

等高线图的高度恢复算法研究

李 辉 宋友厉 张建伟

(四川大学图象图形研究所, 成都 610064)

摘 要 用于等高线图到数字高度图(Digital Elevation Map)转换的转换算法是很多应用都需要的算法, 为了提高这一转换的精度与效率, 一个相应的算法——区域内插法被特别提出来, 该算法利用了等高线图固有的特性, 即“图象被等高线分割成多个区域, 每个区域内的边界只有两个等高线值”的特性. 该算法可以在计算机上快速实现. 经过与现有的象限搜索法的实验比较, 由于它更好地利用了等高线图结构上的特点, 致使其在提高速度的同时, 精度上较象限搜索法也有很大的提高, 因而具有很强的实用价值.

关键词 等高线 数字高度图 插值 Delaunay算法 三维地貌

中图法分类号: P208 **文献标识码:** A **文章编号:** 1006-8961(2002)11-1209-05

Algorithms for Elevation Reconstruction from Elevation Contour

LI Hui, SONG You-li, ZHANG Jian-wei

(Institute of Image & Graphics, Sichuan University, Chengdu 610065)

Abstract Algorithms to reconstruct digital elevation map (DEM) from elevation contour is widely used in many applications. In the process of designing a 3D terrain visualization program, where only elevation contour map is available, a new method, region intra-interpolation algorithm, was proposed. It utilizes the structural features of elevation contour images, that is, image is partitioned into many regions by elevation contours and each region is only bound to at most 2 elevation values. Therefore each point inside this region can be calculated as a linear interpolation from the two boundary values. The problem then turns into finding the two shortest distances to its bounding elevation contours. Fast realization is also given out, and this speed up also achieved by special design utilizing elevation contour structures to save time on searching for the closest points. Compare with current algorithm like Quadrant Search algorithm[3], which is design based on a model of random distributing of known value, it shows good performance on both speed and accuracy.

Keywords Elevation contour, Digital elevation map (DEM), Interpolation, Delaunay algorithm, 3D terrain

0 引 言

随着地理信息系统(GIS)^[1]与三维仿真技术的发展, 反映地形特征的数字高度图(DEM)得到越来越多的应用, 因它是生成三维地貌, 计算地面距离、坡度、土方, 进行地形匹配跟踪等工作的基础. 虽然一些新的测绘方法(如遥感), 可通过飞机/卫星激光扫描直接测绘出DEM, 但仍存在获取费用较高, 精

度不一定满足要求等问题. 为了降低成本, 加快DEM生产速度, 现在已有的地形高度资料应该加以利用, 可惜它们大多是以等高线形式表达的. 等高线图作为一种二维图, 虽适合人眼观察, 但对计算机应用, 则不如DEM方便, 为此需探索一种高效、准确的从等高线图恢复出DEM的算法. 通过对已有的算法, 如基于Delaunay划分的算法, 象限搜索法等分析. 提出了一种新的算法——区域内插法, 并开发了一种快速实现技术. 经过实验比较, 该算法表

现出的较高的精度与实用性。

1 基于 Delaunay 划分的算法

文献[2]中介绍了一种基于 Delaunay 三角形剖分的方法,其核心是根据区域内一些已知高度的点来生成 Delaunay 三角形剖分,进而生成 Voronoi 图(图1),然后利用它们进行插值计算.有关 Delaunay 剖分与 Voronoi 图的介绍参见文献[2]、[3].

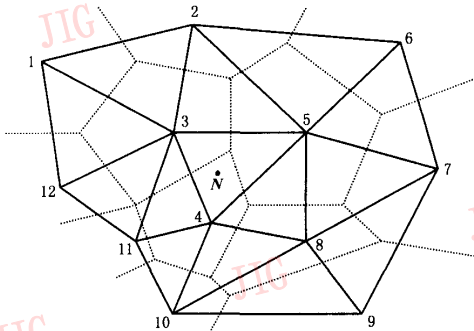


图1 由已知点集构成的 Delaunay 剖分和 Voronoi 图

现假设已生成 Voronoi 多边形,并将与任意一待插点 $N(x_0, y_0, z_0)$ (z_0 为待插值的未知高程值)相邻的 Voronoi 多边形的 Delaunay 三角形顶点设为 $P_1(x_1, y_1, z_1)$, $P_2(x_2, y_2, z_2)$, $P_3(x_3, y_3, z_3)$ (如图1中的3,4,5点),则 N 点与 P_1, P_2, P_3 两两分别构成三角形平面,其中, N 点与 P_1, P_2 点构成的三角平面法线矢量 n 为:

$$n_x = az_0 + b, n_y = cz_0 + d, n_z = e \quad (1)$$

其中

$$\begin{aligned} a &= y_2 - y_1, b = z_2(y_1 - y_0) - z_1(y_2 - y_0) \\ c &= x_1 - x_2, d = z_1(x_2 - x_1) - z_2(x_1 - x_0) \\ e &= (x_1 - x_0)(y_2 - y_0) - (x_2 - x_0)(y_1 - y_0) \end{aligned}$$

将 n_x, n_y, n_z 单位化,并求得所有单位矢量的平均矢量 N 为

$$N_x = Az_0 + B, N_y = Cz_0 + D, N_z = 1 \quad (2)$$

其中

$$\begin{aligned} A &= \sum_{i=1}^n \left(\frac{a_i}{e_i/n} \right), B = \sum_{i=1}^n \left(\frac{b_i}{e_i/n} \right), \\ C &= \sum_{i=1}^n \left(\frac{c_i}{e_i/n} \right), D = \sum_{i=1}^n \left(\frac{d_i}{e_i/n} \right) \end{aligned}$$

由有限元中的“最小曲率原则”可知,任意一个三角形平面的单位矢量与平均矢量之差的平方和应达到最小,可得式(1)中的 z_0 为

$$z_0 = \frac{\sum_{i=1}^n \left[\left(A - \frac{a_i}{e_i} \right) \left(\frac{b_i}{e_i} - B \right) + \left(C - \frac{c_i}{e_i} \right) \left(\frac{d_i}{e_i} - D \right) \right]}{\sum_{i=1}^n \left[\left(\frac{b_i}{e_i} - B \right)^2 + \left(\frac{d_i}{e_i} - D \right)^2 \right]} \quad (3)$$

按照上式计算出的每个网格节点的值,其所组成的二维阵列就形成数字地形模型。

该算法虽可用于任意已知地形数据点集的原高度图恢复,但是,这其中 Delaunay 剖分的生成算法非常复杂,即便使用快速算法仍比较耗时,这还不包括用最小曲率原则来计算插值的时间。

另一个问题是,该算法并不适用于已知点分布在同一等高线上的情况,因为一条等高线上的点互相紧邻,而不同等高线上的点又相距较远,用该算法进行 Delaunay 剖分,将生成大量的细长三角形,另外 Voronoi 图也将是非常奇异的,而且用这种图去推算插值并不能发挥该算法最优化的特性.事实上,该算法更适合已知点呈均匀概率分布的情况,或者说仅适用于已知点呈稀疏分布,且互相粘连较少的图象,因而本文中放弃了使用该算法。

2 象限搜索法及快速实现

文献[4]中介绍了一种基于象限搜索的方法.该算法可简单表述为,对于一个未知高度值的点 N_{ij} ,以它为中心分别在4个象限中,找出离它最近的4个已知高度点 N_1, N_2, N_3, N_4 ,其高程值为 z_1, z_2, z_3, z_4 它们到点 N_{ij} 的距离分别为 r_1, r_2, r_3, r_4 . N_{ij} 点的高程值 z_{ij} 受到 N_1, N_2, N_3, N_4 这4个点的高程值的影响,且其影响程度分别与 r_1, r_2, r_3, r_4 成反比.即

$$z_{ij} = \sum_{k=1}^4 \frac{\frac{1}{r_k} z_k}{\sum_{k=0}^4 \frac{1}{r_k}} \quad (4)$$

