

# 三维物体表面三角划分的快速算法

周 焰<sup>1)</sup> 李德华<sup>2)</sup> 陈振羽<sup>2)</sup> 胡汉平<sup>2)</sup>

<sup>1)</sup>(武汉空军雷达学院六系自动化教研室, 武汉 430010)

<sup>2)</sup>(图象信息处理与智能控制国家教委开放研究实验室, 华中理工大学图象识别与人工智能研究所, 武汉 430074)

**摘 要** 通过三角划分从三维空间点集重建三维表面的方法, 在实际应用中涉及到的数据量常常很大, 因此对算法的计算速度有较高的要求. 基于三维重建时物体截面轮廓线序列中的相邻轮廓线基本相似的事实, 提出了一种物体截面轮廓线点集之间的三角划分的快速算法. 该算法首先寻找轮廓线的控制点, 并对每两条相邻轮廓线的控制点进行配对, 然后将这样两条轮廓线之间的三角划分, 分配在多个小的曲线段对之间进行, 并且由于事先进行了控制点匹配, 因此, 在每个小的区域中就能够进行简化快速的三角划分. 经理论分析和实验, 该算法与常规的全局优化算法相比, 在视觉效果上基本一致, 可在计算时间上却要少得多, 说明本算法是有效的.

**关键词** 控制点 三维表面三角划分 截面轮廓线 匹配

中图法分类号: TP301.6 TP391.41 文献标识码: A 文章编号: 1006-8961(2000)09-0764-05

## A Fast Algorithm of Triangulation on 3D Surface

ZHOU Yan<sup>1)</sup>, LI De-hua<sup>2)</sup>, CHEN Zhen-yu<sup>2)</sup>, HU Han-ping<sup>2)</sup>

<sup>1)</sup>(Teaching & Research Section of Automation of Airforce Radar Academy, Wuhan 430010)

<sup>2)</sup>(State Commission Research Lab of Image Processing and Intelligent Control, Institute of Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology, 430074 Wuhan)

**Abstract** Triangulation from 3D points set is one of often using ways in which 3D surface is restored. But in practice, it needs to process a huge of data, so it is time consuming. Based on the fact in 3D restoration that most sequential section contours of an object are similar to adjacent ones, a quick method on triangulation between two section contours is proposed in this paper. First, the dominant points of every contour are located, then the dominant points of two contours are matched each other, so the area between two contours is segmented into multi smaller areas. And then triangulation is performed in each small area. Because of the pre-matched dominant points, a simple and quick triangulation can be applied. In this paper, the strategy of triangulation we used is: keep the proportion of the numbers of the dots not used on two curves in accordance with that of whole numbers of the dots on the two curves. Analysis by theory and experiments show that the visual effect of this algorithm is almost the same in computing with common global optimizing methods, and it spends much less time. So the algorithm is effective.

**Keywords** Dominant point, Triangulation of 3D surface of object, Cross-section contour, Match

## 0 引 言

通过三角划分的方法, 由物体表面三维空间点集来重建物体表面是计算几何的一个基本问题, 它实际上是在物体表面进行三角形划分. 目前在计算

机三维重建领域中, 许多三维信息的获取方法均是获得物体表面的平行截面线, 因此在这种情况下, 可将物体表面的重建问题转化为一个由物体的一组平行截面线来重建其三维表面的问题.

