中国科学 F辑: 信息科学 2009 年 第 39 卷 第 1 期: 134～144
www.scichina.com info.scichina.com
《中国科学》杂志社
SCIENCE IN CHINA PRESS

# 基于测量数据的植物建模

马伟①*, 项波②, 查红彬①, 刘佳②, 张晓鹏②

① 北京大学机器感知与智能教育部重点实验室, 北京 100871;
② 中国科学院自动化研究所, 模式识别国家重点实验室, 中法信息、自动化与应用数学联合实验室, 北京 100080
* E-mail: mawei@cis.pku.edu.cn

收稿日期: 2008-05-30; 接受日期: 2008-11-17
国家重点基础研究发展规划 (批准号: 2004CB3-18000)、国家高技术研究发展计划 (批准号: 2006AA01Z301, 2006AA01Z302, 2007AA01Z336) 和教育部科学技术研究重大项目 (批准号: 103001) 资助项目

**摘要** 测量数据, 即图像和激光扫描数据, 是建模真实植物的可靠依据. 然而, 由于植物结构复杂、自遮挡严重, 相机或激光扫描仪只能感知部分可靠数据且扫描数据中噪声较多, 导致从该类数据恢复植物模型极为困难. 文中提出通过检测尖点特征, 如带叶片植物的叶尖以及纯枝干植物的枝稍, 重建植物模型的方法. 该方法能够从多视点图像中便捷重建带叶片植物模型, 从含噪声激光数据中重建不带叶片的纯枝干模型. 实验证实了所提特征检测算法的正确性以及植物重建算法的有效性.

**关键词**
植物建模
基于图像的建模
基于激光扫描数据的建模
尖锐特征检测
形状分解
骨架抽取

## 1 引言

在虚拟场景, 如三维游戏场景、计算机重现的古遗迹等, 加入真实植物模型能够提高真实感. 植物的测量数据, 即相机拍摄的照片以及激光扫描仪获取的三维点云, 是真实感建模植物不可或缺的要素. 相机和激光扫描仪是计算机视觉领域被广泛应用的感知器. 相机可以方便地捕捉目标物在任何视角下干净的带纹理数据. 激光扫描仪能够获取具有较简单结构的目标物的三维深度信息, 不需要象基于图像的方法那样经历从二维到三维数据的复杂重建过程. 鉴于此, 本文使用相机捕捉带叶片植物图像, 用激光扫描仪直接获取无叶片植物, 即纯枝干的三维点云数据. 然而, 由于植物几何结构复杂, 其测量数据局限性较大: 在相机拍摄的图像中, 叶片相似且多数叶片只有部分可见[1](如图 1(a) 中所示叶片区域特征); 激光扫描得到的数据噪声严重 (如图 1(b) 所示, 尤其在分叉处附近)[2]. 为了得到实用的植物三维模型, 必须为每个子部分, 即每个叶片和每条枝干, 重建完整的网格模型. 如何克服测量数据的局限性完成以上任务是一个极具挑战性的课题.

为重建每个子部分的网格模型, 本文策略是在输入数据中辨识每个部分. 依据我们的观察: 尖锐特征, 即带叶片植物的叶尖和纯枝干数据的枝梢末端, 较之其他位置更容易被观测到 (图 1). 本文提出一种体素模型上的尖锐特征检测算法, 成功提取叶尖和枝梢特征, 用来标识每个器官. 在此基础上, 分别针对上述两类不同植物开发了便捷的建模方法.

首先, 利用图像和激光数据得到植物体模型. 接着, 在体模型上执行特征检测算法获取每个叶片的叶尖位置和中轴方向, 以及枝梢末端位置. 然后, 分别对两类植物采取不同的建模方法. 对于带叶片植物: 1) 从图像中构建一个通用叶片模型; 2) 在体模型上各个叶片位置处按照叶尖位置和叶片中轴方向摆放一个通用叶片; 3) 依据每个叶片对应的体素数据, 通过优化过程调节叶片尺度和局部形状. 由于局部被遮挡叶片的体模型几何结构不明显, 导致难以分辨其对应的可用体数据, 本文采用从叶尖处出发逐层搜索策略寻找可用体素. 另外, 考虑到只有少量邻域体素可用的情况, 本文利用植物学知识协助尺度和形

状优化过程. 叶片重建完成后, 采用自叶根出发的粒子流模拟生成枝干模型. 对于纯枝干建模, 本文提出 DFMS(distance from multiple source points) 分解策略, 分解体数据得到各个子枝, 然后在考虑噪声情况下抽取每个子枝上的骨架和半径信息. 最终, 依据分解时记录的拓扑连接关系合成完整骨架并重建网格模型.

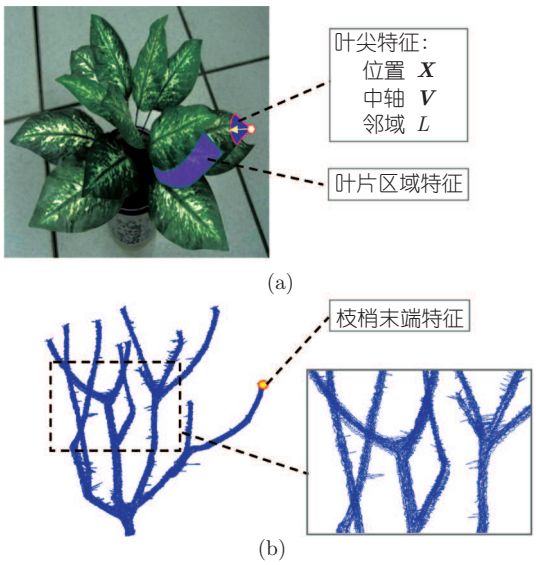



**图 1**

(a) 带叶片植物的特征示意; (b) 纯枝干数据的激光噪声数据以及枝梢末端特征示意

本文主要贡献如下: 1) 提出尖锐特征检测算法并用多类不同数据验证算法有效性. 2) 基于特征检测结果, 提出两种方便的方法分别建模带叶片植物和纯枝干植物. 在建模带叶片植物时, 利用局部可见的叶尖信息推测叶片整体形状, 手工干预较少. 另一方面, 在建模纯枝干植物时, 提出 DFMS 分解方法, 在分解后的子枝上分别建模, 能够消除噪声, 从激光噪声数据中成功重建枝干网格模型.

