

基于绘制-检出的矢量数据栅格化方法研究

李青元¹ 王 涛¹ 朱菊芳^{1,2} 张福浩¹

(1 中国测绘科学研究院,北京市海淀区莲花池西路 28 号,100830)
(2 中国矿业大学(北京)北京地球科学与测绘工程学院,北京市海淀区学院路丁 11 号,100830)

摘 要:对于前人提出但未被重视的绘制-检出栅格化方法进行了阐述,给出了其实现步骤与关键语句,并赋予其一个新的名称——基于绘制-检出的矢量数据栅格化方法。对该方法实现中的一些技术问题,如正确性、精度、内存位图大小限制、分块栅格化、栅格化效率等问题进行了深入的探讨。笔者认为,该方法可以成为一种普适的工业方法。
关键词:矢量数据栅格化;掩膜法;颜色充填法;GDI;绘制-检出
中图法分类号:P208

矢量与栅格是地理空间数据的两种基本类型,这两种数据类型的相互转化几乎是所有 GIS 系统都会遇到的问题,并被认为是 GIS 的关键技术问题之一^[1]。矢量数据栅格化就是将矢量数据转化为栅格数据,简称栅格化。矢量数据栅格化常用于以下场合:① 一些用矢量方法实现非常复杂,但用栅格方法实现非常简单的空间分析(如叠加分析、缓冲区分析),由矢量数据转换成栅格数据进行运算,再将结果转回到矢量数据^[2];② 将传统观念中认为是矢量方法长项的网络分析也转化成栅格方式后,用注水模型、模拟退火模型进行演算后转回到矢量类型^[3,4];③ 虚拟地理环境中,常常需要将矢量要素转化到影像栅格中进行显示;④ 统计数据通常是以行政区划进行统计的,而很多空间分析需要将这些统计数据分摊到地理格网上去。总之,由于栅格数据具有“0”初始化的优势^[5],矢量数据栅格化是 GIS 平台必备的功能之一。

矢量数据栅格化的方法很多,大致可分为两类:基于解析的方法和基于绘制-检出的方法。常用的解析方法有 DDA 法(数字微分分析方法)^[1,6,7]和 Bresenham 法^[1,6-8],后者不仅速度快、效果好,而且可以在理论上证明它是目前同类算法中最优的^[6]。线要素栅格化原则上可分为四方向栅格化、八方向栅格化、全路径栅格化和恒密度栅格化^[9]。笔者将内部点扩散法、射线法、扫描法^[1,6-8,10-12]、边填充算法^[1,6]、复数积分法^[1,8]、边

界代数法^[1,8,11,13]、改进的边界代数法^[1]、无边界游程编码边界跟踪法^[14]、差分边界标志与累加扫描^[15]等线要素、面要素栅格化方法都归为基于解析的方法,是因为它们都需要对线段或多边形边界线段进行求交或扫描运算。本文重点讨论基于绘制-检出的矢量数据栅格化方法。

1 基于绘制-检出的矢量数据栅格化方法

在没有计算机之前,人们对地图栅格化是在地图上打上规则的方格网,然后在地图上读出每一个网格单元所属的区域,当一个网格跨多个区域时,取这个网格的中心点或面积占优的区域的属性值,这可以视为原始的绘制-检出栅格化方法。

文献[10,11]都是摒弃边界追踪、扫描、求交,而采用向实际或虚拟屏幕“绘制”,然后“检出”各像元的颜色(或灰度)值,并转变为栅格单元的属性值。这本是一种更为简单、有效的方法,虽多次被当作不同的方法引用^[1,12,13],但其优越性在很长的时间内没有被大多数同行所认识。在笔者所收集到的所有有矢量数据栅格化的 GIS 算法教科书中^[6-9,14]都没有将其列入,可见该方法并未被同行广泛接受。

以 VC 语言环境下 GDI 图形库的面要素栅格化为例,详细说明该方法的实现步骤和关键语

句。① 根据矢量数据的范围和栅格化的栅格单元代表的实地距离,计算栅格化区域的栅格行列数 m,n ;② 确定面要素的绘制颜色对照表和要素编码对照表;③ 用 CDC 类的 CreateCompatibleBitmap(pDC, m,n) 命令,创建一个大小为 $m \times n$ 的内存位图 memBitmap,该 memBitmap 是 CBitmap 类的实例;④ 由 memBitmap 执行 CreateCompatibleDC(pDC) 命令,创建一个内存绘图设备 memDC,该 memDC 是 CDC 类的实例,用 memDC 执行 SelectObject (memBitmap),将 memBitmap 选入 memDC 中;⑤ 根据矢量数据的空间范围 (mapWidth \times mapHeight) 和内存位图尺寸 ($m \times n$),计算由地图空间向内存位图空间映射的绘图比例 sclae;⑥ 用 scale 为映射比例,将多边形边界坐标映射为内存位图坐标 (注意,计算映射坐标时 $+0.5$ 再取整),由 memDC 调用多边

