋值 $A_{org}=mid$,将末端点 B_{dest} 进行法向偏移,初始化整型变量 $n=0$;

Step3 由式(1)计算点 B_{dest} 的偏移量 $\Delta offset$,式中 L 为体数据的最小间隔, $Moves$ 为移动步数;

$$\Delta offset = \nabla f_{dest} \frac{L/2}{Moves} \quad (1)$$

Step4 计算偏移后的新位置 $B'_{dest} = B_{dest} + \Delta offset$;

Step5 判断 B'_{dest} 点的状态,若与 B_{dest} 相同,则执行下一步,否则跳出循环;

Step6 $n=n+1$,若 $n < Moves$, $B'_{dest} = B_{dest}$,返回 Step2,否则执行下一步;

Step7 $B'_{dest} = B_{dest}$,将末端点 B'_{dest} 进行切向偏移,过程与 Step2-Step5 一样,只是偏移向量采用切向量,计算方式如式(2)所示:

$$\nabla \vec{l}_{dest} = \nabla f_{dest} \times \vec{l} \times \nabla f_{dest} \quad (2)$$

完成上述步骤后,得到经过偏移的活跃边信息,只需在 CUDA 架构下重新分布线程,完成交点的并行计算即可得到等值面模型。

3 算法实例和分析

本节采用显存为 1GB 的 NVidia Quadro NVS290,结合 Visual C++ 和 OpenGL 为实验平台,以氢(128 * 128 * 128)为例进行实验验证和分析。

图 2(a)为采用 CUDA 架构下传统的 MC 算法的实验结果,图 2(b)为在 CUDA 架构中引入点偏移策略的实验结果。

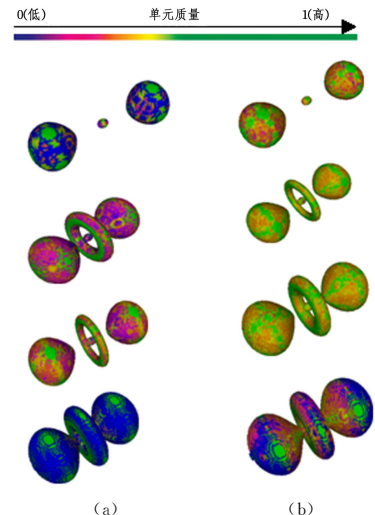


图 2 算法实验图

从图 2 中可以看出,由于采用点偏移策略,图 2(b)得到的网格质量比传统的 MC 算法好。

表 1 列出了与图 2 对应的两种算法的执行时间,c 行代

表 CPU 中的点偏移 MC 改进算法。从表 1 可以看出,所有在 CUDA 架构下的并行计算都极大地提高了执行速度,基本可以实现实时改变阈值的交互式建模。

表 1 实验数据表

算法	阈值			
	$\beta_0=45$	$\beta_1=31$	$\beta_2=25$	$\beta_4=8$
a	13	13	15	16
b	219	560	900	980
c	3195	4245	5054	6776

(单位:ms)

结束语 本文结合 CUDA 架构的可编程性和 Marching Cubes 算法分而治之的特点,合理设计核函数,并行处理 MC 算法中活跃体素以及活跃边的提取,加快执行速度。同时,引入中点投影的方式,改善活跃边与等值面平行的状态,从而达到提高网格质量的目的,研究表明,本文算法的实验效果较好。

但该算法有两点不足之处,首先是投影方法的迭代次数由人为确定,并且只能在 CPU 中进行,算法的大部分时间都集中在这个部分;其次由于硬件条件,该算法无法直接读入数据量稍大的体数据。针对以上两点,需要后续进行进一步的研究。

参 考 文 献

[1] LORENSEN W,CLINE H. Marching Cubes:A High Resolution 3DSurface Construction Algorithm [J]. Computer Graphics (S0097-8930),1987,21(4):163-169.
 [2] 朱恺. 基于改进 MC 算法的脑图谱三维可视化应用研究 [D]. 太原:太原理工大学,2015.
 [3] CHANG M,WOONG O J,CHANG D S,et al. Interactive marching cubes algorithm for intraoral scanners [J]. The International Journal of Advanced Manufacturing Technology,2017,89(5):2053-2062.

