

三维表面模型的快速切割算法

陈矛 唐泽圣 唐龙

(清华大学计算机科学与技术系 北京 100084)

摘要 讨论了一个三维表面模型的快速切割算法,由改进的MC(marching cubes)方法抽取、构成了连续曲面的分组信息以及构成曲面的三角面片的层次包围盒信息,并利用记录了上述信息的B+树模型,完成了三维表面模型的快速切割算法.讨论的算法通过充分发掘和利用MC算法中所隐含的三角面片的包围盒信息和物体各部分间的三维相关信息,在对三维表面模型进行切割计算时,可以快速定位交点,并且根据交点信息,不必遍历模型的所有三角面片,而直接完成对整个被切割模型的分割.研究背景是髌关节整型手术中平面手术刀及球面手术刀对髌关节的切割操作模拟.

关键词 三维表面模型,层次包围盒,三维相关性,快速切割,B+树.

中图法分类号 TP391

医学图象数据的三维重构和手术模拟是当今科学计算可视化的一个研究热点.^[1]在重构后的三维模型基础上,医生们可以进行各种手术方案的模拟,经反复试验后选取最佳方案,从而大大提高手术质量.^[2~4]小儿髌关节手术是常见的手术之一.在计算机上做髌关节手术模拟,需要对重构后的三维模型进行切割、连接及几何变换等操作.为此,我们设计了一个三维表面模型的重构、分组及快速切割算法.这一算法也可以应用于人体其他部位骨骼的整型操作.

MC(marching cubes)方法是一种有效的密集数据场的三维表面模型重构算法.^[5~7]但由以往基于MC的三维重构算法生成的结果为散乱的三角面片,这将给模型切割等全局性操作带来不便.本文的3D重构算法在对3D网格数据的一遍扫描中既完成表面模型的重构又完成3D曲面的分组,并将上述结果和模型的层次包围盒信息记录下来.在此基础上,我们开发了一个3D表面模型的快速切割算法.该算法的核心思想是,将切割刀模型和被切割体模型各自组织到包含层次包围盒信息的B+树中,快速定位交点并分割被切割体模型.图1为系统流程图.

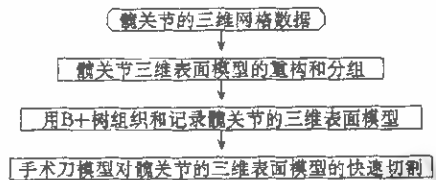


图1 3D髌关节表面模型的重构和快速切割的步骤

1 三维表面模型的重构和分组算法

本项研究的原始数据是螺旋CT产生的小儿髌关节3D网格数据,重构后的3D表面模型由若干组三维曲面表示.每组中的三维曲面是连通的,被称为一个分组,但组间并不连通.

在重构过程中,每生成一个三角面片就判断其与相邻立方体和当前立方体中三角面片的相邻关系,并将其归入所属分组.这时,模型的分组状态会有以下3种变化之一:(1)现有的分组关系不变,当前三角面片被添加到某一分组中,(2)当前三角面片不属于现有的任一分组,需为当前三角面片建立一个新的分组,(3)当前三角面片将原来不相连的多个分组连接成一个新的分组,当模型的所有三角面片生成,并被归入适当的分组中后,整个模型的分组关系便完全确定下来.

2 用B+树组织和记录三维表面模型

2.1 B+树简介及在本算法中的应用

B树的基本思想是,将具体信息存储于叶子节点,而在非叶子节点中存储索引信息.B+树与B树的最大区别在

• 本文研究得到国家863高科技项目基金资助.作者陈矛,女,1972年生,博士生,主要研究领域为计算机图形学,科学计算可视化.唐泽圣,1932年生,教授,博士生导师,主要研究领域为计算机图形学,科学计算可视化,计算几何.唐龙,1938年生,教授,主要研究领域为计算机及其应用,CAD技术,科学计算可视化.