如果对每个点都进行搜索,那么对所有已知点都寻找各象限内的最近点,则将非常费时,因而在文献[3]中又提出了一种快速实现的方法,该方法就是先将待处理的图象分成 $M \times N$ 的块阵列,并统计出每块中已知高程值的点,再判定每个未知高程值的格网节点 $N_{i,j}$ 所在的块;然后在当前块内搜索以 $N_{i,j}$ 为中心的4个象限内,是否有已知高程值的点,如果4个象限都存在这样的点,则由式(4)计算出 $N_{i,j}$ 点的高程 $z_{i,j}$,否则需先确定哪个象限内无满足条件的点,然后再沿该象限方向到与当前块相邻的块中去搜索,但是,这种快速算法的缺点是不一定能在每个象限中找

到一个最近的点(如图 2 所示),因为在一象限中,离待插值点 N 最近的点本应为 P ,但实际结果却是 Q 点.本文改进了该快速算法,使其搜索所有可能找到最近点的块,而不是一旦找到一个点便停止搜索.以后的实验中,均采用改进后的快速算法.

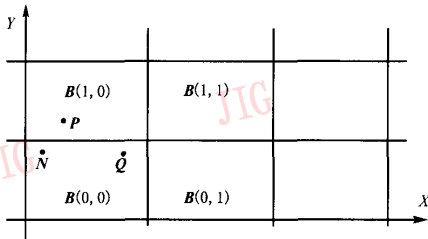


图 2 对 N 点进行搜索,在第 1 象限方向先找到块 $B(0,0)$ 中的 Q 点,实际应该是 P 点



(a) 等高线图

在实践中还发现,这种象限搜索法有一定的弊端(如图 3 所示),即当待插入点在 N 点时,其第 1 象限的最近点在 P 点附近.而当待插点在 N 点向上一个像素的 M 点时,决定其值的第 1 象限的最近点会跳变到 Q 点,由于点 P 与点 Q 距离较大,因而出现一个不应当出现的跳变,其最后恢复出的 DEM 图象如图 4(b)所示(图 4(a)为其等高线图).实践证明,由这种原因引起的伪形状对实际应用是非常不利的.

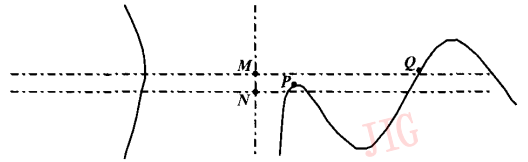
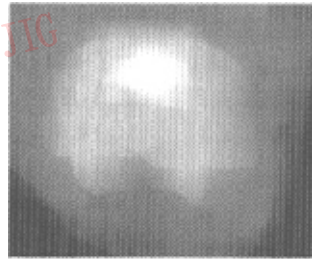
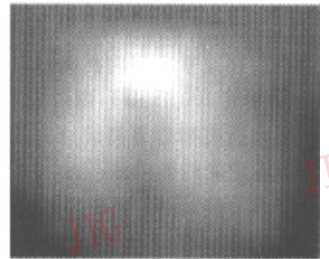


图 3 相邻插值点的各象限参考点可能产生跳变



(b) 象限搜索法生成的 DEM 图象



(c) 区域内插法生成的 DEM 图象

图 4

3 区域内插法及快速实现

通过对等高线图特点的分析可以发现,一幅图象被等高线划分为多个未知高度值的区域,而每个区域的边界要么是图象边界,要么是一条或两条等高线,或者说,与这个区域邻接的等高线值不会超过两个.对于任意待插值点 N 只要在包围其所在区域的两个不同等高线上,找出与 N 最接近的两个点,设为 P_1, P_2 ,设它们对应的高程值为 z_1 和 z_2 ,它们与 N 点的距离为 D_1, D_2 ,则 N 点的高程恢复值 z 为

$$z = \frac{z_1 \times D_1 + z_2 \times D_2}{D_1 + D_2} \quad (5)$$

由此不难看出,用这种算法,图 3 中, N 点和 M 点的高程值不会出现跳变.从理论上分析,两条等高线之间的一个待插值点的最优取值,实际上取决于它周围等高线的分布与走势情况,如果以线性插值作为算法的基础^[5],则可以只考虑包围待插值点的

两条等高线;进一步可以认为待插点的取值是由它到这两条等高线的距离决定.这种方法虽然只是一种线性近似,并不能保证最优,不过后面的测试将证明它相对于象限搜索法还是有很多优越性.