形“绘制”命令,向内存位图 memBitmap 上绘制面要素;⑦ 由 memBitmap 的 GetBitmapBits 方法获取内存位图的指针,再通过行、列两重循环获取各像素对应的颜色值 (也可以由 memDC 用 getImage(i,j) 获取,但效率要低得多),然后根据绘图时为各要素所设的颜色对照表换算成各栅格单元的数据场值,存入到一个 $m \times n$ 整型数组的 (i,j) 单元中。循环结束,栅格化即告完成,结果存在该 $m \times n$ 数组中;⑧ 将 $m \times n$ 数组的值存入标准栅格数据交换格式文件;⑨ 为了方便使用,重新设置栅格单元的边长 (如图 1 为 16 像素),绘制各栅格的颜色,标注要素编号或类型编号,叠加原始面要素边界线或线要素中心线,最后存入图像文件。

上述过程中使用的 GDI 在 GDI+、Java 图形库以及其他语言的图形库中均有类似功能的类或函数库。

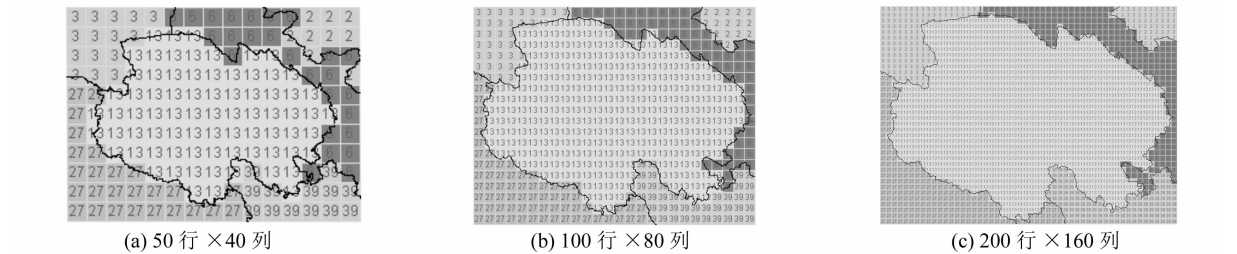


图1 用不同的格网对某区域栅格化后的局部
Fig. 1 A Prat of a Area Rasterization with Different Nets

2 讨论

2.1 正确性问题

由于绘制时使用的是“画家算法”,即对同一位图像素点可能会被不同的要素重复“绘制”好几次,只有最后一次的绘制值能够被“检出”,因而各个不同图层、不同要素“绘制”的顺序对最后结果是有影响的。为了避免上述相互“覆盖”的问题,需要分层“绘制-检出”,并存入不同的栅格图层数组。

地图的面要素配色时,广泛使用地图的三色原理,不同属性的区域可能充填了相同的颜色,因而检出的同一种颜色可能对应多个不同的多边形。有两种方法可以解决这个问题:① 绘制时,不采用地图原来配置的颜色,而采用要素的顺序号 (惟一的标识码) 或用户标识码 (同样的省份或相同属主的地块具有相同的标识码) 为颜色值进行绘制,检出后,绘制各栅格的颜色时,再采用地图原配的颜色值;② 对于要素逐个地“绘制-检出”,即一次只绘制一个要素后,就对要素所在的外接矩形框内的栅格进行检出,并将检出结果叠

加到总结果中,这种方法可以适应大数据量、高精度的栅格化的需要,但程序结构较复杂。

2.2 精度问题

对于很多应用来说,矢量数据栅格化的精度问题是至关重要的。下面分别使用 50×40 、 100×80 、 200×160 的栅格尺度对中国省级行政区划进行栅格化后取某区的结果。根据实验结果得出以下结论:① 在多边形边界处,各栅格单元归属是遵从“中心点归属法”,并基本符合面积占优的原则^[12],尽管在个别单元处还有归属不合理的现象,但基本可以接受;② 随着栅格单元的缩小,栅格化的精度随之提高;③ 笔者交换绘制多边形的顺序,得到的绘制结果是一样的,说明同一拓扑层内相邻的多边形“绘制”的先后顺序不同,“检出”形成的栅格化结果是一样的。

2.3 内存位图的大小限制问题

从上面的实验可以看出,要提高栅格化的精度,就要增大内存位图的 $m \times n$ 值,因而系统能够申请的内存位图的大小是提高栅格化精度的瓶颈。根据实验,用 VC 在 1 G 内存的机器上允许开出的彩色内存位图的大小是 $4\,090 \times 4\,090$ 左

右,黑白内存位图的大小是 $24\ 000 \times 24\ 000$;在 3.25 G 内存的机器上能够开出的彩色内存位图的大小是 $8\ 050 \times 8\ 050$ 左右,黑白内存位图的大小是 $46\ 000 \times 46\ 000$ 左右。显然,计算机能开出的内存位图的大小是有限的,但基本可以满足通常的栅格化需要,尤其是当设置成黑白影像时,能达到 $46\ 000 \times 46\ 000$ 的量级。如果欲栅格化区域的量级还大于这个量级,就需要采用下面的分块栅格化或设置虚拟内存。另外还需要注意,在 § 1 第⑨步可视化时,需重新设置栅格单元大小(如 16 个像素,为在单元上写要素编号),如果还是绘在内存位图上,这时所需创建的内存位图的大小是 $(m \times 16) \times (n \times 16)$,可能会超出彩色位图的限制而需采用单色位图,也可采用不在内存位图绘制,直接向屏幕绘制的方法,使单元大小随放大倍率进行调整。