由于重建表面的构造并不唯一, 需要从中选择与物体表面接近的解, 因此, 很多人都将其转化为组合优

化问题, 并分别提出了各自的目标函数与相应的重建算法<sup>[1,2]</sup>. 它们按实现方法划分, 大致为两类: 一类是全局优化方法, 例如, Keppel 提出的以重建物体体积最大为目标, 并把三维重建转化为图的最小费用路径搜索<sup>[3]</sup>的算法, 又如 Fuchs 等提出的以重建物体的表面积最小为目标<sup>[4]</sup>; 另一类为基于推理的局部优化方法, 这类方法首先选择轮廓线上的两点以构成初始连接边, 然后选择使某种局部度量最优的两点进行连接, 以逐步构成物体的表面, 如 Christian 等提出的以连接边边长最小为准则<sup>[2]</sup>的算法. 但每个算法都有局限性, Sloan 等指出, 对每个重建算法总可以构造反例, 使重建表面不是人们通常所期望的解. 目前的任何目标函数尚难以保证在所有的情况下都能得到期望解. 另外, 第一类局部优化算法的计算量虽比较小, 但对整体信息没有考虑, 故很难在视觉上或整体评价上得到比较满意的效果; 而第二类全局优化算法的计算量一般比较大, 可是实际的曲面并不一定要求重建后在某种评价上达到全局最优, 故这类方法实用性还有一定问题.

文献[5]将模拟退火法应用于物体表面的三维重建, 即通过有向图路径顶点的局部交换进行求解, 从而得到一种三维重建的统一方法. 但是由于模拟退火法本身对“温度”下降速度有一定要求, 而且实际应用中三维数据一般都很大, 这将使算法达到最优解的时间很长.

文献[6]将遗传算法应用到物体表面的三角划分上, 业已证明, 利用遗传算法具有能够在概率上达到全局最优的优点, 因而较好地解决了物体表面三角划分的问题.

但是, 这两种算法的计算时间都很长, 虽然遗传算法能够并行地计算, 但要求有并行计算机, 故在普通串行机上对实际问题仍难以求解. 基于以上考虑, 为了既能体现整体的要求, 又能提高重建速度, 在两条连续的截面轮廓曲线之间的三角划分的问题上, 我们利用物体两个截面轮廓曲线间一般有较大的相似性的特点, 提出了一种将三角划分限制在局部分块中进行的三维物体表面重建的快速算法.

## 1 基本思想

在三维重建工作中, 常常能够得到物体对象的截面图象, 如 CT 图象等, 这种情况下的三维重建就是首先得到每张图片中物体的截面轮廓曲线, 然后,

在每两条轮廓曲线之间进行三角划分. 由于每相邻两个截面的距离较近, 故得到的两条轮廓曲线大致是相似的, 当然要排除这样的情况, 即两个截面正好处于物体对象表面的突变位置, 这时得到的两条轮廓曲线将不是相似的, 对于这种情况将另外处理.

本文的基本方法是: 首先计算两条轮廓曲线各自的控制点, 然后对两条轮廓线进行控制点配对(对于未匹配上的控制点将不予考虑); 之后, 如果两轮廓线控制点的配对程度小于某一给定阈值, 则说明两轮廓线的相似程度小, 此时对其需采用一般的全局优化方法进行三角划分; 否则, 就可以将两条轮廓曲线按匹配好的控制点分成多个曲线段对, 即两轮廓线之间的区域被分割成多个小的区域; 最后, 在每个小区域中就可以按一般的分割方法进行三角划分. 如图1所示, 其中  $A、B、C、D、E、F、G、H$  是轮廓曲线1的控制点, 而其中  $A'、B'、C'、D'、E'、F'、G'、H'$  是轮廓曲线2的控制点; 这样, 就形成了控制点对  $AA'、BB'、CC'、DD'、EE'、FF'、GG'、HH'$ , 最后在曲线段对  $AB$  与  $A'B'$  之间、 $BC$  与  $B'C'$  之间、 $CD$  与  $C'D'$  之间、 $DE$  与  $D'E'$  之间、 $EF$  与  $E'F'$  之间、 $FG$  与  $F'G'$  之间、 $GH$  与  $G'H'$  之间、 $HA$  与  $H'A'$  之间分别进行三角划分.