## 2 相关工作

植物建模方法可分为两大类, 一类是基于规则生长的方法, 另一类是基于测量信息的方法. 基于规则的方法, 如 L-system[3] 以及各种改进版本 [4], 用一系列参数控制并模拟植物生长过程. 为忠实于植物学知识, Reffye 等人 [5] 提出通过分析真实植物建立基本模型的 AMAP 系统. 然而, 即使对于经验丰富的用户, 也很难用这些系统快速建模植物. 因此, Deussen 等人开发了 X-Frog 系统 [6], 内含丰富的植物学规则而给用户提供一个简单的操作界面. Ijiri 等人 [7] 同样基于植物学知识, 给出一种操作方便的花卉建模系统. 上述各类系统能够生成各类型植物模型并得到广泛使用 [8,9].

近年来, 基于测量数据 (即图像和激光扫描数据) 的方法受到越来越多的关注. 其根本原因在于这类方法能够以较强的真实感重现自然界中存在的特定植物. Shlyakher 等人 [10] 用图像恢复的 visual hull 及其骨架控制 L-system 相关参数, 生长得到较符合实际的树木模型. Han 等人 [11] 利用 Bayes 方法结合简单的空间分布先验和图像观测信息, 能够从纯枝干的单幅图像中推测被遮挡部分并恢复完整的枝干模型. 该方法证明即使用数量有限的图像也能产生具有较逼真外观的植物模型. Neubert 等人 [12] 利用图像上方向场的约束, 用粒子流模拟生成枝干模型. Tan 等人 [13] 采用 SFM(structure from motion) 技术, 从多幅图像中得到三维点云模型, 以此为基础重建可见枝干, 并合成不可见枝干. Reche 等人 [14] 没有重建叶片和枝干的网格模型, 而是从有限幅图像中估计一个体模型, 用于基于视点的布告板绘制. 其绘制结果能够满足一定距离外观测的需要. 由于该方法没有生成网格模型, 限制了其在实际中的应用. 与图像相比, 激光扫描仪能够直接获得植物的三维点云数据. 基于激光数据的典型工作是文献 [2]. 该方法在每个激光扫描点云与其局部邻域点之间建立一个加权图, 得到一系列不连续的子图. 从包含根部点的子图开始构造骨架, 量化从根点到其他点的最短路径, 连接相邻类的质心得到该子图所包含的树干骨架, 然后将其扩展直至包含所有子图得到整个树干骨架.

与本文方法最相关的工作是文献 [1,2], 分别是从图像中重建带叶片植物, 从激光数据中重建枝干模型. 前者 [1] 结合图像和从图像中恢复的三维点云信息, 同时加以手工辅助, 分割得到每个叶片 (有些叶片只可观测到局部), 建立叶片模型, 并手工建模枝干. 该方法能够生成真实感较强的植物模型. 相比之下, 本文方法较智能地检测叶片位置和中轴方向, 从局部可见信息推断被遮挡叶片整体形态, 进而从叶

根出发便捷生成枝干模型, 过程简单, 交互量少. 另一方面, 相比文献 [2] 的方法, 本文对分解后的子枝分别抽取骨架和半径信息, 能够从含噪声激光数据中重建可靠的三维枝干模型.

本文首要步骤是尖锐特征检测. 该算法所需输入是空间体素数据. 给定目标植物的一系列无背景图像和相应的相机参数, 本文采用 voxel coloring 方法以及 NCC(normalized cross correlation) 的图像一致性约束重建体素模型 [15]. Voxel coloring 方法在恢复体数据, 尤其是凹区域部分时, 局限性较大. 然而, 尽管目前体重建方法已经得到长足发展, 植物结构复杂、表面不连续, 导致多数方法 [16] 难以应用. 此外, 本文所提植物建模方法一定程度上不受限于体重建方法: 1) 用于检测叶片位置的叶尖附近体数据, 通常直接由图像轮廓确定, 并非由对光照和遮挡较敏感的 NCC 一致性度量决定 [15]; 2) 对局部被遮挡叶片采取的特殊处理, 在遇到凹区域未完全恢复的体数据时能够发挥同样作用. 本文所用激光数据由 6 个不同视角数据融合而成. 激光数据体模型获取方式为: 递归细分其包围盒, 并在每一次细分后去掉偏离原数据的体素. 在此过程中, 与主部分不相连接的体素被视为噪声去除.

文章结构安排如下: 第 3 节详细介绍特征检测算法, 并验证该算法的有效性. 第 4 节给出分别针对带叶片植物和纯枝干植物的建模方法, 并在介绍完每种方法之后, 展示和分析重建结果. 第 5 节是结论.

## 3 尖锐特征检测

鉴于叶尖特征和树枝末端较其他部位形状尖锐, 本文提出一种从几何上描述体素局部尖锐度的方法. 通过计算体模型上每个体素的尖锐度, 得到一个空间得分场. 然后依据得分场, 用户简单指定分值范围即可得到叶尖和枝梢特征.

### 3.1 尖锐度定义

给定体模型上一个体素 $C_0$(其中心空间坐标 $\boldsymbol{X}_0$), 选择其周围连续的 $L$ 层邻域 (体素的第一层邻域是该体素的 26 邻域). 假定位于第 $l$ 层领域的某体素坐标为 $\boldsymbol{X}_n^l$, 其中, $n \in N^l$ 是第 $l$ 层领域的体素个数. 则自 $\boldsymbol{X}_0$ 出发, 沿着某一方向 $\boldsymbol{V}_i$, 到 $\boldsymbol{X}_n^l$ 的逐层累加伸展量为 (依据矢量加法规则)

$$r_n^l(\boldsymbol{V}_i) = (\boldsymbol{X}_n^l - \boldsymbol{X}_0)^{\mathrm{T}}\boldsymbol{V}_i. \tag{1}$$

则 $L$ 层邻域体素沿着 $\boldsymbol{V}_i$ 的总累加伸展量为

$$S(\boldsymbol{V}_i) = \sum_{l=1}^{L}\frac{1}{N^l}\sum_{n=1}^{N^l} r_n^l(\boldsymbol{V}_i), \tag{2}$$

