

体素重建中的快速移动立方体方法

徐晓玲, 李现民, 李桂清, 石 睿, 李 华

(中科院计算技术研究所智能信息处理开放实验室, 北京 100080)



摘 要: 利用二维切片数据重建三维结构, 在医学等领域有着重要的应用。在大数据量的情况下, 等值面抽取的计算效率是富有挑战性的课题。本文提出了一种改进的等值面抽取算法。该算法采用区域增长和一种优化的搜索策略加速边界体素的寻找, 避免了整个体素空间的搜索, 从而加速等值面抽取。实验结果表明, 本算法大大加速了传统的移动立方体 (Marching Cubes) 方法, 特别是在大数据量时, 加速的效果更加明显。

关键词: 医学图像处理; 三维重建; 等值面; 区域增长; Marching Cubes

中图分类号: TP391.9

文献标识码: A

A Fast Algorithm of Marching Cubes in Volume Reconstruction

XU Xiao-ling, LI Xian-min, LI Gui-qing, SHI Rui, LI Hua

(Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100080, China)

Abstract: There are important applications in medical field by reconstructing 3D structures by 2D slices. Computational efficiency of iso-surface extraction is a challenge for large scale of data. This paper gives an iso-surface extraction algorithm. We use region growing and an optimized searching strategy to speed up boundary voxels searching speed. The experiment results show that this algorithm accelerates the speed of Marching Cubes algorithm, especially for large scale of data.

Keywords: medical image processing; 3D reconstruction; iso-surface; region growing; marching cubes

1 引言

医学图像数据经过先期获取阶段的预处理之后, 一般形成一个非常规整的分布在某一空间上的数据, 称为体数据。有了这套体数据, 可根据需要对其进行三维建模, 即建立完整的三维模型。对医学体数据进行三维建模是非常有意义的, 例如根据三维模型, 医生可以对病情做出正确的判断, 可以进行手术规划和手术模拟, 进行计算机辅助人体器官 (如假肢) 的设计和制造等。

体数据是 3D 实体, 在这个实体中含有丰富信息。可视化就是一种从体数据中采用交互图形和图像的手段抽取有意义信息的方法, 它涉及到体数据的表示、建模、处理和绘制^[1]。

大量采样的三维体数据蕴藏着丰富细腻的物体结构。物体的表面重建的任务就是要恢复出蕴藏在体数据中的物体结构的几何信息, 以便更加方便地对其进行定量分析、处理和显示。因为体数据本身并不包含几何信息, 它只是某种物理属性的空间采样, 因此, 要从三维体数据中重建物体表

面, 首先必须根据采样值分布的变化确定物体的边界, 然后用适当的数学表示方法来描述其表征的物体或器官的表面。根据重建的过程所操作的对象, 重建方法可分为从轮廓重建物体表面^[2-4]和从体素重建物体表面^[5-8]。

从轮廓重建物体表面是以断层图像作为处理对象, 首先在每一个切片图像上抽取物体的轮廓, 然后用轮廓拼接构造通过这组轮廓的表面。^[4]为了使重建问题有解, 人们引入新的约束, 如体积最大^[2]、表面积最小^[3]、对应方向一致、跨度最小等。虽然引入约束除去了原始重建问题的多义性, 但不能保证重建出的曲面很准确地表示真实物体的表面。另一棘手的问题是对分叉没有可靠的处理方法。从轮廓重建物体表面方法是以各个切片上抽取的轮廓作为输入, 但是至今没有鲁棒的方法保证边缘轮廓抽取的正确性, 人机交互是不可缺少的。

从体素重建物体表面是在物体表面通过的每一个体素内构造小面片, 这些小面片一般是三角片, 物体表面是由许许多多的小三角片组成。因为它不考虑分叉问题, 全局的拓扑已经由局部拓扑所确定, 所以它比从轮廓重建物体表面方法具有更高的精度和可靠性^[8]。