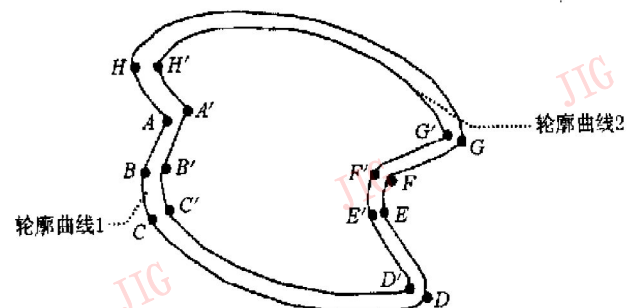


图1 两轮廓线分成多个曲线段示意图

本文中, 由于已经考虑根据轮廓曲线的控制点进行了分组, 即将两条轮廓曲线间的三角划分问题分配在多组曲线段之内进行, 因此就可以将每组内的三角划分算法简化. 这里由于使用了最简单的等比例划分方法, 故可以节约大量的运算时间和运行空间.

## 2 实现方法

### 2.1 控制点的计算

控制点的计算有很多方法, 其中两种主要的方法是角点检测法和多边形近似法, 但是角点检测方法容易漏检控制点, 而多边形近似方法则由于对起始点敏感, 因而会将曲线上不同的点作为起始点, 所以所得到的控制点不会完全一样. 这里使用一种将角点法和

多边形法结合起来的方法<sup>[7]</sup>, 并且为了提高检测速度, 事先对轮廓曲线上所有的点进行了预处理, 在预处理中, 将位于直线中间的点(即直线点)标记出来, 以便在以后进行角点检测和多边形近似时, 都不处理这些点, 因为这些直线点显然不会是控制点.

控制点的检测过程如下:

(1) 断点的标记

为了节省处理时间, 首先要将轮廓曲线上不可能成为控制点的点排除在外, 这里利用曲线的 Freeman 编码对其进行处理. 若进入该点的方向链码与出去方向链码相同, 则该点是直线点, 否则就是断点.

(2) 角点的检测

对曲线上每个断点  $P_i$ , 首先在半径  $k=5$  的邻域内, 计算其曲率  $c_i$ , 这里用  $P_i$  点的偏差  $d_i$  与  $P_{i-k}$   $P_{i+k}$  的弦长  $L_i$  的比来近似  $c_i$

$$c_i = d_i / L_i$$

然后将具有局部极大值曲率的点作为角点(如图 2).

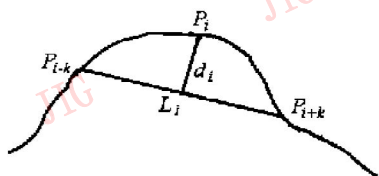


图 2 通过偏差与弦长的比  $d_i/L_i$  来估计  $P_i$  的曲率

(3) 多边形近似

如图 3 所示, 对于上面得到的角点, 可在每两个相邻的角点之间进行多边形近似, 假设这两个角点分别是  $P_s$  和  $P_e$ .



图 3 通过最大曲率来分割曲线

本文的多边形近似方法是这样的:

① 计算曲线  $P_e P_s$  中所有的断点  $P_{(j)}$  的曲率  $c_{(j)}$ , 其中  $s < j < e$

② 将这些曲率从大到小排序, 设:  $c_{(1)} \geq c_{(2)} \geq \dots \geq c_{(m)}$ , 即点  $P_{(j)}$  有第  $j$  大的曲率  $c_{(j)}$ , 而且其偏差为  $d_{(j)}$ , 其中  $m$  是曲线  $P_e P_s$  中断点的个数.

③ 分割的规则是:

如果  $d_{(1)} > T_d$  (这里  $T_d$  是一个预先定义好的阈值, 一般取  $T_d=3$ ), 那么, 就可在  $P_{(1)}$  处分割曲线, 否则检测  $d_{(2)}$ . 如果  $d_{(2)} > T_d$ , 就在  $P_{(2)}$  处分割曲线,

否则就检测  $d_{(3)}$ , 直到条件  $d_{(j)} > T_d$ , 并且  $d_{(i)} \leq T_d$ ,  $i=1, 2, \dots, j-1$  得到满足, 或者不需要进一步分割. 如果没有一个点满足条件, 则该段曲线不分割.

那么过程(2)中的角点和本过程中的分割点就是我们需要的控制点.