其约束项为

$$\boldsymbol{V}_i^{\mathrm{T}}\boldsymbol{V}_i = 1. \tag{3}$$

本文定义 $C_0$ 的局部尖锐程度为 $S(\boldsymbol{V}_i)$ 的最大值. 该定义方式符合直观理解, 本文在 3.4 小节用实验验证了该定义的准确性. 使得 $S(\boldsymbol{V}_i)$ 值最大的矢量 $\boldsymbol{V}_0$ 可表示为

$$\boldsymbol{V}_0 = \arg\max_{\boldsymbol{V}_i}(S(\boldsymbol{V}_i)). \tag{4}$$

本文应用中, 只有凸特征在考虑范围之内, 因此在求解 $\boldsymbol{V}_0$ 时, 加入如下约束

$$r_n^l(\boldsymbol{V}_i) > 0,\ \forall \boldsymbol{X}_n^l. \tag{5}$$

### 3.2 尖锐度求解

在 (4), (5) 式的约束下, 最大化 (2) 式可以得到 $C_0$ 的得分值 $S$, 其求解过程如下. 首先, 用 (1) 式替代 (2) 式中的 $r_n^l(\boldsymbol{V}_i)$ 并整理, 得到

$$S(\boldsymbol{V}_i) = \boldsymbol{V}_i^{\mathrm{T}}\sum_{l=1}^{L}\frac{1}{N^l}\sum_{n=1}^{N^l}(\boldsymbol{X}_n^l - \boldsymbol{X}_0). \tag{6}$$

从 (6) 式可以看出, $S(\boldsymbol{V}_i)$ 是 $\boldsymbol{V}_i$ 的线性函数, 公式右侧是 $\boldsymbol{V}_i$ 与另外一个矢量的内积. 因此, 当 $\boldsymbol{V}_i$ 与该矢量同方向时, $S(\boldsymbol{V}_i)$ 达到最大. 那么, $\boldsymbol{V}_0$ 即如下形式:

$$\boldsymbol{V}_0 = \frac{\sum_{l=1}^{L}\frac{1}{N^l}\sum_{n=1}^{N^l}(\boldsymbol{X}_n^l - \boldsymbol{X}_0)}{||\sum_{l=1}^{L}\frac{1}{N^l}\sum_{n=1}^{N^l}(\boldsymbol{X}_n^l - \boldsymbol{X}_0)||}. \tag{7}$$

如果 $\boldsymbol{V}_0$ 满足方程 (5) 的约束, 则 $C_0$ 的尖锐度值为

$$S(\boldsymbol{V}_0) = \left\|\boldsymbol{V}_0^{\mathrm{T}}\sum_{l=1}^{L}\frac{1}{N^l}\sum_{n=1}^{N^l}(\boldsymbol{X}_n^l - \boldsymbol{X}_0)\right\|. \tag{8}$$

否则, 置为零.

假定邻域体素均匀分布在一个轴对称 (穿过 $C_0$) 的凸体上 (如圆锥), 用几何方法可以很容易证明 $\boldsymbol{V}_0$ 与对称轴重合. 由于本文假定叶片关于其中轴对称, $\boldsymbol{V}_0$ 即是叶片中轴. 然而, 因为空间体素并非均匀分布, 导致 $\boldsymbol{V}_0$ 与叶片中轴有一定的偏离. 为解决这个问题, 我们重新计算累加伸展量 $r_n^l$: 把相邻体素 (不

论是面邻、角邻还是边邻) 的距离全部设置为单位 1. 计算过程如下:

$r_n^l = 0$
$l' = l$
while $l' > 0$
  find a closest neighbor $\boldsymbol{X}_n^{l'-1}$ for $\boldsymbol{X}_n^{l'}$ in ring $l'-1$
  $r_n^l := r_n^l + \frac{(\boldsymbol{X}_n^{l'} - \boldsymbol{X}_n^{l'-1})^{\mathrm{T}} \boldsymbol{V}_i}{||\boldsymbol{X}_n^{l'} - \boldsymbol{X}_n^{l'-1}||}$
  $l' := l' - 1$
endwhile

所有体素的得分值形成一个空间得分场, 如图 2(a) 所示. 从图中可看出, 分值较高的体素位于叶尖附近. 图 2(b) 显示: 最优向量与叶片中轴方向吻合.

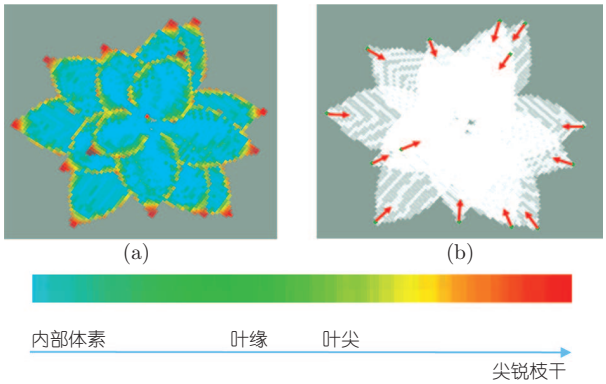



**图 2 叶尖特征检测**

(a) 采用底部伪彩色条 (从左到右分值升高) 绘制的得分场; (b) 叶尖特征以及对应的中轴矢量. 绘制每个体素所用 alpha 值为 0.3

### 3.3 特征选择

根据 (2) 式中尖锐度的定义以及实验过程显示, 细而尖锐的枝干得分最高, 然后是叶尖处. 基于此规律, 本文抽取特征 (叶尖或枝梢点) 步骤如下: 1) 在空间得分场中进行局部极大值抑止 ($L$ 层邻域内), 以免叶尖附近得分较高的体素参与竞争; 2) 对剩余特征按照得分从高到低排序; 3) 用户在三维可视化界面中指定包含正确特征的合适范围. 范围变化由两个滑动块控制, 并将选择结果直接显示在三维界面中的体素模型上, 以方便用户快速决定范围.

如此即可简单地获得叶尖和枝梢特征点, 每个叶尖特征伴随一个指定对应叶片中轴方向的向量. 然而, 并非所有待抽取特征都连续分布在队列中: 体模型可能包含噪声 (尤其是带叶片植物的体模型), 或者存在悬空的较粗枝干得分值低于叶尖得分值. 在这种情况下, 只决定覆盖多数正确特征的连续范围. 为避免特征丢失或错误选择, 我们提供给用户另一个界面: 点击体模型相应位置以去除或者添加特征. 范围选定和错误特征矫正过程只需少量交互, 可以在数秒钟之内完成.

### 3.4 算法验证

本文在 4 种类型数据上分别测试了所提出的特征检测算法, 分别是合成的带叶片植物、纯枝干植物, 以及实际的带叶片和纯枝干植物. 合成数据先由 AMAP Gensis™ 软件生成然后采用与激光数据一样的处理方法得到体模型. 合成数据主要用来验证算法在无噪声情况下的准确性. 实际数据 (包括从约 20~30 幅图像中恢复的带叶片植物体模型和细分激光点云得到的的纯枝干体数据) 通常掺杂噪声, 被用来验证算法在实际应用中的有效性.