从体素重建物体表面可以直接对原始灰度体数据进行操作, 构造物体的等值面。在这里, 分割是简单的门限分割, 它不是在重建之前独立进行, 而是隐含到重建的过程中。最有影响的体素级重建方法是 Lorensen 的移动立方体法

收稿日期: 2001-01-27

修回日期: 2001-09-11

基金项目: 国家自然科学基金(69673020); 973 项目(G1998030607); 中科院计算所创新基金项目(20006160) 和国家高技术研究发展计划专项经费资助。

作者简介: 徐晓玲(1976-), 女, 山东济南人, 硕士, 研究方向为三维医学图像建模和计算机图形学; 李现民, 男, 博士, 研究方向为三维医学图像建模和计算机图形学。

(Marching Cube)^[8], Herman 的立方体法^[9], Cline 的分解立方体法^[10]等。但是它们都扫描整个个体数据空间。由于个体数据空间一般非常大,所以非常耗时。针对这一个问题,本文提出一种等值面抽取的改进算法,利用区域增长的方法搜索边界体素,以加速等值面的抽取。

2 相关工作

2.1 Marching Cubes 算法

2.1.1 六面体体素模型

规则的体数据是三维空间中某一个区域中的采样,且采样点在 x, y, z 三个方向的分布是均匀的。因此体数据可以用三维数字矩阵来表示,即 $v(i, j, k)$ 。每八个相邻的采样点所定义的立方体区域就构成了一个体素,这八个采样点称为该体素的八个角点,如图 1 所示。对于体素内的任意一点 $p(x, y, z)$ 的值,采用三线性插值的办法计算可得如下公式:

$$f(x, y, z) = a_0 + a_1x + a_2y + a_3z + a_4xy + a_5xz + a_6yz + a_7xyz \quad (1)$$

其中 $a_i (i = 0, 1, \dots, 7)$ 是由体素 8 个角点唯一确定的常数。

等值面是空间的具有某个相同值的点的集合,它可表示成:

$$\{(x, y, z) | f(x, y, z) = T\}, T \text{ 为常数} \quad (2)$$

在该体素模型下可以看出等值面是三次代数曲面。当体素足够小时可用三角面片表示。并不是每个体素内都有等值面。只有那些既有大于常数 T 又有小于常数 T 的角点的体素才含有等值面,称这些体素为边界体素。

2.1.2 六面体体素内等值面提取

Lorenson 等人提出的 Marching Cubes 方法是对体数据中的体素逐个进行处理。对每个被处理的体素,以三角面片表示其内部的等值面。当给定一个阈值,一个边界体素内的等值面就确定了^[8]。

体素的八个顶点的编码顺序如图 1 所示。对于 Marching Cubes 方法中所用的 256 种构型,经过化简,最后只用十五种构型如图 2 就可以表示^[8]。要把 256 种构型之一转换为 15 种构型之一,只要经过以体素中心为中心,绕 x 轴、 y 轴和 z 轴的适当旋转就可以了。对于每一个标准构形,只要求出其等值面,那么所有构型的等值面都可求出。如果一个边界体素的一条棱边的两个端点的值一个大于 T 另一个小于 T ,

