

计算机地图制图

Computer Aided Cartography

艾自兴 龙毅 编著

武 汉 大 学 出 版 社

前 言

计算机地图制图是随着计算机技术在地图学领域中的应用而发展起来的新技术,是伴随着计算机及其外围设备的发展而产生和发展起来的,已成为多学科集成并应用于各领域的基础平台,成为地学空间信息表达的基本手段和工具。它的理论、方法和技术得到了广大地学、信息技术工作者及相关专业学生的青睐。因此,简明阐述计算机地图制图的基础理论和方法,为深入学习和发展计算机地图制图技术奠定良好基础是本书的基本出发点。

本书系统地介绍了计算机地图制图的基本原理与方法,并结合国内外该领域技术应用的最新进展和作者长期的教学、科研成果,进一步阐述了计算机地图制图系统的开发与应用思路。在编写过程中,本书力求内容具体、由浅入深、循序渐进,理论与应用并重。全书共分为七章。第一章介绍计算机地图制图的基本过程、系统构成和相关科学技术;第二章介绍地图数据的描述方法,地图的数据结构及其相互关系;第三章描述地图数据的分类编码、采集及地图数据库;第四章论述矢量、栅格两种基本地图数据类型的方法;第五章介绍可视化、符号化及地图符号库;第六章从软件工程的角度讨论计算机地图制图系统软件开发的基本方法和过程,并分别应用结构化方法和面向对象方法进行了计算机地图制图系统的分析、设计与程序实现;第七章介绍计算机地图制图系统应用的一般流程和常用的计算机地图制图软件。

本书可作为普通高等学校测绘、地理、地质、城市规划、土地管理、区域开发、资源与环境等专业本科生和研究生教材,也可供从事相关工作的科技人员和管理人员参考。

本书的出版,得到了武汉大学资源与环境科学学院的大力帮助与支持。杨新文、王霜、贾玉明、牛瑞芳等研究生也参与了本书的资料收集与文字整理工作,在此表示衷心的感谢。

由于作者水平所限,书中不足之处在所难免,敬请读者提出宝贵意见。

作 者
2005 年 8 月

目 录

第1章 绪 论	1
1.1 计算机地图制图概述	1
1.1.1 计算机地图制图的概念	1
1.1.2 计算机地图制图的基本过程	2
1.2 计算机地图制图的发展	3
1.2.1 国际发展状况	3
1.2.2 国内发展状况	4
1.3 计算机地图制图系统的构成	5
1.3.1 硬件	5
1.3.2 软件	9
1.3.3 地图数据	10
1.4 计算机地图制图与相关科学技术	12
1.4.1 与计算机地图制图相关的科学技术	12
1.4.2 计算机地图制图与相关科学技术的联系与区别	12
第2章 地图数据结构	14
2.1 地图数据的描述方法	14
2.1.1 地图对地理空间的描述	14
2.1.2 地图数据的基本特征	21
2.1.3 地图数据的基本类型	21
2.2 地图的数据结构	23
2.2.1 矢量数据结构	24
2.2.2 栅格数据结构	34
2.3 两种数据结构的比较及转换	42
2.3.1 两种数据结构的比较	42
2.3.2 两种数据结构的转换	44
第3章 地图数据的采集和地图数据库	51
3.1 地图数据源及数据分类编码	51
3.1.1 数据源的种类	51
3.1.2 数据的分类编码	52
3.2 地图数据的采集	55

3.2.1	几何数据的采集	56
3.2.2	属性数据的采集	57
3.3	地图数据的编辑和数据质量分析	58
3.3.1	数据编辑	58
3.3.2	数据质量分析	59
3.4	地图数据库	61
3.4.1	数据库的概念	62
3.4.2	数据模型	64
3.4.3	地图数据库的设计与建立	72
第4章	地图数据处理	74
4.1	矢量数据处理	74
4.1.1	数据预处理	74
4.1.2	图形编辑	79
4.1.3	数据变换	82
4.1.4	曲线光滑	85
4.1.5	矢量符号的生成	90
4.1.6	数字地形模型	94
4.2	栅格数据处理	98
4.2.1	栅格数据运算	99
4.2.2	栅格数据处理应用	104
第5章	地理信息可视化	108
5.1	可视化的概念与形式	108
5.1.1	可视化的概念	108
5.1.2	可视化的形式	109
5.2	符 号 化	109
5.2.1	地图符号	110
5.2.2	符号化的方法	112
5.3	地图符号库	112
5.3.1	符号库设计的原则	112
5.3.2	矢量符号库	113
5.3.3	栅格符号库	115
5.3.4	符号库的引用	115
5.4	电 子 地 图	119
5.4.1	电子地图的基本特征	119
5.4.2	电子地图的设计与制作	121
5.4.3	电子地图集的设计与制作	123
5.4.4	电子地图集系统的开发	124

第 6 章 计算机地图制图系统的软件开发	128
6.1 CAC 系统开发的基础知识	128
6.1.1 系统开发的基础条件	128
6.1.2 系统开发过程	129
6.1.3 软件开发模型	131
6.1.4 软件设计与实现方法	132
6.2 结构化方法在 CAC 系统开发中的应用	135
6.2.1 系统分析	135
6.2.2 系统设计	136
6.2.3 系统的软件实现	149
6.3 面向对象方法在 CAC 系统开发中的应用	166
6.3.1 CAC 系统的面向对象分析设计	167
6.3.2 CAC 系统的面向对象程序实现	175
第 7 章 计算机地图制图系统的应用	180
7.1 地图制图的一般过程	180
7.1.1 地图设计和编辑准备阶段	180
7.1.2 地图编稿和编绘阶段	180
7.1.3 地图整饰和制印阶段	181
7.2 CAC 系统的应用流程	181
7.2.1 编辑准备	181
7.2.2 数据采集	185
7.2.3 数据处理	186
7.2.4 图形输出	188
7.3 常用 CAC 软件介绍	188
7.3.1 CorelDraw 软件	188
7.3.2 AutoCAD 软件	189
7.3.3 ArcGIS 软件	190
7.3.4 MapInfo Professional 软件	191
7.3.5 Genamap 软件	194
7.3.6 SICAD 软件	195
7.3.7 GeoStar 软件	196
7.3.8 MAPCAD 软件	197
7.4 计算机地图制图的应用研究进展	199
7.4.1 数学理论方法在计算机地图制图中的应用研究	199
7.4.2 与其他技术结合的应用研究进展	204
7.4.3 数字地图的新产品	205
参 考 文 献	209

第1章 绪论

1.1 计算机地图制图概述

1.1.1 计算机地图制图的概念

地图制图学是研究地图编制及其应用的一门学科。作为一门技术性学科,随着现代信息科学及计算机技术的发展,它正在向计算机地图制图方向发展。

计算机地图制图又称机助地图制图或数字地图制图,它是以传统的地图制图原理为基础,以计算机及其外围设备为工具,采用数据库技术和图形数据处理方法,实现地图信息的采集、存储、处理、显示和绘图的应用科学。

计算机地图制图是伴随着计算机及其外设的产生和发展而兴起的一门正在得到迅速发展的应用技术学科。它的诞生为传统的地图制图学开创了一个崭新的计算机图示技术领域,并有力地推动了地图制图学理论的发展和技术进步的过程。

计算机地图制图已在普通地图制图、专题地图制图、数字高程模型、地理信息系统等方面得到了广泛的应用,成为地图制图学的发展趋势,得到了广大地图制图工作者和地图用户的高度重视。

计算机地图制图(或数字地图制图)处理的是数字化的地图(简称数字地图)。数字地图的图形可以显示在计算机屏幕上,也可以通过绘图机输出到纸上。数字地图的数据可以保存在数字存储介质上(如磁盘、光盘等)。数字地图显示出来的内容是动态可调的,用户能方便地找出感兴趣的内容并控制显示的方式。

计算机地图制图不是简单地把数字处理设备与传统制图方法组合在一起,而是地图制图领域内一次重大的技术变革。与传统的地图制图相比,计算机地图制图具有如下的优越性:

1. 易于编辑和更新

传统的纸质地图一旦印刷完成即固定成型,不能再变化,而数字地图是在人机交互过程中动态产生出来的,可以方便地根据地图用户的要求改编地图,以增加地图的适应性。例如,用户可以指定地图的显示范围,设定显示的比例尺并可以选择地图上出现的地物要素种类等。根据用户的指令,可以随时生成改编后的新地图。

2. 提高绘图速度和精度

计算机绘图显著提高了绘图的速度,缩短了成图周期,把制图人员从烦琐的手工制图中解放出来,同时也减少了制图过程中由于制图人员的主观随意性而产生的偏差。

3. 容量大且易于存储

数字地图的容量大 ,它一般只受计算机存储器的限制 ,因此可以包含比传统地图更多的地理信息。数字地图易于存储 ,并且由于存储的是数据 ,所以不存在传统地图中常见的纸张变形等问题 ,保证了存储中的信息不变性 ,提高了地图的使用精度。

4. 丰富地图品种

计算机地图制图增加了地图品种 ,可以制作很多用传统制图方法难以完成的图种 ,如坡度图、坡向图、通视图、三维立体图等。

5. 便于信息共享

数字地图具有信息复制和传播的优势 ,容易实现共享。数字地图能够大量无损失复制 ,并可以通过计算机网络进行传播。

1.1.2 计算机地图制图的基本过程

地图设计与生产过程通常与地图资料、地图类型、地图比例尺、地图用途等多种因素有关。计算机地图制图的过程还与使用的设备和软件、数据源以及图形输出的目的有密切的联系。无论是制作普通地图还是其他类型的地图 ,计算机地图制图的基本过程可分为数据采集、数据处理和数据输出三个阶段。它们的主要内容及相互关系如图 1-1 所示。



图 1-1 计算机地图制图的基本过程

1. 数据采集阶段

地图数据的采集包括图形数据、属性数据及关系信息等的采集。

采集后的地图数据必须按一定的数据结构进行存储和组织 ,建立有关的数据文件或地图数据库 ,以便后续的数据处理和图形输出。

计算机地图制图的主要数据源是地图及有关的地图数据 ,另外 ,遥感像片、影像数据、野外测量、地理调查资料和统计资料也可作为数据源。其中地图的图形以及图像资料必须通过某种图-数转换装置转换成数字 ,以便计算机识别和处理 ,该过程又叫数字化。

地图资料通常可以用手扶跟踪数字化仪 ,按一定的规则和编码系统进行数字化 ,也可以用扫描数字化仪进行图-数转换 ,还可以在扫描数字化的基础上进行屏幕跟踪数字化。

图像资料常用扫描数字化仪进行图-数转换 ,也可进一步在此基础上进行屏幕跟踪数字化。

2. 数据处理阶段

数据处理阶段是指在计算机地图制图过程中,在数据采集后、图形输出前对地图数据进行各种处理的阶段。数据处理既可采用人机交互的处理方式,也可采用批处理方式,工作主要在某种编辑系统或相应软件中进行。

数据处理的主要内容有:地图数据的预处理、地图投影变换、地图编辑、制图综合及地图数据的符号化等。

3. 数据输出阶段

数据输出是计算机地图制图过程的最后一个阶段。地图数据处理阶段得到的结果一般是绘图数据文件,数据输出的形式有两类:一类为图形方式,又可分为屏幕显示方式和绘图机绘图;另一类即数据文件本身。

数据输出时,应根据地图数据的格式、目的和用途选择屏幕显示、矢量绘图机、栅格绘图机和适当的存储介质。

1.2 计算机地图制图的发展

计算机的发明和应用极大地推动了人类社会的发展和进步。随着社会的进步,人们逐步认识到赖以生存的地球资源是有限的,不能随意开发,而应把资源的开发和保护结合起来,进行科学的规划、管理和应用。在此背景下,人们对各类地图制品的需求与日俱增,传统的手工制图方式已远远不能满足时代的要求。伴随着技术的进步和社会需求的不断发展,计算机地图制图迅速发展起来。计算机地图制图的发展可按国内外情况分为不同的阶段。

1.2.1 国际发展状况

1. 计算机地图制图的探索期

20世纪50年代末至60年代初,计算机及其外部设备逐步得到广泛应用,并很快被用于数据的存储和处理,使计算机成为图形数据存储和计算处理的有力工具。随着图-数转换装置和自动绘图机的出现,地图可转换为能被计算机识别和处理的数字,并于1964年在绘图机上绘出了首张地图。60年代中后期,以英国牛津自动制图系统为代表的第一批计算机地图制图系统投入运行,用模拟人工制图的方法绘制了一些较为简单的地图作品。在此期间,对一些计算机地图制图的理论和技术问题进行了初步的探讨。所有这些奠定了计算机地图制图发展的基础。

2. 计算机地图制图的发展期

20世纪70年代,计算机发展到第三代,运算速度加快,内存容量增大,输入输出设备较为齐全,具有大容量直接存取功能的磁盘设备的应用,为地图数据的采集、存储、检索和输出提供了有力手段。特别是人机对话和随机操作的应用,屏幕和图形、图像卡的发展,高质量图形显示功能,使得可以通过屏幕直接监视数字化的操作,可以进行实时编辑,并且地图数据处理的结果也能很快看到,促使计算机地图制图朝着实用化方向迅速发展。

这一时期,在新技术的条件下,人们对许多计算机地图制图的理论和技术问题进行了深入的研究,其中包括地图图形的数学描述和数字表达,地图要素属性的数学描述和数字表

示 地图资料的数字化方法 地图数据的组织结构和处理 以及地图数据库和图形输出等方面的问题。在很多计算机地图制图关键问题得到解决之后 许多国家相继建立了人机交互式的计算机地图制图系统 并在此基础上进一步推动了地理信息系统的发展。

3. 机助地图制图的发展成熟期

20 世纪 80 年代 出现了大规模和超大规模集成电路 计算机发展到第四代。特别是微型计算机的出现 为计算机的普及应用创造了条件。远程通信传输设备的应用和计算机网络的建立 使地图信息的传输效率得到了极大提高 在软件方面 数据库技术、系统软件和工具软件日臻完善。这些都推动了计算机地图制图的数据处理、地图的输入输出、地图编辑和人机交互等技术的进一步发展和日渐成熟。在这期间 各种类型的地图数据库和地理信息系统不断涌现 计算机地图制图 特别是计算机辅助的专题地图制图得到了广泛的应用和很大的发展。1982 年 美国地质调查局建成了可用于绘制 1:200 万 ~ 1:1 000 万比例尺地图的 1:200 万美国国家地图数据库 并在 1983 年开始建立 1:10 万的美国国家地图数据库。在此基础上 进一步深入了对从传统地图生产模式向数字地图制图体系转变的程序和技术问题的研究 并于 1988 年颁布了数字制图数据标准。

4. 机助地图制图的普及应用期

20 世纪 90 年代以来 计算机的硬件和软件均得到了飞速的发展 随着各种计算机地图制图系统的建立 数字地图产品得到了广泛的应用。各国相继建成了大、中、小比例尺的国家地图数据库。例如 英国先后建立了可供编制 1:1 万、1:5 万及 1:25 万比例尺地形图的大比例尺国家地形图数据库 美国环境系统研究所还建成了全世界 1:100 万地图数据库。计算机地图制图在各部门、各行业得到了广泛应用 并已逐步取代了传统地图制图的生产模式。

1.2.2 国内发展状况

我国计算机地图制图的研究工作起步于 20 世纪 60 年代末 主要采用了引进、消化、改造和自主研制的方法 先后经过了硬件设备研制、软件开发以及系统建立等发展阶段。

在硬件设备方面 计算机地图制图所需的各种设备和仪器从无到有 从少到多 包括各种型号的计算机、数字化仪、图形显示器和绘图机等。

在软件开发以及系统建立方面 通过引进和消化 研制出了大量的基本绘图程序、应用绘图程序以及相应的软件系统。并在引进和自主研制相结合的基础上 陆续建立了各种计算机地图制图系统 分别用于普通地图制图、专题地图制图及地图集的设计和生 内容涵盖从各种地图投影的自动展绘、图廓整饰、符号库的建立、地图要素的自动绘制、数字地图接边、地图数据库、地图自动综合到大小比例尺地形图及各种专题地图和地图集等数字地图产品 例如“中国人口地图集”、“深圳市地图集”、中国 1:400 万地图数据库及海南 1:5 万地图数据库等。

近年来 我国的数字化测绘生产体系正在逐步建立和完善。我国测绘事业已基本实现了从传统测绘技术体系向数字化测绘技术体系的转变 国家测绘局于 1995 年 12 月成立了国家基础地理信息中心 以适应现代测绘业不断发展的需要 并积极推进国家空间数据基础设施建设。目前 该中心已相继建设了全国 1:100 万地形数据库、全国 1:50 万数字地理底图数据库、全国 1:25 万地形数据库、数字高程模型、地名数据库 完成了全国七大江河流域

重点防范区 1:1 万数字高程模型数据库建设,承担了全国 1:5 万数据库建设的前期 4D 产品试验和积极进行 1:5 万数据库建设工作。所有这些将有力地推动我国计算机地图制图的进一步发展。

1.3 计算机地图制图系统的构成

完整的计算机地图制图系统包括四个基本组成部分,即硬件、软件、地图数据和制图人员,其中硬件、软件是系统最主要的部分。各基本组成部分之间的关系如图 1-2 所示。

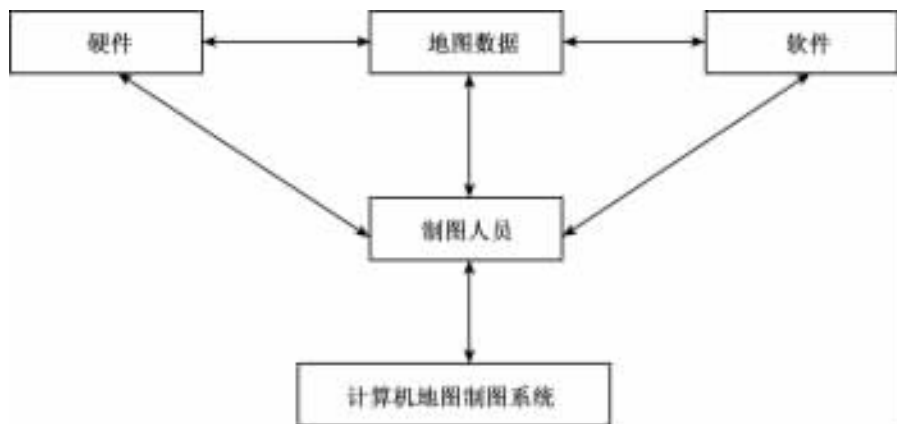


图 1-2 计算机地图制图系统的基本组成部分及其关系

1.3.1 硬件

硬件是计算机地图制图系统中用于采集、存储、处理和输出地图数据的各种仪器和设备。计算机地图制图系统的设备主要可分为数据输入、数据处理和数据输出三个部分,如图 1-3 所示。



图 1-3 计算机地图制图系统的主要设备

1. 计算机

计算机是计算机地图制图系统的主要设备,是硬件部分的核心,主要用于整个系统的数据计算、处理和管理。计算机性能的高低对计算机地图制图系统有很大的影响,正是计算机技术的飞速发展推动了计算机地图制图的不断进步。

2. 输入设备

计算机地图制图系统的输入主要分为两类:一类是图形(图像)的输入;另一类是数据的输入。对后者而言,可通过键盘、撰储设施及通信方式等直接输入计算机,无需专用设备;而对于前者,则需专用设备。

计算机地图制图系统中输入设备的主要作用是将图形(图像)转换成数字,以便计算机识别和处理。输入设备主要包括图形跟踪数字化仪和自动图形扫描仪。

(1) 图形跟踪数字化仪

图形跟踪数字化仪的类型有很多,常用的有平台式手扶跟踪数字化仪(如图 1-4 所示)。手扶跟踪数字化仪由数字化仪平板、游标(数字化鼠标)及电子线路组成。其平台的有效面积大小不等,常见的规格有 $900\text{mm} \times 1\,200\text{mm}$ 及 $1\,200\text{mm} \times 1\,800\text{mm}$,分辨率可达 0.025mm 。

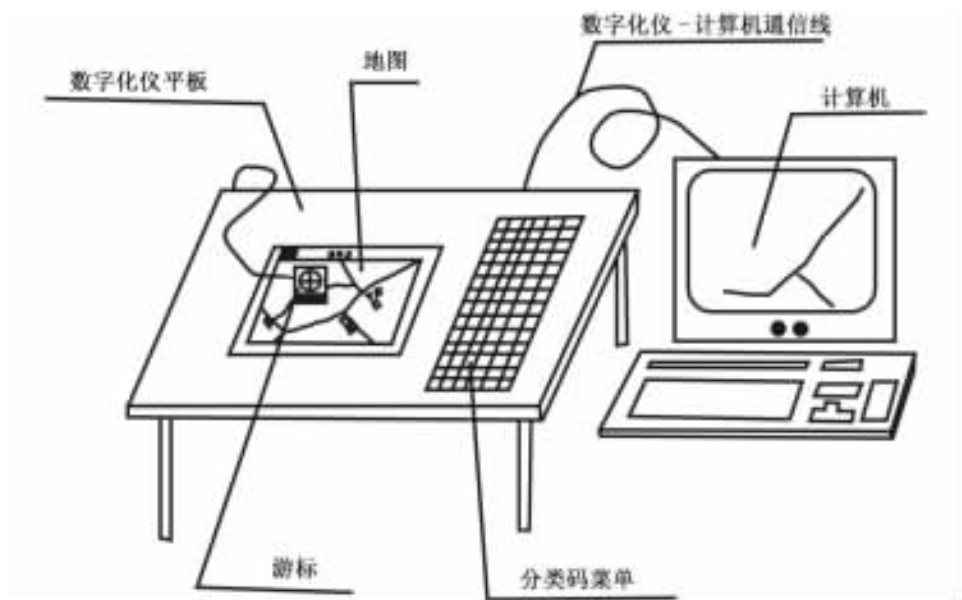


图 1-4 图形跟踪数字化仪

数字化仪平板实际上是一块电磁感应板,其表面平整光滑,在表面下的平板中有许多与 x 、 y 方向平行的印刷电路,呈规则网格状。游标又称鼠标,其内装有一个线圈,中间嵌有一个用于定位的十字丝。常用的有 4 键,乃至 16 键,每个键都可以赋予特定的功能。在进行数字化操作时,将图纸放在操作平台的有效范围内,移动游标到图上的指定位置,并将十字丝的中心对准所需数字化的点位,操作相应的按钮,此时线圈中会产生相应的磁场,从而使其正下方的印刷电路栅格上产生相应的感应电流。根据已产生电流的印刷电路栅格的位置,就可以判断出十字丝的中心定位点当前所处的几何位置。将这种位置信息以坐标(x , y)的形式传送给计算机,就实现了数字化的功能。

数字化仪的主要性能指标有:

①有效面积:指能够有效地进行数字化操作的最大面积。有效面积一般从 12 英寸 \times 12

英寸(305mm×305mm)到 44 英寸×60 英寸(1 118mm×1 524mm),有多种配置可供选择。也可按工程图纸的规格来划分,如 A3、A2、A1 等。

②分辨率:分辨率是指数字化仪的输出坐标显示值增加 1 个单位的最小可能距离。一般定位点的精度可达 0.005~0.001 英寸(0.13~0.025mm)。

数字化仪还提供多种操作模式供用户选择,可用命令设置数字化板的菜单和鼠标器的按键设定。操作模式有点方式、连续方式(流方式)、相对坐标方式等。这样,用户可方便地获取不同图形的坐标数据。数字化仪至今仍为空间数据采集的主要工具。

(2) 自动图形扫描仪

自动图形扫描仪是直接把图形(如地图)和图像(如航片、照片)扫描输入到计算机中,以栅格数据(像元、像素)形式进行存储、表示的设备。根据图形扫描仪所支持的颜色,图形扫描仪可分为单色扫描仪和彩色扫描仪;也可根据其所采用的器件分为电荷耦合器件(CCD)扫描仪、MOS 电路扫描仪等。

扫描仪的基本工作原理是:将照射原稿的光线,经过一组光学镜头投射到光敏器件上,再经过模-数转换器、数据存储器等,输入到计算机。在黑白扫描仪中,每个像元用 1 个二进制位来表示。而在灰度扫描仪中,每个像元有多个灰度层次,需要用多个二进制位表示,如 4 位精度的模-数转换器可以输出 16 种灰度值。彩色扫描仪需要提取原稿中的彩色信息,其基本工作原理与灰度扫描仪的工作原理类似。

扫描仪的分辨率是指在原稿的单位长度(英寸)上取样的点数,单位是 dpi,常用的分辨率为 300~1 000dpi。扫描图像的分辨率越高,所需的存储空间就越大。

现在新型大幅面图形扫描仪可提供高分辨率、真彩色、近乎完美的图像效果,是一种快速图形、图像数据录入和采集的有效工具。例如 ANATech 公司的 Evolution 3840 大幅面扫描仪扫描一张 A0 幅面的图纸仅需 15s,精度为 0.05%,失真率小于 0.1%。用户可在 800dpi 范围内任选扫描分辨率,可以按黑白二值或 256 级灰度方式扫描,可以边显示边扫描,并具有实时消蓝去污功能。地图扫描数字化得到的图像信息,可经过目标识别由栅格数据转换为矢量数据。