(4) 控制点的配对

由于得到的控制点可按在轮廓曲线上的顺序排列, 所以配对后的控制点不能破坏这个顺序, 即控制点的配对不能发生交叉. 设两个轮廓曲线  $C_1, C_2$  上的控制点集分别是  $S_1, S_2$ , 其中  $S_1 = \{p_1, p_2, \dots, p_n\}$ ,  $S_2 = \{q_1, q_2, \dots, q_m\}$ , 假定  $m \geq n$ , 且每个控制点用一个特征向量  $(c, x, y)$  来描述, 其中  $(x, y)$  是控制点的坐标,  $c$  是控制点的曲率, 则

$p_i$  的特征就是  $(c_{p_i}, x_{p_i}, y_{p_i}), i=1, 2, \dots, n$

$q_j$  的特征就是  $(c_{q_j}, x_{q_j}, y_{q_j}), j=1, 2, \dots, m$

这里, 定义  $p_i$  与  $q_j$  之间的匹配程度为

$$d_{ij} = \alpha |c_{p_i} - c_{q_j}| + \beta (|x_{p_i} - x_{q_j}| + |y_{p_i} - y_{q_j}|)$$

其中,  $\alpha$  和  $\beta$  为权系数, 它们的取值是由曲率与距离的相对重要程度确定的. 从上式可以看出,  $d_{ij}$  越小, 则  $p_i$  与  $q_j$  之间的匹配程度越高. 但这里除了考虑匹配程度以外, 另一个必须考虑的因素是不能出现交叉匹配, 即当  $k < i, l > j$ , 如果  $p_i$  与  $q_j$  已经匹配, 那么  $p_k$  与  $q_l$  匹配是非法的. 控制点匹配算法如下:

①  $i=1, j=1$

② 计算  $d_{ik}(k=j, \dots, m)$

③ 选出最小的  $d_{ik}(k=j, \dots, m)$ , 假如是  $d_{ik'}(j \leq k \leq m)$ , 并且  $d_{ik'} < T$  ( $T$  为一预先定义的阈值), 则相应的  $p_i q_{k'}$  配成一对

④  $i=i+1, j=j+1$

⑤ 如果  $i < n$ , 则返回②

⑥ 结束

通过以上的处理, 就将轮廓曲线  $C_1, C_2$  上的控制点集  $S_1$  和  $S_2$  配对了, 同时也去掉了  $S_1$  和  $S_2$  中多出来的控制点. 这里我们假设得到的点对集为  $D = \{(p_1, q_1), (p_2, q_2), \dots, (p_k, q_k)\}$ , 其中,  $k \leq n, m$ , 并且  $p_1, p_2, \dots, p_k \in S_1; q_1, q_2, \dots, q_k \in S_2$ .

2.2 两条轮廓曲线之间的三角划分算法

(1) 相似性判断

判断两个轮廓形状是否相似的方法很多, 对特殊形状, 如多边形一般可用计算角或边的比例的方法; 对于规则的曲线可用参数判决法进行判断; 但用得最多的是模式匹配的方法.

这里需要判断的是两条任意形状的轮廓曲线的相

似情况, 由于上面已获得了它们的控制点, 且对其上的控制点进行了配对, 所以这里可利用两条轮廓曲线控制点的匹配情况来判断这两条曲线是否相似.

令  $N_d = \min\{M, N\}$ ,  $N_D = |D|$  ( $|D|$  表示集合  $D$  中元素的个数), 其中  $M$  为轮廓曲线  $C_1$  中断点的个数,  $N$  为轮廓曲线  $C_2$  中断点的个数;

则匹配率  $P_r = N_D / N_d$

如果  $P_r \geq P_T$  ( $P_T$  为一预设阈值), 则判定  $C_1$  与  $C_2$  相似, 否则为不相似.

### (2) 三角划分

如果判定  $C_1$  与  $C_2$  不相似, 则轮廓曲线  $C_1, C_2$  之间的三角划分可使用一般的全局优化的方法进行, 如引言中提到的遗传算法、模拟退火方法以及 Keppel、Fuchs 等的算法. 如果判定  $C_1$  与  $C_2$  相似, 则将使用下面的等比例划分方法, 以提高三角划分的效率.