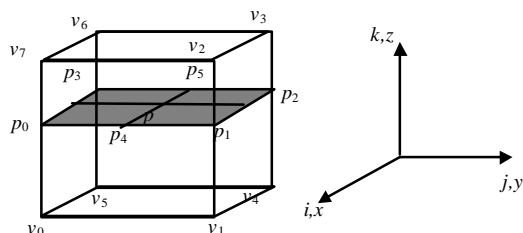


图 1 六面体体素的模型

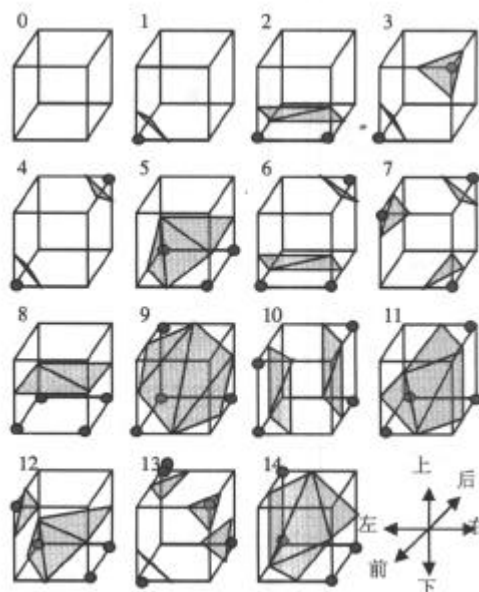


图 2 六面体体素的 15 种标准构形

那么在这条棱边上一定有等值面的一个顶点。不妨设这个棱边的顶点分别为 $v_1(i', j', k')$ 和 $v_2(i'', j'', k'')$, 且顶点的值分别为 $f(v_1)$ 和 $f(v_2)$, 不失一般性, 不妨设 $(i', j') = (i'', j'')$, 那么等值面片的顶点的 k 坐标可由下式求出:

$$k = \frac{(f(v_2) - T)k' + (T - f(v_1))k''}{f(v_2) - f(v_1)} \quad (3)$$

求出在边界体素上所有棱边的所有等值面顶点, 然后连接成三角面。这样一个边界体素内的等值面便被求出了。另外根据边界体素的定义可以看出 15 种标准构形中除了 0 标准构形外, 都是边界体素。

2.2 区域增长

基于区域增长的分割方法是和一系列的种子点相关的。这个过程是从种子点集合出发, 根据某一相似规则将相邻的像素加入到这个种子点集合中去^{[11][12]}。

本文是要找到边界体素, 然后在边界体素中计算等值面。为了搜索边界体素, 使用区域增长的方法来搜索边界体素。具体地说, 如果在边界体素的一个面上有一条等值边, 则在与这面上相邻的体素就是边界体素, 将其加入边界体素的集合。

3 基于区域增长边界体素搜索

本文主要是对边界体素采用了基于区域增长的搜索策略进行搜索, 采用区域增长的策略主要涉及到种子点选取和其它元素加入种子点集合的策略。

3.1 种子点选取和阈值计算

在本算法中采用区域增长的方法来寻找边界体素, 因此首先要解决的问题就是种子边界体素选取的问题。种子边界体素选取就是根据指定阈值选取一个或一些边界体素, 然后从这个或这些边界体素出发, 寻找所有的边界体素。选择

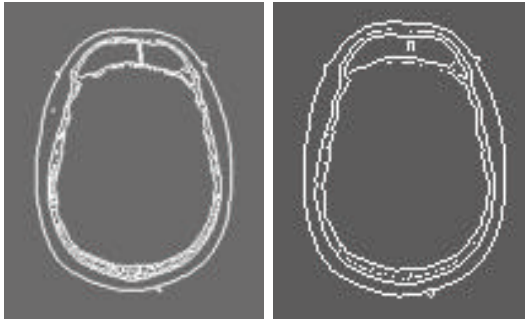


图 3 一个片层图像 图 4 边沿提取后的图像

边界体素的一种办法就是给定一个阈值,根据该阈值在体数据中寻找一些边界体素作为种子体素。本文给出了一种交互的方法进行种子边界体素选取和阈值计算。

所处理的图像是基于片层的 CT 图像或者其它断层图像,先找到具有要抽取等值面的器官或者物体轮廓的片层。比如我们要抽取脑骨的等值面,那么我们找到一个如图 3 所示的片层,对该片层进行边沿提取得到图 4,根据图 3 和图 4 用鼠标选取骨骼外表面的一些边沿点,这些边沿点和这些边沿点所在层次的上一层的点形成的体素为候选种子边界体素。阈值的计算方法为:设我们选择的边沿点为 p_1, p_2, \dots, p_n , 各点的灰度值分别为 $f(p_1), f(p_2), \dots, f(p_n)$, 那么等值面的阈值为:

$$T = \frac{\sum_{i=1}^n f(p_i)}{n}$$

(4)