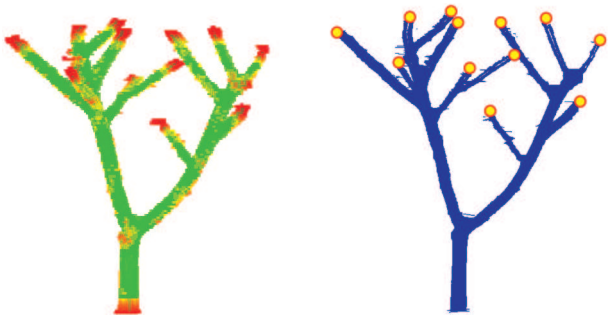



**图 3 枝梢特征检测**

(a) 采用与图 2(a) 同样的伪彩色条绘制的得分场; (b) 在原始激光噪声数据上显示的枝梢特征

表 1 列出了特征检测过程中的统计数据. 其中, 第 1 列给出了植物名称及其对应的本文中的图片. 第 2 列指示植物类型, R 表示真实植物, S 表示合成植物, F 表示带叶片植物, B 表示无叶片植物. 第 3 列是植物体数据所包含的体素数目. 第 4 列是用于控制邻域层数的参数 $L$. 第 5 列则是执行完 3.3 小节中描述的 3 步选择步骤之后, 用户必须手工添加或者删除的特征数目. 第 6 列是最终抽取到的特征数目. 作为对比, 第 7 列给出了真实特征数目. 图 2 和 3 分别显示了在两类真实植物体素数据中抽取到的特征: 叶尖特征和枝梢特征. 为清晰表述检测结果, 图 3(b) 以及图 4 和 5 中显示的特征都是在检测结果上手工做了突出标注.

**表 1　特征检测过程中的统计数据. PT 表示类型; VN 表示体素数目; REN 表示重编辑特征数目; EFN 表示所恢复的特征数目; RFN 表示真实特征数目**

| | PT | VN | $L$ | REN | EFN | RFN |
|---|---|---|---|---|---|---|
| Nep.(图 9,10) | R&F | 10588 | 8 | 2 | 11 | 11 |
| Die.(图 1,7,11) | R&F | 20385 | 8 | 3 | 14 | 14 |
| Ant.(图 12) | R&F | 155249 | 12 | 4 | 15 | 15 |
| Rub.(图 2) | S&F | 11413 | 8 | 0 | 14 | 14 |
| Sun.(图 8) | S&F | 6299 | 8 | 0 | 15 | 15 |
| Bam.(图 6) | S&F | 6264 | 6 | 1 | 190 | ≈250 |
| Wil.(图 4) | S&B | 49883 | 10 | 0 | 18 | 18 |
| SuB.(图 4) | S&B | 12367 | 8 | 0 | 13 | 13 |
| Mur.(图 5) | R&B | 3788 | 8 | 0 | 12 | 12 |
| Tel.(图 3,14) | R&B | 11965 | 10 | 0 | 12 | 12 |

从表 1 中可以看出, 合成的数据以及真实的纯枝干数据不涉及手工增减特征, bamboo(图 6) 的数据是个例外. 另外, 对于两种纯枝干数据, 在实际操作中不需要指出特征的覆盖范围, 而只需提供参数 $N$, 指示计算机只保留前 $N$ 个特征即可. 实验结果证明了特征检测算法的正确性. 对于实际的带叶片数据, 由于该类体模型重建算法的限制, 导致恢复的体模型不准确且噪声较多, 直接影响了特征检测效果. 正如表 1 中显示, 需要手工地添加或去除少量特征. 然而, 此类操作很方便, 只需在三维界面相应位置处点击鼠标, 计算机搜寻附近的局部极大值作为新添加的特征, 或是去除局部极大值从而达到去除特征的目的.

以上实验显示: 用来区分并定位叶片和枝干的特征检测算法有效且方便实用. 当然, 在尖锐特征不明显或者完全被遮挡的情况下, 则无法被检测到. 典型的例子就是 bamboo 模型, 重建后丢失了部分叶片, 原因是这些叶片的叶尖在原模型上并不明显.

## 4　植物重建

本节依据第 3 节得到的叶尖以及枝梢特征对两类不同植物: 带叶片植物和纯枝干数据分别设计了不同的重建方法.

### 4.1　带叶片植物重建

重建带叶片植物, 本文首先从图像中选取一个完整叶片 (手工选取) 并沿其中轴剖分得到一个带纹理的平面状通用叶片网格模型 (图 7). 选择叶片时, 要求该叶片满足从三维到二维的较正视投影. 然后, 在各个叶尖位置处放置一个通用叶片, 对齐中轴方向, 并对叶尖处 $L$ 层邻域体素做 PCA(principle component analysis) 分析定向叶片朝向. 图 7(c) 给出了叶片摆放之后的效果. 以下, 详细描述叶片尺度、局部形状求解以及基于叶根位置的枝干建模过程.

#### 4.1.1　叶片尺度

在调整尺度和局部形状时, 需要利用体素数据. 而对于局部被遮挡叶片, 除开叶尖附近小的局部范围内的体素数据外, 其他部分的体素数据作为叶片的结构性并不明显, 很难用来计算叶片形状. 因此, 给定一个三维叶片, 以初始的 $L$ 层邻域为基础, 逐层递增以选择更多邻域用于拟合并求解叶片的真实尺度. 在每次迭代过程中, 增加一层邻域, 并通过离散化的穷尽搜索得到针对当前所有邻域的最佳拟合尺度. 当前拟合误差大于给定阈值时, 终止迭代过程并以前一次迭代得到的最优尺度作为该叶片的最终尺度. 根据邻域的选择标准, 严重被遮挡叶片对应的可用体素不多, 造成尺度计算或局部形状计算的困难. 为克服这个问题, 在尺度计算过程中加入一个经验式约束, 即当无法依据足够的邻域体素计算尺度时, 利用周围邻居叶片尺度来约束当前叶片尺度, 以防出现过大或过小的叶片.