在上面得到控制点对的基础上, 对轮廓曲线  $C_1, C_2$  之间的三角划分就可以在每两个连续控制点对之间进行, 因为我们知道, 匹配好的控制点对所构成的连线, 必须是三角划分后的三角形中的边. 这相当于事先通过控制点的匹配已将两个轮廓曲线配准了, 或者说已将两条轮廓曲线进行了预分割. 又由于两条轮廓曲线基本相似, 所以可以在两对连续的控制点对之间使用简易的三角划分算法, 而且不会影响分割的效果, 并能大大提高三角划分的速度.

如图 4 所示, 设  $(p_{di}, q_{di}), (p_{di+1}, q_{di+1}) \in D$  是两个连续的控制点对, 相应地, 在轮廓曲线  $C_1$  上, 设  $p_{di}$  到  $p_{di+1}$  之间有  $m$  个点, 可设这  $m$  个点为  $P_1, P_2, \dots, P_m$ , 并且  $p_{di} = P_1, p_{di+1} = P_m$ ; 同理, 在  $C_2$  上, 设  $q_{di}$  到  $q_{di+1}$  之间有  $n$  个点, 可设这  $n$  个点为  $Q_1, Q_2, \dots, Q_n$ , 并且  $q_{di} = Q_1, q_{di+1} = Q_n$ , 那么, 从  $P_1$  到  $P_m$  的曲线段为  $C'_1$ , 从  $Q_1$  到  $Q_n$  的曲线段为  $C'_2$ .

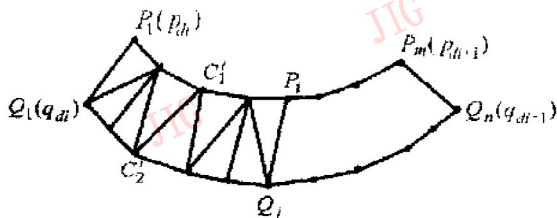


图 4 曲线段  $C'_1$  与  $C'_2$  之间等比例三角划分方法示意图

等比例三角划分方法的基本思想, 就是在曲线段  $C'_1$  与  $C'_2$  之间进行三角划分的过程中, 应保持  $C'_1$  和  $C'_2$  中剩下的点数之比, 与  $C'_1$  和  $C'_2$  中总的点数之比的差别尽量小, 以使  $(m - k) / (n - j)$  与  $m / n$  尽量接近. 其算法如下 (假设  $m \leq n$ ):

① 初始化:  $k = 1; j = 1$ , 连接  $P_1$  和  $Q_1$ ;

② 如果  $\frac{m-k}{n-j} \leq \frac{m}{n}$ , 则  $k = k + 1, j = j$ , 否则  $k = k, j = j + 1$ ;

③ 连接  $P_k$  和  $Q_j$ , 则线段  $P_k Q_j$  与前一条连接线段, 以及  $P_{k-1} P_k$  或  $Q_{j-1} Q_j$  构成一个三角形. 这里, 是与  $P_{k-1} P_k$ , 还是与  $Q_{j-1} Q_j$  构成三角形, 需视第②步的操作情况而定: 如果第②步中有  $k = k + 1$ , 则用  $P_{k-1} P_k$ , 否则, 如果有  $j = j + 1$ , 则使用  $Q_{j-1} Q_j$ ;

④ 如果  $k < m$  并且  $j < n$ , 则返回②;

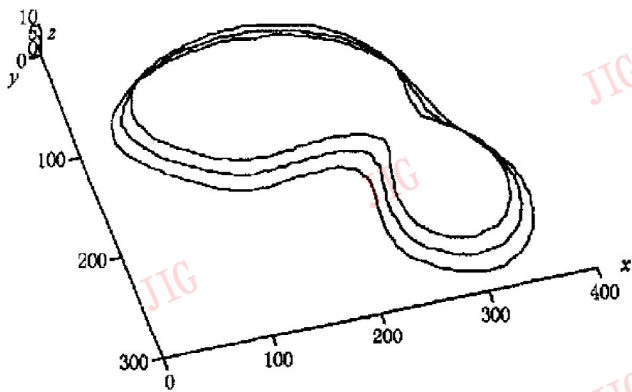
⑤ 将  $C'_1$  (或  $C'_2$ ) 中剩下的点  $P_{k+1}, \dots, P_m$  (或  $Q_{j+1}, \dots, Q_n$ ) 依次与  $C'_2$  (或  $C'_1$ ) 中的最后一个点  $Q_n$  (或  $P_m$ ) 连接成线段, 并与相应的边线和前一条连接线段一起构成三角形, 则结束.