2.4 分块栅格化方法

分块栅格化是为了解决由于栅格细分后可能出现的计算机内存不足而引入的一种方法。分块的方法有两种:① 分块基于空间域,即将欲栅格化的区域划分为若干规则方块,使每一块区的栅格数量在计算机内存所能开出的内存位图范围之内,然后分块进行矢量数据栅格化,这种方法尤其适合原本就分幅存储的空间数据,最好一个图幅作为一个分块。② 分块基于要素,即前面正确性讨论时说的第二种方法,一次“绘制-检出”一个面要素或线要素。

2.5 效率测试

以 1.73 主频 CPU/1 G 内存的 Dell Inspiron 6000 便携式计算机进行实验。对全国 1:400 万的省/地/县级行政区划(面状)、铁路/公路和经纬网(线状)的矢量数据(大地坐标)进行栅格化,用 $1\ 000 \times 800$ 网格(可视化时,栅格单元边长为 16 像素,位图大小为 $16\ 000 \times 14\ 400$,只能用单色位图),各层均耗时约 52 s(包括栅格化和 § 1 第⑨步生成可视图像文件);用 310×250 网格(可视化时用彩色位图),耗时约 13 s;用 100×80 网格,耗时约 2 s。当用全国 1:100 万 76 个图幅的行政区划、公路、经纬网 3 个图层的矢量数据(为经纬度坐标)进行栅格化时,每个图幅用 320×240 的栅格(1:100 万图幅为 $6^\circ \times 4^\circ$,即每个栅格单元边长 $1'$),各层耗时相近,约为 16 min 30 s,即平均每幅耗时 13 s。总结实验结果可以得出,不同复杂程度的矢量数据栅格化耗时差不多,若栅格化所用的行列数不同,则耗时差别明显。

3 结 语

矢量数据栅格化的方法中,自动化程度和结果的合理化程度是至关重要的。基于“绘制-检出”过程的矢量数据栅格化方法虽然早就被提出,但还没发展成为矢量栅格化的一种通用方法。其中还有很多的技术细节需要根据不同的栅格化目的及其栅格化准则进行调整。但笔者相信,这是一种很有前途的方法,值得进一步讨论。

参 考 文 献

- [1] 王建,杜道生. 矢量数据向栅格数据转换的一种改进算法[J]. 地理与地理信息科学, 2004, 22(1): 22-23, 33, 44
- [2] 杨存建,张增祥. 矢量数据在多尺度栅格化中的精度损失模型探讨[J]. 地理研究, 2001, 20(4): 416-421
- [3] 王海军. 基于注水模型的网络分析方法研究[J]. 武汉大学学报·信息科学版, 2007, 32(5): 23-24, 124, 224
- [4] Albert P. Finding Minimal Cost Paths in Raster Geographic Information System Map Representations, Genetic Algorithms, Simulated Annealing and Tabu Search[D]. USA: Kent State University, 2004
- [5] 胡鹏,杨传勇,胡海,等. GIS 的基本理论问题——地图代数的空间观[J]. 武汉大学学报·信息科学版, 2002, 27(6): 616-621
- [6] 郭伦,刘瑜,张晶,等. 地理信息系统原理方法和应用[M]. 北京:科学出版社, 2001: 150-154
- [7] 吴立新,史文中. 地理信息系统原理与算法[M]. 北京:科学出版社, 2003
- [8] Bresenham J E. Algorithms for Computer Control of a Digital Plotter[J]. IBM Systems Journal, 1965, 4(1): 25-33
- [9] 张宏,温永宁,刘爱利,等. 地理信息系统算法基础[M]. 北京:科学出版社, 2006: 70-77
- [10] 陈艳. 地图矢量数据栅格化的一种简便方法[J]. 江苏测绘, 1994(1): 9-12
- [11] 黄波,陈勇. 矢量、栅格相互转换的新方法[J]. 遥感技术与应用, 1995(3): 61-65
- [12] 吴华意,龚健雅,李德仁. 无边界游程编码及其矢栅直接相互转换算法[J]. 测绘学报, 1998, 27(1): 63-68
- [13] 王晓理,孙庆辉,江成顺. 面积误差最小约束下矢量数据向栅格数据转换的优化算法[J]. 测绘学报, 2006, 35(8): 471-472, 672, 772
- [14] 胡鹏,黄杏元,华一新. 地理信息系统教程[M]. 武汉:武汉大学出版社, 2002: 110-114

第一作者简介:李青元,博士,研究员,主要从事矿山、地质三维空间数据模型与方法以及地理信息标准研究。

E-mail: LiQY@casm. ac. cn