本文通讯联系人:唐泽圣,北京100084,清华大学计算机科学与技术系

本文1998-02-28收到原稿,1998-04-20收到修改稿

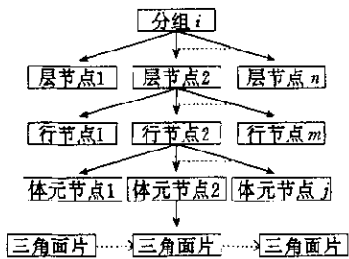


图2 三维表面模型的数据结构

即为每一面片建立多层次索引。

3 手术刀模型对髌关节模型的快速切割算法

3.1 髌关节模型分组 B+ 树中的各种相交节点

切割算法过程中被扩展到的节点为相交节点。

3.1.1 相交体元节点

这里的“相交”指三角面片的最小包围盒的相交。相交体元节点分为 3 种。

(1) 如图 3(a) 所示, 在该体元内, 手术刀模型的三角面片不与髌关节模型的任一三角面片相交, 且髌关节模型的所有三角面片位于手术刀模型的三角面片的同一侧。因此, 该体元内髌关节模型的所有三角面片属于切割操作后的同一个新分组。

(2) 如图 3(b) 所示, 在该体元内, 手术刀模型的三角面片不与髌关节模型的任一三角面片相交, 但手术刀模型的三角面片将该体元内髌关节模型的三角面片集分为两个集合, 每一集合分属切割操作后的不同分组。

(3) 如图 3(c) 所示, 在该体元内, 手术刀模型的三角面片与髌关节模型的 1 个或多个三角面片相交。被切割的髌关节模型的三角面片分成两部分, 每一部分分别属于切割操作后的不同分组。该体元内未被切割的三角面片集也根据在手术刀模型的哪一侧分为两个集合, 分属切割操作后的不同分组。

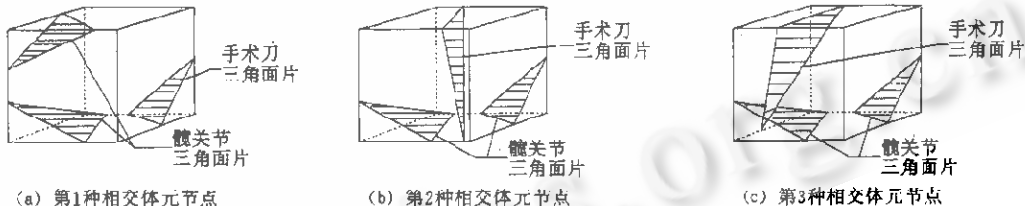


图3

3.1.2 相交行节点

两棵 B+ 树中同一行节点均不为空时, 该行节点可能为相交行节点。设 $X1$ 和 $X2, X3$ 和 $X4$ 分别为手术刀模型和髌关节模型的 B+ 树中该行节点的子节点体元号的下、上界。相交行节点有 3 种。

(1) 如图 4(a) 所示, 区间 $(X1, X2)$ 和 $(X3, X4)$ 无交集。因此, 髌关节模型的 B+ 树中, 该行节点属于切割后的同一分组。图示为 $X2 < X3$ 的情形, $X4 < X1$ 的情形类似。对应切割算法中的操作为不改变被切割分组的 B+ 树和新分组的 B+ 树, 或将被切割分组的 B+ 树中该行节点整个移入新分组的 B+ 树。

(2) 如图 4(b) 所示, 区间 $(X1, X2)$ 和 $(X3, X4)$ 有交集, 但该行节点的子节点中不含相交体元节点。因此, 髌关节模型的 B+ 树中, 该行节点的子节点集分裂为两个集合, 分属切割后的不同分组。在切割算法中的操作为, 分裂被切割分组的 B+ 树和增添新分组的 B+ 树。

(3) 如图 4(c) 所示, 区间 $(X1, X2)$ 和 $(X3, X4)$ 有交集, 且该行节点的子节点中含相交体元节点。在处理了相交体元节点后, 该行节点的子节点集以相交体元节点为界, 分裂为两个集合, 分属切割后的不同分组。在切割算法中的操作为, 修改被切割分组的 B+ 树和新分组的 B+ 树。

3.1.3 相交层节点

两棵 B+ 树的同一层节点均不为空时, 该层节点可能为相交层节点。设 $Field1$ 和 $Field2$ 分别为髌关节模型和手术

于 B+ 树中所有叶子节点由一有序链连接了起来。因此, B+ 树提供了两条搜索路径: 一条由根至叶, 另一条是沿叶节点链。

本文讨论的层次包围盒由顶向下依次为, 分组、层 (CT 切片间形成的立方体层)、行和体元 (逻辑立方体)。这种结构为快速定位三角面片提供了条件。图 2 表示一个分组对应的 B+ 树。多分组模型可表示为 B+ 树组成的森林。B+ 树还提供了同一分组内各部分间的 3D 相关信息。