## 3 实验结果和讨论

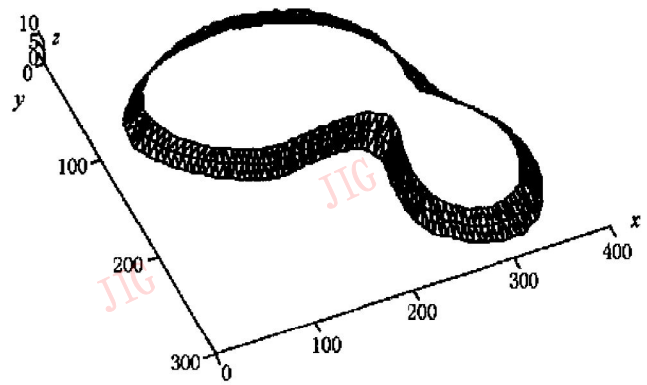
算法中的两个参数  $T, P_T$  是依赖于经验的, 事实上, 它们主要与两个连续轮廓之间的距离、物体表面沿切割垂直方向变化的程度以及轮廓曲线本身的复杂度这三个因素相关, 并且  $T$  还与参数  $\alpha, \beta$  有关. 但在实际工作中, 我们可以将问题分类, 例如在 CT 中可以将人的各部分分别处理, 由于所涉及的  $T$  和  $P_T$  对不同部位是不一样的, 于是可以分别给出相应的经验值, 而对不同的人的相同部分的参数可以是一样的. 具体地说, 如果  $T$  取得太小, 则能够匹配的控制点太少, 从而使初始分段失去意义; 而  $P_T$  取值一般在 0.5 到 1 之间, 如果小于 0.5, 就很难说是相似的了. 为了进一步提高算法的鲁棒性, 一种方法就是根据前述的三个因素具体计算相应的  $T$  和  $P_T$ , 使它们成为自适应参数. 图 5(a) 是某地形等高线图 (相当于物体的截面线轮廓图); 图 5(b) 是使用模拟退火算法, 并以面积最小为全局最优条件进行三角划分的结果; 图 5(c) 标出了等高线图配对后的控制点, 图 5(d) 是使用本文算法得到的结果. 这里控制点匹配算法中  $\alpha = 1, \beta = 2, T = 15$ , 相似性判断中  $P_T = 0.75$ .

从图 5 中可以看出, 本文算法在整体效果上与全局优化算法没有大的区别, 但在局部细节上, 如在轮廓线的控制点处, 本文算法划分比较准确.

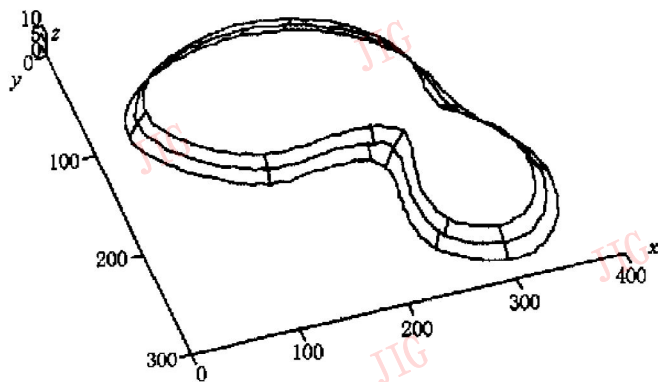
对于两条相似轮廓曲线之间的三角划分, 本算法的计算量主要集中在控制点的处理阶段, 一般情况下它的计算复杂度是  $O(n \log n)$ , 而在极端情况下, 即若所有的点均是控制点的情况, 这时的计算复杂度为  $O(n^2)$ , 由此可见, 计算量主要集中在控制点匹配的地方. 实际应用中, 由于控制点占轮廓线点数