3. 输出设备

输出设备主要包括各种图形显示终端、绘图仪和打印机等。

(1) 图形显示终端

图形显示终端用于图形的交互式输入、编辑、分析、处理和输出。多数图形设备中的显示终端采用标准的阴极射线管,也有采用其他技术的显示器,如等离子显示器、液晶显示器等。目前有多种系列和型号的显示终端,如 Teltronix 公司生产的 4128、4335 型等。

表 1-1 列出了各种显示器技术性能的比较。

表 1-1

各种显示器技术性能的比较

性质	阴极射线管	等离子显示器	液晶显示器
色彩	丰富	中	中
屏幕	大	中	小

续表

性质	阴极射线管	等离子显示器	液晶显示器
厚度	大	小	小
平面度	一般	中	好
亮度	好	好	适中
分辨率	中	好	一般
对比度	中	好	差
灰度等级	好	差	差
视角	大	中	一般
功耗	大	中	小
价格	低	中	低

(2)绘图仪

绘图仪有矢量式绘图仪和栅格式绘图仪。

矢量式绘图仪分为平板式和滚筒式。平板式绘图仪是在一块平板上画图,绘图笔分别在 x、y 两个方向进行运动(如图 1-5(a)所示);滚筒式绘图仪是在一个滚筒上画图,图纸向一个方向(如 x 方向)滚动,而绘图笔向另一个方向(如 y 方向)移动(如图 1-5(b)所示)。绘图仪的主要性能指标包括有效绘图幅面、绘图速度及精度等。

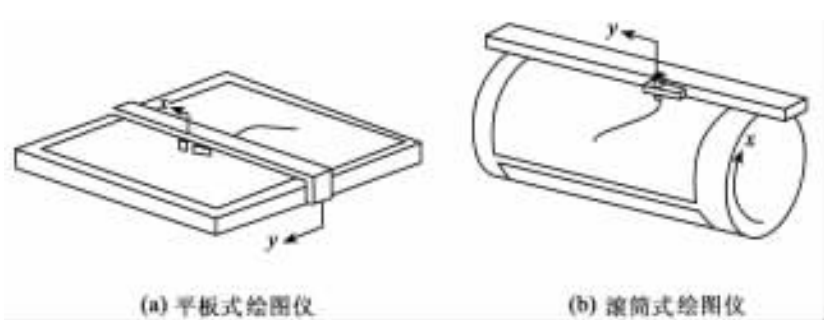


图 1-5 矢量式绘图仪

栅格式绘图仪主要有喷墨式绘图仪。喷墨式绘图仪的关键部件是喷墨头,常用的喷墨头有压电式、气泡式、静电式和固体式四种。目前广泛使用的有 HP 系列的多种系列和型号喷墨绘图仪。如 Design Jet 750C 彩色喷墨绘图仪,是一种快速、可靠、可在多种介质上进行高质量输出的绘图仪,它采用根据对象空间分布形式和输出产品的特征,选择适当的图形表示方法,结合色彩、线条、符号、文字等表示手段,具有 600dpi 分辨率的高精度黑白输出,彩色输出在 300dpi 时,颜色可达 1 600 多种,可获得极高清晰度的绘图质量。

(3)打印机

打印机的类型有针式打印机、激光打印机、液晶打印机等。激光打印机是一种既可用于

打印文字又可用于绘图的设备。HP Design Jet 彩色打印机还能打印出丰富绚丽的彩色图形和精细的文字。

1.3.2 软件

计算机地图制图系统软件是系统的核心,用于执行计算机地图制图功能的各种操作,包括数据输入、处理、数据库管理和图形用户界面(GUI)等。按照其功能分为计算机地图制图专业软件、数据库软件和系统管理软件等(如图 1-6 所示)。

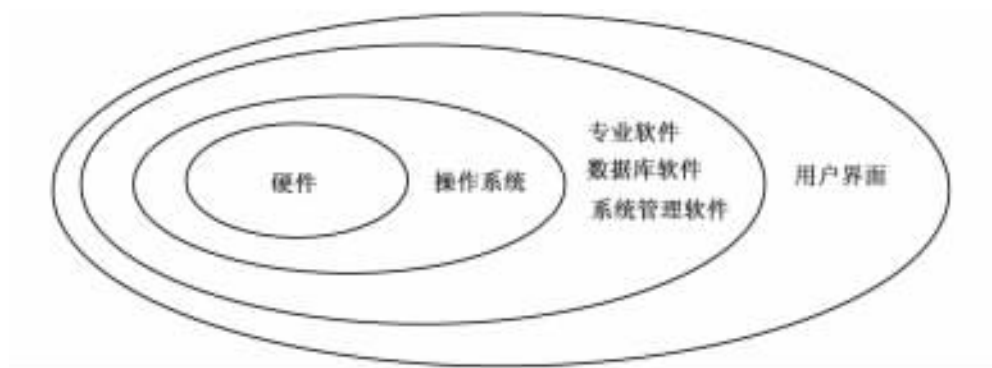


图 1-6 计算机地图制图系统软件层次

1. 计算机地图制图专业软件

计算机地图制图专业软件一般指具有丰富功能的通用计算机地图制图软件,它包含了处理地图信息的各种高级功能,一般具有以下核心模块:

(1)数据输入和编辑模块

该模块用于支持数字化仪手扶跟踪数字化、图形扫描及矢量化,以及对图形和属性数据提供修改和更新等编辑操作。

(2)空间数据管理模块

该模块能对用户数据库进行有效的存储、检索和管理。

(3)数据处理模块

该模块能完成地图投影转换,转换各种标准的矢量格式和栅格格式的数据等。

(4)数据输出模块

该模块提供符号生成、汉字生成、图形显示和地图制作等功能。

(5)用户界面

该模块提供生产图形用户界面的工具,使用户无须编程就能制作友好和美观的图形用户界面。

2. 数据库软件

数据库软件除了在计算机地图制图专业软件中用于支持地图空间数据的管理软件以外,还包括服务于以非空间属性数据为主的数据库系统。由于这类数据库软件具有快速检索、满足多用户并发和数据安全保障等功能,目前已实现在现成的关系型商业数据库中存储地图空间数据,例如 SDE(Spatial Database Engine)就是最好的解决方案。

3. 系统管理软件

系统管理软件主要指计算机操作系统 ,目前使用的操作系统主要有 MS-DOS、UNIX、Windows 98/2000/2003、Windows NT、VMS 等。它们关系到计算机地图制图软件和开发语言使用的有效性 因此也是计算机地图制图软件环境的重要组成部分。

1.3.3 地图数据

计算机地图制图系统的操作对象是地图数据 ,它描述了地理实体的空间特征、属性特征、时间特征和地理实体之间的相互关系。空间特征是指地理实体的空间位置(几何图形) ;属性特征表示地理实体的名称、类型和数量等 ;时间特征指实体随时间而发生的变化。根据地图上地理实体的空间图形表示形式 ,可将地图空间数据抽象为点、线、面三类元素 ,它们的数据表达可以采用矢量和栅格两种组织形式 ,分别称为矢量数据结构和栅格数据结构。

在计算机地图制图系统中 地图数据是以结构化的形式存储在计算机中的 称为地图数据库。地图数据库由数据库实体和数据库管理系统组成。数据库实体中存储着许多数据文件和文件中的大量数据 ,而数据库管理系统主要用于对数据进行统一管理 ,包括查询、检索、增删、修改和维护等。由于地图数据库存储的数据包含空间数据和属性数据 ,它们之间具有密切的联系 ,因此 ,如何实现两者之间的连接、查询和管理 ,是地图数据库管理系统必须解决的重要问题。常用的解决方法有以下几种 :

1. 紧凑式

紧凑式数据库管理系统是将空间数据和属性数据紧密结合在一起。例如将属性数据作为空间数据的悬挂体 ,把属性数据作为空间数据的一部分进行存储(如图 1-7 所示)。

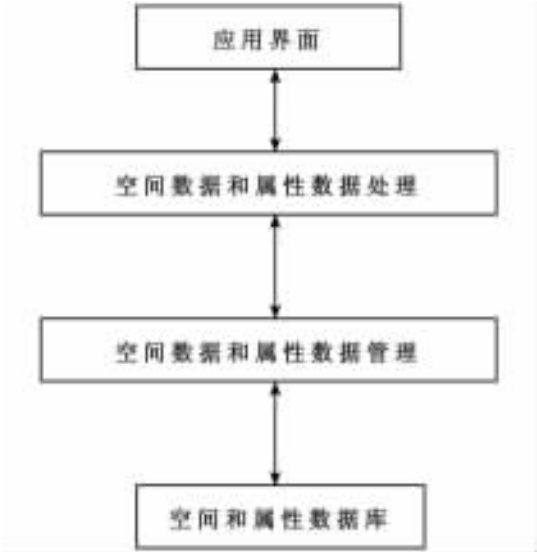


图 1-7 紧凑式数据库管理系统

2. 混合式

混合式数据库管理系统是利用两个子系统分别存储空间数据和属性数据,其中,空间数据存储在在线状或面状实体的弧段文件中,属性数据存储在关系数据库管理系统中,两个子系统之间通过标志码(ID)进行连接(如图 1-8 所示)。

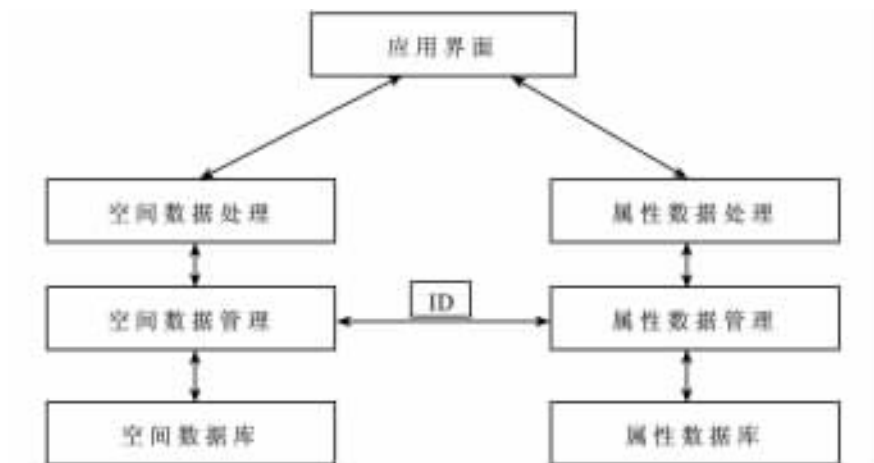


图 1-8 混合式数据库管理系统

3. 扩展式

扩展式数据库管理系统是在标准 RDBMS 的顶层,通过将地理结构查询语言(GeoSQL)转化成标准的 SQL 查询,借助索引数据的辅助关系实施空间索引操作(如图 1-9 所示)。

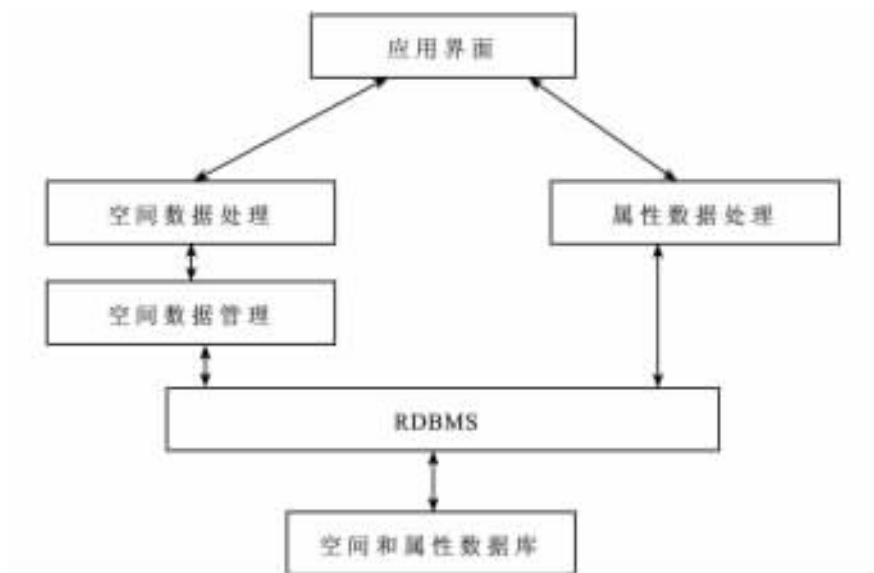


图 1-9 扩展式数据库管理系统

1.4 计算机地图制图与相关科学技术

1.4.1 与计算机地图制图相关的科学技术

1. 计算机科学

计算机科学为地图空间信息的表达、存储、处理、分析和应用提供了有利的工具。数据库技术提供数据的管理、更新、查询和维护功能;计算机图形学提供算法基础;CAD 对于辅助设计提供支持;软件工程对于计算机地图制图的系统设计提供科学的方法。

2. 地图学

计算机地图制图源于传统地图制图,地图学理论与方法对计算机地图制图系统的发展有着重要的影响。计算机制图为地图特征的数字表达、操作和显示提供了一系列方法,为计算机地图制图系统的图形输出提供技术支持。

3. 遥感技术

遥感(RS)作为空间数据的采集手段,已经成为计算机地图制图的重要信息源与数据更新途径。可以从遥感图像中快速而可靠地提取地面目标的空间信息和属性信息。我国卫星遥感地面站的实际工作经验证明,每 1~2 年 TM 图像可覆盖我国国土 1 次,其数据的现势性比常规的地图资料要好。因此,利用遥感图像实施地图空间数据的更新具有重要的现实意义和价值。

4. 全球卫星定位系统

全球卫星定位系统(GPS)作为一种新型的定位数据的采集和更新手段,具有高精度、高效益、全天候、低成本、高灵活性、实时性等优势,因此在计算机地图制图中具有重要的应用价值。

5. 电子地图

电子地图是以地图数据库为基础,通过一定的硬件和软件在电子屏幕上显示的可视地图,是数字地图在电子屏幕上的符号化显示。

1.4.2 计算机地图制图与相关科学技术的联系与区别

1. 计算机地图制图与地理信息系统

计算机地图制图是地理信息系统的技术基础,它涉及地理信息系统中的空间数据采集、表示、处理、可视化甚至空间数据的管理。它们的主要区别在于空间分析方面:计算机地图制图系统具有强大的地图制图功能,而完善的地理信息系统可以包含计算机地图制图系统的基本功能,此外还应该具有丰富的空间分析能力,特别是对图形数据和属性数据进行深层次的空间分析能力。

2. 计算机地图制图与数据库

现代数据库的主流是关系数据库。一些关系数据库管理系统不仅是一般事务管理系统的基础软件,同时也是计算机地图制图系统中属性数据管理的基础软件。目前有些计算机地图制图系统的图形数据也交给关系数据库管理系统管理,而关系数据库管理系统也向空间数据管理方面扩展。计算机地图制图系统除了需要强大的空间数据管理功能之外,还需

要具有图形数据的采集、空间数据的处理和可视化等功能。同时,计算机地图制图系统比一般的事务处理更加复杂,在功能上也更加丰富。

3. 计算机地图制图与遥感图像处理

遥感图像处理是专门针对遥感图像进行分析处理的软件,图像分析处理功能强大。而一般计算机地图制图的图像分析处理功能较弱。

4. 计算机地图制图与计算机辅助制图

计算机辅助制图(CAD)主要用来代替或者辅助工程师进行各种设计。两者的共同点是都有坐标参考系统,都能描述和处理图形数据及其空间关系,也都能处理非图形属性数据。它们的主要区别是:计算机辅助制图多为规则的几何图形及其组合,图形功能极强,属性功能相对较弱;计算机地图制图处理的图形及其关系更为复杂,空间数据与属性数据的相互操作频繁,空间数据的处理和符号化功能较强。

第 2 章 地图数据结构

2.1 地图数据的描述方法

地图数据是地图诸要素的数字化表示,是以点、线、面等方式采用编码技术对地理空间物体进行特征描述及在物体间建立相互联系的数据集。

2.1.1 地图对地理空间的描述

地图是现实世界的模型,它按照一定的比例和投影原则,有选择地将复杂的三维地理空间的某些内容投影到二维平面介质上,并用符号将这些内容要素表现出来。

1. 地理空间模型

为了研究地理现象,需要建立地球表面的几何模型。根据大地测量学的研究成果,地球表面几何模型常用的有以下三类:

(1) 地球的自然表面

地球的自然表面是十分不规则的表面,起伏不定,如马里亚纳海沟海拔 - 11 022m,珠穆朗玛峰海拔 8848.13m。地球的自然表面非常复杂,难以用一个简单的数学表达式描述出来,所以不适合于数学建模。

(2) 大地水准面

大地水准面是一个相对抽象的面,即假设当海水处于完全静止的平衡状态时,从海平面延伸到所有大陆下部,而与地球重力方向处处正交的一个连续、闭合的水准面。它有高达百米以上的起伏变化。

(3) 地球椭球体模型

该模型为绕地球自转轴旋转而成的椭球体(如图 2-1 所示),它是一个规则的数学表面。长半径、短半径和扁率成为地球椭球体的基本元素。

双轴椭球体模型(旋转椭球体):

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1$$

椭球体参数有:长半径(a),短半径(b),扁率(α),第一偏心率(e),第二偏心率(e')。其中:

$$\alpha = (a - b)/b;$$

$$e^2 = (a^2 - b^2)/a^2;$$

$$e'^2 = (a^2 - b^2)/b^2。$$

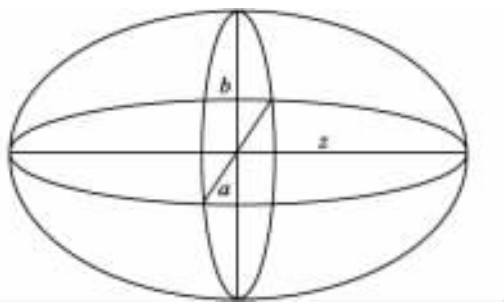


图 2-1 地球椭球体模型

国际大地测量联合会提出的 80 坐标系椭球体元素为：

$$a = 6\,378\,160\text{m}, \quad b = 6\,356\,775\text{m}$$

84 坐标系椭球体元素为：

$$a = 6\,378\,137.0\text{m}, \quad b = 6\,356\,752.3\text{m}$$

1952 年之前,我国采用海福特椭球,1953 年起改用克拉索夫斯基椭球,1978 年后开始采用国际椭球,并依此建立了我国新的、独立的大地坐标系。

2. 地理坐标系

建立地理坐标系主要的目的是确定地面点的位置,也就是求出地面点与大地水准面的关系,它包括地面点在大地水准面上的平面位置和地面点到大地水准面的高度。

地面上任一点的位置通常用经度和纬度来表示。经线和纬线是地球表面上两组正交(相交为 90°)的曲线,这两组正交的曲线构成的坐标称为地理坐标系(如图 2-2 所示)。地球上任意一点都可以用该点的经度和纬度来表示,例如北京在地球上的位置可由北纬 $39^\circ 56'$ 和东经 $116^\circ 24'$ 来确定。地表面某两点的经度值之差称为经差,某两点的纬度值之差称为纬差。

3. 平面直角坐标系

由于地理坐标系是一种球面坐标,难以进行距离、方向、面积等参数的计算,故运用地图投影的方法,建立地球表面和平面上点的函数关系,使得地球表面上任意一个由地理坐标确定的点,在平面上必有一个与其相对应的点。地图投影变换引起了地理空间要素在平面形态上的变化,包括长度变化、方向变化和面积变化。但是,平面直角坐标系却建立了对地理空间良好的视觉感,并易于进行距离、方向和面积等空间参数的量算,以及进一步进行空间数据处理和分析。

4. 高程系

高程(也称绝对高程、海拔高程)即由高程基准面起算的地面点的高度。高程基准面是根据多年观测的平均海水面确定的。高程是指地面点至平均海水面的垂直高度。地面点之间的高程差称为相对高程,简称高差。

不同地点的验潮站所得的平均海水面之间存在着差异,选用不同的基准面就有不同的高程系统。

一个国家通常只能采用一个平均海水面作为统一的高程基准面。我国的高程基准原来

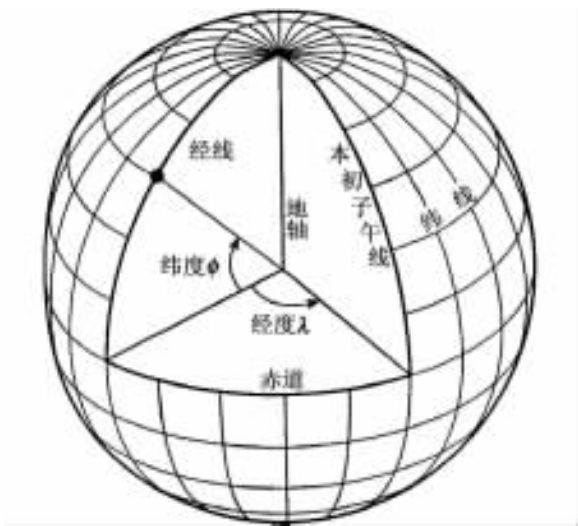


图 2-2 地理坐标系

采用“1956 年黄海高程系” ,由于观测数据的积累 ,黄海平均海水面发生了微小的变化 ,因此启用了新的高程系 ,即“1985 国家高程基准”。

5. 地图投影

地图是一个平面 ,而地球椭球面是不可展曲面 ,将地球椭球面上的点映射到平面上的方法称为地图投影。

地图投影的方法是建立在地球椭球面上的经纬线网与平面上相应的经纬线网相对应的基础上的 ,实质上就是建立地图平面上点的坐标(x ,y)与地球椭球面上对应点的坐标(λ ,φ)之间的函数关系 ,其数学表达为：

$$\begin{cases} x = F_1(\lambda , \varphi) \\ y = F_2(\lambda , \varphi) \end{cases}$$

根据不同的具体条件 ,可得到不同种类的投影公式。

对于较小区域范围 ,可以视地表为平面 ,这样就可以认为投影没有变形。但这种投影方法不太适合于大区域范围。

对于大区域范围 ,甚至是半球或全球来说 ,由于要将不可展的地球椭球面展开成平面 ,且不能有断裂 ,那么图形必将在某些地方被拉伸 ,某些地方被压缩 ,因此投影变形是不可避免的。

有三种常见的投影变形 ,即长度变形、角度变形和面积变形。根据地图投影中可能引入的变形的性质 ,可以将地图投影分为等角投影、等面积投影和任意投影三种。其中：

等角投影保证了投影后任意点的由任意两条微分线段构成的角度不产生变形 ,这种投影可以使得区域投影前后的形状保持不变。

等面积投影保证了投影前后面积保持不变 ,对微分面积如此 ,对整个区域的较大面积亦如此。

任意投影在投影后可能同时存在着长度、角度和面积的变形。在任意投影中 ,如果存在某一方向上长度不变时 ,称之为等距离投影。

等角投影与等面积投影是相互排斥的,等角是以牺牲等面积为代价的,同样,等面积也是以牺牲等角为前提的。任意投影虽然存在着各种变形,但各种变形比较均衡。

在地图投影的过程中,首先将不可展的地球椭球面投影到一个可展的曲面上,然后将该曲面展开成为一个平面,得到所需要的投影。常用的可展曲面有圆锥面、圆柱面、平面,相应地可以得到圆锥投影、圆柱投影、方位投影。

还可以根据投影面与地球轴向的相对位置将投影进一步区分为正轴投影(投影面的中心轴与地轴重合)、斜轴投影(投影面的中心轴与地轴斜向相交)、横轴投影(投影面的中心轴与地轴相互垂直)。

各种投影都有一定的变形和局限性,一般而言,距投影面越近,变形就越小。为了控制投影的变形分布,可以调整投影面与椭球体的相交位置,根据这个相交位置,可以进一步得到各种投影相应的切投影(投影面与椭球体相切)和割投影(投影面与椭球体相割)。

在计算机地图制图中,地图数据的显示往往可以根据用户的需要,指定各种投影。当显示的地图与国家基本地图系列的比例尺一致时,往往采用国家基本地图系列所用的投影。

我国常用的地图投影的情况为:

基本比例尺地形图(1:100万、1:50万、1:25万、1:10万、1:5万、1:2.5万、1:1万、1:5000)除1:100万外均采用高斯-克吕格投影作为地理基础。1:100万地形图采用了Lambert投影,其分幅原则与国际地理学会规定的全球统一使用的国际百万分之一地图投影保持一致。大部分省区图以及大多数这一比例尺的地图也多采用Lambert投影。