#### 4.1.2　叶片局部形状

目前得到的三维叶片都是平面状. 本节通过优化一个能量函数调节每个叶片 $F$ 的局部几何形状该能量函数定义为

$$E(F) = w_1 E_b(T) + w_2 E_s(T) + w_3 E_m(M). \quad (9)$$

$E_b(T)$ 是一个平衡项, 其对应的函数为

$$E_b(T) = \sum_{T_i \in T} \sum_{\boldsymbol{X}_j^{T_i} \in \boldsymbol{X}^{T_i}} ((\boldsymbol{X}_j^{T_i} - \boldsymbol{C}_{T_i})^{\mathrm{T}} \boldsymbol{O}_{T_i})^2. \quad (10)$$

其中, $T$ 是叶片 $F$ 的三角网格集合. $\boldsymbol{X}^{T_i}$ 表示在三角形 $T_i$ 所在平面上的投影位于 $T_i$ 内部的体素. $C_{T_i}$ 是 $T_i$ 的中心. 该平衡项在位于各个叶片上方和下方的体素达到平衡时取最小值.

$E_s(T)$ 是约束叶片表面的平滑项, 其定义为

$$E_s(T) = \sum_{T_i \in T} \sum_{T_j \in \mathrm{Nei}^{T_i}} \left(\tan \frac{\theta_{i,j}}{2}\right)^2. \quad (11)$$

$\mathrm{Nei}^{T_i}$ 表示三角形 $T_i$ 的所有 1 邻域. $\theta_{i,j}$ 表示 $T_i$ 和 $T_j$ 的法向夹角.