2.2 B+ 树的构造算法

在重构过程中记录了每个三角面片所属的体元号, 由体元号确定了其所属的层和行后, 可将面片的层次包围盒信息记入 B+ 树中适当的非叶子节点,

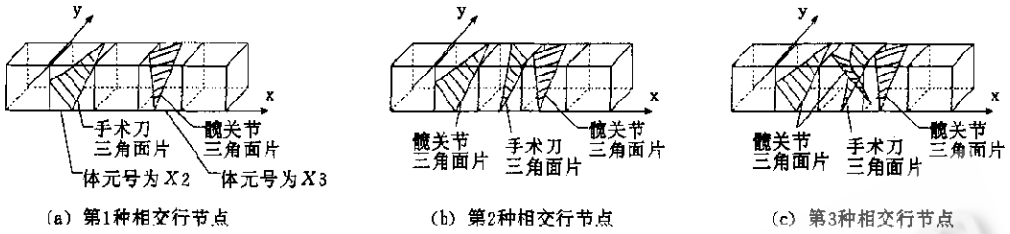


图4

刀模型中该层节点的 2D 包围盒,有 3 种相交层节点。

- (1) Field1 和 Field2 无交集,该层节点的子孙节点属于切割后的同一分组。
- (2) Field1 和 Field2 有交集,但该层节点的子节点中不含相交行节点,不需扩展该层节点的子节点,直接将其子节点集分裂为两个集合,分属切割后的不同分组。
- (3) Field1 和 Field2 有交集,且该层节点的子节点中含相交行节点,在对该层节点的子孙节点进行扩展和回溯后,将其子节点集分裂为两个集合,分属切割后的不同分组。

3.1.4 相交分组节点

若髓关节模型中某一分组的最大包围盒与手术刀模型的最大包围盒相交,则该分组的 B+ 树根节点为相交分组节点,有两种相交分组节点:

- (1) 其子节点中没有相交层节点,这种情况发生在平面手术刀模型刚好与某一 Z 平面重合时。
- (2) 其子节点中有相交层节点,在对其子孙节点进行扩展和回溯后,层节点集分裂为两个集合,分属切割后的不同分组。

3.2 切割算法概述

将手术刀模型离散为三角面片,并用 B+ 树组织,切割运算通过遍历手术刀模型的 B+ 树和髓关节模型的 B+ 树林完成,利用模型的层次包围盒信息及三维相关信息,遍历中只对相交节点进行扩展,因此,这是一种不完全遍历。

4 有关算法效率的讨论

下面分析和对比一下当物体模型的表示方法不同时,几种切割算法的计算复杂性。在下面的讨论中,假设 m 为 3D 髓关节表面模型的三角面片数,而 n 为 3D 手术刀模型的三角面片数。

4.1 整个髓关节表面模型和手术刀模型均为散乱三角面片集

此时的切割计算必须直接面向三角面片,三角面片间的求交运算的时间复杂度为 $O(m \times n)$ 。

4.2 含分组信息的由散乱三角面片表示的髓关节表面模型

假设手术刀模型和髓关节表面模型的每一分组都有包围盒信息,最好情况下,被切割分组的规模远远小于 m ,算法的时间复杂度为 $O(n)$,最坏情况下,各分组的片数与 m 为同一数量级,此时,算法的时间复杂度为 $O(m \times n)$ 。

4.3 采用 B+ 树结构表示髓关节表面模型和手术刀模型

设髓关节模型共有 k 个分组,图 5 表示出髓关节模型和手术刀模型的 B+ 树中各层的节点集规模。

算法的时间复杂性取决于对相交节点的递归和回溯次数,最好情况下,手术刀模型只与一个分组相交,且此分组的 B+ 树根节点为第 1 类相交分组节点,可用二分法搜索分界点,算法的时间复杂性为 $O(\log(m^{\frac{1}{3}}))$,即 $O(\log(m))$ 。最坏情况下,髓关节模型的每一个分组都被手术刀模型切割,且每一被切割分组的 B+ 树中所有节点都需遍历,需要访问的节点数 C 的数量级可由下式表示。

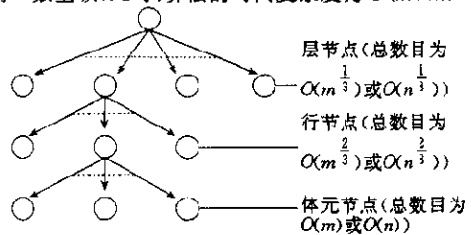


图5 髓关节和手术刀模型的 B+ 树各层节点的数目

$$O(C) = k \times [O(n^{\frac{1}{3}}) + O(n^{\frac{2}{3}}) + O(n)] + O(m^{\frac{1}{3}}) + O(m^{\frac{2}{3}}) + O(m) = O(m + k \times n) \xrightarrow{k \ll n} O(m + n)$$

4.4 小结

上面讨论的都是算法的时间复杂性,就空间复杂性来说,不采用 B+ 树时,除三角面片以外,所需要的存储空间为

常数;而采用 B+树结构需要 $O(m+n)$ 的额外空间储存包围盒等信息.