[4] 周筠,樊晓平,蒋富. 医学仿真中一种高效的生物组织几何建模方法 [J]. 系统仿真学报,2012,24(1):6-10.
 [5] SUN L N,TIAN H Q,WU D M,et al. Three-Dimensional Geometric Modeling of the SpineBased on Reverse Engineering Technology [C]//3rd International Conference on Biomedical Engineering and Informatics. 2010:1292-1295.
 [6] 王明,冯洁青,杨贲. 移动立方体算法与移动四面体算法的对比与评估 [J]. 计算机辅助设计与图形学学报,2014,26(12):2009-2106.
 [7] CIZNICKI M,KIERZYNKA M,KUROWSKI K,et al. Efficient Isosurface Extraction Using MarchingTetrahedra and Histogram Pyramidson Multiple GPUs [C]// International Conference on Parallel Processing and Applied Mathematics. 2011:343-352.
 [8] RECK F,DACHSBACHER C,GROSSO R,et al. Realtime isosurface extraction with graphics hard-ware [R]. Eurographics Short Presentations,2004.
 [9] 汤颖,嵇海锋,盛风帆,等. 大规模森林多精度生长仿真模型及其计算加速算法[J]. 小型微型计算机系统,2016,37(5):1033-1038.
 [10] HAN S Q,LEI Z,SHEN W F,et al. An Approach to Improving the Performance of CUDA in Virtual Environment [C]//IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD). 2016:585-590.
 [11] DIETRICH C A,SCHEIDEGGER C E,SCHREINER J,et al. Edge transformations for improving mesh quality of marching cubes [J]. IEEE Transactions on Visualization and Computer Graphics,2009,15(1):150-159.
 [12] DIETRICH C A,SCHEIDEGGER C E,COMBA J L D. Edge groups:an approach to understanding the mesh quality of marching methods[J]. IEEE Transactions on Visualization and Computer Graphics,2008,14(6):1651-1666.

(上接第 541 页)

价误差)两种指标的计算,结果取两位有效数字,如表 2 所列。

表 2 各模型的预测误差对比结果

	原始序列	分段处理	镜像处理
MSE	14.35	4.38	7.68
MAE	45.63	9.52	25.86

从结果中可以看出,分段处理和镜像处理都比原始序列直接预测的结果更为精确,说明两种处理方法都能够有效地减小误差,提高预测的精确度。其中分段预测的结果最为准确,镜像处理后使得非线性误差变大,因此预测效果不如分段处理。

结束语 本文针对舆情预测中拐点存在的造成预测不准确的问题,首先判断了舆情拐点存在的位置,提出了基于 ARIMA 和灰色预测的分段和镜像处理模型,对分段处理和镜像处理分别进行了研究和讨论,最后用实例验证了分段和镜像处理能够在一定程度上提高预测的准确率。今后在舆情预测上可以考虑优化非线性部分,减小非线性误差。

参 考 文 献

[1] 周耀明,王波,张慧成. 基于 EMD 的网络舆情演化分析与建模方法[J]. 计算机工程,2012,38(21):5-9.

[2] 舒予,张黎俐. 基于小波分析与人工神经网络的网络舆情预测 [J]. 情报科学,2016,34(4):40-42.
 [3] 高辉,王沙沙,傅彦. Web 舆情的长期趋势预测方法[J]. 电子科技大学学报,2011,40(3):440-445.
 [4] 何炎祥,刘健博,孙松涛. 基于神经网络的微博舆情预测方法 [J]. 华南理工大学学报(自然科学版),2016,44(9):47-52.
 [5] 王努努,张伟佳,钮亮. 基于 ARIMA 和 BP 神经网络模型的舆情情感预测[J]. 电子科技,2016,29(5):83-87.
 [6] 杜智涛,谢新洲. 利用灰色预测与模式识别方法构建网络舆情预测与预警模型[J]. 图书情报工作,2013,57(15):27-33.
 [7] 陈福集,张燕. 基于 E-Divisive 的网络舆情演化分析[J]. 情报杂志,2016,35(4):75-79.
 [8] 黄惠新,陈越,李超零,等. 基于 OLAP 技术的网络舆情分析[C]//河南省计算机学会 2011 年学术年会. 2011.
 [9] JAMES N A,MATTESON D S. ecp: An R Package for Non-parametric Multiple Change Point Analysis of Multivariate Data [J]. Journal of Statistical Software,2013,62(7):1-25.
 [10] 荣自瞻. 网络舆情预警与预案系统的研究与实现[D]. 北京:北京邮电大学,2013.
 [11] 陈婷,曲霏,陈福集. 基于时间片划分的舆情话题演化模型研究 [J]. 武汉:华中师范大学学报(自然科学版),2015,49(6):890-894.