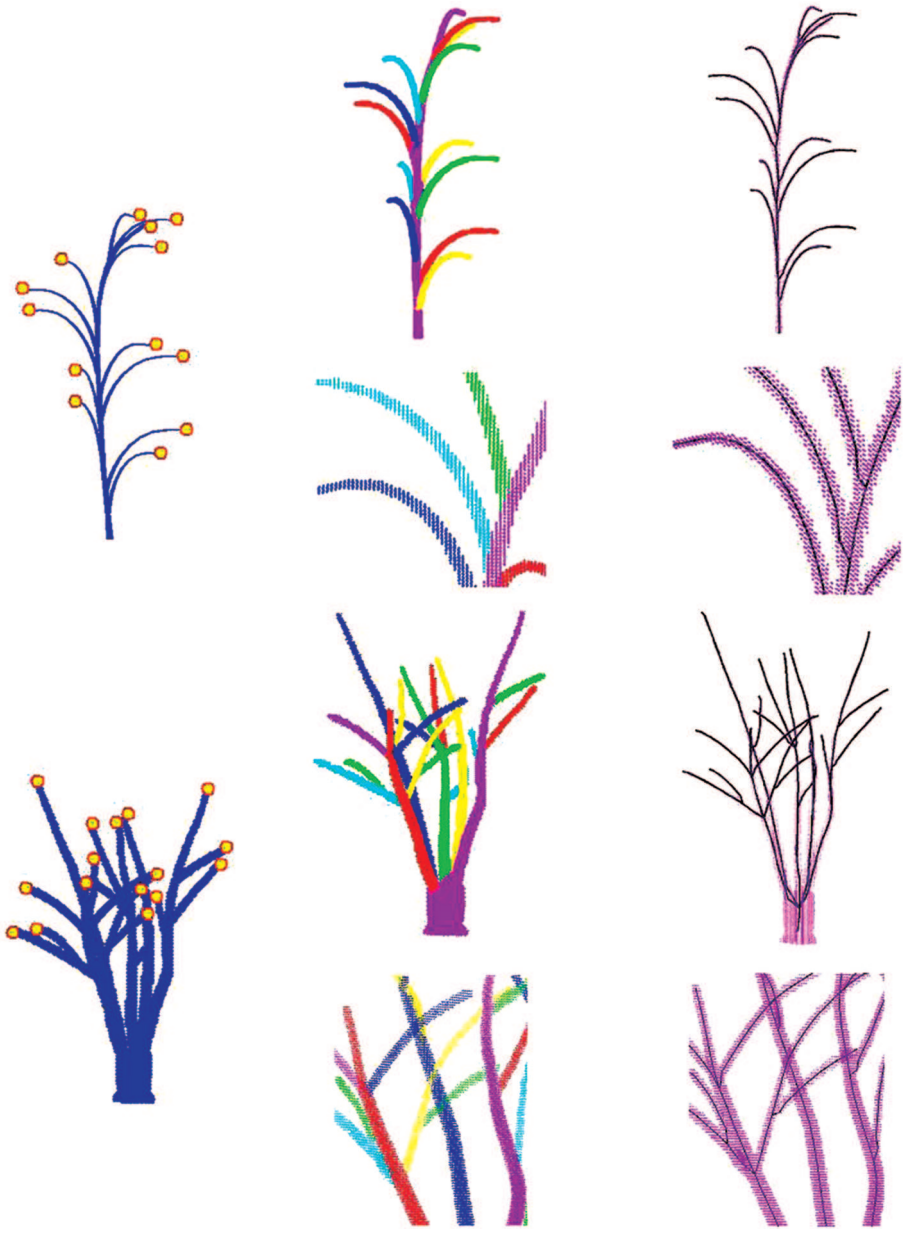


**图 4 合成数据 (上面 2 行: sunflower branch; 下面 2 行: willow branch)**

从左到右图中 3 列分别为: 原始数据以及检测到的特征点、分割结果, 以及骨架抽取结果

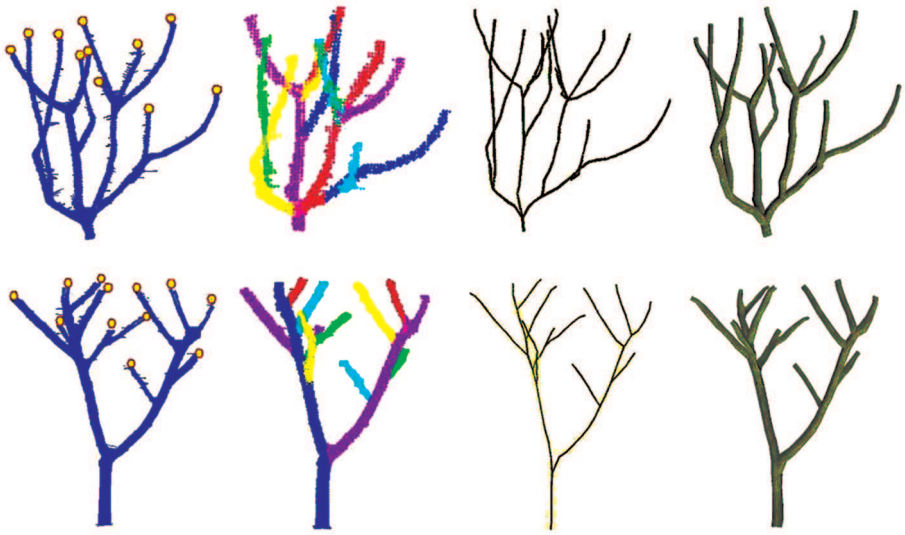


**图 5　真实数据 (上面 1 行: murraya paniculata; 下面 1 行: telosma cordata)**

从左到右图中 4 列分别为: 原始数据及特征点、分割结果、骨架抽取结果, 以及重建的网格模型




**图 6　Bamboo**

(a) 原始模型; (b) 重建模型

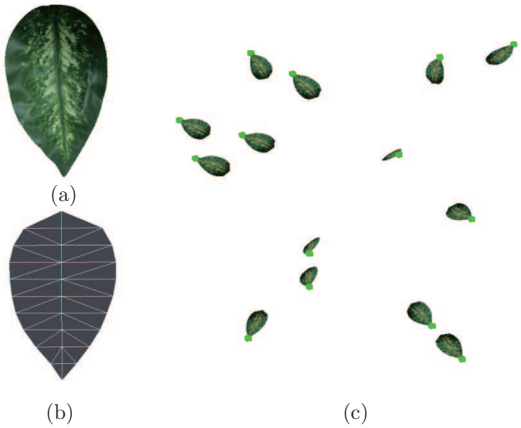



**图 7　初始化叶片**

(a) 通用叶片图像; (b) 通用叶片几何模型; (c) 初始化所有叶片

最后一项 $E_m(M)$ 是中轴约束项, 用来避免中轴随意弯折, 采用了与文献 [17] 相同的定义方式

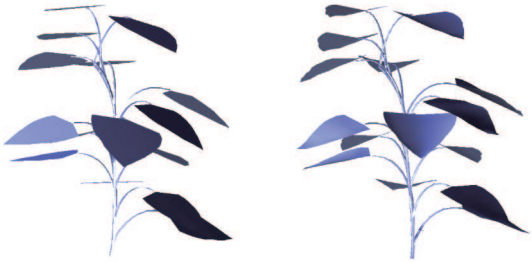
$$E_m(M) = \sum_{M_i \in M} \left( \left( \frac{M_i M_{i+1}}{M_{\text{savg}}} \right)^2 - 1 \right)^2 . \qquad (12)$$

$M_i$ 是位于中轴 $M$ 上的点. $M_i M_{i+1}$ 是相邻两点之间的间隔长度. $M_{\text{savg}}$ 是平均间隔. 当所有相邻间隔相等时, 该中轴能量最小. 该项可以有效地约束中轴上的不必要变化. (9) 式中, $w_1, w_2$ 以及 $w_3$ 是调节 3 个不同能量项的权重.

与 4.1.1 小节一样, 考虑到局部被遮挡叶片, 本文以初始的 $L$ 层邻域为基础, 逐层递增来选择更多较为可靠的邻域体素. 与之不同的是, 这次的选择是垂直于叶片方向的有效邻域的选择. 在迭代选择过程中, 新加入的邻域层厚度 (在叶片垂直方向的投影高度) 应当小于一个预先给定的阈值. 该阈值的选取取决于实际叶片的厚度以及体素的尺寸. 图 8 给出了叶片优化前后的效果. 优化方程中中轴平滑约束和叶面平滑约束 (两个约束均满足实际叶片的要求), 使得仅仅依靠部分体素就可以得到比较真实的叶片形状.

### 4.1.3　树枝建模

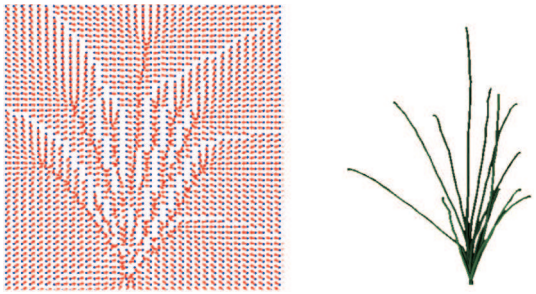
枝干建模采用文献 [12] 中粒子流方法. 与之不同的是, 本文采用透视投影而非平行投影. 另外, 粒子的初始位置采用叶片模型的叶根位置而非随机采



(a) (b)

**图 8 叶片局部形状优化前后对比**

(a) 优化前; (b) 优化后



(a) (b)

**图 9 枝干建模**

(a) 引导枝干生长的方向场; (b) 生成的枝干模型

样的方法. 得益于已经重建好的叶片模型, 该枝干重建过程简单方便. 图 9 给出了引导枝干生长的方向场以及生成的枝干模型.

#### 4.1.4 实验结果

实验对象包括 3 棵真实植物, 分别为 nephthytis(图 10), dieffenbachia (图 1(a) 和图 11), 和 anthurium scherzerianum (图 12). 拍摄照片约 20~30 幅. 表 1 中列出了重建相关数据. 涉及到的手工交互限于指定特征范围, 添加或去除少量叶尖点 (不多于 4 个, 见 3.4 小节), 以及选择叶片通用模型. 用户可以很快完成这些操作. 图 10~12 显示了重建之后的模型.

另外, 利用上述重建方法, 本文方便地完成了复杂合成数据 bamboo 的重建. 整个过程手工参与仅为: 确定通用叶片和去除一个错误叶片. 重建后的模型采用了原模型的枝干. 图 6 给出了与原始模型的对比结果.

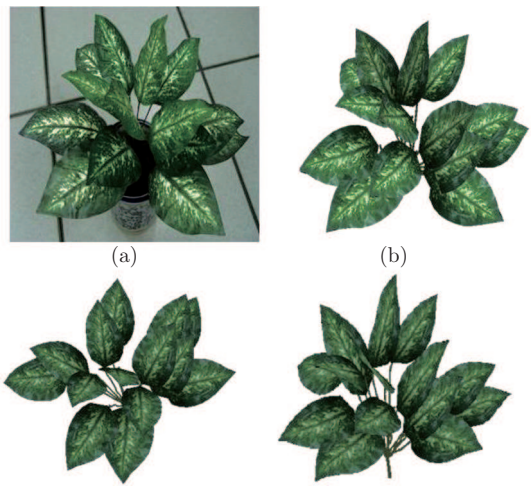
实验证明所提通过叶尖推测叶片的方法可以方便地产生具有真实感的模型. 然而, 当体模型上的叶



(a) (b)