对于我们所选择的候选种子边界体素先根据我们计算出的阈值进行等值面抽取,那些不含有等值面的体素从我们所选择体素中剔除,剩余的体素为种子边界体素。

3.2 边界体素集合增长策略

在给出一个种子点集合后,一般的区域增长方法将不属于该集合的元素加入种子点集合的方法为:找出一个和种子点集合相邻但不在种子点集合中的元素,计算该元素的某个特性和种子点集合中所有元素该特性平均值之差,如果该值小于指定的值,则将该元素加入种子点集合中,直到所有满足条件的元素都加入种子点集合为止^{[11][12]}。

设一个边界体素上的一个面上有一条等值面三角片的一个边(称为等值边)如图 5 所示,显然 a 点和 b 点的值一个大于 T 一个小于 T ,在和图 5 体素上表面相邻的体素中 8 个角点包括 a, b 两点,所以该相邻体素一定不能转化为 0 标准构形。证明:如果该体素能够转化为 0 标准构形,那么

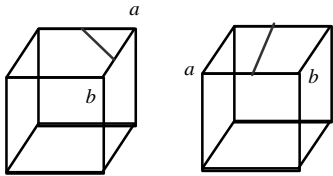


图 5 上表面的边为等值边

该体素的 8 个角点值都大于常数 T 或者都小于常数 T ,这与八个角点中 a, b 两点值一个大于 T 一个小于 T 矛盾,所以该体素不能转化为 0 标准构形,只能转化其它 14 种标准构形,因此该体素是边界体素。即一个边界体素的一个面上有一条等值边,则和该面相邻的体素一定是边界体素。所以当求出一个边界体素的等值面之后,可以根据该体素的每个面上是否有一个等值三角面的一条边,来确定该面相邻的体素便是否为边界体素。如果给出 15 种标准构形中每一构形中相邻的边界体素,那么 256 种构形中任一构形相邻的边界体素都可求出。使用前、后、左、右、上和下六个方向表示一个体素的相邻方向,如图 2 所示,对于如图 2 的 15 种标准构形,根据每个面上是否有一个等值三角面的一条边,将与该面相邻的体素标记为是否边界体素(表 1)。

表 1 15 种标准构形的相邻边界体素的位置

构形	0	1	2	3	4	5	6	7
前	否	是	是	是	是	是	是	是
后	否	否	否	否	是	是	是	是
左	否	是	是	是	是	是	是	是
右	否	否	是	是	是	是	是	是
上	否	否	否	是	是	否	是	是
下	否	是	是	是	是	是	是	是
构形	8	9	10	11	12	13	14	
前	是	是	是	是	是	是	是	
后	是	是	是	是	是	是	是	
左	是	是	是	是	是	是	是	
右	是	是	是	是	是	是	是	
上	否	是	是	是	是	是	是	
下	否	是	是	是	是	是	是	

这样当给出一个种子边界体素集合时,将其它体素加入种子体素集合的规则:对于种子边界体素集合中每一个体素,求出其等值面,根据(表 1)确定和每个面是否相邻的边界体素,若有边界体素并且没有被加入过种子边界体素集合,则加入种子边界体素集合,直到没有新的边界体素为止。

4 算法实现

在算法实现中,将上面的搜索规则记录到各个标准构形的子程序中,这样在等值面处理完毕之后,将没有被处理的相邻的边界体素加入边界体素集合。本算法中边界体素的集合使用了队列的数据结构。

算法的步骤为:

- (1) 选取种子点,计算阈值,形成种子边界体素,加入边界体素队列;
- (2) 按层次顺序从边界体素队列中取出边界体素,计算出其等值面;
- (3) 根据边界体素等值面的信息找到没有入队的相邻的边界体素入队,转(2);
- (4) 当边界体素队列中的所有元素为空时,算法结束。

设每个层次一个队列,这样形成一个队列数组 s ,并且 k 为层次计数器,具体做法如下面伪算法所示:

```

从第  $k$  层选取种子点, 计算阈值;
将种子边界体素进入队列  $s[k]$ , 并置空其它的所有队列;
所有队列为空标志为假;
层次增加标志为真;
while (所有队列为空标志为假) {
    置所有队列为空标志为真;
    if (层次增加标志为真) {
         $k=0$ ;
        while ( $k$  不是最后一层)
            while ( $s[k]$ 不为空) {
                置所有队列为空标志为假;
                取出  $s[k]$ 中的体素;
                计算等值面;
                对于每一个与该体素相邻没有入队的
                边界体素{
                    根据它的相邻位置将其加入
                     $s[k], s[k-1]$ , 或  $s[k+1]$ ;
                }
            }
         $k=k+1$ ;
    }
    置层次增加标志为假;
} else {
     $k$ =最后一层减一;
    while ( $k$  不是第一层)
        while ( $s[k]$ 不为空) {
            置所有队列为空标志为假;
            取出  $s[k]$ 中的体素;
            计算等值面;
            对于每一个与该体素相邻没有入队的
            边界体素{
                根据它的相邻位置将其加入
                 $s[k], s[k-1]$ , 或  $s[k+1]$ ;
            }
        }
         $k=k-1$ ;
    }
    置层次增加标志为真;
}
}

```

5 算法完整性的证明

对于复杂且有连续表面的器官或物体, 本算法可完全不重复的求出其表面。如图 6 所示的器官或物体表面的剖面, a, c 是种子边界体素, 1) 沿着向下的方向搜索, ad, ce 和 cf 上的体素依层次入队, 并求出等值面, 出队; 2) 到达 e 点所在层次时有向上的体素要入队, 在这个体素入队之前, 将 e 点所在体素上面四个边上的等值面的顶点记录在入队体素的顶点缓存中, 对 d 点同样处理; 3) 当搜索方向改为向上时, 第一个不为空的队列为 d 点所在层次的上一层队列, 根据边界体素顶点缓存设置访问过标志, 当层次到 e 点层次上一层队列同 d 一样处理, 直到 db 和 eb 的体素处理完毕; 4) 在向下时, 所有队列为空, 程序结束。

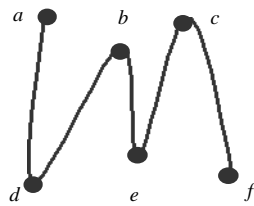


图 6 一个物体或器官的剖面图, a, c 点是种子点

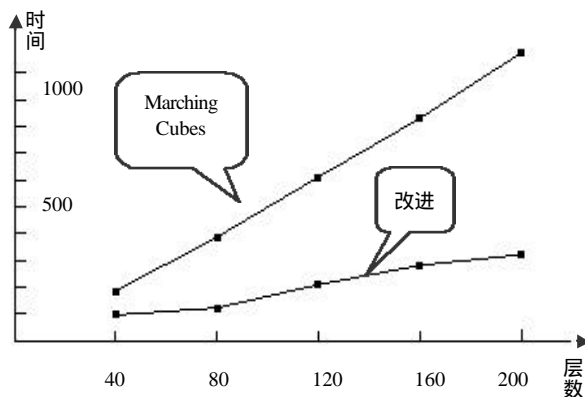


图 7 运行时间和数据量的关系

可以看到每一个边界体素都被处理一次且只被处理入队一次。所以本算法可以毫不遗漏的把物体所有的边界体素搜索出来。

6 实验结果分析和讨论

算法所处理的数据是源自美国国家医学图书馆的 Visible Human 的男子的头部 CT 数据。该数据大小为 $341 \times 587 \times 200$ 。我们在 PIII 微机平台上利用 Visual C++ 实现了上述算法。在阈值选取一样时, 实验得到的 Marching Cubes 算法和改进算法的数据如表 2 所示。