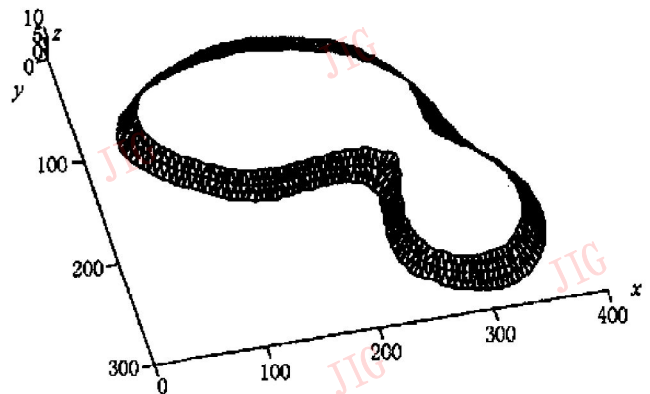
(a) 某处地形等高线图(相当于物体截面轮廓线)



(b) 用模拟退火算法以面积最小为全局最优条件进行划分的结果



(c) 在等高线图中将匹配后的控制点两两连接



(d) 使用本文算法的三角划分结果

图5 某地形等高线间三角划分示例

的比例很小,并且连续的物体,其截面轮廓线大多数是两两近似的,故本算法是实用的.而且本算法克服了传统三角划分的全局优化与问题本身没有必然联系的缺点.

从另一个角度上讲,可以说本算法是在两个层次上进行两条相似轮廓曲线之间的三角划分的,即首先划分出大块,然后在大块内部进一步划分.如果从约束的角度上讲,则本算法是在保证大多数控制点(或关键点)配对的约束条件下所进行的快速三角划分.

### 参考文献

- 1 Choi Y K, Park K H. A heuristic triangulation algorithm for multiple planar contours using an extended double branching procedure. *Visual Computer*, 1994, 10: 372~ 387.
- 2 Christian H N, Sederberg T W. Conversion of complex contour line definition into polygonal element mosaics. *Computer Graphics*, 1978, 12(3): 187~ 192.
- 3 Keppel E. Approximating complex surface interpolation technique for reconstruction 3D objects from serial cross-sections. *CVGIP*, 1989, 48(1): 124~ 143.
- 4 Fuchs I I, Keddem Z M, Uselton S P. Optimal surface reconstruction from planar contours. *Communication of the ACM*, 1977, 20(10): 693~ 702.
- 5 陈凌均, 金建荣, 汪国昭. 三维重建的统一方法——模拟退火法. *计算机学报*, 1997, 20(12): 1133~ 1136.
- 6 周焰, 李德华, 王祖喜等. 三维物体表面三角划分的遗传算法.

中国图象图形学报, 1999, 4A(5): 357~ 362.

- 7 Wu Wen-yen, Wang Mao-jiun. Detecting the dominant points by the curvature-based polygonal approximation. *Graphical Models and Image Processing*. *CVGIP*, 1993, 55(2): 79~ 88.



周焰 1966年生,空军雷达学院六系自动化教研室讲师,博士.主要研究领域为模式识别、计算机视觉、图象压缩、人工智能、多媒体以及密码学.

李德华 1946年生,华中理工大学图象识别与人工智能研究所教授,博士生导师.主要研究领域为模式识别、图象处理、立体视觉、人工智能、思维科学.

陈振羽 1958年生,第二炮兵指挥学院,副教授,华中理工大学图象识别与人工智能研究所博士研究生.研究方向为模式识别与智能系统、图象理解、人工智能、立体视觉.

胡汉平 1960年生,华中理工大学图象识别与人工智能研究所副教授,博士.主要研究方向为模式识别与智能系统、非线性动力学、网络安全.