**图 10 Nephthytis**

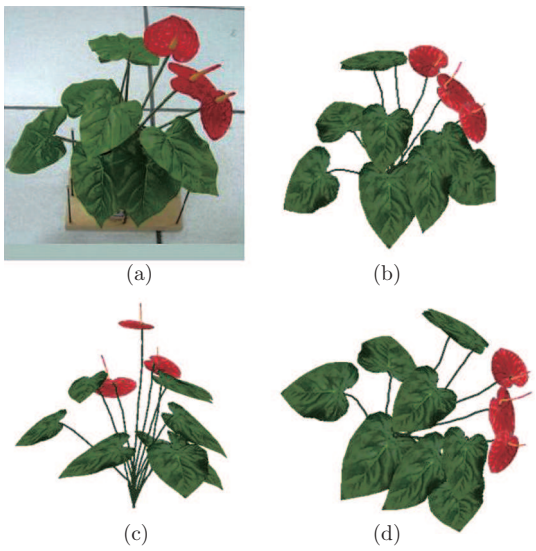
(a) 一幅原始图像; (b) 重建结果



(a) (b) (c) (d)

**图 11 Dieffenbachia**

(a) 一幅原始图像; (b) 与图像视点相近的模型绘制结果; (c),(d) 新视点下绘制的三维模型



(a) (b) (c) (d)

**图 12 Anthurium scherzerianum**

(a) 一幅原始图像; (b) 与图像视点相近的模型绘制结果; (c),(d) 新视点下绘制的模型

尖不可见时方法失效. 例如, 在重建 bamboo 模型时, 只有 190 个叶片 (原有约 250 个) 被恢复.

### 4.2 纯枝干模型重建

本文采用基于分解 [18] 的方法从激光数据中抽取纯枝干骨架, 进而重建网格模型. 基于分解的策略在骨架提取 [19~21] 中用的较多. 相比此类文献, 本文方法典型特点是采用多个特征点作为其对应类的标志点和起始点, 更适合在噪声数据上分解并抽取骨架. 该方法过程如下: 首先通过 DFMS 将体素数据分解成若干子枝. DFMS 概念源于单点出发的距离变换 [4](distance from a source point, 简称 DFS) 方法 (DFS 的缺陷是对噪声敏感). 然后, 抽取每个子枝骨架及半径信息, 依据分割时记录的拓扑连接关系合成总体骨架. 最后, 依据骨架和半径信息生成表面网格模型.

#### 4.2.1 DFMS 变换

DFS 变换是层层扩散的过程, 并返回每个体素到源点的距离值 [19]. 本文所提 DFMS 变换从每个枝梢点出发同时进行多个 DFS 变换, 扩张枝梢点对应类的领域. 每个 DFS 在区域扩散过程中遇到分叉处则停止并等待该分叉处其他子枝对应的 DFS 到来. 然后, 所有 DFS 在分叉处竞争继续生长的机会. 竞争失败者终止生长, 只有唯一获胜的 DFS 可以继续扩张领域直至遇到下一个分叉处或者根部. 图 13 给出了基于 DFMS 分解的简单示意图.

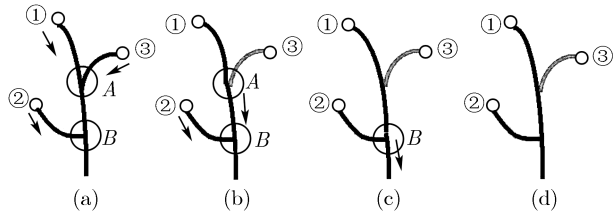



**图 13 基于 DFMS 的分解过程示意**

#### 4.2.2 分叉处检测

对于从某枝梢特征出发的 DFS 变换, 具有相同距离值的体素形成一个等值面. 若等值面中的所有体素点彼此相连, 则将这些体素全部归到该枝梢决定的子枝类. 反之, 则可确定此等值面到达一个分叉处. 假定该等值面包含 $M$ 组 ($M > 1$) 不相连的体素 (每组内部体素是连通的), 则表示此分叉处包含 $M$ 个子枝类. 决定分叉处之后的主要问题为: 分界面处理和类簇竞争.

#### 4.2.3 分界面处理

分叉处的有效处理能够保证光滑的边界面. 如上, 自某枝梢点出发的 DFS 变换遇到分叉处, 分叉面被分为 $M$ 个自连接的组, 如图 14(a) 中绿色部分被分为 2 组. 连接这 2 组体素的中心, 得到一个方向矢量 (图 14(a)). 沿着这个方向, 搜索所有在分叉面之前的等值面内体素, 能够以最短路径连接这 2 组体素的那些被排除在枝梢点对应类外. 图 14(b) 给出一个实际分叉处的分割效果. 上述算法假定各分枝都是局部圆柱形.

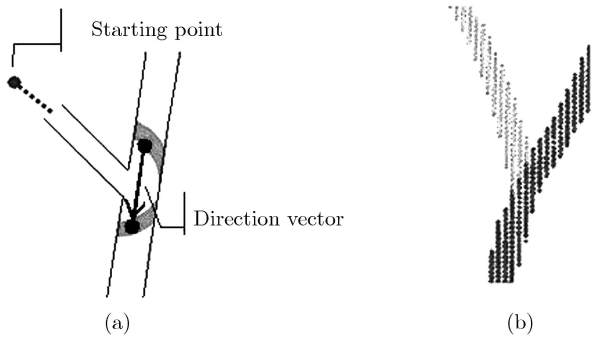



**图 14 分界面的优化**

(a) 示意图; (b) 真实分解结果

#### 4.2.4 分叉处的竞争

给定 $N$ 个树梢点, 我们可以获得从各个树梢点出发进行 DFS 变换并扩散的 $N$ 条支路. 不同支路在分叉处相遇并停止, 为得到继续扩张的机会发生竞争. 此处, 规定具有最大半径的支路将在竞争中获胜. 此规定符合植物学知识.