表 2 算法运行的实验数据

层数		40	80	120	160	200
改进	运行时间(秒)	96	122	209	281	321
	三角面数	73070	154254	309274	452348	591536
	顶点数	36806	77425	155413	226432	295646
	体素数	36369	76728	153938	224118	292534
	运行时间(秒)	185	387	611	830	1075
标准	三角面数	74146	187872	327790	577124	728018
	顶点数	36999	96269	168833	285836	372762

从表中可以看出改进算法的运行时间比传统算法时间要短得多。有趣的是改进算法在可以完整地将器官或物体的外表面求出时, 所获得的三角面数和顶点数少于 Marching Cubes 算法所产生的结果。这一事实说明 Marching Cubes 方法将一部分本不是物体外表面的三角面片计入了统计。图 7 给出了改进算法和 Marching Cubes 算法的运行时间和体素数目的关系。两者都随着体素数目的增加而线性增长, 但是 Marching Cubes 的运行时间比改进后的算法要长得多。图 8 给出了改进算法运行时间和边界体素数目之间的关系, 可以看出二者是近似的线性关系, 即运行时间随着边界体素的增

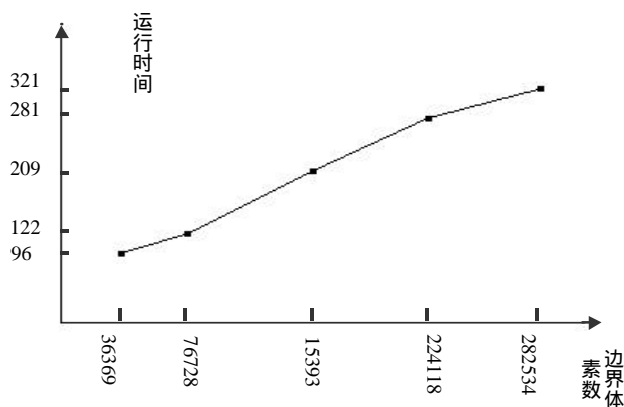
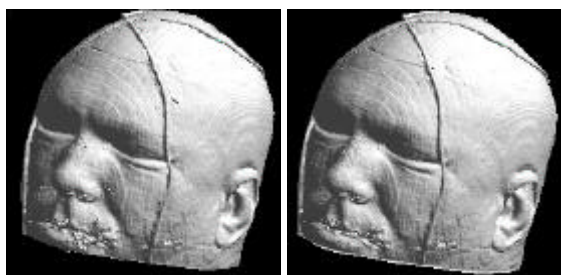


图 8 改进算法运行时间和边界体素间的关系



(a) Marching Cubes 方法, (b) 本文的方法
图 9 Visible Man 的头部表面重建

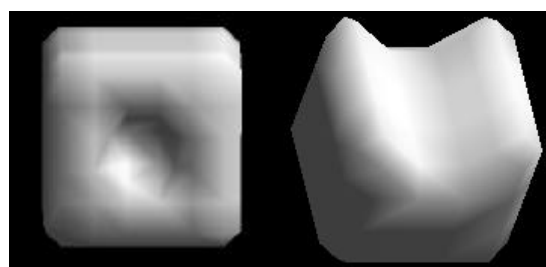


图 10 带有凹陷的立方体的重建

加而以近似固定的比例增加。由此可以看出本算法在时间复杂度和运行时间要优于 Marching Cubes 算法。图 9 是对比实验结果的图片, 图 10 说明本算法也适用于有凹陷的物体的表面重建。

本文给出的算法的特点是采用区域增长技术作为搜索的依据, 在等值面抽取的过程中大大提高了计算效率。在大于 200 层的数据处理中, 计算速度的提高达到 2-3 倍。实验表明, 这一算法也适合于带有凹面的物体。

(上接第 505 页)

参考文献:

- [1] 徐秉铮, 张百灵, 韦岗. 神经网络理论与应用 [M]. 广州: 华南理工大学出版社, 1994.
- [2] 胡守仁, 余少波, 戴葵. 神经网络导论 [M]. 长沙: 国防科学技术大学出版社, 1993.