其中,高斯-克吕格投影是一种横轴等角切椭圆柱投影(见图2-3)。该投影是将一椭圆柱横切于地球椭球体上,椭圆柱面与椭球体表面的切线为一经线,投影中将其称为中央经线,然后根据一定的约束条件(即投影条件),将中央经线两侧规定范围内的点投影到椭圆柱面上,从而得到点的高斯投影。

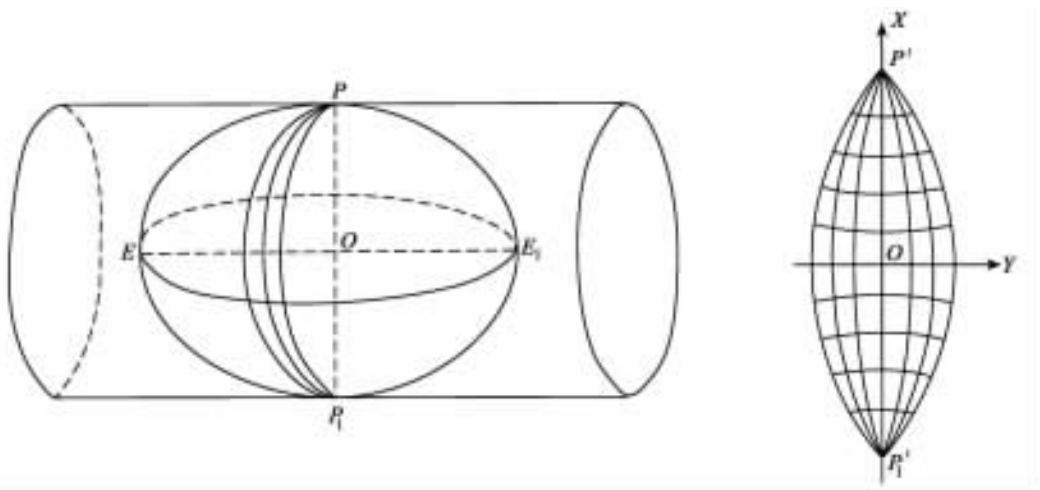


图2-3 高斯-克吕格投影示意图

高斯投影的条件为：

- ①中央经线和地球赤道投影成为直线且为投影的对称轴；
- ②等角投影；
- ③中央经线上没有长度变形。

根据高斯投影的条件推导出的高斯-克吕格投影的计算公式为：

$$\begin{cases} x = s + \frac{\lambda^2 N}{2} \sin \varphi \cos \varphi + \frac{\lambda^4 N}{24} \sin \varphi \cos^3 \varphi (5 - \tan^2 \varphi + 9\eta^2 + 4\eta^4) + \dots \\ y = \lambda N \cos \varphi + \frac{\lambda^3 N}{6} \cos^3 \varphi (1 - \tan^2 \varphi + \eta^2) + \frac{\lambda^5 N}{120} \cos \varphi (5 - 18 \tan^2 \varphi + \tan^4 \varphi) + \dots \end{cases}$$

其中 x 、 y 为点的平面直角坐标系的纵、横坐标；

φ 、 λ 为点的地理坐标，以弧度计， λ 从中央经线起算；

s 为由赤道至纬度 φ 处的子午线弧长；

N 为纬度 φ 处的卯酉圈曲率半径。

$$\eta^2 = e^2 \cos^2 \varphi$$

$$e'^2 = (a^2 - b^2) / b^2$$

式中 e' 为地球的第二偏心率， a 、 b 则分别为地球椭球体的长、短半轴。

高斯投影是等角投影，没有角度变形，其沿任意方向的长度比都相等，其面积变形是长度变形的两倍。

高斯投影变形具有以下特点：

- (1) 中央经线上无变形；
- (2) 同一条纬线上，离中央经线越远，变形越大；
- (3) 同一条经线上，纬度越低，变形越大。

高斯投影的最大变形处为各投影带在赤道边缘处。我国地形图采用分带投影方法来控制变形，即将地球按一定间隔的经差（ 6° 或 3° ）划分为若干相互不重叠的投影带，各带分别投影。1:2.5 万 ~ 1:50 万的地形图均采用 6° 分带方案，即从格林尼治零度经线起算，每 6° 为一个投影带，全球共分为 60 个投影带。我国领土位于东经 $72^\circ \sim 136^\circ$ 之间，共包括 11 个投影带（13 带 ~ 22 带）。1:1 万及更大比例尺地形图采用 3° 分带方案，全球共分为 120 个投影带（如图 2-4 所示）。

6. 坐标网

在地图上通常都绘有一种或两种坐标网，即经纬线网、方里网。

(1) 经纬线网

经纬线网又称地理坐标网，是指由经线和纬线所构成的坐标网。

我国 1:1 万 ~ 1:20 万比例尺的地形图上，经纬线只以图廓线的形式直接表现出来，并在图角处注出相应度数。为了在用图时加密成网，在内外图廓间还绘有加密经纬网的加密分划短线（图式中称“分度带”），必要时对应短线相连就可以构成加密的经纬线网。1:25 万地形图上，除内图廓上绘有经纬网的加密分划外，图内还有加密用的十字线。

对 1:50 万 ~ 1:100 万地形图，在图面上直接绘出经纬线网，内图廓上也有供加密经纬线网的加密分划短线。

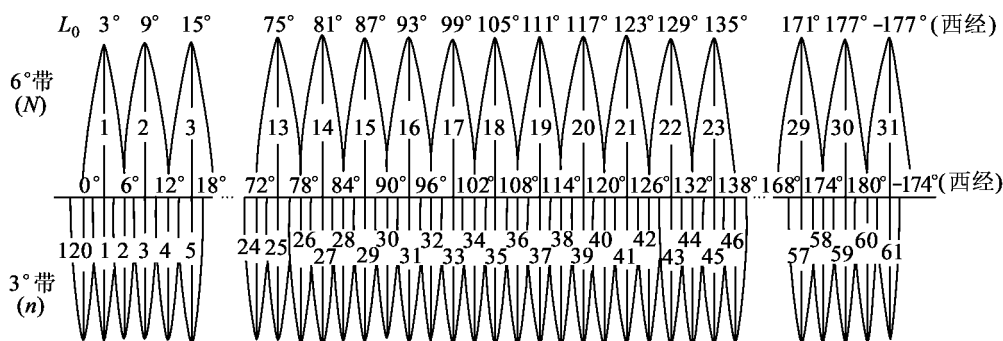


图 2-4 高斯-克吕格投影 3°与 6°带分带方案示意图

(2) 方里网

方里网是由两组分别平行于投影坐标轴的平行线所构成的方格网。因为是每隔整公里绘出坐标纵线和坐标横线,所以称之为方里网。由于方里线同时又是平行于直角坐标轴的坐标网线,故又称直角坐标网。

方里网的坐标系以中央经线投影后的直线为 x 轴,以赤道投影后的直线为 y 轴,它们的交点为坐标原点。这样,坐标系中就出现了四个象限。纵坐标从赤道算起向北为正,向南为负;横坐标从中央经线算起,向东为正,向西为负。

因为我国位于北半球,故全部 x 值都是正值。而在每个投影带中,则有一半的 y 坐标值为负。为了避免 y 坐标出现负值,规定纵坐标轴向西平移 500km(半个投影带的最大宽度不超过 500km)。这样,全部坐标值都表现为正值了。

7. 空间实体

在地图学中,把地理空间的实体分为点、线、面三种要素(对象),分别用点状、线状、面状符号来表示。

将研究的整个地理空间看成一个空域,地理实体和现象分布在该空域中。按照空间特征,地理实体可分为点、线和面三种基本对象。对象也可能是由其他的对象构成的复杂对象,并且与其他的对象保持着特定的关系。每个对象对应着一组相关的属性,以区分出各个不同的对象。将空间要素嵌入在一个坐标空间之中,一般是欧氏空间,在该空间中可以利用公式进行距离、方位和面积的测量。空间要素在欧氏空间中主要形成以下三类空间实体:

(1) 点实体

点实体是有特定的位置、维数为 0 的实体(如图 2-5 所示)。

实体点:用来代表一个实体;

注记点:用于定位注记;

内点:用于记录多边形的属性,存在于多边形内;

节点:表示线的终点和起点;

拐点:表示线段和弧段的内部点。

(2) 线实体



图 2-5 点实体

线实体是维数为 1 的实体(如图 2-6 所示),由一系列坐标点表示,有以下特征:

- 实体长度:从起点到终点的总长;
- 弯曲度:用于表示如道路拐弯时弯曲的程度;
- 方向性:如河流从上游到下游,公路有单双向之分;

线实体包括:线段、线列、弧段、链、路径、多边线等。

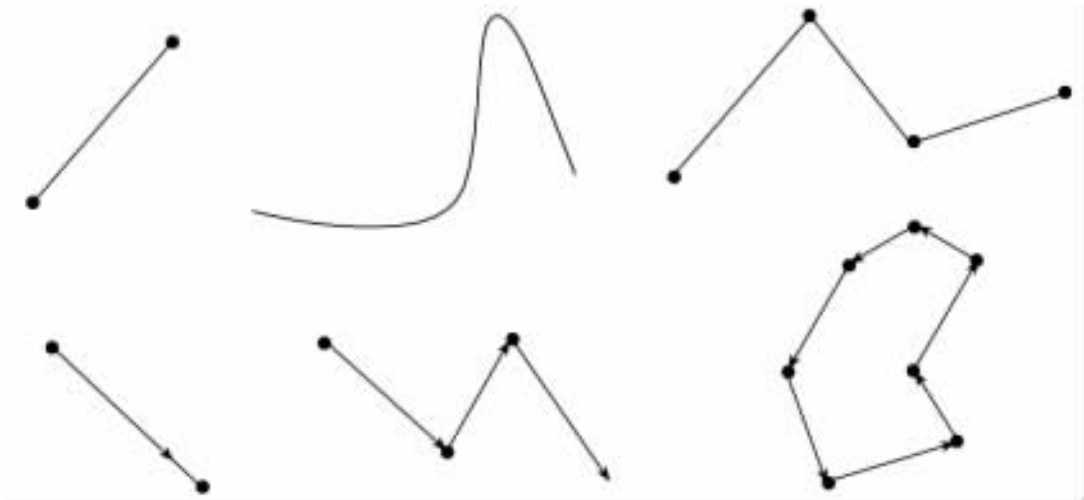


图 2-6 线实体

(3)面(多边形)实体

面(多边形)实体是维数为 2 的实体(如图 2-7 所示),由一个封闭的坐标点序列外加内点表示,是对湖泊、岛屿、地块等现象的描述。面具有以下特征:

- 周长;
- 面积;
- 独立或与其他地物的邻近性:如湖北及周边省市;
- 内岛或齿状外形:如岛屿及海岸线;
- 重叠性与非重叠性。



图 2-7 面实体

2.1.2 地图数据的基本特征

地图数据具有空间特征、属性特征及时间特征。

1. 空间特征

(1) 空间位置

空间位置用以描述事物或现象的地理位置,又称几何特征、定位特征,如界桩的经纬度等。地理空间实体在一定的坐标参考系中的空间位置,通常用地理坐标系、平面直角坐标系来表示,也称几何特征,包括空间实体的位置、大小、形状、分布状况等,表明“在哪里”。

(2) 空间关系

空间关系指地理空间实体之间存在的一些具有空间特性的关系,主要包括:

拓扑关系 拓扑变化下的拓扑不变量,如邻接关系、关联关系和包含关系等;

方位关系 实体在地理空间中的某种顺序,如左右、东南西北等;

度量关系 用地理空间中的度量来描述的实体之间的关系,如实体之间的距离。

2. 属性特征

属性特征用以描述事物或现象的特性,如事物或现象的类别、等级、数量、名称等,用来说明“是什么”。

属性特征通常分定性和定量两种:定性特征包括名称、类型等;定量特征包括数量、等级等。

3. 时间特征

时间特征用以描述地理实体随着时间而变化的特征。地理空间实体的空间位置和属性相对于时间来说,常常呈现出相互独立的变化。由于地图数据具有上述特征,所以在计算机地图制图中的数据表示是非常复杂的。目前的计算机地图制图还较少考虑到地图数据的时间特征,只考虑其属性特征与空间特征的结合。由于地图数据具有时间维,过时的信息虽不具有现势性,却可以作为历史性数据保存起来,但这样做会增大计算机地图制图表示和处理数据的难度。

2.1.3 地图数据的基本类型

根据地图数据的特征,可以把地图数据分为空间数据、关系数据、属性数据三类。

1. 空间数据

空间数据是描述地图要素中空间特征部分的数据,也称几何数据,即描述地理现象或地理实体的空间位置、形状、大小等的的数据。

根据地理要素的空间分布特征和空间实体分类,可以将地理空间数据分为点、线、面三种类型。

(1)点类型

点类型可以描述如城乡居民地、工厂、学校、医院、机关、车站、山峰、隘口等现象。这里,“点”是一个相对的抽象概念,即从较大的空间规模上来观测这些地物,就能把它们都归结为点状分布的地理现象,因此能用一个点的坐标(或栅格像元)来描述其空间位置。而如果从较小的空间尺度上来观察这些地理现象,它们中的多数将可以用一个面状特征来描述。例如同一个城市,在小比例尺地图上表现为点状分布,而在大比例尺地图上则可表现为面状分布,其内部表示了十分详细的城市街道分布状况。

(2)线类型

线类型描述如河流、运河、海岸、铁路、公路、地下管网、行政边界等线状分布的地理现象。这里的“线”(有时也称“弧”)也是一个相对的抽象概念。它们的空间位置数据是一线状坐标串(或栅格像元集合)。

(3)面类型

面类型描述如土地、水域、森林、草原、沙漠等具有大范围连续(或断续)面状分布特征的现象。这里的“面”是一个相对的抽象概念,有时实地上不一定有明显的边界。描述面状特征的空间数据是一封闭的面坐标串(或栅格像元集合),通常称之为多边形。

2. 关系数据

关系数据是描述空间数据之间的空间关系的数据。上述点、线、面空间位置数据之间存在着某种特定的拓扑关系。拓扑关系是一种对空间关系进行明确定义的数学方法,可以用某种特定的数据来表达,称之为拓扑关系数据(简称拓扑数据)。这类数据表达了各类地理实体空间位置之间的相互关系,如空间数据的相邻、关联、包含等。

在地图上,地理要素间相互关系信息并没有直接标示,只能依靠判断明显的地图要素之间的关系来识别。在计算机地图制图技术中,这种空间关系通常采用图论中的概念,使用节点、路径和图之间的拓扑结构来描述。可把这种关系看做一类“图”,就比较容易建立起地理实体空间位置数据的空间拓扑关系。

地理实体的空间拓扑关系具有重要意义,它不随地图投影而变化,比一般几何关系具有更大的稳定性,能本质地反映实体之间的逻辑结构关系。

利用地理实体的空间拓扑关系可以解决许多实际应用问题。例如,在空间要素的查询中有时需要查询:某区域(面或多边形)的邻接区域(面或多边形)是哪些?而如果某供气管网系统中某段供气管破裂,要找关闭它的阀门就需要查询该管道(线段或弧段)与哪些阀门(点)关联等。

应用拓扑关系还可重建地理实体。如根据弧(线)段构建多边形,实现面域的选取。根据弧(线)段与节点的关联关系重建道路网络,进行最佳路径选择等。

空间实体间相互关系的描述是非常重要的。各种地理要素的空间位置数据在地图上的关系,可以概括为点、线、多边形之间的9种形式的拓扑关系:点与点、点与线、点与面、线与

点、线与线、线与面、面与点、面与线、面与面。

设有两个空间实体 A 和 B。用 $B(X)$ 表示实体 X 的边界, $I(X)$ 表示实体 X 的内部, 用 $E(X)$ 表示实体 X 的外部。根据上述概念, Egenhofer 在 1993 年为空间实体间的拓扑关系描述构造了“9 交空间关系模型”(9-Intersection Model 9-IM), 见表 2-1。

表 2-1 9 交空间关系模型

$B(A) \cap B(B)$	$B(A) \cap I(B)$	$B(A) \cap E(B)$
$I(A) \cap B(B)$	$I(A) \cap I(B)$	$I(A) \cap E(B)$
$E(A) \cap B(B)$	$E(A) \cap I(B)$	$E(A) \cap E(B)$

表 2-1 中, 每个元素的取值可为“空”或“非空”, 9 个元素一共可以产生 $2^9 = 512$ 种情形, 即可描述 512 种空间关系。

最常用的空间实体关系有 6 种, 即: 点-点、点-线、点-面、线-线、线-面、面-面。它们之间的相互关系见表 2-2。

表 2-2 常用的空间实体关系

	邻接	关联	相交	相离	包含	重合
点-点	*			*		*
点-线		*	*	*	*	
点-面		*	*	*	*	
线-线	*		*	*	*	*
线-面		*	*	*	*	
面-面	*		*	*	*	*

3. 属性数据

属性数据是描述空间实体属性特征的数据, 也称非几何数据, 即描述地理现象或地理实体的定性或定量指标, 包括语义与统计数据, 如类型、等级、名称、状态等。有时也把描述时间特征的数据纳入该类。

属性数据中的定性(或定量)指标通常要经编码转换才能被计算机接受。为了方便计算机存储、管理和使用这些编码, 需要研究统一的分类系统和编码。有关这方面的详细内容将在本书第 3 章中进行介绍。

2.2 地图的数据结构

地图的数据结构主要是指地图数据中空间数据的结构, 即指空间数据适合于计算机存储、管理及处理的几何数据的逻辑结构。换句话说, 是指几何数据以什么形式在计算机中存储和处理。地图的数据结构主要分为矢量数据结构和栅格数据结构。

2.2.1 矢量数据结构

1. 矢量数据的概念

矢量是具有一定长度和方向的量。一个矢量在二维空间里可表示为 (dx, dy) ,其中 dx 表示沿 x 方向移动的距离 dy 表示沿 y 方向移动的距离。

矢量数据结构是表达地图空间数据的一种常见的数据结构 ,它通过记录坐标值的方式尽可能精确地表示呈点、线或面状分布的地理实体。在计算机地图制图中 ,各地图要素在二维平面上的矢量数据表示为(如图 2-8 所示) :

- 点 :由一对 x, y 坐标表示 ;
- 线 :由一串有序的 x, y 坐标对表示 ;
- 面 :由一串有序的且首尾坐标相同的 x, y 坐标对表示。

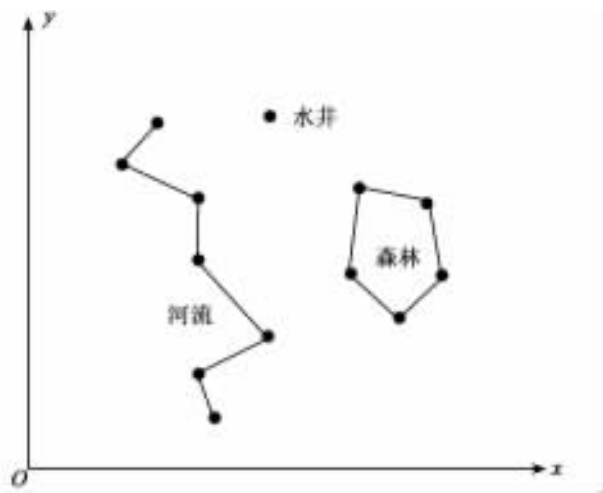


图 2-8 矢量数据的表示

矢量数据可以表示地图中各种复杂的地理实体 ,当问题可描述成线、边界和网时 ,特别有效。矢量数据还有冗余度低、结构紧凑、具有空间实体的拓扑信息、便于深层次分析及输出质量好、精度高等优点。

2. 矢量数据结构的表示

矢量数据结构是最早用于表达地图空间数据的一种常见的数据结构 ,在计算机地图制图中 ,表示矢量数据的结构时应考虑以下问题 :

- ① 矢量数据的存储和处理 ;
- ② 与属性数据的联系 ;
- ③ 矢量数据之间的拓扑关系。

表示矢量数据的方法有多种 ,但基本上类似。下面按考虑问题的多寡分别介绍矢量数据的简单结构和拓扑结构及其有关的编码方法。

(1) 简单数据结构及编码

简单的矢量数据结构不考虑拓扑关系 ,可用于矢量数据的存储、处理、显示、输出以及一

般的查询和检索。有点、线、面三种基本的矢量数据结构形式。

①点数据结构形式。点的矢量数据结构形式可表示为：

标志码	属性码	(x, y)坐标对
-----	-----	-----------

其中 标志码具有惟一性,是按某种原则进行的编码,如在一般情况下可按记录顺序编号。

属性码是与点实体有关的基本属性(如等级、类型、大小等)的编码,属性码可以有一个或多个。

(x, y)坐标对:是点实体的定位点坐标,如果是有向点,则可以有二个坐标对。

在点的矢量数据结构中也可不包含属性码,而将属性数据单独存放在数据库中,通过标志码联系矢量数据和与其对应的属性数据。

②线(弧、链)数据结构形式。线(弧、链)的矢量数据结构可表示为：

标志码	属性码	坐标对数 n	坐标串(x_1, y_1) ... (x_n, y_n)
-----	-----	--------	--------------------------------------

标志码和属性码的含义与点的数据结构相同。在线数据结构的属性码中还可含有表示线的类型、等级以及是否要加密、光滑等信息。

坐标对数 n 构成该线(弧、链)的坐标对个数。

坐标串是构成线(弧、链)的矢量坐标对序列,共有 n 对。也可采用把线(弧、链)的坐标串单独存放,而只给出指向该线(弧、链)坐标串的首地址指针的方法。

③面(多边形)数据结构形式。面(多边形)的矢量数据结构可以像线的数据结构一样表示,但其中坐标串的首尾坐标相同。也可采用其他方式,如弧段索引编码的面(多边形)的矢量数据结构为：

标志码	属性码	弧段数 n	弧段标志码集
-----	-----	-------	--------

其中 标志码与属性码的含义同点和线的矢量数据结构；

弧段数 n 是指构成该面(多边形)的弧段的数目；

弧段标志码集是指所有构成该面(多边形)的弧段的标志码的集合,共有 n 个。

一个面(多边形)可由多条弧段构成,每条弧段的坐标可由弧(线、链)的矢量数据结构获取。该方法还保证了多边形公共边的惟一性。

④简单数据结构的编码形式。因在矢量的简单数据结构中不考虑拓扑关系,故其编码方法仅记录空间实体的位置、标志及属性信息,而不记录拓扑关系。常见的编码方法有独立实体法和点位字典法。

• 独立实体法

在独立实体法中,每个点、线、面实体都直接跟随它的空间坐标,即：

点实体:惟一标志码,实体编码,空间坐标(X, Y)；

线实体 :唯一标志码 ,实体编码 ,空间坐标($X_1 ,Y_1 \dots X_n ,Y_n$) ;
面实体 :唯一标志码 ,实体编码 ,空间坐标($X_1 ,Y_1 \dots X_n ,Y_n ,X_1 ,Y_1$)。
如图 2-9 所示 ,多边形实体可用表 2-3 的编码数据来表示。其中 A、B、C、D、E 为多边形的标志码。

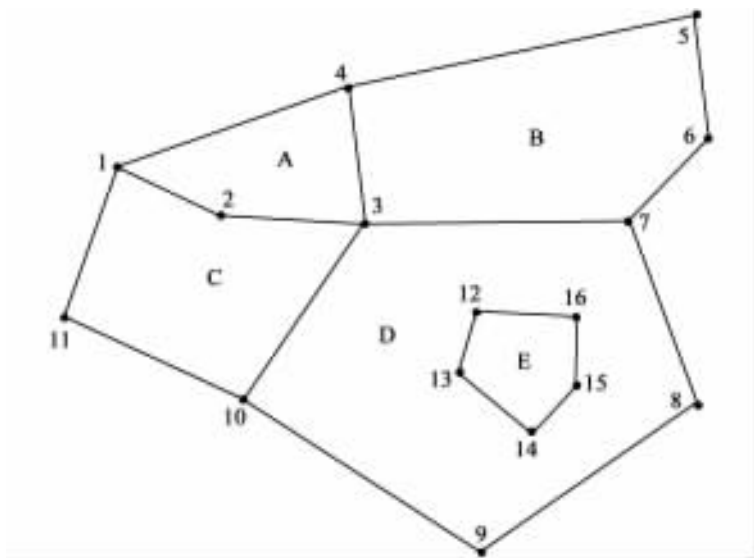


图 2-9 多边形实体

表 2-3 独立实体编码

多边形	数 据 项
A	$X_1 ,Y_1 ,X_2 ,Y_2 ,X_3 ,Y_3 ,X_4 ,Y_4 ,X_1 ,Y_1$
B	$X_3 ,Y_3 ,X_4 ,Y_4 ,X_5 ,Y_5 ,X_6 ,Y_6 ,X_7 ,Y_7 ,X_3 ,Y_3$
C	$X_1 ,Y_1 ,X_2 ,Y_2 ,X_3 ,Y_3 ,X_{10} ,Y_{10} ,X_{11} ,Y_{11} ,X_1 ,Y_1$
D	$X_3 ,Y_3 ,X_7 ,Y_7 ,X_8 ,Y_8 ,X_9 ,Y_9 ,X_{10} ,Y_{10} ,X_3 ,Y_3$
E	$X_{12} ,Y_{12} ,X_{13} ,Y_{13} ,X_{14} ,Y_{14} ,X_{15} ,Y_{15} ,X_{16} ,Y_{16} ,X_{12} ,Y_{12}$