表 1 3种算法计算复杂性的比较

	时间复杂性比较		空间复杂性比较
	最好情况	最坏情况	
散乱面片模型	$O(m \times n)$		$O(1)$
分组散乱面片模型	$O(n)$	$O(m \times n)$	$O(1)$
分组 B+树模型	$O(\text{Log}(m))$	$O(m+n)$	$O(m+n)$

5 结果

图 6 和图 7 分别为平面手术刀和球面手术刀切割 3D 髌关节表面模型的结果. 上述算法都是在 SGI Indigo2 Extreme 机器上实现的. 由前面的讨论可知, 本切割算法的基本思想适用于任意离散为三角面片并组织成 B+树的 3D 表面模型间的切割运算.

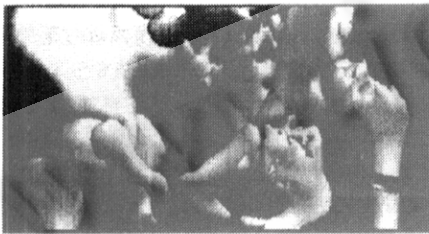


图 6 平面手术刀切割 3D 髌关节表面模型

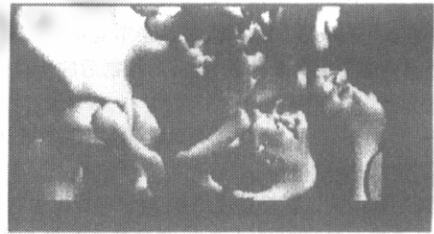


图 7 球面手术刀切割 3D 髌关节表面模型

参考文献

- 1 Rhodes Michael L, Roberstson Douglas D. Applications in surgery and therapy. *Computer Graphics and Applications*, 1996, 16(1):28~29
- 2 Brewster Linda J, Sushma S, Tuy Heang K *et al.* Interactive surgical planning. *Computer Graphics and Applications*, 1984, 4(3):31~40
- 3 Delp Scott L, Loan J Peter, Hoy Melissa G *et al.* An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. *IEEE Transactions on Biomedical Engineering*, 1990, 37(8):757~767
- 4 Koch R M, Gross M H, Carls R R *et al.* Simulating facial surgery using finite element models. *Computer Graphics, Annual Conference Series*, New Orleans, USA: Addison-Wesley, 1996. 421~428
- 5 Lorensen William E, Cline Harvey E. Marching cubes: a high resolution 3D surface construction algorithm. *Computer Graphics*, 1987, 21(4):163~169
- 6 Hoppe H. Progressive mesh. *Computer Graphics, Annual Conference Series*, New Orleans, USA: Addison-Wesley, 1996. 99~108
- 7 Zhou Yong, Chen Wei-hai, Tang Ze-sheng. An elaborate ambiguity detection method for constructing isosurfaces within tetrahedral meshes. *Computer & Graphics*, 1995, 19(3):355~364

Fast-Cutting Algorithm for 3D Surface Model

CHEN Mao TANG Ze-sheng TANG Long

(Department of Computer Science and Technology Tsinghua University Beijing 100084)

Abstract In this paper, the authors discuss the algorithms for fast-cutting on 3D surface model. Using a 3D reconstruction algorithm based on MC(marching cubes) method, information about grouping and hierarchical bounding boxes are extracted in the generation of 3D surface model. With a B+ tree data structure recording all information, cutting on 3D surface model can be accomplished without tracking all triangular patches. The key point of the algorithms lies in sufficiently employing the information according bounding boxes and 3D (three-dimensional) coherence inherent in MC. It provides a way to fast locate the intersection points and accomplish dividing. The result of this paper can be applied to simulate cutting operation using plane and sphere knife in coxae orthopedics.

Key words 3D surface model, hierarchical bounding boxes, 3D coherence, fast cutting, B+ tree.