如何处理随之而来的存储资源的耗费, 以及自动地寻找种子边界体素是值得进一步探讨的问题。此外, 等值面抽取的结果, 一般得到大量的三角片, 适当简化, 以利于编辑, 传输和绘制等, 是另一个近来受到关注的课题^{[13][14]}。

致谢: 感谢首都医科大学罗述谦老师, 中科院自动化所的田捷老师和张晓鹏老师提供的 Visible Man 数据。

参考文献:

- [1] Kaufman A E. Introduction to Volume Graphics [A]. SIGGRAPH'99 COURSE NOTES'41, Volume Graphics.
- [2] Keppel E. Approximating Complex Surfaces by Triangulation of Contour Lines [J]. IBM Journal of Research and Development, 1975, 19(1): 2-11.
- [3] Fuchs H, Kedem Z M, Uselton S P. Optimal Surface Reconstruction from Planar Contours [J]. Communications of the ACM, 1977, 20 (10): 693-702.
- [4] Boissonnat J -D. Shape Reconstruction from Planar Cross Sections [J]. Computer Vision and Image Processing, 1988, 44: 1-29.
- [5] Jane Wilhelms, Allen Van Gelder. Octrees for Faster Isosurface Generation [J]. ACM Transactions on Graphics, 1992, 11(3): 201-227.
- [6] Michael Laszlo. Fast Generation and Display of Iso-surface Wireframes [J]. Computer Vision Graphics and Image Processing, 1992, 54(6): 473-483.
- [7] Jane Wilhelms, Allen Van Gelder. Topological Considerations in Iso-surface Generation [J]. In a special issue of the ACM Siggraph Quarterly, 1990, 24(5): 79-86.
- [8] Lorensen W E, Cline H E. Marching Cubes: A High Resolution 3D Surface Construction Algorithm [A]. SIGGRAPH'87 Proceedings [C], 1987, 163-169.
- [9] Herman G T, Liu H K. Three Dimensional Display of Human Organs from Computed Tomograms [J]. Computer Graphics Image Processing, 1979, 9: 1-21.
- [10] Cline H E, et al. Two Algorithms for Three-Dimensional Reconstruction of Tomograms [J]. Med Phys, 1988, 15(3): 320-327.
- [11] Steven W Zucker. Region Growing: Childhood and Adolescence [J]. Computer Graphics and Image Processing, 1976, (5): 382-399.
- [12] Rolf Adams, Leanne Bischof. Correspondence Seeded Region Growing [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1994, 16(6): 641-647.
- [13] 李捷. 三角网格模型的简化及多分辨率表示 [D]. 北京: 清华大学, 1998.
- [14] Peter Lindstrom. Out-of-Core Simplification of Large Polygonal Models [A]. SIGGRAPH'2000, 259-262.

- [3] 许力, 蒋静坪. CSTR 系统的基于 CMAC 神经网络的学习控制研究 [J]. 控制与决策. 1992, 7(2): 130-135.
- [4] 渠川瑾. 反应釜 [M]. 北京: 高等教育出版社, 1992.
- [5] Pao Y-H, Philips S M, Sobajic D J. Neural-net computing and intelligent control systems [J]. Int J Contr, 1992, 56(2): 263-289.
- [6] Pao Y-H. Adaptive Pattern Recognition and Neural Networks [M]. Reading, MA: Addison-Wesley, 1989.