该方法的优点是编码容易 ,数字化操作简单 ,数据编码直观 ,显示速度快。该方法的缺点是相邻多边形的公共边界数字化两次 ,造成数据的冗余 ,可能出现重叠或裂缝 ,引起数据不一致 (如图 2-10 所示) ,缺少拓扑关系 ,空间分析困难。

• 点位字典法

点位字典法中 ,点坐标作为一个文件 ,点、线和面 (多边形)实体目标都由点号组成 ,即 :
点位字典 :点号、(X ,Y) ;
点实体 :唯一标志码 ,地物编码 ,点号 ;
线实体 :唯一标志码 ,地物编码 ,(点号 1 ... 点号 n) ;
面实体 :唯一标志码 ,地物编码 ,(点号 1 ... 点号 n ,点号 1)。

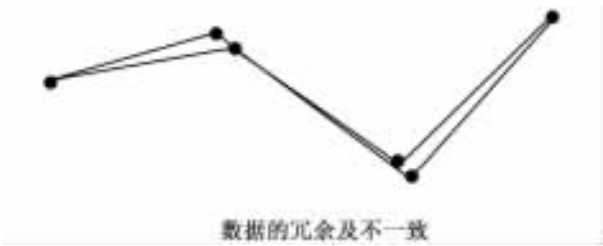


图 2-10 公共边界两次数字化

例如对图 2-11 中所示的多边形实体可用表 2-4 及表 2-5 的编码数据来表示。其中 A、B、C 分别为点、线和面(多边形)的标志 阿拉伯数字为点号。

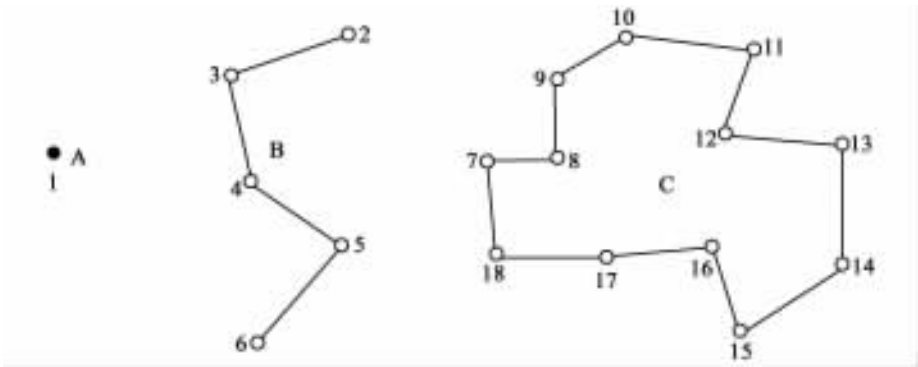


图 2-11 点、线、面(多边形)实体目标

表 2-4 点 位 表	
点 号	坐 标
1	X_1, Y_1
2	X_2, Y_2
3	X_3, Y_3
4	X_4, Y_4
5	X_5, Y_5
...	...
18	X_{18}, Y_{18}

表 2-5 点位字典编码	
目标标志	点 号
A	1
B	2 3 4 5 6
C	7 8 9 10 11 12 13 14 15 16 17 18

(2)拓扑数据结构及编码

地图上两点间距离或方向会随地图投影的不同而发生变化 ,故仅用距离或方向不能很好地描述地图要素间的空间关系。

拓扑学(又称“橡皮几何学”)是研究图形在保持连续状态下变形时的那些不变的性质。拓扑关系是一种对空间结构关系进行明确定义的数学方法。如果引用拓扑关系来描述地图要素间的空间关系 ,则不论地图投影如何变化 ,其邻接、关联和包含关系都不改变。可见拓扑关系能从本质的方面来描述地图要素间的空间关系。

具有拓扑关系的矢量数据结构就是拓扑数据结构 ,拓扑数据结构是现代计算机地图制图系统所必需的。尽管拓扑数据结构的表示方式还没有固定的格式 ,也没有形成标准 ,但其基本原理是相同的。

①拓扑元素。地图要素可由点、线、面三类来概括表示 ,其矢量数据可抽象为点(节点)、线(链、弧段、边)、面(多边形)三种要素 ,称为拓扑元素。点(节点)包括孤立点、线的端点、面的首尾点、链的连接点等 ;线(链、弧段、边)为两节点间的有序弧段 ;面(多边形)为若干条线(链、弧段、边)构成的闭合多边形。

②基本拓扑关系。基本拓扑关系是邻接、关联和包含。

邻接是相同拓扑元素之间的关系(如图 2-12 所示)。如节点与节点、链与链、面与面等。邻接关系是借助于不同类型的拓扑元素描述的 ,如点通过链而邻接。

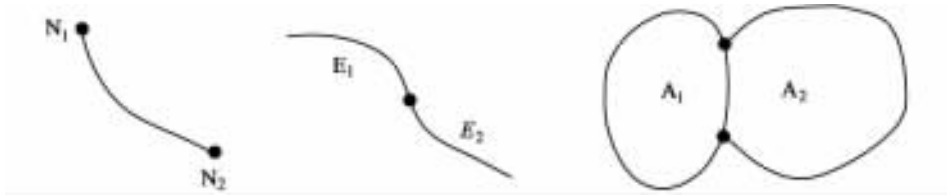


图 2-12 拓扑邻接

关联是不同拓扑元素之间的关系(如图 2-13 所示)。如节点与链、链与多边形等。

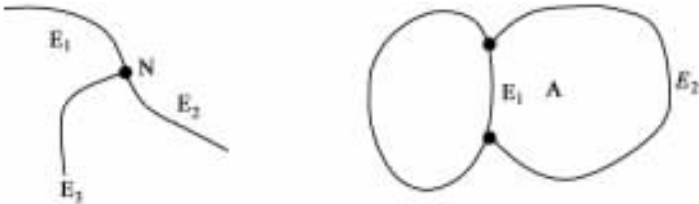


图 2-13 拓扑关联

包含是面与其他拓扑元素之间的关系。如果点、线、面在该面内 ,则称为被该面包含 ,如某省包含的城市、河流及湖泊等。

在计算机地图制图系统中 ,也可能用到其他关系 ,如层次关系即相同元素之间的等级关系。如国家由省(自治区、直辖市)组成 ,省(自治区、直辖市)由县组成等。

③拓扑关系的表示方法。拓扑关系的表示方法分为隐式和显式两种：

•隐式表示

隐式表示不直接存储拓扑关系 ,而是在需要拓扑关系时通过几何数据临时推导生成。

•显式表示

显式表示是将拓扑元素(点、线、面)之间的拓扑关系数据化 ,作为地图数据的一种类型予以存储。

如何表示拓扑关系是拓扑数据结构的关键 ,其中几何数据的表示可参照矢量数据的简单数据结构。在目前的计算机地图制图系统中 ,主要表示的是拓扑元素(点、线、面)之间基本的拓扑关系 ,表示方法多种多样。下面以图 2-14 为例介绍一些常用的显式表示拓扑关系的方法。

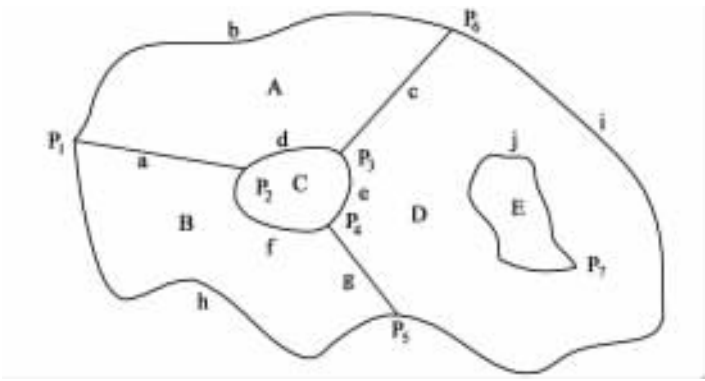


图 2-14 拓扑关系示例

在图 2-14 中 $P_1, P_2, P_3, P_4, P_5, P_6$ 和 P_7 为节点 $a, b, c, d, e, f, g, h, i$ 和 j 为弧段(线) ;A , B , C , D 和 E 为多边形(面)。为了表示出节点、弧段及多边形之间的拓扑关系 ,可以使用如下 4 个关系表。

节点与弧段关系见表 2-6。

表 2-6

节点与弧段的拓扑关系

节 点	弧 段
P_1	a, b, h
P_2	a, d, f
P_3	d, c, e
P_4	e, f, g
P_5	g, h, i
P_6	b, i, c
P_7	j

弧段与节点关系表见表 2-7。

表 2-7 弧段与节点的拓扑关系

弧段	节点(始节点 终节点)
a	P ₁ P ₂
b	P ₁ P ₆
c	P ₆ P ₃
d	P ₃ P ₂
e	P ₃ P ₄
f	P ₂ P ₄
g	P ₄ P ₅
h	P ₅ P ₁
i	P ₅ P ₆
j	P ₇ P ₇

弧段与多边形关系表见表 2-8。

表 2-8 弧段与多边形的拓扑关系

弧段	多边形	
	左	右
a	A	B
b	0	A
c	D	A
d	C	A
e	D	C
f	C	B
g	D	B
h	0	B
i	D	0
j	D	E

其中 ,以弧段的前进方向区分左右 0 为制图区域外部的多边形 称为包络多边形。
多边形与弧段关系表见表 2-9。

表 2-9 多边形与弧段的拓扑关系

多边形	弧段
A	a b c d
B	a f g h
C	d e f
D	c i g e , - j
E	j

其中“-”表示面域中含有岛。

有时也可将弧段与节点关系表(表 2-7)和弧段与多边形关系表(表 2-8)合并成一个表。

④拓扑数据结构的编码形式。在矢量的拓扑数据结构中包含了拓扑关系,故其编码方法不仅记录空间实体的位置、标志及属性信息,还要记录拓扑关系。记录拓扑关系的编码方法有多种,常见的有双重独立地图编码和链状双重独立式编码。

● 双重独立地图编码

双重独立地图编码(Dual Independent Map Encoding ,DIME)最早是由美国人口统计系统采用的一种编码方法,它以城市街道为编码的主体。以图 2-15 为例介绍如下:

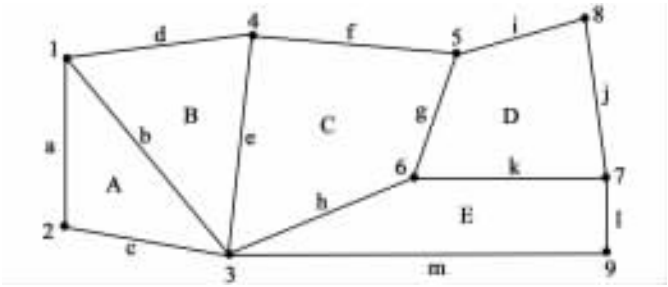


图 2-15 双重独立地图编码示例

图中 A、B、C、D、E 为面域标号, a、b、c、d、e、f、g、h、i、j、k、l、m 为线段号,阿拉伯数字为点号。双重独立地图编码的编码方式由两个主要表格文件组成(见表 2-10 及表 2-11)。

表 2-10 节 点 表

节点号	坐 标
1	X ₁ ,Y ₁
2	X ₂ ,Y ₂
3	X ₃ ,Y ₃
4	X ₄ ,Y ₄
5	X ₅ ,Y ₅

续表

节点号	坐 标
6	X_6, Y_6
7	X_7, Y_7
8	X_8, Y_8
9	X_9, Y_9

线段表

线段号	起点	终点	左多边形	右多边形
a	1	2	A	0
b	3	1	A	B
c	2	3	A	0
d	1	4	0	B
e	4	3	C	B
f	4	5	0	C
g	5	6	D	C
h	6	3	E	C
i	5	8	0	D
j	8	7	0	D
k	7	6	E	D
l	7	9	0	E
m	9	3	0	E

- 链状双重独立式编码

链状双重独立式编码由美国计算机图形及空间分析实验室研制,是当今各种图形数据结构的基本框架。该框架主要由4个(或3个)文件构成:

节点文件 其结构与上述节点表相似,即 标志码, (X,Y);

弧段坐标文件 标志码 弧段中间点；

弧段文件 标志码 起始节点 终止节点 左多边形 右多边形 内点(指向中间点坐标的指针或坐标);

多边形文件 标志码 组成多边形的弧段号及面积、周长及中心点坐标等。

链状双重独立式编码表示的拓扑关系有:节点与节点之间的邻接关系,多边形与多边形之间的邻接关系,节点与线段之间的关联关系,线段与多边形之间的关联关系。

该编码方法的优点是:数据结构紧凑、数据冗余小、拓扑关系明晰,使得拓扑查询、拓扑分析效率高。

该编码方法的缺点是 :对单个地理实体的操作效率低 ,难以表达复杂的地理实体 ,查询效率低 ,局部更新困难。

下面以图 2-16 为例 ,对链状双重独立式编码进行介绍。

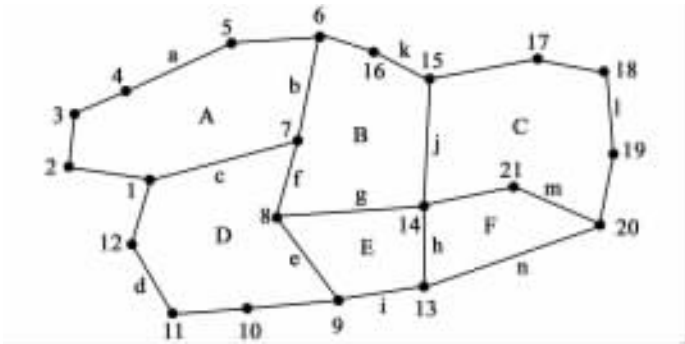


图 2-16 链状双重独立式编码示例

图中 A B C D E F 为面域号 a b c d e f g h i j k l m n 为弧段号 ;阿拉伯数字为点号。

节点文件与上述节点表(表 2-10)相似 ,在此不再赘述。

弧段坐标文件见表 2-12。

表 2-12	弧段坐标表
弧段号	点 号
a	1 2 3 4 5 6
b	6 7
c	7 1
d	9 10 11 12 1
e	8 9
f	7 8
g	8 14
h	13 14
i	9 13
j	14 15
k	6 16 15
l	15 17 18 19 20
m	20 21 14
n	13 20

弧段文件见表 2-13。

表 2-13弧 段 表

弧段号	起点	终点	左多边形	右多边形	内点
a	1	6	0	A	
b	6	7	B	A	
c	7	1	D	A	
d	9	1	0	D	
e	8	9	E	D	
f	7	8	B	D	
g	8	14	B	E	
h	13	14	E	F	
i	9	13	E	0	
j	14	15	B	C	
k	6	15	0	B	
l	15	20	0	C	
m	20	14	F	C	
n	13	20	F	0	

多边形文件见表 2-14。

表 2-14多 边 形 表

多边形号	弧段号	面积	周长	中心点坐标
A	a b c			
B	b k j g f			
C	j l m			
D	d c f e			
E	e g h i			
F	h m n			

2.2.2 栅格数据结构

1. 栅格数据的概念

可将地图制图区域的二维平面表像按行和列作规则划分 ,形成一个栅格阵列 ,其中各栅格阵列元素又称为“ 像元 ”(或“ 像素 ”)。各个像元可用不同的灰度值来表示相应的属性值。在各个像元内部 ,其属性值被认为是一致的。栅格数据是由二维平面表像对应位置上像元灰度值所组成的阵列形式的数据。

栅格数据表示的是地图平面上地理数据的离散化数值。在栅格数据中 ,地表被分割为规则排列、相互邻接的方形地块 ,每个地块与一个像元相对应。因此 ,栅格数据的比例尺就

是像元(栅格)的大小与地表相应单元的大小之比。每个像元的属性是地表相应区域内地理数据的近似值,当像元所表示的面积较大时,对长度、面积等的量测有较大影响,因此有可能产生属性方面的偏差。栅格数据记录的是属性数据本身,而位置数据可以由属性数据对应的行列号转换为相应的坐标。

栅格数据的阵列方式很容易为计算机存储和操作,不仅直观,而且易于维护和修改。由于栅格数据的数据结构简单,定位存取性能好,因此在计算机地图制图中发挥着越来越重要的作用。

在计算机地图制图中,各地图要素在二维平面上的栅格数据表示方法如下(如图 2-17 所示)。

点:用一个栅格单元表示;

线:用沿其走向的一组相邻栅格单元表示;

面:用其所覆盖的相邻栅格单元的集合表示。

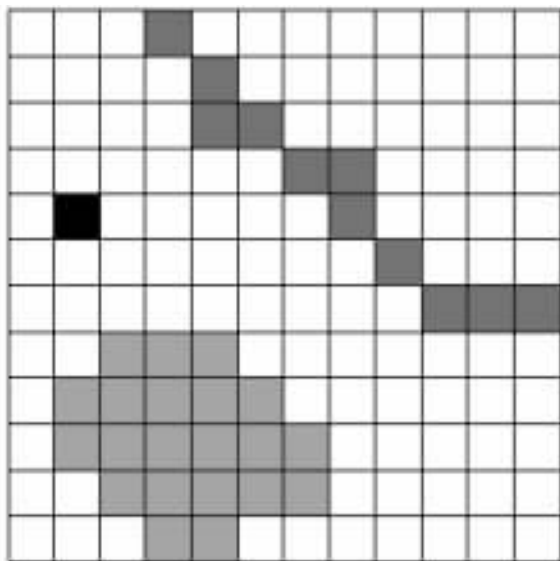


图 2-17 栅格数据的表示

其中,有关相邻栅格单元的概念又分为四方向相邻和八方向相邻(如图 2-18 所示)。一般而言,四方向相邻的栅格图形线画显得粗壮、结实,但阶梯效应(栅格化“抖动”)较为明显,而八方向相邻的栅格图形线画显得纤细,位置过渡较自然,阶梯效应也相对较弱。

2. 栅格数据结构的表示

栅格数据结构是以规则的像元阵列来表示地图上空间地物或现象的分布的数据结构,其阵列中的每个数据表示地物或现象的属性特征。可以说,栅格数据结构就是像元阵列,像元的行列号确定实体的空间位置,像元的值表示实体的类型、等级等属性编码。

(1)简单栅格数据结构。最简单的栅格数据结构是将栅格数据看做一个数据矩阵,逐行记录各像元代码,可以每行都从左到右记录,也可以奇数行从左到右,偶数行从右到左进行记录。如图 2-19 所示的栅格数据矩阵可存储记录为:AAAAABBBBAABBAABB。

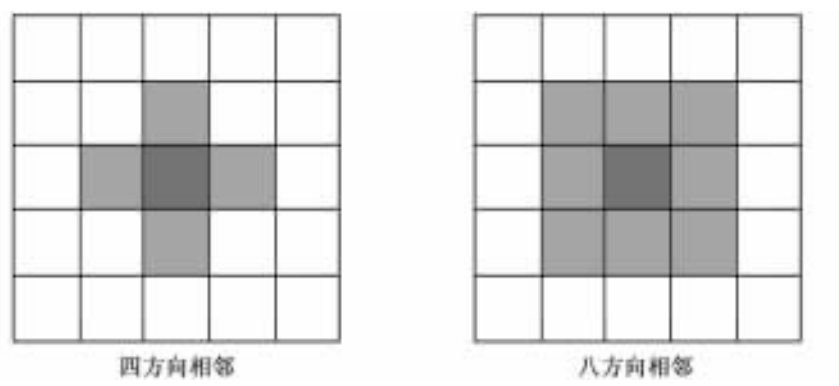


图 2-18 相邻栅格单元的概念

A	A	A	A
A	B	B	B
A	A	B	B
A	A	B	B

图 2-19 栅格数据矩阵

这种记录栅格数据的编码方法又称直接栅格编码,它是最简单、直观而又非常重要的一种栅格结构编码方法。记录这种栅格数据的文件称为栅格文件,通常在文件头中还存在有该栅格数据的长和宽,即行数和列数。其特点是直观和处理方便,但因为没有压缩,需占用较多存储空间。该方法的优点是编码简单,信息无压缩、无丢失,缺点是数据量较大,如图 2-20 所示。

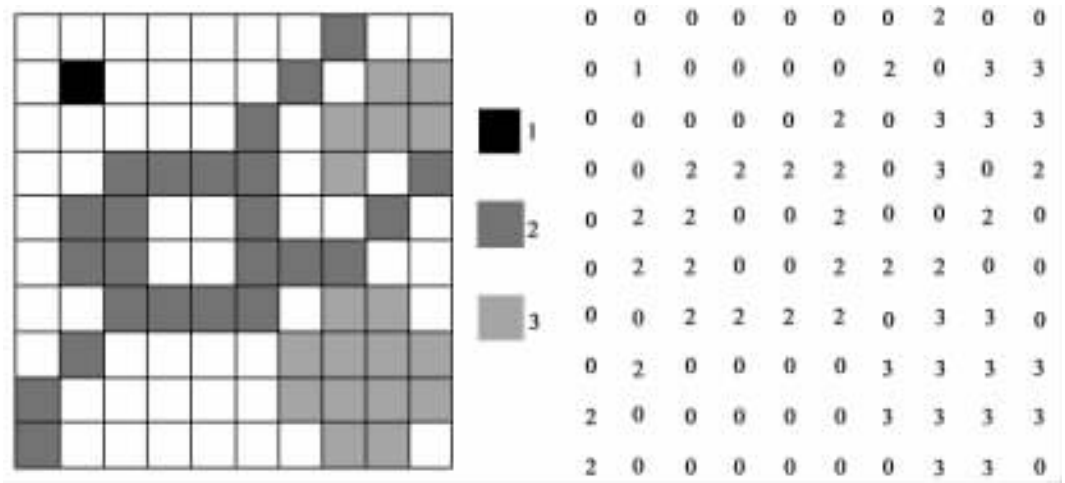


图 2-20 直接栅格编码

3. 栅格数据压缩编码

地理数据具有较强的相关性,相邻像元的值往往是相同的,因此可以通过栅格数据的压

缩存储来节省计算机的存储空间。

栅格数据压缩是重要的编码方法。它的基本思路是:对于一个栅格图形,常常有行(列)方向上相邻的若干栅格单元具有相同的属性代码,因此可采取某种方法压缩那些重复的内容。常见的栅格数据压缩编码方法有链码(Chain Encoding)、游程长度编码(Run-length Encoding)、块状编码(Block Encoding)和四叉树编码(Quadtree Encoding)分别介绍如下。

①链码。链码又称 Freeman 编码或边界编码,主要记录线状地物或面状地物的边界。它把线状地物或面状地物的边界表示为:由某一起始点开始并按某些基本方向确定的单位矢量链。前两个数字表示起点的行列号,从第三个数字开始的每个数字表示单位矢量的方向。单位矢量方向的确定如图 2-21 所示。

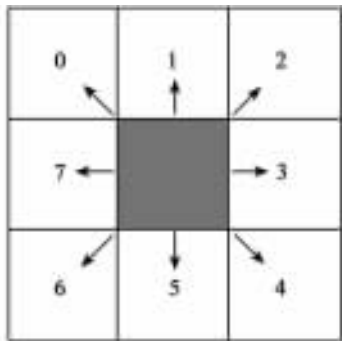


图 2-21 单位矢量方向

对于如图 2-22 中的线状地物和面状地物,其链码分别为:1 4 4 4 4 4 4 3 3 和 8, 3 3 3 4 4 5 7 6 7 0 0 1。

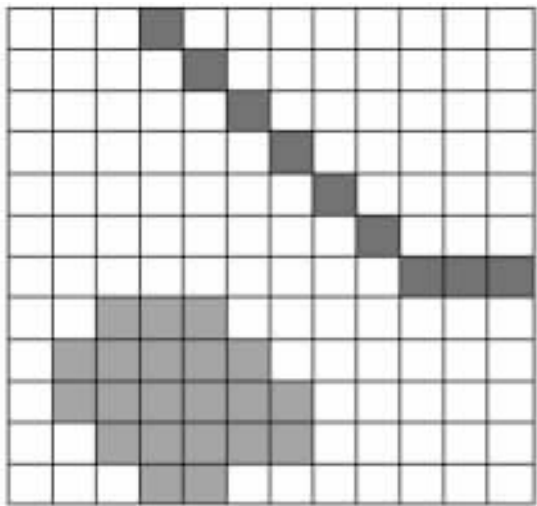


图 2-22 链码示例

该方法的优点是 :有很强的数据压缩能力 ,并具有一定的运算功能 ,如面积、周长等的计算 ,类似于矢量数据结构 ,比较适合于存储图形数据。其缺点是 :叠置运算(如组合、相交等)较难实施 ,对局部的改动会影响整体结构 ,而且相邻区域的边界重复存储。

②游程长度编码。游程长度编码(又称行程编码)的基本思想是 :按行扫描 ,将相邻等值的代码像元合并 ,并记录代码的重复个数。

有两种游程长度编码方法 :

第一种方法是在各行(列)栅格单元的代码发生变化时依次记录该代码以及相同代码重复的个数 ,即 :各行的代码 ,个数 ,代码 ,个数 ,...(如图 2-23 所示)。

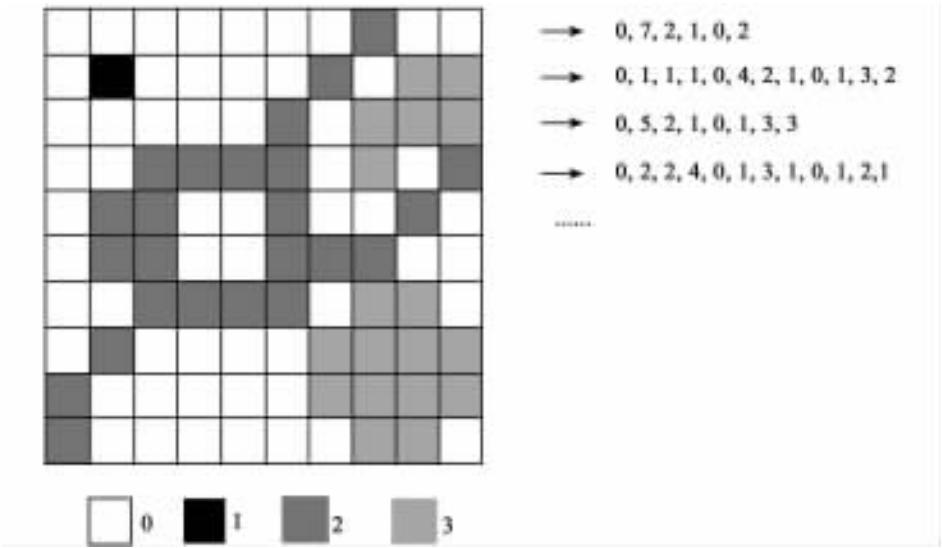


图 2-23 游程长度编码方法一(代码 ,个数)

第二种方法是在各行(列)栅格单元的代码发生变化时依次记录代码以及变化的位置(列数) ,即 :代码 ,位置 ,代码 ,位置 ,...(如图 2-24 所示)。

上述编码方法是按各行分别进行的 ,有时也可采用在行与行之间不间断地连续编码。

对于游程长度编码 ,区域越大 ,数据的相关性越强 ,则压缩比例越大。故其特点是地物属性的变化越少 ,行程越长 ,压缩比例越大 ,即压缩比的大小与图的复杂程度成反比。

游程长度编码的优点是 :压缩效率高(保证原始信息不丢失) ,易于检索 ,叠加、合并等运算操作简单 ,编码和解码的速度快。其缺点是 :只顾及单行单列 ,没有考虑周围其他方向的代码值是否相同 ,压缩受到一定限制。

③块状编码。块状编码是将游程长度编码扩展到二维的情况 ,采用方形区域作为记录单元 ,每个记录单元包含相邻的若干栅格 ,数据结构由初始位置(行号、列号)和半径 ,再加上记录单元的代码组成 ,即 :行号 ,列号 ,半径 ,单元代码 ,行号 ,列号 ,半径 ,单元代码 ,...