#### 4.2.5 骨架提取

至此, 枝干模型体数据已经被分解为 $N$ 个子枝. 接着, 为每个子枝提取骨架. 给定一个子枝, 通常将组成骨架的节点定义为等值面的中心 (此处使用距离量化 [2] 来避免节点过密). 考虑到噪声的存在, 本文计算骨架点 $\boldsymbol{X}_t$ 的公式如下:

$$\boldsymbol{X}_t = w_1\left(\frac{(\boldsymbol{X}_{t-1}-\boldsymbol{X}_{t-2})||\boldsymbol{C}_t-\boldsymbol{X}_{t-1}||}{||\boldsymbol{X}_{t-1}-\boldsymbol{X}_{t-2}||}+\boldsymbol{X}_{t-1}\right) + w_2\boldsymbol{C}_t, \tag{13}$$

其中, $\boldsymbol{C}_t$ 是等值面中心, $\boldsymbol{X}_{t-1}, \boldsymbol{X}_{t-2}$ 分别表示前 2

个等值面中的骨架点. 公式第 1 项是光滑项, 第 2 项是数据项. $w_1$, $w_2$ 是调节数据项和光滑项的权重. 每个骨架点对应的半径采用将等值面所包含体素投影在垂直于骨架方向的平面上计算得到. 同样考虑到噪声影响, 若某节点半径较大, 参考附近节点的半径重新计算该半径值.

得到每个子枝单独的骨架点和骨架点处对应半径之后, 依据分解时记录的拓扑关系融合所有骨架成一个完整的骨架模型. 接着, 采样每个骨架点和半径决定的圆环作为枝干表面点, 将各表面点顺次连接生成表面网格.

#### 4.2.6 实验结果

我们在 4 组纯枝干数据上验证分解以及骨架抽取算法, 包括两组 AMAP Gensis$^{TM}$ 生成的合成数据 (图 4) 和 2 个由激光扫描 (6 个视角) 得到的真实植物 (图 5). 图 4 展示了合成数据的结果, 可以看到, 算法分解得到的分枝有光滑合理的边界, 并且提取的骨架准确度高. 图 5 给出了激光数据的结果. 可以看出, 重建后的模型与原始模型具有较高的相似度.

## 5 结论

本文给出了一种基于测量数据的植物建模方法. 首先, 详细描述并验证了所提的特征检测算法. 该算法可以提供每个叶尖或者枝梢的位置. 针对带叶片植物, 本文通过分析局部叶尖几何特性并辅助以有限的用户交互, 能够便捷重建该类模型. 对于纯枝干植物, 本文采用基于 DFMS 分解的方法, 能够成功从含噪声的扫描数据中重建该类植物模型. 本文方法缺陷在于只适用于具有明显尖锐特征的植物.

未来工作主要侧重以下两个方面: 首先, 在体模型上估计尖锐特征的实验结果体现了该算法与表面曲率计算的相似性, 两者的内在关联值得进一步探究; 其次至今仍没有相关算法能够量化评价植物建模质量. 目前, 正在考虑的方法思路为: 比较重建模型与多视点图像 (事先采集而不参与重建过程的那些) 的差别, 具体的衡量算法仍需进一步研究.

## 参考文献


1 Quan L, Tan P, Zeng G, et al. Image-based plant modeling. ACM T Graphic, 2006, 25(3): 599–604

2 Xu H, Gossett N, Chen B. Knowledge and heuristic-based modeling of laser-scanned trees. ACM Trans Graph, 2007, 26(4): 19: 2–13

3 Prusinkiewicz P, Lindenmayer A, Hanan J. The Algorithmic Beauty of Plant. New York: Springer-Verlag, 1990. 1–228

4 Soner I, Day A M. Modeling trees and their interaction with the environment: a survey. Comput Graph, 2005, 29(5): 805–817

5 Reffye P D, Edelin P C, Francon J, et al. Plant models faithful to botanical structure and development. Comput Graph, 1988, 22(4): 151–158

6 Deussen O, Lintermann B. Interactive modeling of plants. IEEE Comput Graph, 1999, 25(3): 56–65

7 Ijiri T, Owada S, Okabe M, et al. Floral diagrams and inflorescences: interactive flower modeling using botanical structural constraints. ACM Trans Graphic, 2005, 24(3): 720–726

8 Qin X, Nakamae E, Tadamura K, et al. Fast photo-realistic rendering of trees in daylight. Comput Graph Forum, 2003, 22(3): 243–252

9 Behrendt S, Colditz C, Franzke O, et al. Realistic real-time rendering of landscapes using billboard clouds. Comput Graph Forum, 2005, 24(1): 507–516

10 Shlyakher I, Rozenoer M, Dorsey J, et al. Reconstructing 3D tree models from instrumented photographs. IEEE Comput Graph, 2001, 21(3): 53–61

11 Han F, Zhu S C. Bayesian reconstruction of 3D shapes and scenes from a single image. In: Bilof R, ed. Proceedings of IEEE Workshop on Higher-Level Knowledge in 3D Modeling and Motion Analysis. Washington: IEEE Computer Society, 2003. 12–20

12 Neubert B, Franken T, Deussen O. Approximate image-based tree modeling using particle flows. ACM Trans Graph, 2007, 26(3): 88: 1–8

13 Tan P, Zeng G, Wang J D, et al. Image-based tree modeling. ACM Trans Graph, 2007, 26(3): 87: 1–7

14 Reche A, Martin I, Drettakis G. Volumetric reconstruction and interactive rendering of trees from photographs. ACM Trans Graph, 2004, 23(3): 720–727

15 Seitz S M, Dyer C R. Photorealistic scene reconstruction by voxel coloring. Int J Comput Vision, 1999, 25(3): 151–173
16 Seitz S M, Curless B, Diebel J, et al. A comparison and evaluation of multi-view stereo reconstruction algorithms. In: Huttenlocher D, Forsyth D, Fitzgibbon A, et al., eds. IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE Computer Society, 2006. 519–562
17 Manh A G, Rabatel G, Assemat L, et al. Weed leaf image segmentation by deformable templates. J Agr Eng Res, 2001, 80(2): 139–146
18 Ma W, Xiang B, Zhang X P, et al. Decomposition of branching volume data by tip detection. In: Marjunath B S, Ortega A, eds. IEEE International Conference on Image Processing. California: IEEE Computer Society, 2008. 1948–1951
19 Zhou Y, Toga A W. Efficient skeletonization of volumetric objects. IEEE Trans Vis Comput Gr, 1999, 5(3): 196–209
20 Bucksch A, Appel van Wageningen H. Skeletonization and segmentation of point clouds using octrees and graph theory. In: Miis J P, Chandler J H, eds. ISPRS Commission V Symposium: Image Engineering and Vision Metrology. Great Britain: Blackwell Publishing, 2006. Vol XXXVI. 1–6
21 Lien J M, Keyser J, Amato N M. Simultaneous shape decomposition and skeletonization. In: Kobbelt L, Wang W, eds. Proceedings of the 2006 ACM Symposium on Solid and Physical Modeling. New York: ACM Press, 2006. 219–228