在实践中,由于发现通过直接遍历 N 点所在区域所有相邻的等高线点去找 P_1 与 P_2 点,非常费时.于是本文也提出了一种快速算法,与上节中提到的分块方法不同,该算法是按由近到远的顺序直接在图象上搜索最近的等高线点,直到找到前文所述的 P_1 与 P_2 点为止.在图象网格中,由于一个点,其邻域点由近到远的顺序及它们到这一点的距离是固定的,因此可以将它们预先算好,并以相对位置与不同距离的形式存储在一个大型数组中,这个预存搜索数组的形式如下:

- $(-1,0,1.0),(1,0,1.0),(0,-1,1.0),(0,1,1.0),$
- $(-1,-1,1.414),(1,-1,1.414),(-1,1,1.414),$
- $(1,1,1.414),(-2,0,2.0),(2,0,2.0) \dots$

这样,以后就不需要重复计算,只需依次判断这些相邻点是否是一个等高线点即可.这就省去了更

耗时的浮点数运算. 实际上, 这种算法还可以加速, 若 N 点和 M 点相邻, 并设在已知等高线中, 相对 N 点, 两个已知高程值的最近点分别为 P_1 和 P_2 , 其与 N 点的距离分别为 D_1 和 D_2 , 不难推出, 与 M 点相邻最近的两个等高线点 P'_1, P'_2 , 其到 M 点的距离 D'_1, D'_2 满足下式

$$D_1 - 1 \leq D'_1 \leq D_1 + 1$$

$$D_2 - 1 \leq D'_2 \leq D_2 + 1$$

这样在对 M 点进行由近到远地搜索 P'_1, P'_2 时, 可以分别只从距离为 $D_1 - 1$ 和 $D_2 - 1$ 的点开始搜索, 由此又可以大幅提高运算的速度. 在算法当中, 预存搜索数组的大小与搜索距离的平方成正比, 以搜索距离为 100 个像素为例, 其所占存储空间为 320K, 这对现有的计算机系统来说, 完全可以承受, 而 100 个像素的预存搜索距离, 对绝大多数情况来说, 已可以覆盖绝大多数待插值点了, 而对于极少数不能在预存搜索范围内找到的点(它到最近等高线的距离大于 100pixels), 则可以再通过遍历该点所在区域的所有相邻等高线点去找最近的等高线点, 这对运算速度的影响并不大.

下面给出简单的算法描述:

(1) 对一图象上未求出高程值的待插值点 N , 先进行 4 邻域种子点充填, 并记录下充填过的点列表 L , 同时在列表 L 中记下每个点由前面哪一个点延伸而来, 然后记下边界点列表 B (等高线点), 并在

待处理图中, 为每个列表 B 中的点打上标记.

(2) 对 L 中每个点:

① 根据前面描述的方法, 按预存搜索顺序寻找最近的等高线点 P_1, P_2 , 这里需要注意, 对每个点, 要考虑其前面一个邻域点到两条等高线的最近距离 D_1, D_2 , 然后只从 $D_1 - 1, D_2 - 1$ 开始搜索.

② 若在预存搜索范围内找不到要求的等高线点, 则换用遍历 B 的方法求出 P_1, P_2 .

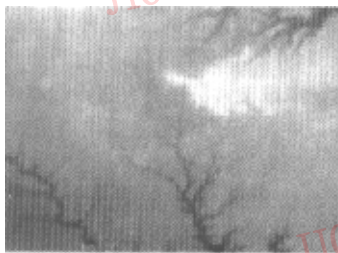
③ 按式(5)求出待插值点的高程值, 并在 L 中记下它到相邻两条等高线的距离 D_1, D_2 .

(3) 清除 L 和 B , 转步骤 1 直到所有待插点都外理完为止.

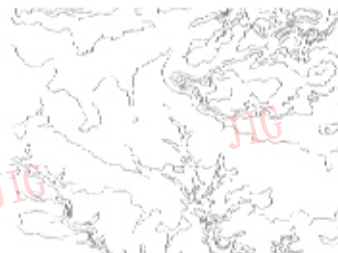
4 实验对比

为了对比象限搜索法与区域内插法的效果, 选取了一幅 DEM 图象(640×480)及由其生成的等高线图, 用不同算法进行了恢复 DEM 图象的实验, 图 5(a)和图 5(b)为本文测试用的 DEM 和相应的等高线图; 然后, 用不同的算法来恢复 DEM, 其恢复 DEM 的速度与效果见表 1. 由表 1 及图 5(c), 图 5(d)可以看出, 区域内插法的效率较高, 不但失真较小, 且不会出现伪形状.