例如图 2-25 的块状编码为 :1 1 1 4 , 1 2 2 4 , 1 4 1 7 , 1 5 1 7 , 1 6 2 7 , 1 8 1 7 , 2 1 1 0 , 2 4 1 4 , 2 5 1 4 , 2 8 1 7 , 3 1 1 4 , 3 2 1 4 , 3 3 1 4 , 3 4 1 4 , 3 5 2 8 , 3 7 2 7 ,...

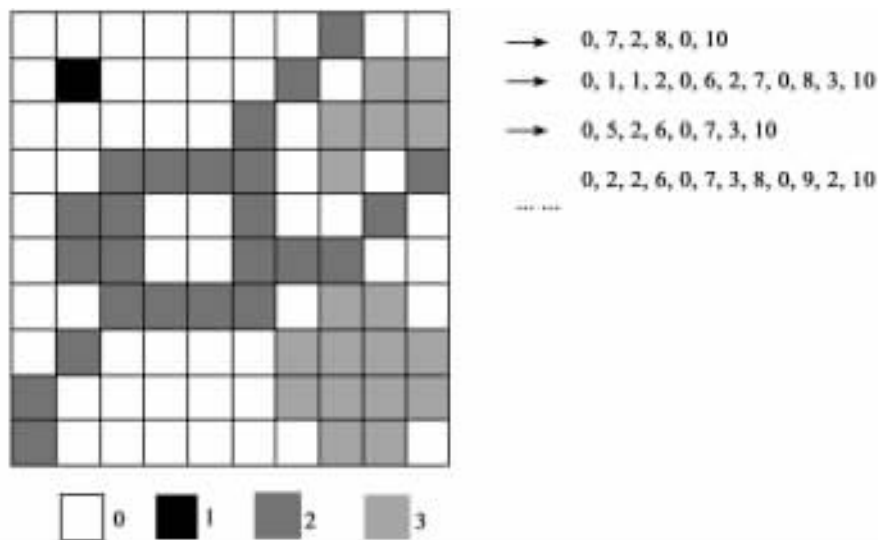


图 2-24 游程长度编码方法二(代码 位置)

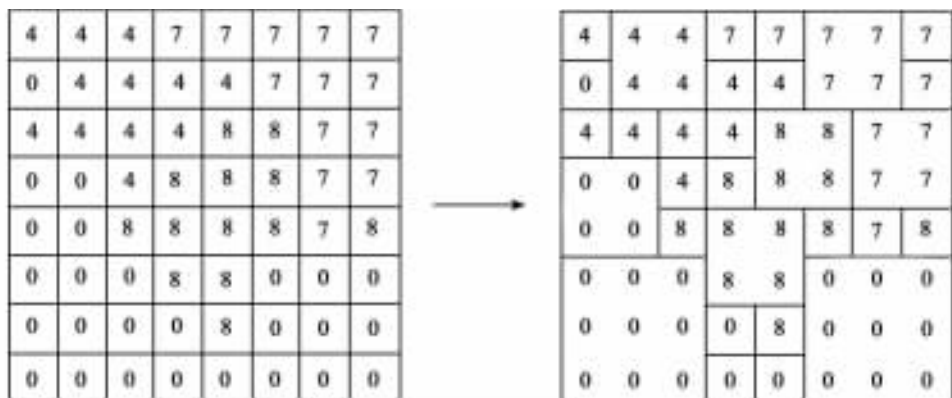


图 2-25 块状编码示例

块状编码对大块成片分布的地物有较高的编码效率。如果图形过于破碎，则效果不是很好。

④四叉树编码。四叉树编码是最有效的栅格数据压缩编码方法之一。其基本思路是将 $2^n \times 2^n$ 个象元组成的图像(不足的用背景补上)所构成的二维平面按 4 个象限进行递归分割,直到子象限的数值单调为止。也即将一幅栅格图像等分为 4 部分,逐块检查其网格属性值。如果某一子区所有栅格的属性值相同,则这个子区不再继续分割,否则还要将这个子区再分割为 4 个子区,这样依次分割,直到每个子区都具有相同的属性值。最后得到一棵四分叉的倒向树,该树最高为 n 级。对于图 2-26 所构成的图像,可用四叉树编码法得到如图 2-27 所示的四叉树。

常规四叉树除了要记录叶节点外,还要记录中间节点,节点之间的联系靠指针。因此,

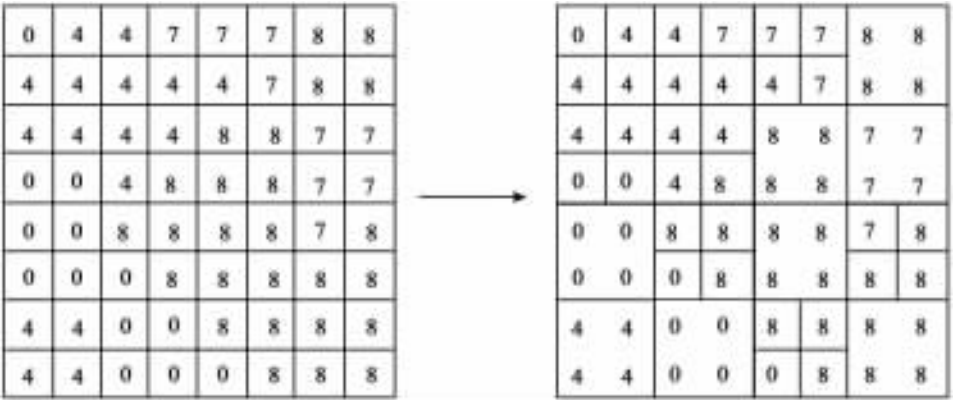


图 2-26 四叉树编码示例

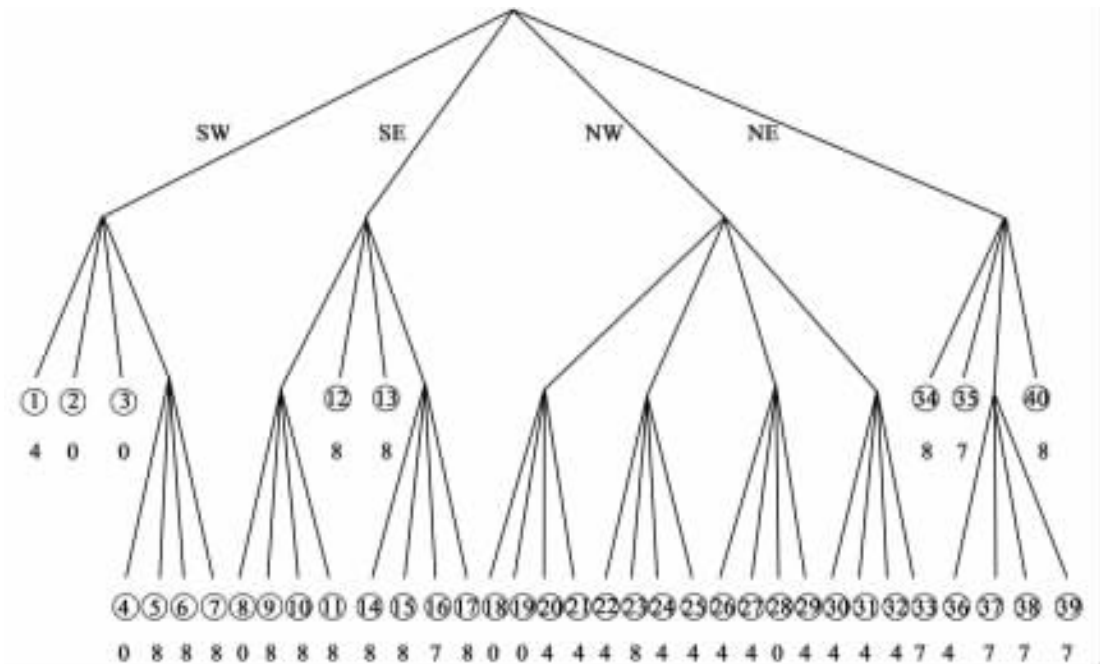


图 2-27 四叉树

为了记录常规四叉树,通常每个节点需要 6 个变量,即父节点指针、4 个子节点指针和本节点的属性值。常规四叉树多用于数据索引和图幅索引等方面。

常规四叉树中,节点所代表的图像块的大小可由节点所在的层次决定,层次数由从父节点移到根节点的次数来确定。节点所代表的图像块的位置需要从根节点开始逐步推算下来,因此常规四叉树的算法是比较复杂的。为了改进常规四叉树的算法及节省存储空间,引入了一些不同的四叉树编码,其中最常用的有线性四叉树编码,介绍如下。

线性四叉树叶节点的编号须遵循一定的规则,这种编号称为地址码,在地址码中隐含了叶节点的位置和深度信息。

线性四叉树编码的基本思想是:不需记录中间节点和使用指针,仅记录叶节点,并用地址码表示叶节点的位置。也即线性四叉树只记录最后叶节点的信息。

线性四叉树编码有四进制和十进制两种,通常使用的是十进制四叉树编码。十进制四叉树编码使用的地址码又称 Morton 码。获取线性十进制四叉树地址码的方法是:首先将栅格阵列数据的行列号转化为二进制数,然后交叉放入 Morton 码中,即为线性十进制四叉树的地址码。

Morton 码的计算如图 2-28 所示。图中显示了对于栅格阵列中第 5 行、第 5 列的 Morton 码的计算过程。

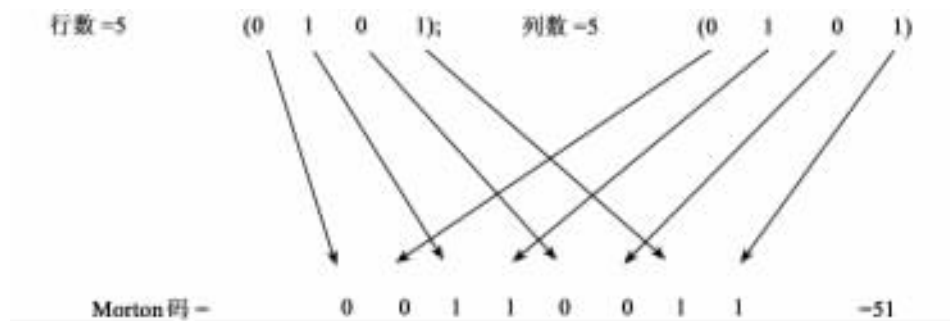


图 2-28 Morton 码的计算示意

同理,针对栅格阵列图像中的每个像元点都可算出一个 Morton 码。对一个 $2^n \times 2^n$ 的栅格阵列,当 $n=3$ 时,其 8×8 栅格阵列图像中的每个像元点相应的 Morton 码如图 2-29 所示。

0	1	4	5	16	17	20	21
2	3	6	7	18	19	22	23
8	9	12	13	24	25	28	29
10	11	14	15	26	27	30	31
32	33	36	37	48	49	52	53
34	35	38	39	50	51	54	55
40	41	44	45	56	57	60	61
42	43	46	47	58	59	62	63

图 2-29 Morton 码

这样就可将用行列表示的二维栅格阵列图像,用 Morton 码写成一维数据,通过 Morton 码就可确定像元的位置。

图 2-30(a)为某二维栅格阵列图像示例(其中 A、B 为像元属性值),其 Morton 码如图

2-30(b)所示。



图 2-30 二维栅格阵列图像及其 Morton 码示例

压缩处理的具体过程为：

第一步 按 Morton 码读入一维数组：

Morton 码 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

像元值： A A A B A B B B A A A A B B B B

第二步 如相邻四像元的值相同 则合并该相邻四像元 只记录第一个像元的 Morton 码：

Morton 码 0 1 2 3 4 5 6 7 8 12

像元值： A A A B A A B B A B

在实际应用中 对上述线性四叉树的编码方法所形成的数据还可游程长度编码进一步压缩 压缩时只记录第一个像元的 Morton 码。故上述数据还可游程长度编码进一步压缩为：

Morton 码 0 3 4 6 8 12

像元值： A B A B A B

因此 把一个 $2^n \times 2^n$ 的栅格阵列图像压缩成线性四叉树的过程可归纳为：首先把栅格阵列图像按 Morton 码读入一维数组 然后比较相邻的 4 个像元 如果一致 则将此 4 个像元合并 只记录第一个像元的 Morton 码 再比较所形成的大块 相同的合并 直到不能合并为止。如有必要 再用游程长度编码进行进一步压缩。

在对线性四叉树的编码解码时 可根据 Morton 码获知像元在图像中的位置(左上角) , 本 Morton 码和下一个 Morton 码之差即为像元个数。知道了像元的个数和像元的位置 就可恢复出图像了。

该编码方法的优点是：压缩效率很高 且压缩和解压缩也比较方便 阵列各部分的分辨率可变 既可减少存储量 又可精确地表示图形结构 易于进行大部分图形操作和运算。该编码方法的缺点是：具有图形编码的不定性 相同形状和大小的多边形可得出完全不同的四叉树结构 不利于形状分析和模式识别。

2.3 两种数据结构的比较及转换

2.3.1 两种数据结构的比较

矢量数据结构和栅格数据结构是模拟地图空间信息的两种不同的方式。

1. 矢量数据结构的特点

在计算机地图制图系统中 矢量数据结构可具体分为点、线、面三类 以构成地理现实空

间中各种复杂的实体。当实体可描述成线或边界时特别有效。矢量数据的结构紧凑,容易定义和操作单个空间实体,冗余度低;具有空间实体的拓扑信息,便于网络分析;图形输出质量好、精度高。但是矢量数据结构较为复杂,导致了操作和算法的复杂化,不能有效地进行点集的集合运算(如叠加);同时,空间实体的查询十分费时,需要逐点、逐线、逐面地查询。

2. 栅格数据结构的特点

栅格数据结构通过空间点密集而规则的排列表示整体的空间现象。其数据结构简单,定位存取性能好;可以与影像和 DEM 数据进行联合空间分析,数据共享容易实现;对栅格数据的操作比较容易。但栅格数据的数据量与格网间距的平方成反比,较高的几何精度的代价是数据量的极大增加。因为只使用行和列来作为空间实体的位置标志,故难以获取空间实体的拓扑信息,难以进行网络分析等操作。栅格数据结构不是面向实体的,各种实体往往是叠加在一起反映出来的,因此难以识别和分离。对点实体的识别需采用匹配技术,对线实体的识别需采用边缘检测技术,对面实体的识别则需采用影像分类技术,这些技术不仅费时,而且不能保证识别的结果完全正确。

3. 矢量数据结构与栅格数据结构的比较

归纳起来看:

矢量数据结构具有“位置明显、信息隐含”的特点。它的数据表达精度高,工作效率高,数据存储量小,输出图形美观;但它操作起来比较复杂,许多分析操作(如叠置分析)用矢量数据结构难以实现。

栅格数据结构具有“属性明显、位置隐含”的特点。它的数据表示直观,易于实现,操作简单,有利于空间信息的分析和处理;但数据表达精度不高,数据存储量大,工作效率低。对基于栅格结构的应用来说,需要根据应用项目的自身特点及其精度要求,恰当地平衡栅格数据结构的表达精度和工作效率两者之间的关系。

两种数据结构的对比见表 2-15。

表 2-15 两种数据结构的对比

比较内容	矢量结构	栅格结构
数据结构	复杂、紧凑,冗余度低	简单,冗余度高
数据量	小	大
图形运算	复杂	简单
图形精度	高	低
影像格式	不一致	一致或接近
数据共享	不易实现	容易实现
拓扑和网络分析	容易实现	不易实现
叠置分析	不易实现	容易实现

通过以上的对比分析可以看出,矢量数据结构和栅格数据结构各有优缺点。两者结合,可以优势互补。在计算机地图制图系统中,有时为了高效地实现某项功能,需要实现两种数据结构的转换,并能同时使用两种数据结构。

2.3.2 两种数据结构的转换

1. 矢量数据向栅格数据的转换

(1)点的转换

首先确定变换的坐标系及栅格单元的大小(如图 2-31 所示)。

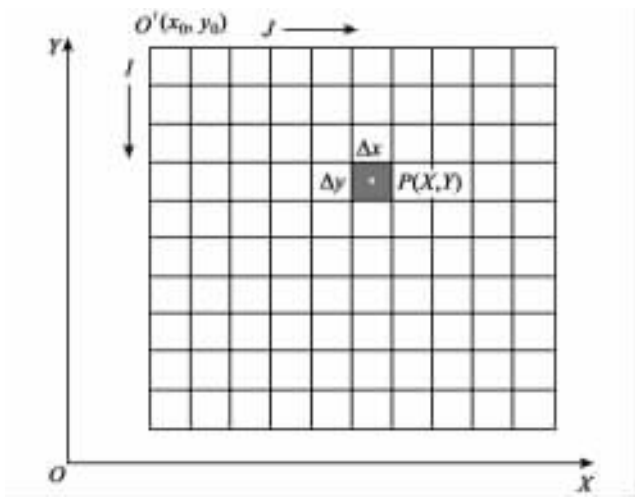


图 2-31 点的转换

图中：

$$\Delta X = (X_{\max} - X_{\min}) / N$$

$$\Delta Y = (Y_{\max} - Y_{\min}) / M$$

式中 X_{\max} 、 X_{\min} 及 Y_{\max} 、 Y_{\min} 分别为栅格阵列在矢量坐标系中的最大值及最小值；M、N 分别为全图网格的行列数。

矢量数据的点到栅格数据的点只是简单的坐标变换。

设点的矢量坐标为 $P(X, Y)$ ，则其转换为栅格行列号的公式为：

$$\begin{cases} I = 1 + \text{INT}\{(Y_0 - Y) / \Delta Y\} \\ J = 1 + \text{INT}\{(X - X_0) / \Delta X\} \end{cases}$$

式中 INT 表示取整。

(2)线的转换

线的矢量数据是由多个直线段数据组成的，因此 线矢量数据向栅格数据转换的核心就是对任一直线段如何将矢量数据转换为栅格数据。常用的转换方法有多种，选择介绍如下。

①八方向栅格化法。设直线段两端点的矢量坐标分别为 $P_1(X_1, Y_1)$ 、 $P_2(X_2, Y_2)$ (如图 2-32 所示)。首先将直线段两端点按上述点转换方法得到相应的行列号，其次求出两端点的行数差和列数差。为避免转换后栅格像元线段可能出现的间断现象，须分两种情况进行：

第一种情况，若行数差大于列数差，则逐行分别地求出该行中心线与直线段的交点，即

$$\begin{cases} Y = Y_i \\ X = (Y - Y_1) \cdot (X_2 - X_1) / (Y_2 - Y_1) + X_1 \end{cases}$$

式中 Y_i 为该行中心线的 Y 坐标。

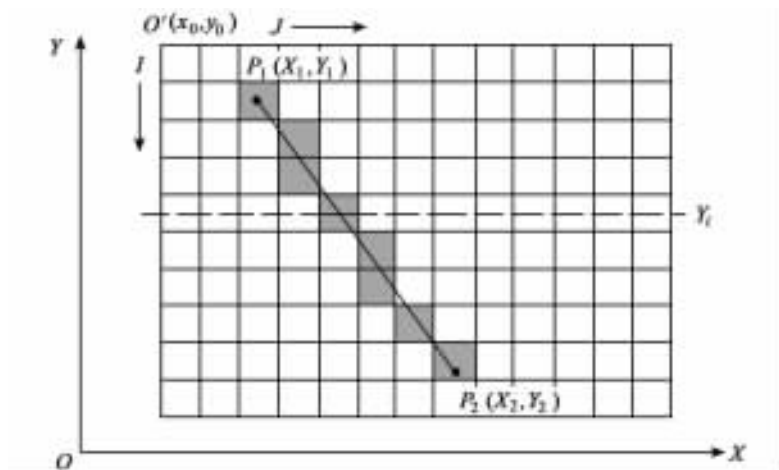


图 2-32 八方向栅格化法

然后,再将所求出的交点按上述点转换方法得到相应的行列号。

第二种情况,若列数差大于行数差,则逐列分别地求出该列中心线与直线段的交点,即

$$\begin{cases} X = X_i \\ Y = (X - X_1) \cdot (Y_2 - Y_1) / (X_2 - X_1) + Y_1 \end{cases}$$

式中 X_i 为该列中心线的 X 坐标。

之后,再将所求出的交点按上述点转换方法得到相应的行列号。

②分带法。分带法的基本思想是:根据直线段的情况按行(或列)分带,计算各个分带的起点和终点列号(或行号),再用本线段的栅格属性值去填充起点和终点之间的栅格。

通常,当 $|X_2 - X_1| \geq |Y_2 - Y_1|$ 时,按行分带;

当 $|X_2 - X_1| < |Y_2 - Y_1|$ 时,按列分带。

下面以第一种情况为例进行介绍(第二种情况类似,仅将行列互换即可)。

设直线段如图 2-33 所示。这时应按行分带,线段的首、末点所在的列号应分别作为第一行的起点和最后一行的终点。对其他各行,可计算该行所在栅格的上下两条栅格线与直线段的交点,即可求出该行的起点和终点列号。具体方法如下:

设当前处理行为 i (如图 2-33 所示),像元边长为 N ,则起点 b 的列号

$$J_b = \text{INT}\{((Y_0 - (i - 1) \cdot N - Y_1)(X_2 - X_1) / (Y_2 - Y_1) + X_1 - X_0) / N\} + 1$$

终点 e 的列号

$$J_e = \text{INT}\{((Y_0 - i \cdot N - Y_1)(X_2 - X_1) / (Y_2 - Y_1) + X_1 - X_0) / N\} + 1$$