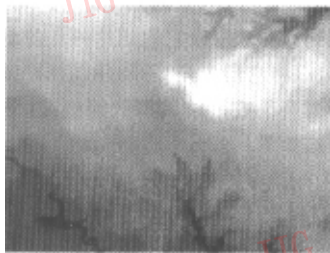
测试条件为: 处理器: Pentium 600E, 内存 128M, 操作系统为 Windows 2000.



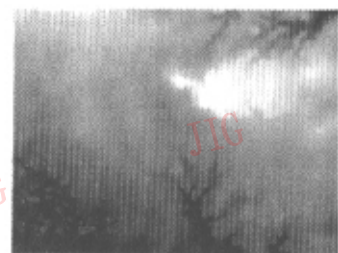
(a) 原 DEM 图象



(b) 原图的等高线



(c) 象限搜索法生成的 DEM 图象



(d) 区域内插法生成的 DEM 图象

图 5 用不同算法生成的数字高程图象(DEM)

表 1 象限搜索法与区域内插法结果对比

	象限搜索法 (不用快速算法)	象限搜索法 (用快速算法)	区域内插法 (不用快速算法)	区域内插法 (用快速算法)
用时(s)	719.334	112.041	264.199	12.487
均方差	17.654 334	17.683 342	13.271 815	13.270 572
信噪比(dB)	45.963 750	45.947 239	48.796 140	48.797 059

注:采用 100 像素的预存搜索距离,算法用时不包括预先生成搜索数组的用时 7.8s.

5 总 结

本文介绍了一种用于从等高线恢复出数字高度图的算法——区域内插法,并研究出了其在计算机上快速实现的技术.一般现有算法大都更适合于用离散已知高度点,对数字高度图进行插值恢复,而区域内插法则由于利用了等高线图上,已知高度点分布的独有属性,故而可以大大提高高程插值的速度与精度,有一定的实用价值.为了使该算法能更好的应用,应进行如下两个后续的工作:(1)由于采用区域内插,所以当一个区域的边界线仅有一条时(例如在山顶处),其插值结果必然退化成等值填充,此时应增加外插处理;(2)由于等高线的断线或缺失,可能造成本区域内插法的前提条件(即每一区域只有最多两个边界点取值)不能成立,从而无法应用,此时应增加自动修复功能或修正算法,以使其达到实用的目的.

参 考 文 献

- 1 Teffrey Star, Jone Estes. Geographic information system—An introduction [M]. Englewood Cliffs, New Jersey, USA: Prentice Hall, 1990.
- 2 Macedonio G. An algorithm for the triangulation of arbitrarily distributed points: Applications to Volume Estimate and Terrain Fitting[J]. Computers & Geosciences, 1991, 17(7): 859~874.

- 3 Victor J D Tsai. Fast topological construction of delaunay triangulation and voronoi diagrams [J]. Computers and Geosciences, 1993, 19(10): 345~350.
- 4 王永明, 林行刚. 一种快速 DEM 生成算法[J]. 计算机应用与软件, 1998, 15(4): 28~33.
- 5 Nagy, Wagle. Geographic data processing [J]. Computing Surveys, 1979, 11(2): 139~181.



李 辉 1970 年生,副教授,1997 年获加拿大 Simon Fraser 大学计算机硕士学位,1998 年至今在四川大学计算机系图象图形研究所任教.主要从事实时仿真系统、图形系统及 GIS 系统的研究与开发.



宋友厉 1976 年生,2002 年获四川大学计算机硕士学位,现在深圳华为公司工作.



张建伟 1972 年生,副教授,1998 年获四川大学计算机硕士学位,现在四川大学计算机系图象图形研究所任教.主要从事实时仿真系统、图形系统及 GIS 系统的研究与开发.