一般情况下,本行的终点也即下一行的起点。特殊情况下,当 $|Y_2 - Y_1| = 0$ 时,线段的首、末点所在的列号即作为起点和终点的列号。

(3)面的转换

面的矢量数据由其轮廓多边形数据组成,因此,面矢量数据向栅格数据转换的核心就是将多边形矢量数据及其内部区域转换为栅格数据。有多种转换方法,选择介绍如下。

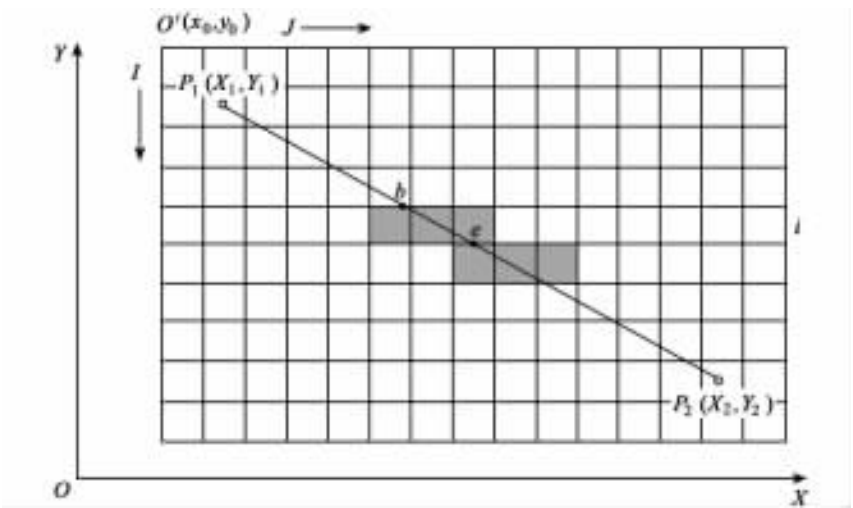


图 2-33 分带法

①行填充法。行填充法的基本思想是：根据面(多边形)的矢量数据，计算栅格图形每行中心线与面边界(多边形)的交点，对交点进行排序、配对，并用本面域的栅格属性值去填充每对交点(含)之间的栅格。

图 2-34 是该算法的示意。

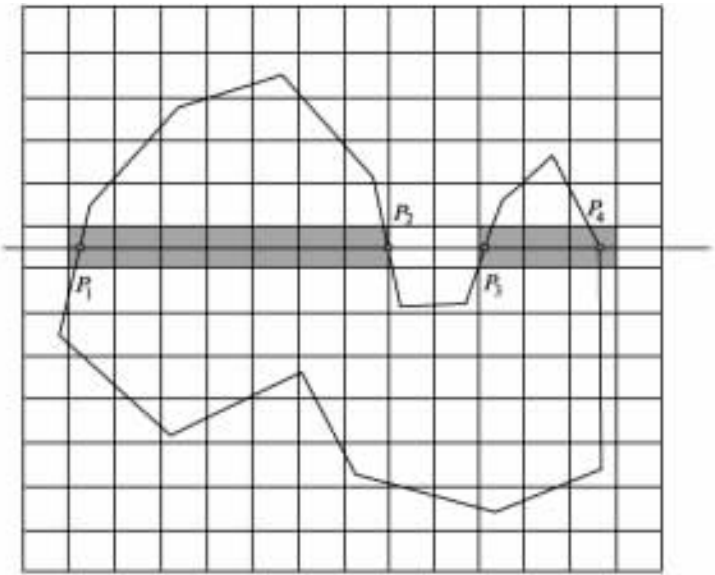


图 2-34 行填充法

其中 P_1 、 P_2 、 P_3 、 P_4 是该行中心线与多边形的交点。要注意的是对于第一行和最后一行，应分别用第一行的下栅格线和最后一行的上栅格线与多边形求交点。

②点填充法。该方法的基本思想是:首先将多边形边界转换为栅格数据,然后由多边形内部的一个点(种子点),向其4个方向的相邻点扩散。判断新加入的点是否在多边形边界上,如果是,不作为种子点,否则当做新的种子点,直到区域填满,无种子点为止。

图 2-35 是该算法的示意。

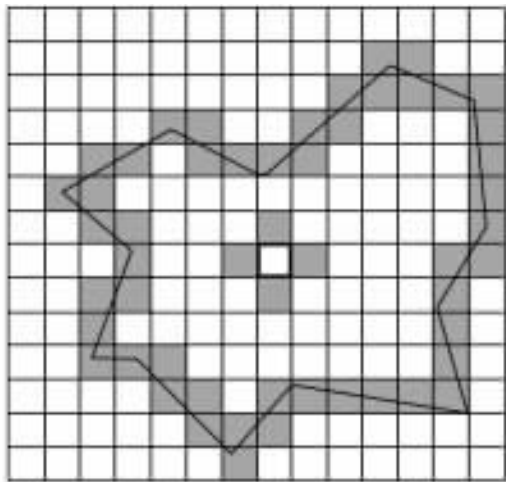


图 2-35 点填充法

2. 栅格数据向矢量数据的转换

为了便于进行栅格数据向矢量数据的转换,需要对原始栅格图像进行二值化处理。这是因为通常扫描后的原始图像是以不同灰度级存储的,而太多的灰度级不便于进行矢量化处理,故需进行压缩,通常压缩为两级(0 和 1),这个过程就称为二值化。

所谓二值化,就是在栅格图像灰度级的最大值和最小值之间选取一个阈值,当灰度级小于阈值时,取值为 0;当灰度级大于阈值时,取值为 1。以下假定在栅格数据矢量化转换过程中使用的初始栅格图像都是二值化过的。

(1) 点的转换

栅格数据点到矢量数据点的坐标转换实际上是矢量数据的点到栅格数据的点变换的逆运算。

设栅格点行列号为 I、J,相应栅格中心点的矢量坐标为 $P(X, Y)$ (如图 2-36 所示),则其转换公式为:

$$\begin{cases} X = X_0 + (J - 0.5) \cdot \Delta X \\ Y = Y_0 - (I - 0.5) \cdot \Delta Y \end{cases}$$

(2) 线的转换

线状栅格图像通常具有一定的宽度且往往宽窄不一,不便于直接进行矢量化,须在细化处理,提取中轴(骨架)线的基础上再矢量化。因此,线的矢量化转换包括细化、跟踪和拓扑化等过程。

①细化。细化就是将线状栅格像元阵列逐步剥除轮廓边缘的点,使之成为线画宽度只有一个像元的中轴线(骨架)图形。细化后的图形骨架既保留了原图形的绝大部分特征,又

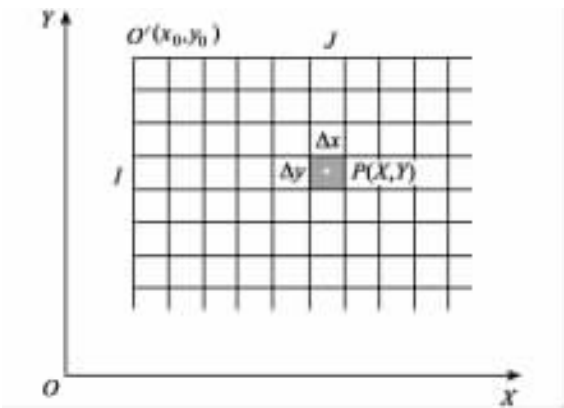


图 2-36 栅格点到矢量点的坐标转换

便于下一步的矢量化处理。

对图像进行细化处理 ,应符合下列基本要求 :

- 线宽只有一个像元 ;
- 细化后的骨架线应是原线画的中轴线 ;
- 保持原有线画的连通性及特征。

有关细化的算法有很多 ,各有优缺点 ,主要过程包括 :确定需细化的像元集合 ,移去不是骨架的像元 ,重复该过程直到仅剩骨架像元。简要介绍如下。

• 经典的细化算法

经典的细化算法是通过 3×3 的栅格像元阵列来确定如何细化的。其基本原理是 ,在 3×3 的像元阵列中 ,凡是去掉后不会影响原栅格图像拓扑连通性的像元都应该去掉 ;反之 ,则应保留。 3×3 的像元阵列共有 2^8 即 256 种情况 ,但经过旋转 ,去除相同情况 ,共有 51 种情况 ,其中只有一部分是可以将中心点去掉的 ,如图 2-37 中 ,(a)、(b)的中心点是可去掉的 ,而(c)、(d)的中心点是不可去掉的。通过对每个像元点进行如此反复处理 ,最后可得到应保留的骨架像元。

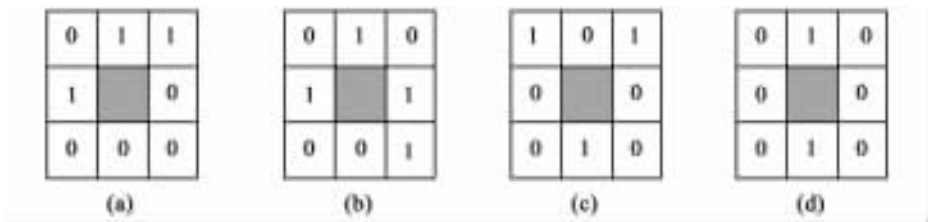


图 2-37 中心像元点去留举例

• 边缘跟踪与剥皮算法

边缘跟踪与剥皮算法的基本思想是 :以线状栅格影像边缘的一个像元为起点 ,检测其八邻域 ,以找到下一个相邻的边缘像元作为新的起点 (即跟踪) ,并确定本像元点是否应去掉 (即剥皮) ,重复该跟踪和剥皮过程 ,直到仅剩骨架像元。

算法举例如下：

在图 2-38(a)、(b)中,从正北方向起按顺时针检测中心点的八邻域,最后检测到的(非起始检测点)像元值为“1”的点即为被跟踪到的新边缘点。

在确定本中心像元点是否应剥掉时,需要计算除中心像元外 8 个邻域的连通像元块数。若连通像元块数等于 1,则该中心像元应剥掉(如图 2-38(a)所示);若连通像元块数大于 1,则该中心像元应保留(如图 2-38(b)所示)。

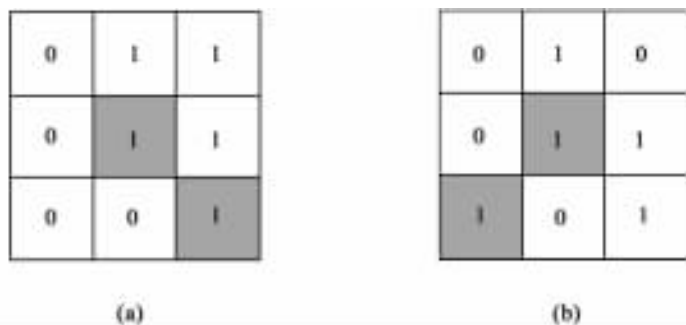


图 2-38 边缘跟踪剥皮法

②跟踪骨架线。细化后的二值线状栅格像元图像形成了骨架线图,跟踪就是把骨架线转换为矢量图形的坐标序列。其主要步骤为：

第一步 从上向下,从左向右搜索骨架线起始点,并记下其按上述点转换方法所得的(X,Y)坐标。

第二步 按该点的 8 个邻域方向跟踪下一点,若没有,则本条线的跟踪结束,转第一步进行下一条线的追踪;否则,记下该点按上述点转换方法所得的(X,Y)坐标。

第三步 把搜索点移到新取的点上,转第二步。

在跟踪过程中,对已跟踪的点应作标记,以防止重复跟踪。

③拓扑化。为了在跟踪时得到拓扑化的矢量数据,需找出线的端点、节点和孤立点等。

- 端点 8 个邻域中只有 1 个值为 1 的像元,如图 2-39(a)所示。
- 节点 8 个邻域中有 3 个或 3 个以上值为 1 的像元,如图 2-39(b)所示。
- 孤立点 8 个邻域中没有值为 1 的像元,如图 2-39(c)所示。

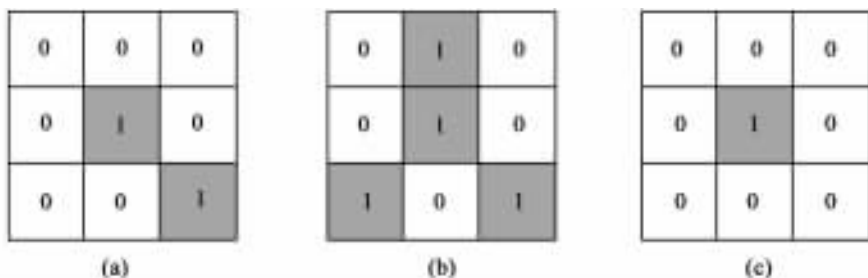


图 2-39 拓扑化

有了这些信息 ,在跟踪时就可形成节点和弧段等拓扑要素 ,从而得到拓扑化的矢量数据。

(3)面的转换

面状栅格数据向矢量数据的转换实际上只需进行轮廓的转换即可 ,其基本思想是 :以面状栅格影像边缘(内沿或外沿)的一个像元为起点并记下其按上述点转换方法所得的(X , Y)坐标 ,检测其 8 个邻域 ,以找到(跟踪)下一个相邻的边缘像元作为新的起点 ,记下其(X , Y)坐标 ,重复该跟踪过程直到结束(封闭)。

具体方法可参见边缘跟踪与剥皮算法。

第3章 地图数据的采集和地图数据库

3.1 地图数据源及数据分类编码

3.1.1 数据源的种类

计算机地图制图的数据源,是指建立计算机地图制图系统的数据库所需的各种数据的来源。其种类多种多样,主要包括地图、遥感影像数据、实测数据、文字与统计资料、已有数字数据等。

1. 地图

各种现有纸质地图是大多数计算机地图制图系统的主要数据源。

地图的内容丰富、直观、一览性强,不仅含有实体的类别和属性,而且含有实体间的空间关系。地图数据主要通过对地图的跟踪和扫描数字化获取。在使用地图数据时,应考虑到纸质地图可能因存放条件不同而引起的变形,因此要进行必要的纠正。另外,还应注意地图投影所引起的变形,在必要时进行投影变换。

2. 遥感影像数据

遥感影像数据是计算机地图制图的重要数据源。

遥感影像数据具有很强的现势性,是一种大面积、动态、近实时的数据源。遥感影像数据含有各种丰富的专题资源信息,是计算机地图制图系统数据更新的重要手段。

3. 实测数据

实地测量(包括GPS全球定位系统)等获取的数据可以通过转换直接进入计算机地图制图系统,因此实测数据也是计算机地图制图系统的一个重要数据来源和数据更新的强有力手段。

4. 文字与统计资料

文字资料是指各部门的有关法律文档、行业规范及条例等。

统计资料是指国家各部门和机构拥有的不同领域的大量统计数据(如人口数据、国民经济数据等)。这些都是计算机地图制图系统中重要的属性数据源。

5. 已有数字数据

已有数字数据是指从其他已建成的系统和数据库中获取的相应数字图形数据和属性数据。随着各种与地图有关的信息系统和数据库(包括多媒体数据)的建立以及规范和标准的推广,不同系统间的数据共享和可交换性越来越强。它已成为计算机地图制图的又一重要数据源。

3.1.2 数据的分类编码

地图数据主要有几何数据和属性数据之分。由于在计算机地图制图系统中几何数据是地理数据的主导因素,前者的数据类型、形式往往决定了后者的类型和形式,所以人们常说把地理实体分为矢量数据和栅格数据两种类型。实际上这是根据几何数据在计算机中存储、处理的逻辑结构,即数据结构来分类的。这里,主要介绍地图中地理实体的属性数据编码。

1. 属性数据的概念

属性数据是用来描述实体的属性(如质量、数量、等级等)特征的数据。例如,河流可以数字化为矢量表示的一串有序的(X,Y)坐标或栅格表示的一组连续的像元,而河流的属性数据则是指河流的宽度、等级、流量等。

属性数据可以和几何数据存放在一起,也可以单独存储在数据库中,通过惟一的标志码与相应的几何数据联系起来。如果属性数据的数据量较大,则常与几何数据分开输入。可以先将属性数据输入到一个文件,经编辑、检查后存入数据库。

在计算机地图制图系统中,通常把那些与几何数据有密切联系的属性数据用编码的形式表示,并与几何数据一起管理起来。编码的实质就是将属性信息转换成数字编码(代码)。为此需对所要表示的属性信息进行分类分级。

属性数据的分类分级与编码是许多地理信息进入数据库前必须完成的一项任务。分类分级与编码既有紧密的联系,又有较大的区别。建立要素的分类分级体系是认识地理要素的基本方法,编码则是为了获得科学的存储、管理和快速查询地图数据的效果。两者的联系体现在编码反映了分类分级体系的特征,而分类分级系统可以从编码中产生。

虽然目前人们对各种地理要素的研究深度与广度不同,提供的分类分级与编码的依据也不一样,还没有一个统一的分类编码方案,但还是有一些应共同遵循的原则和常用方法。

2. 分类分级的原则和方法

(1) 分类的原则和方法

分类是人们认识事物的一种方法,是将具有共同属性特征的事物(或现象)归并在一起,而把具有不同属性特征的事物(或现象)分开的过程。

分类的基本原则是:

- 科学性:选择事物或现象最稳定的属性和特征作为分类的依据。
- 完整性和系统性:应形成一个完整的分类体系,低级的类应能归并到高级的类中。
- 实用性:应考虑对信息分类所依据的属性特征的获取方式和获取能力,应与有关的标准协调一致。
- 可扩性:应能容纳新增加的事物和现象,而不致打乱已建立的分类系统。

分类中最常用的方法是层次分类法。

层次分类法是将初始的分类对象按所选定的若干个属性或特征依次分成若干层目录,并编排成一个有层次的、逐级展开的分类体系。其中,同层次类目之间存在并列关系,不同层次类目之间存在隶属关系,同层次类目互不交叉、互不重复。层次分类法的优点是层次清晰,使用方便,缺点是分类体系确定后,不易改动,当分类层次较多时,代码位数较长。

(2) 分级的原则和方法

分级是对事物(或现象)的数量(或特征)进行等级的划分,主要过程为分级数和分级界线的确定。基本原则和方法如下:

- 分级应符合数值估计精度的要求。分级数多,数值估计的精度就高,在满足精度的前提下,应尽可能选择较少的分级数。在任何一个等级内都必须有数据,任何数据都必须落在某一个等级内。
- 分级应符合数据的分布特征。对于呈聚群状分布的数据,应以数据的聚群数作为分级数。应使级内差异尽可能小,各级代表值之间的差异尽可能大。分级界线应当凑整,并尽可能采用有规则变化的分级界线。
- 分级还应顾及可视化的效果。因等级的划分在计算机地图制图中要以图形的方式表示出来,考虑到人对图形符号等级的感受,分级数不宜超过8级。
- 分级时主要使用数学方法,如比例分级、最优分割分级、数列分级等。有时也可以定性地分级,如按行政级别分为国家、省、市、县、乡等。另外,如有统一规定的分级方法时,应采用规定的分级方法。

3. 数据编码

这里,数据编码是指确定属性数据代码的过程。代码是一个易于被计算机识别与处理的符号,是计算机地图制图中定性查询信息的主要依据和手段。

编码的基础是分类分级,而编码的结果就是代码。

(1) 代码的类型和功能

代码有数字、字母以及数字和字母混合三种类型。其中,数字代码是用若干个阿拉伯数字表示对象的代码,其特点是结构简单,便于计算机处理,但直观性较差;字母代码是用若干个字母表示对象的代码,其特点是便于识别,易于记忆,但比数字代码多占用计算机空间;数字、字母混合代码是用数字和字母混合组成的代码,兼有数字类型和字母类型的优点,但处理较复杂。

计算机地图制图中的代码又可以分为两种,即分类码和标志码。

分类码(特征码)是指根据地理信息分类体系设计的专业信息的分类代码,表示不同类别的数据。在计算机定性查询信息时,可以根据它查出所需类别的全部数据。

标志码(识别码)是指在分类码的基础上,对每类数据设计出其全部(或主要)实体的识别代码,表示某一类数据中的某个实体,以实现诸如某个乡镇、某条道路等进行个体查询。标志码是联系实体的几何信息和属性信息的关键字。

代码的功能体现在三个方面,即:代码表示对象的名称,是对象的惟一标志;代码也可作为区分分类对象类别的标志;代码还可作为区别对象排序的标志。

(2) 编码的原则和方法

在进行编码时一般应遵循以下基本原则:

- 科学性:代码要与科学的分类体系相适应,要便于数据库管理。
- 惟一性:一个代码惟一地表示一类对象。
- 完整性和可扩充性:代码必须完整地反映分类体系,并留有足够的备用代码,以适应扩充的需要。
- 适用性和规范性:代码应尽可能反映对象的特点,结构简单,便于记忆,格式规范、统一。

根据上述原则 ,可以有多种分类编码方法 ,简要介绍如下。

①国土基础信息分类编码。该编码有 9 大类 ,再依次分小类和一、二级代码及标志码。分类代码由 6 位数字组成 ,从左到右依次为 :大类码、小类码、一级代码、二级代码、标志码 (如图 3-1 所示)。

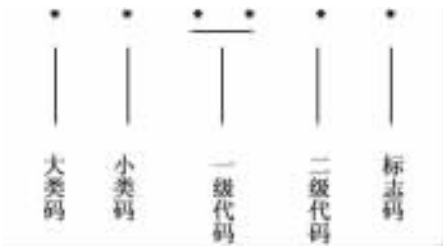


图 3-1 国土基础信息分类编码

②土地利用分类编码。农业部全国农业区划委员会制定的土地利用分类系统包括 8 个一级类型和 46 个二级类型 ,全面描述了我国现有的土地利用状况。其编码采用三位数字 ,左起第一位表示一级分类 ,后两位数字表示二级分类。土地利用分类编码举例见表 3-1。

表 3-1 土地利用分类编码举例

分类系统	代 码
I. 耕地	100
1 灌溉水田	101
2 望天田	102
3 水浇地	103
4 旱地	104
5 菜地	105
II. 园地	200
1 果园	201
2 桑园	202
3 茶园	203
4 橡胶园	204
5 其他园地	205
III. 林地	300
1 有林地	301
2 灌木林	302
3 疏林地	303
4 未成林造林地	304
5 迹地	305
6 苗圃	306
IV. 牧草地	400
1 天然草地	401

续表

分类系统	代 码
2 改良草地	402
3 人工草地	403
V. 居民点及工矿用地	500
1 城镇	501
2 农村居民点	502
3 独立工矿用地	503
4 盐田	504
5 特殊用地	505
VI. 交通用地	600
1 铁路	601
2 公路	602
3 农村道路	603
4 民用机场	604
5 港口码头	605
VII. 水域	700
1 河流水面	701
2 湖泊水面	702
3 水库水面	703
4 坑塘水面	704
5 苇地	705
6 滩涂	706
7 沟渠	707
8 水上建筑物	708
9 冰川及永久积雪	709
VIII. 未利用土地	800
1 荒草地	801
2 盐碱地	802
3 沼泽地	803
4 沙地	804
5 裸土地	805
6 裸岩、石砾地	806
7 田坎	807
8 其他	808

3.2 地图数据的采集

地图数据具有定位、定性和时间性三大特点。与此相适应 ,地图数据包含了几何数据 (空间数据)和属性数据两大类。其中 :几何数据描述了地理物体或现象的空间分布 ,即定

位特性,而属性数据描述了地理物体或现象的种类、质量和数量等特征,即定性特性。有时也可将时间信息隐含在属性数据中表示(如时令河等)。

地图数据采集的任务就是将地理实体的几何数据和属性数据输入到地图数据库中去。

3.2.1 几何数据的采集

由于地图数据源的多样性,在几何数据采集中,可以使用不同的方法。对于由外业测量仪器获取的几何数据,只需把测量仪器的数据传输进入数据库即可。如果是已有的数字数据(包括栅格形式的数据等),可经过转换后输入数据库。而从遥感影像上提取专题信息,必须使用几何纠正、图像变换、影像分类和信息提取等技术,这些主要属于遥感图像处理的内容。这里主要介绍对纸质地图(图件、航片等)的数据采集,其中包括手扶跟踪数字化与扫描跟踪数字化两种方式。

1. 手扶跟踪数字化

手扶跟踪数字化是在数字化软件的支持下应用手扶跟踪数字化仪来完成的。它通过记录数字化仪面板上点的平面坐标来获取矢量数据,是目前广泛采用的一种地图数据采集方式。

手扶跟踪数字化的基本过程是:先将所需数字化的地图(图件、航片等)固定在数字化板上,然后设定数字化范围,设置代码清单,输入有关参数及选择数字化方式,最后按地图要素的类别分别实施图形数字化即可。具体步骤如下:

(1)确定数字化方案

数字化方案是数字化过程的指导性文件。数字化方案的制定与计算机地图制图系统的数据结构、数据库和处理功能有很大的关系。数字化方案的内容主要包括:

①选择分类编码系统。分类编码系统的选择与用户的目的、要求有关。尽量选用成熟、规范的分类型编码体系,必要时进行扩充、修改。

②确定各类地图要素的跟踪输入方式。各类地图要素的跟踪输入方式应有明确的说明。例如:

对点状地图要素,应明确其定位点(中心、左上角、左下角等)的位置,以及有向点的定位等;对线状地图要素,应确定其定位线(轴线、走向等),有时还需确定是分段输入还是连续输入;对面状地图要素,应确定在输入其轮廓多边形时是否同时输入内点信息或拓扑关系等。

(2)设置代码(特征码)清单

代码清单内容的定义应与所选定的分类编码系统相匹配。代码清单是指由地图要素代码构成的格网状清单,每种类型的地图要素代码占据清单中的一格。代码清单可以安放在数字化仪台面上有效范围内,在数字化时只要点击代码清单区中地图要素代码所在的网格,就可知道所数字化要素的编码。

(3)设定有关参数

根据不同的数字化仪和数字化软件确定有关参数。其中包括确定数字化仪与主机的通信接口,设置数字化标志器上各按键的功能,确定地图的比例尺、图幅号、成图时间、坐标系、投影类型以及选择数字化的方式(点方式、流方式)等。

(4) 手扶跟踪操作

手扶跟踪操作由操作员按地图要素分类逐项进行。数据的质量和可靠性在很大的程度上取决于操作员的经验和技能。操作员的经验和技能主要体现在是否能选择最佳点位来数字化地图上的点、线、面要素及判断标示器十字丝与目标重合程度等的的能力。

在手扶跟踪数字化过程中,应及时发现错漏并加以改正。

2. 扫描跟踪数字化

(1) 扫描跟踪数字化的基本思想

扫描跟踪数字化的基本思想是:首先通过扫描将地图(图件等)转换为栅格数据,并进行适当处理,然后采用模式识别技术识别出点和注记,最后使用栅格数据矢量化的技术追踪出线 and 面,并根据地图内容和地图符号的关系,自动给矢量数据赋属性值。

扫描跟踪数字化与手扶跟踪数字化相比,具有速度快、精度高、自动化程度高等优点,是目前较为先进的地图数字化方式,也是今后的发展方向,正在逐步成为计算机地图制图中最主要的地图数字化方式。

虽然扫描跟踪数字化大大提高了地图数字化的自动化程度,但离实现完全自动化还有很长的距离,还须作巨大的努力。

(2) 扫描跟踪数字化软件

目前使用的扫描跟踪数字化软件都是半自动化的,还需做相当的人机交互工作。例如,经常需要进行一些人机交互编辑,以保证自动跟踪和识别的进行(如处理断线、确定属性值等),有时甚至要人工在屏幕上进行与手扶跟踪数字化类似的操作,即进行手扶屏幕跟踪数字化。

地图扫描跟踪数字化方式自动化程度的高低取决于扫描跟踪数字化软件功能的强大与否。一个较为完善的扫描跟踪数字化软件应具备如下的主要功能:

①地图的扫描输入和分版功能。能使用各种扫描仪把地图通过扫描数字化为栅格数据,在需要时能够将所扫描的彩色地图栅格数据分解成不同要素版的单色图像栅格数据,以便于识别和跟踪。

②图形、图像编辑功能。必须具有基本的图形、图像编辑功能。

③符号及注记的自动识别功能。能对大部分点状符号和注记进行自动识别。

④要素的矢量化功能。具有相当程度的自动提取线状要素中心线及面状要素轮廓线的功能。

⑤属性编码的设置与赋值功能。能根据不同的要求进行地图要素的编码设置。能在一定程度上自动对已数字化的要素根据其符号特征赋予相应的编码。

⑥屏幕跟踪数字化功能。具有在必要时能人工在屏幕上进行跟踪数字化的功能。

根据目前的技术水平,上述功能要完全实现自动化仍有相当困难,在数字化的过程中还经常需要进行人机交互。例如,对某些特殊符号的识别、属性编码的赋值以及诸如在多条线的交叉点找到粘连及断开处,原实体连续但图形中断处(被符号及注记等压断),等等,均需人机交互处理。

3.2.2 属性数据的采集

地图数据主要包括几何数据和属性数据,属性数据主要用来描述地图要素的属性特征

(如质量、数量、等级等)。例如,在计算机地图制图中,河流可以数字化为矢量表示的一串有序的坐标或以栅格表示的一组连续的像元,即相应的几何数据,而河流的属性数据则是指河流的宽度、等级、流量等,这些数据都与河流这一空间实体相关。属性数据可以通过给予一个公共标志符与空间实体的几何数据联系起来。

1. 属性数据的采集方法

属性数据的采集较为简单,主要使用键盘输入的方式。分两种情况:

①当属性数据的数据量较小时,可以在输入几何数据的同时,根据数字化软件的提示用键盘输入。

②当属性数据的数据量较大时,可与几何数据分开输入,经检查修改后再转入到数据库中。

在计算机地图制图系统中,应提供方便灵活的手段以对属性数据进行增加、删除、修改等操作。

2. 属性数据与几何数据的联系方法

当地图要素的属性数据与几何数据分开输入时,把属性数据与几何数据联系起来的方法是在属性数据与几何数据之间建立一个惟一的公共标志码。该标志码可以在输入几何数据或属性数据时手工输入,也可以由系统自动生成。

当几何数据或属性数据没有公共标志码时,必须通过人机交互的方法,如选取一个地图要素,再指定其对应的属性数据来确定两者之间的关系,并生成公共标志码。

在数据库中,只有当地图要素的几何数据与属性数据有一共同的数据项时,才能将几何数据与属性数据自动地连接起来。在地图要素的几何数据与属性数据连接起来以后,就可以进行各种诸如定性、定位等查询操作与运算了。

3.3 地图数据的编辑和数据质量分析

3.3.1 数据编辑

在数据采集的过程中,无论是地图的几何数据数字化还是属性数据数字化,都不可能完全正确,常见的错误主要有:地图要素的遗漏、地图要素的重复或多余、几何数据的位置不正确或不完整、属性数据的遗漏、重复、几何数据与属性数据的连接错误等。因此,必须对所采集的数据进行编辑修改。为此,首先要显示数据,然后才能进行编辑修改。

1. 数据显示

数据显示是指在屏幕上或通过绘图机把有关信息显示出来,以便与原图进行比较,找出数字化过程中的差错,加以编辑修改。为了能对采集到的数据进行较为完整的显示,计算机地图制图系统应提供如下基本功能:

①图形显示功能。图形显示功能是指能将所采集的几何数据(包括注记)显示在屏幕上,并能进行放大、缩小、漫游等操作。

②符号设置及符号库功能。符号设置是指为每一类地图要素指定相应的地图符号,其中包括符号的形状、色彩、尺寸、图案等。为了能以不同符号表示不同类型的几何数据,计算机地图制图系统必须具有符号设计与符号库建库功能。

③图形输出功能。图形输出功能是指能通过绘图机把数字化的地图以符号化的形式输出。

2. 数据编辑修改

数据编辑修改的前提是数字化定位(参见本书第4章)。所谓数字化定位是指一旦发现图形上的错误,就可以在数据库中找到相应的数据。定位的依据主要是坐标、代码及序列号等,这些都与数据结构及数据库有关。

数据编辑修改是通过发布编辑命令来进行的。编辑的方式主要有两种,即批处理方式(如投影变换、节点匹配等)和人机交互处理方式。编辑系统一般具有以下主要功能:

①投影变换。为了保持地图数据库中几何数据的一致性,必须将原地图投影下的几何数据转换为指定投影下的几何数据。

②数据匹配。数据匹配包括节点匹配和数字图形接边。

③数据查询。可以由几何数据查询其属性信息,也可以由属性信息查询其几何数据。

④图形编辑。强大的图形编辑功能是几何数据的修改得以实施的基础。图形编辑功能应具有良好的人机交互界面和较快的响应速度,应能对点、线、面数据进行增加、删除、移动、分割、合并等修改。

⑤属性编辑及注记配置。属性编辑包括编码的增删等修改;注记配置重要的是确定其定位点,并确定相关的字体、大小、间隔、色彩、排列等。

3.3.2 数据质量分析

1. 数据质量的概念

(1)地图数据质量的基本含义

地图数据具有定位、定性和时间性三大特征。地图数据的质量是指用该数据来表达其三大特征时所能达到的准确性、一致性和完整性,以及它们之间统一性的程度。

(2)地图数据质量的基本内容

地图数据质量的基本内容主要有以下几个方面:

①准确性。准确性主要指地图数据的几何位置精度和属性精度,其中:位置精度反映几何数据的质量,如平面精度、高程精度和数学基础等;属性精度描述属性数据的质量,主要反映要素分类的正确性、属性编码的正确性、注记的正确性等。

②一致性。这里,一致性主要是指数据的逻辑一致性,如节点匹配、多边形闭合及拓扑关系的正确性等。

③完整性。完整性包括诸如地图要素类型的完整性、数据分类的完整性、属性数据的完整性以及注记的完整性等。

④现势性。现势性可以由数据的采集时间、数据的更新时间等来体现。

⑤统一性。统一性主要是指各类数据之间的统一协调等。

2. 数据质量的评价

(1)数据质量的评价方法

地图数据的质量评价主要是对其几何数据和属性数据评价它们的准确性、一致性、完整性和现势性,以及它们之间统一性的程度等(见表3-2)。

表 3-2 地图数据的质量评价

地图要素	空间定位特征	属性特征
位置精度
属性精度
一致性
完整性
现势性
统一性

具体采用的数据质量评价方法可分为以下几种：

①直接法。即直接用计算机程序进行自动检测。这是一种较为先进的智能化方法 ,是今后的发展方向。其基本思想是用计算机软件自动发现数据中某些类型的错误 ,并自动算出数据中不符合要求的数据项的百分率或平均质量等级等。

另外 ,随机抽样检测也是目前常用的方法。在确定抽样方案时 ,应充分考虑数据的空间相关性。

②间接评价法。间接评价法是根据外部信息进行推理来确定地图数据质量的方法。用于推理的外部信息包括地图的目的和用途、数据源的质量、数据采集时间及方法等。

(2)地图数据误差

地图数据的质量通常用误差来衡量。对地图数据的质量评价通常包括两部分 ,即地图数据采集过程中的数据质量评价和后续数据处理过程中的数据质量评价。这里着重介绍前者。

地图数据采集的方法除直接从野外采集 ,以获取观测数据、图像外 ,主要是对已有的地图进行数字化采集。在地图数字化中 ,地图原有误差和数字化过程中引入的误差是两个主要的误差源。

①地图原有误差。地图原有误差主要包括：

- 控制点和碎部点误差

该误差来源于地面测量中的控制测量误差和碎部测量误差。

- 地图制图综合误差与编绘误差

地图制图综合误差的大小取决于特征的类型和复杂程度 ,极难量化 ,与采用的制图综合方法 (如取舍、移位、夸大)等有很大的关系。在编绘误差中 ,点状要素的编绘精度通常优于线状要素的编绘精度 ,线状要素的编绘精度还与其分辨率或宽度有关。自然界中的许多特征并无明确的界线 ,例如 ,海岸线的实际位置、森林和草地的边界等 ,但它们在地图上却有明确的位置。

- 地图清绘误差

地图清绘误差是在清绘地图过程中产生的 ,误差范围一般为 0.06 ~0.18mm。

- 印刷误差

在地图印刷过程中产生的误差主要包括地图复制误差及分版套合误差。其中 ,复制误

差的均方差为 $0.1 \sim 0.2\text{mm}$;分版套合误差的均方差为 $0.17 \sim 0.30\text{mm}$ 。另外,由于在印刷时纸张的湿度和温度变化,纸张会伸长或收缩。因此,在地图印刷完成后,图纸在长、宽方向上分别会有 0.99% 和 1.73% 的变化。

- 图纸变形误差

一般情况下,随着温度和湿度的变化,地图纸张的尺寸也会变化。例如,温度不变,湿度增加 25% ,则纸张的尺寸改变 1.6% 。

②数字化误差。地图数字化方式目前主要有手扶跟踪数字化和扫描跟踪数字化两种。地图数字化过程中引入的误差主要有:

- 仪器差

仪器差主要与手扶跟踪数字化仪或扫描仪的精度有关。

- 人员差

人员差与操作员的水平、态度等有关。数据的精度在很大程度上取决于操作员选择最佳点位来数字化地图上的点、线、面等要素的能力。

- 数字化方式

数字化方式(如点方式、流方式等)对地图要素(尤其是特征点)的定位精度有一定的影响。

- 数字化软件

数字化软件的各种算法也会产生一定程度上的误差。

3. 数据质量的控制

数据质量控制是一个复杂的过程,其目的就是减少误差以提高精度。数据质量控制的基本方法是找出数据的误差及其原因,并采取适当措施加以消除(或减少、限制)。常见数据质量控制方法有以下几种:

(1)误差带法

误差带法是指在数字化的过程中,在一条数字化线的两侧,各定义宽为 s 的范围,作为该数字化线的误差带,也就是说用 s 的值来控制误差的范围。该方法的关键是如何确定合理的 s 值的大小。

(2)比较法

用绘图机绘出数字化后的数据,与原图进行叠加比较,选择相应要素及明显地物点,进行对照检查,以确定误差。

(3)相关法

根据地图要素自身的相关性来分析数据的质量。例如,根据河流与等高线的套合程度,可以确定数据误差的情况。为此,可以建立地图要素的相关关系知识库,以便进行相关性检查。

3.4 地图数据库

计算机地图制图是地理信息处理的过程。在地理信息的采集、存储、检索、分析处理及图形输出等过程中,都需要地图数据库的支持。地图数据库与一般的数据库相比,不仅数据量特别大,而且既要管理属性数据又要管理空间数据。地图数据库是计算机地图制图系统

的重要组成部分。

3.4.1 数据库的概念

1. 数据库技术的产生

数据库技术是在文件系统的基础之上,于20世纪60年代末期发展起来的。它是计算机领域中最重要技术之一,是一种先进的对数据进行组织、存储、检索和维护的数据管理技术。

计算机数据管理技术主要经历了三个发展阶段,即人工管理阶段(20世纪50年代中期)、文件系统阶段(20世纪50年代后期到60年代中期)、数据库系统阶段(从20世纪60年代后期开始)。

在文件系统中,存在着文件之间无联系、只能在文件级共享数据、数据不完全独立于程序等缺陷。为了克服这些缺陷,数据库技术应运而生。

数据库是存储和管理某个领域信息数据的系统。它根据数据间的自然联系而构成,实现了数据共享,数据冗余少且具有较高的数据独立性。

2. 数据库系统的构成

数据库系统由4个基本部分构成:

①数据集合:即一个结构化的相关数据体,包括数据本身和数据间的联系。数据集合独立于应用程序而存在,是数据库的核心和管理对象。

②数据库硬件:主要包括计算机及内、外存储器。其中,外存储器负责存储数据;内存储器负责存储操作系统和数据库管理系统。

③数据库软件:主要指数据库管理系统(Data Base Management System, DBMS)。DBMS是数据库系统运行的核心,是用户和数据库之间的接口。其主要任务是对数据库进行管理和维护,具有对数据进行定义、描述、操作和维护等功能,接受并完成用户程序和终端命令对数据库的请求,负责数据库的安全。

④人员:包括设计管理人员和众多用户。

3. 数据库的主要特征

与文件系统相比,数据库具有更强的数据管理能力。数据库具有如下主要特征:

(1)数据集中控制与共享

在文件系统中,文件是分散的,这些文件之间一般是没有联系的,每个用户或每种处理都有各自的文件,因此,不能多用户共享数据,不能按照统一的方法来控制、维护和管理数据。而数据库则很好地克服了这一缺点,它可以集中控制、维护和管理有关数据,数据可以供多个用户使用,每个用户只与数据库中的一部分数据发生联系,用户数据可以重叠,用户可以同时存取数据而互不影响。这大大提高了数据的使用效率,节省了存储空间。

(2)数据冗余少

所谓数据冗余是指数据的重复存储。在文件系统中,每个用户或每种处理都有各自的文件,数据冗余太大且数据不易保持一致性。而数据库中的数据不是面向各自应用,而是面向系统,数据统一定义、组织和存储,集中管理,避免了数据冗余,提高了数据的一致性。

(3)数据独立性

这里,数据独立性是指数据与应用程序的相互独立,也即应用程序不因数据的改变而改

变,而数据也不因应用程序的改变而改变。数据库中的数据独立于应用程序,包括数据的物理独立性和逻辑独立性,给数据库的应用、调整和进一步优化提供了方便,提高了数据库系统的稳定性。

(4)数据结构(模型)化

数据库按一定的数据结构(模型)形式构成,数据在记录内部和记录类型之间相互关联,用户可通过不同的路径存取数据。常用的数据模型有三种:层次模型、网络模型及关系模型。

(5)数据保护

在数据库系统中,多用户共享数据资源的情况下,数据保护具有重大意义。必须对用户使用数据进行严格的检查,对数据库规定密码或存取权限,拒绝非法用户进入数据库,以确保数据的安全性、一致性和并发控制。另外,还能发现故障和恢复正常。

4. 数据库的抽象分级

数据库作为一个复杂的系统,一般划分为三个抽象的分级结构,即子模式、模式和存储模式(见图3-2)。

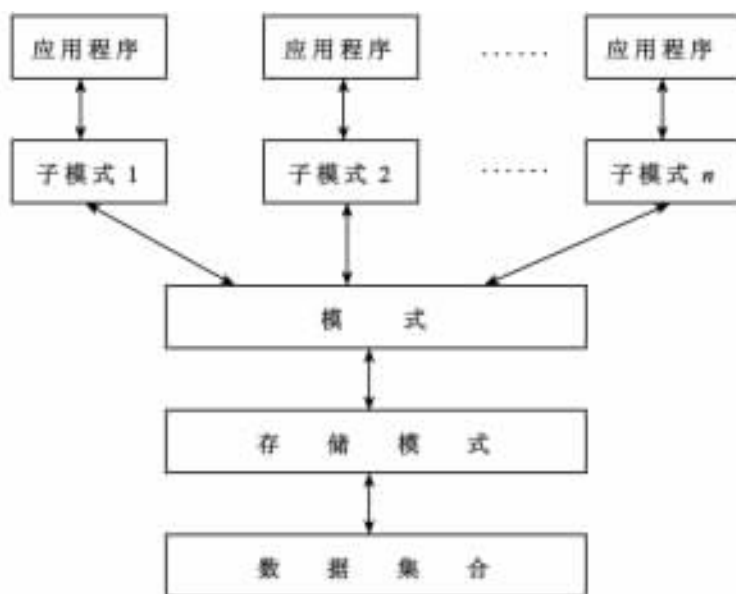


图3-2 数据库的分级结构

(1)子模式

子模式又称外模式,是数据库与用户的接口,也是数据库用户的数据视图。它是数据库的一个子集,属于模式的一部分,描述用户数据的结构、类型、长度等。用户的应用程序都是根据子模式中对数据的描述而不是根据模式中对数据的描述而编写的。每个用户应用程序只能使用一个子模式,但在一个子模式中 can 编写多个应用程序。根据不同的应用,一个模式可以对应多个子模式,子模式也可以互相覆盖。子模式由子模式描述语言(SDDL)进行具体描述。

(2) 模式

模式也称概念模式。模式的主体是数据模型,是数据库的总框架。它描述数据库中关于目标存储的逻辑结构和特性,它们之间的联系和依赖性,以及对数据的安全性、完整性等方面的定义。模式由模式描述语言(DDL)进行描述。

(3) 存储模式

存储模式又称内模式,是对数据的物理描述,用来描述数据在物理存储器上的具体实现。它涉及文件组织技术,主要包括数据在存储介质上的物理组织方式、存储块的大小、记录寻址技术及溢出处理方法等。存储模式由数据存储描述语言(DSDL)进行描述。

在数据库系统的三级模式结构之间进行了两次映射:

- ①子模式与模式之间的映射:反映了它们之间的对应关系,保证了数据的逻辑独立性;
- ②模式与存储模式之间的映射:反映了数据的逻辑结构和物理存储之间的对应关系,使全局逻辑数据独立于物理数据,保证了数据的物理独立性。

这样,用户可以有各自的数据视图,所有用户的数据视图集中在一起统一组织,消除冗余数据,得到全局数据视图。用存储描述语言来定义和描述全局数据视图数据,并将数据存储于物理介质上。

数据库系统的三个抽象分级结构将数据库的全局逻辑结构与用户的局部逻辑结构和物理存储结构区分开来,给数据库的组织和使用带来了极大的方便。

3.4.2 数据模型

1. 数据模型的基本概念

数据模型是对现实世界部分现象的抽象,它描述了数据的基本结构及其相互之间的关系以及在数据上的各种操作。

在数据库系统中,现实世界中的事物及联系是用数据模型来描述的,数据库中各种操作功能的实现是基于不同的数据模型的,因此数据库的核心问题是模型问题。

数据模型是数据库系统中关于数据和数据间联系的逻辑组织的形式表示,以抽象的形式描述和反映一个部门或系统的业务活动和信息流程。每一个具体的数据库都由一个相应的数据模型来定义。选择与建立数据模型的目的是用最佳的方式反映本部门的业务对象及信息流程,以最佳的方式为用户提供访问数据库的逻辑接口。

数据模型反映了现实世界中实体之间的各种联系,这种联系通过数据记录及记录间的逻辑联系反映出来。实体间的联系主要有两类:一类是实体与实体之间的联系,反映在数据上是记录之间的联系;另一类是实体内部属性间的联系,反映在数据上是记录内部的联系。

实体与实体之间的联系是错综复杂的,可以分为以下4种:

(1) 一对一的联系(记为 1:1)

一对一的联系是最简单的一种实体之间的联系,它表示两个实体集中的个体间存在的一对一的联系(见图 3-3)。

(2) 一对多的联系(记为 1:N)

一对多的联系是实体间存在的较普遍的一种联系,表示一个实体集 A 中的每个实体与另一个实体集 B 中的多个实体间存在联系;反之,B 中的每个实体都至多与 A 中的一个实体发生联系(见图 3-4)。

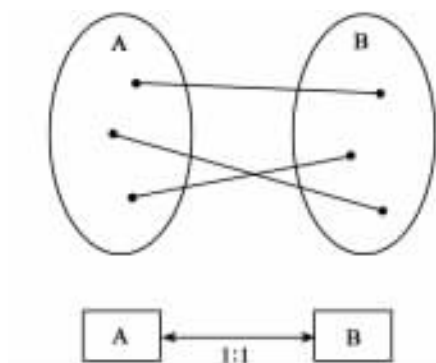


图 3-3 实体间一对一的联系

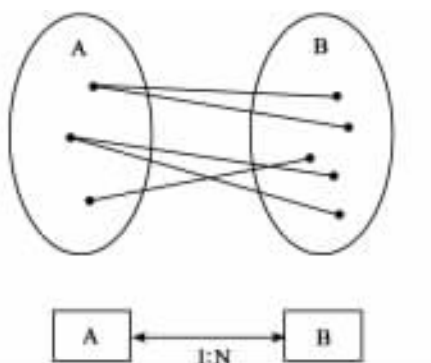


图 3-4 实体间一对多的联系

(3) 多对多的联系(记为 $M:N$)

多对多的联系是实体间存在的更为普遍的一种联系,表示多个实体集之间的多对多的联系。其中,一个实体集中的任何一个实体与另一个实体集中的实体间存在一对多的联系;反之亦然(见图 3-5)。

(4) 条件联系

条件联系是实体间存在的一种特殊联系,表示在一定条件 C 下,存在 A 到 B 的联系;若不符合条件,则不存在 A 到 B 的联系(见图 3-6)。

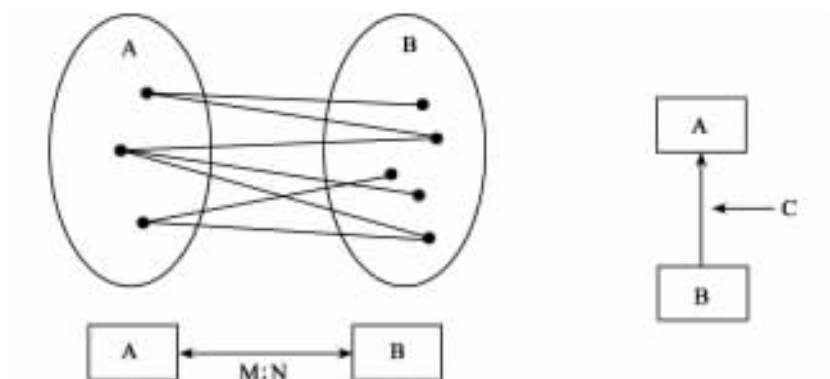


图 3-5 实体间多对多的联系

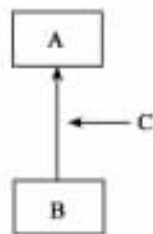


图 3-6 实体间的条件联系

实体内部的联系也可以分为一对一的联系、一对多的联系、多对多的联系和条件联系 4 种。

2. 传统数据模型

数据模型是一种在计算机系统中描述现实世界的信息结构及其变化的抽象方法。数据模型不同,描述和实现的方法也不相同。

数据模型主要可以分为两种类型:一种是独立于计算机之外的,如实体-联系模型(即 E-R 模型)、语义数据模型等,它们不涉及信息在计算机中如何表示,又称概念模型;另一种模型是直接面向计算机的,它们以记录为单位构造数据模型,目前,数据库中常用的有层次

模型、网状模型和关系模型等 ,其中关系模型应用最为广泛。

(1)层次模型

层次模型是数据库技术中发展最早、技术上比较成熟的一种数据模型。层次模型把数据按自然的层次关系组织起来 ,以反映数据之间的隶属关系(一对多的联系) ,它是一种树结构模型。

层次模型的特点是不同节点(记录)间只有简单的层次联系 ,地理数据组织成有向有序的树结构。该结构中的节点代表数据记录 ,用有向的连线描述位于不同节点数据间的从属关系。由于通常表示树结构时都是根节点在上 ,叶节点在下 ,表示联系的箭头都向下 ,故连线时箭头一般都省去(如图 3-7 所示)。

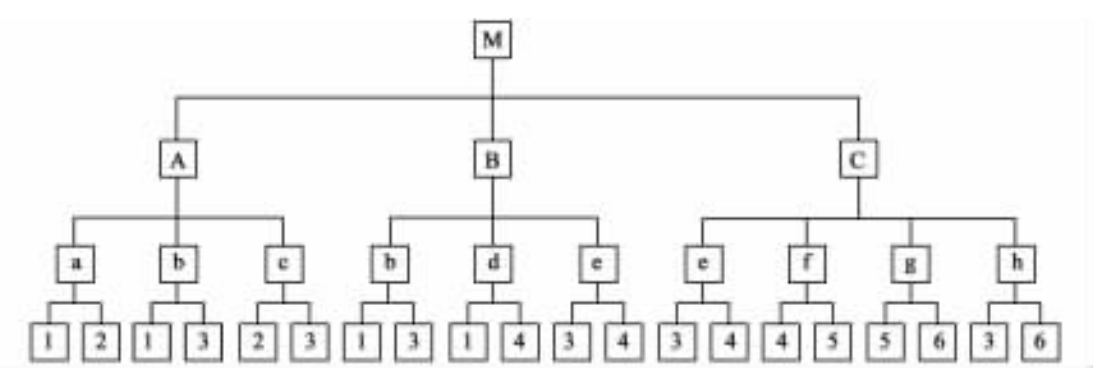


图 3-7 层次模型

根据树的定义 ,一棵树中 ,有且仅有一个无双亲的节点称为根节点 ;其余节点有且仅有一个双亲节点 ,它们可分为 $N(N \geq 0)$ 个互不相交的有限集 ,其中每一个集合本身又是一棵树 ,将其称为子树。

以图 3-8 示意的地图 M 及其所包含的点、线、面空间要素为例 ,其中 :A、B、C 为多边形(面) ;a、b、c、d、e、f、g、h 为线(边) ;1、2、3、4、5、6 为点。其空间关系所构成的层次模型如图 3-7 所示。这是一棵有向有序树 ,节点表示不同层次的空间要素 ,连线描述空间要素之间的从属关系。

层次模型的优点是层次分明 ,结构清晰 ,易于实现 ,并在一定程度上支持数据的重构。层次模型用于地图数据库时存在的问题主要是 :

- ①对任何对象的查询都必须从其层次结构的根节点开始 ,低层次对象的查询效率很低 ,很难进行反向查询 ,插入和删除操作比较复杂 ,父节点的删除意味着其下层所有子节点均被删除。
- ②数据独立性较差 ,数据更新涉及许多指针。
- ③层次命令具有过程性性质 ,用户须了解数据的物理结构 ,并在数据操作命令中显式地给出数据的存取路径。
- ④难以描述复杂的地理实体之间的联系 ,描述多对多的关系时导致物理存储上的冗余。

(2)网状模型

网状模型反映现实世界中实体间更为复杂的联系(多对多的联系) ,它将数据组织成有

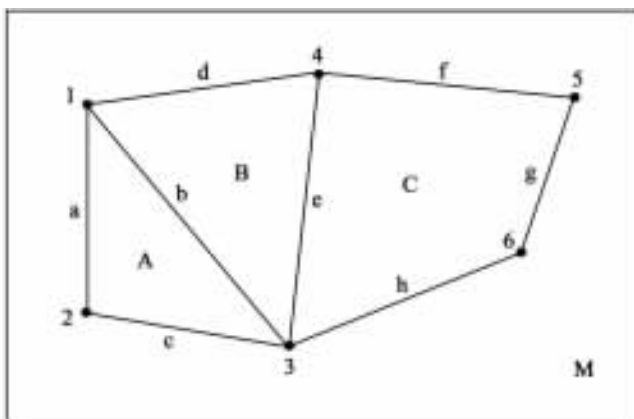


图 3-8 地图 M 及其所包含的点、线、面空间要素

向图结构,图中的节点代表数据记录,连线描述不同节点数据间的联系。网状模型的基本特征是:任一节点可与其他多个节点建立联系,即节点之间的联系是任意的,任何两个节点之间都能发生联系,且节点数据之间没有明确的从属关系。

以图 3-9 示意的地图 M 及其所包含的点、线、面空间要素为例,其空间关系所构成的层次模型如图 3-10 所示。

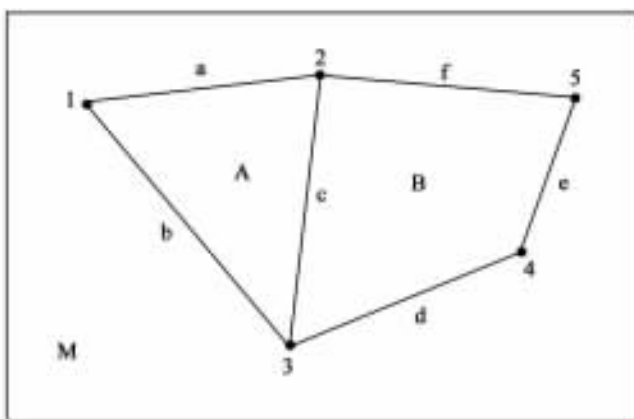


图 3-9 地图 M 及其空间要素

网状模型反映了地理世界中常见的多对多关系,在一定程度上支持数据的重构,具有一定的数据独立和数据共享特性,且运行效率较高。它用于地图数据库时存在的主要问题如下:

- ① 网状模型的结构比较复杂,增加了用户查询和定位困难,用户须熟悉数据的逻辑结构,知道自己所在的相应位置。
- ② 网状模型不直接支持对于层次结构的表达。
- ③ 网状模型的数据操作命令具有过程式性质。

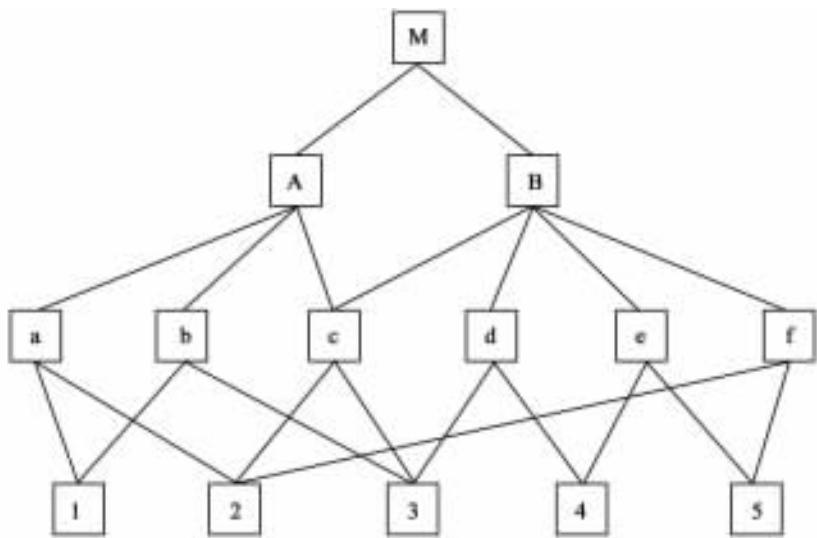


图 3-10 网状模型

(3)关系模型

关系模型把数据模型看成是数学关系的集合 ,它将数据的逻辑结构归结为满足一定条件的二维表形式。在关系数据库系统中 ,对数据的操作是通过关系代数实现的 ,具有严格的数学基础。

关系模型中二维表具有固定的列数和任意的行数 ,这在数学上称为“ 关系 ”。二维表是同类实体的各种属性的集合 ,每个实体对应于表中的一行 ,称为一个元组 ,相当于通常的一条记录 ;表中的每一列表示同一种属性 ,称为域 ,相当于通常记录中的一个数据项。若二维表中有 n 个域 ,则每一行叫做一个 n 元组 ,这样的关系称为 n 度(元)关系。表中的行对应于对象的实例 ,表中行列交点就用来存储简单值。满足一定条件的规范化关系的集合 ,就构成了关系模型。

关系数据模型要求每一属性都必须赋予一个域 ,域可允许更多的语义检查 ,表中的每一个值都必须属于其属性的域或者为空。

关系模型可以方便、灵活地表示各种实体及其关系 ,其数据描述具有较强的一致性和独立性。关系数据库结构简单 ,操作方便 ,有坚实的理论基础 ,发展很快 ,目前绝大部分的数据库管理系统是关系型的。

在关系模型中 ,每个表有一个明确的主关键字 ,主关键字是一个或多个属性的集合。外来关键字与它相应的主关键字一致 ,外来关键字是一个表的主关键字 ,但它又嵌入另一个表中。外来关键字是表间关联的桥梁。

范式是关系模型中的一个重要概念。所谓范式 ,就是用来避免表修改操作的逻辑不一致性的一些规则集。范式有多种层次 ,高级范式可以对低级范式增加约束条件。常用的范式主要有 Codd 在 1971 年前后提出第一范式(1NF)、第二范式(2NF)和第三范式(3NF)。

第一范式(1NF) 如果存在一关系模式 R ,当且仅当它的每一个属性都是单纯域时 ,则此关系模式 R 为 1NF ;

第二范式(2NF) 若关系模式 $R \in 1NF$,且每个非主属性完全函数依赖于主关键字 ,则此关系模式 $R \in 2NF$;

第三范式(3NF) 若关系模式 $R \in 2NF$,且每个非主属性都非传递依赖于主关键字 ,则此关系模式 $R \in 3NF$ 。

以图 3-11 示意的地图 M 及其所包含的点、线、面空间要素为例。其中 :A ,B ,C ,D 为多边形(面) a ,b ,c ,d ,e ,f ,g ,h ,i 为线(边) 1 ,2 ,3 ,4 ,5 ,6 为点 ,可用关系模型构成如下关系 :

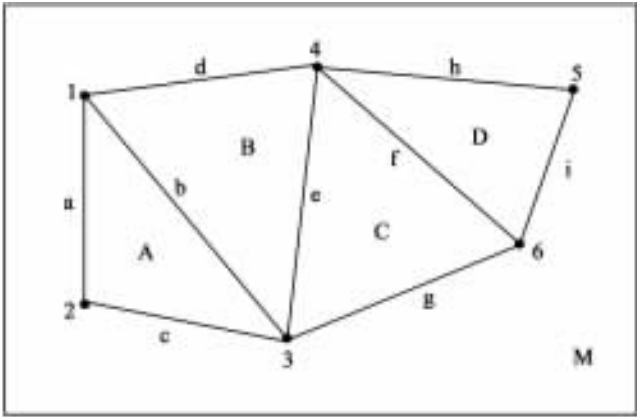


图 3-11 地图 M 及其所包含的空间要素

地图-多边形(面)关系 :

$M(A ,B ,C ,D)$

多边形-线(边)关系 :

$A(a ,b ,c)$

$B(b ,d ,e)$

$C(e ,f ,g)$

$D(f ,h ,i)$

边-节点关系 :

$a(1 ,2)$

$b(1 ,3)$

$c(2 ,3)$

$d(1 ,4)$

$e(4 ,3)$

$f(4 ,6)$

$g(3 ,6)$

$h(4 ,5)$

$i(5 ,6)$

由上述空间关系所构成的二维关系表分别如表 3-3、表 3-4、表 3-5 所示。

表 3-3 地图-多边形关系表

M	A
M	B
M	C
M	D

表 3-4 多边形-线关系表

A	a	b	c
B	b	d	e
C	e	f	g
D	f	h	i

表 3-5 边-节点关系表

a	1	2
b	1	3
c	2	3
d	1	4
e	4	3
f	4	6
g	3	6
h	4	5
i	5	6

在关系数据库管理系统(RDBMS)中,常用索引及排序等各种协调技术手段来加速对表的访问,这些协调手段是透明的,只在读写表的命令中是不可见的。由关系数据库管理系统来决定在处理查询的过程中使用协调信息的时间,如果需要,则自动执行协调。每当修改相应表时,关系数据库管理系统自动修改协调信息。

关系模型能简单、灵活地表示各种地理实体及其相互间的关系,支持数据重构;关系操作和关系演算具有非过程式特点,具有严格的数学基础和一定的演绎功能。目前,关系数据库管理系统的理论和技术都非常完善和成熟,商品化的产品大量涌向市场,基本上占据了当今数据库市场中全部的传统应用领域和 90% 以上的非传统应用领域。但是,关系模型用于地图数据库也还存在一些不足,表现在以下三个方面:

- ①关系之间的联系需要执行系统开销较大的连接操作,运行效率不够高。
- ②关系模型描述复杂的地理对象时,需对地理实体进行不自然的分解,导致存储模式、查询途径及操作等方面均显得语义不甚合理。
- ③模拟和操作复杂地理对象的能力较弱,无法用递归和嵌套的方式来描述复杂关系的

层次和网状结构。

3. 面向对象数据模型

(1) 面向对象的基本思想

如前所述,传统的数据模型如层次模型、网状模型和关系模型,用于地图数据库都有一定的局限性,其根本问题是较难有效地管理复杂的地理对象。

面向对象方法是近年来迅速发展起来的一种新颖和具有独特优越性的方法。其基本思想是通过对问题领域进行自然的分割,以接近人类通常思维的方式建立问题领域的模型,并进行结构模拟和行为模拟,从而使设计出的软件能尽可能地直接表现出问题的求解过程。

面向对象方法在描述复杂的地理对象时,用更接近人类通常思维方式的思想,将客观世界的地理实体化为对象。每一种对象都有各自的内部状态和运动规律,不同对象之间的相互联系和相互作用就构成了各种不同的系统。因此,面向对象数据模型能较为有效地管理复杂地理对象。

面向对象的地图数据模型的核心是对复杂对象的模拟和操作。这里,复杂对象是指具有复杂结构和操作的对象。复杂对象可以由多种关系聚合抽象而成,或由不同类型的对象构成,或具有复杂的嵌套关系等。

地图上的一个复杂地理实体(如城镇)可能含有矢量数据、栅格数据、属性数据,甚至多媒体数据,并且可以认为是由其他较简单的实体(如道路、街区等)组成的。又如,地图的地理实体所具有的矢量数据也可以认为是一个复杂对象,因为它包含了几何数据和属性数据,而几何数据又是由点、线、面等简单对象组成的。

地图中的复杂地理对象通常具有三个重要特征:

- ①一个复杂的地理对象由多个成员对象构成,每个成员对象又可参与其他对象的构成。
- ②复杂的地理对象具有多种数据结构,如矢量、栅格、关系表等。
- ③一个复杂地理对象的不同部分可由不同的数据模型所支持,即可以分布于不同的数据库中。

(2) 面向对象的地图数据模型

要有效地描述地图上复杂的地理事物或现象,需要在更高层次上综合利用和管理多种数据结构 and 数据模型,并用面向对象的方法进行统一的抽象。这就是面向对象数据模型的含义,其具体实现就是面向对象的数据结构。

在传统的数据模型中,数据结构是与其数据模型分离的。对于面向对象的数据模型,其数据模型和数据结构是一致的,数据模型的实现就是数据结构。

在计算机地图制图系统的面向对象数据结构中,通常把地图要素的空间数据抽象为点、线、面三种简单的地物要素类型,以及由这三种要素类型所混合组成的复杂要素类型,并以此作为相应的4种对象,其中:

点状要素 如桥梁、烟囱、车站等。点状要素具有标志码、特征代码、定位点坐标等数据项,同时还有显示、修改、增加、删除等操作。另外,当点状要素是有向地物时,需要有两对定位点坐标。

线状要素 如境界、道路、河流等。线状要素由一条或多条弧段组成,弧段还涉及两端的节点。线状要素具有标志码、特征代码、组成线状要素的弧段等数据项,并具有显示、修改、增加、删除、计算长度等操作。

面状要素 如行政区划、湖泊、地块等。面状要素由一条或多条弧段构成,具有标志码、特征代码、弧段串等数据项,有时还含有面的内点坐标、面积,面状要素外接矩形的坐标等,并具有显示、修改、增加、删除及计算面积等操作。

复杂要素 如大学、农场、城镇等。复杂的地理实体可以由多种简单要素对象构成,也可以由其他复杂要素对象构成。

一个地理实体可以由上述 4 种要素对象之一构成。为了描述空间数据的拓扑关系,在点、线、面这三种简单要素类型中涉及了孤立点、节点、弧段等拓扑数据类型,另外还应该与注记有紧密的联系。

每个地理实体都可以通过其标志码与其属性数据联系起来。若干个地理实体可以作为一个图层,若干个图层可以组成一个工作区。在计算机地图制图系统中,可以开设多个工作区。

面向对象的地图数据模型具有如下优点:

①面向对象的数据模型是一种基于抽象的模型,允许设计者在基本功能上选择最为适用的技术,可充分吸收和利用传统数据模型的优点。如可以把栅格数据结构和矢量数据结构统一为一种高层次的对象结构,这种结构可以具有栅格结构和矢量结构的特点,但实际的操作仍然是栅格数据用栅格算法,矢量数据用矢量运算。

②可以较为有效地模拟和操纵复杂对象。传统的数据模型是面向简单地理实体的,无法直接模拟和操纵复杂实体对象,而面向对象的数据模型具备对复杂对象进行模拟和操纵的能力。

③具有良好的可扩充性。因为对象是相对独立的,故可以很自然和容易地增加新的对象,并且对不同类型的对象具有统一的管理机制。

另外,任何一种模型都无法完全反映出现实世界的所有方面,对于复杂的事物和现象更是如此,因此很难设计出一种通用的数据模型和数据结构来适应所有的情况。某种模型常常是在描述一类问题时,具有很高的效率,而在描述另一类问题时,却是低效的。

3.4.3 地图数据库的设计与建立

数据库设计应根据不同用户的目的和要求,在一个给定的应用环境中,确定最优的数据模型、处理模式、存储结构、存取方法,建立能反映现实世界的实体间信息的联系,满足用户要求,能被一定的数据库管理系统接受,同时又能实现系统目标并有效地存取、管理数据的数据库(本节仅作原理性介绍,进一步阐述见本书第 6 章)。

1. 地图数据库的设计

数据库因不同用户的目的和要求会有各种各样的组织形式。数据库设计就是把现实世界中已存在着的数据抽象成一个数据库的具体过程。

地图数据库的设计是指在现有数据库管理系统的基础上建立地图数据库的整个过程。其中最重要的是需求分析和结构设计两个步骤。

需求分析是地图数据库设计与建立的基础。首先,应了解用户的要求和特点,使得设计者与用户对需求有一致的看法;其次,在此基础上收集和分析所需数据,包括数据的内容、特征、存储、处理等要求;最后,应编制用户需求说明书,包括需求分析的目标、任务、具体需求

说明、系统功能与性能、运行环境等。

结构设计的目的是得到一个合理的地图数据模型,这是地图数据库设计的关键。结构设计分为概念设计、逻辑设计和物理设计三个阶段。

(1)概念设计

在概念设计阶段,对需求分析中所收集的信息和数据进行分析、整理,以确定地理实体、属性及它们之间的联系,形成与具体的数据库管理系统无关、结构稳定、能较好地反映用户观点的概念模式。

实体-联系模型(E-R模型)是表示概念模型的有效工具。该模型包括实体、联系和属性三个基本成分,比一般的数据模型更接近于现实地理世界。用它来描述现实地理世界,无须考虑数据的存储结构、存取路径等与计算机有关的问题,具有自然、直观、语义丰富等特点,在地图数据库设计中得到了广泛应用。

(2)逻辑设计

逻辑设计是在概念设计的基础上进行的,是按照不同的转换规则将概念模型转换为具体数据库管理系统支持的数据模型的过程。也即导出具体数据库管理系统可处理的地图数据库的逻辑结构(或外模式),包括数据项、记录的确定以及它们之间的联系、安全性和完整性等。

以E-R模型向关系模型转换为例,其主要过程包括各实体的主关键字确定,找出实体内部属性之间的数据关系表达式,即某一数据项决定另外的数据项,把经过消冗处理的数据关系表达式中的实体作为相应的主关键字并形成新的关系等。

导出的逻辑结构应与概念模式一致,应能满足用户要求,并且还需要对其功能和性能进行评价,予以优化。

(3)物理设计

物理设计的目的是在物理存储器上有效地实现空间数据库的逻辑结构,也即确定数据在物理介质上的存储结构,其结果是导出地图数据库的存储模式(内模式)。

物理设计的主要内容包括选择文件的存储结构,确定记录的存储格式,并决定存取路径和分配存储空间。一般数值型数据可用十进制或二进制形式表示。通常二进制形式所占用的存储空间较少。字符型数据可以用字符串的方式表示,也可以利用代码值的存储代替字符串的存储。常采用数据压缩技术来节约存储空间。

物理设计在很大程度上与选用的数据库管理系统有关。设计中应根据需要,选用系统所提供的功能。一个好的物理存储结构应该是地理数据占有的存储空间较小,且对数据库的操作具有尽可能快的处理速度。

2. 地图数据库的建立

完成地图数据库的设计之后,利用数据库管理系统提供的数据库描述语言描述逻辑设计和物理设计的结果,得到概念模式和外模式,编写功能软件,经编译、运行后形成目标模式,并输入数据,就可以建立起实际的地图数据库了。

第4章 地图数据处理

地图是按照一定的比例和投影原则,有选择地将复杂的现实地理空间世界的某些内容投影到二维平面上,并用符号将这些内容要素表现出来。

地图数据是地图要素的数字化表示,是采用某种编码技术方式对地理空间物体进行描述及在它们之间建立相互联系的数据集。

地图数据按其结构主要可分为矢量数据结构和栅格数据结构两种。因此,对地图数据的处理也可分为两类,即矢量数据处理和栅格数据处理。

4.1 矢量数据处理

矢量数据处理可分为批量式处理和人机交互式处理两种方式,主要包括数据的预处理、图形编辑、数据变换、数据插值、矢量符号的生成和拓扑关系的建立等。

4.1.1 数据预处理

数据预处理是使数据便于存储、管理和进一步分析应用而进行的变换、加工等,主要包括坐标变换、数据压缩等。

1. 坐标变换

在地图数据的采集中,由数字化过程产生的明显误差可以在编辑修改过程中消除,但是图纸变形产生的坐标数据误差却难以纠正,因此要对其进行几何改正。而当所使用的数据是来自不同地图投影的图幅时,则需将一种投影的几何数据转换成所需的另一种投影的几何数据,这就需要进行地图投影变换。下面分别对几何改正和投影变换进行介绍。

(1) 几何改正

常用的几何改正方法有高次变换、二次变换和仿射变换等。

①高次变换和二次变换。高次变换采用的是多项式拟合方法,公式如下:

$$\begin{cases} x' = a_0 + a_1x + a_2y + a_{11}x^2 + a_{12}xy + a_{22}y^2 + A \\ y' = b_0 + b_1x + b_2y + b_{11}x^2 + b_{12}xy + b_{22}y^2 + B \end{cases}$$

式中 x' 、 y' 为变换后的坐标, x 、 y 为变换前的坐标, a 、 b 为待定系数, A 、 B 代表二次以上高次项之和。上式是高次变换方程,符合上式的变换称为高次变换。在进行高次变换时,需要有 6 对以上控制点的坐标和理论值,才能求出待定系数。

在上述高次变换方程中,当不考虑 A 和 B 时,则变成二次变换方程,称为二次变换。二次变换适用于原图有非线性变形的情况,需要 5 对控制点的坐标及其理论值,才能求出待定系数。在具体应用中,为提高精度,可取多于 5 对控制点的坐标及其理论值,用最小二乘法求解。

②仿射变换。仿射变换实质上是一种一次变换,是使用得最多的一种几何改正方法,公

式如下：

$$\begin{cases} x' = a_1 x + a_2 y + a_3 \\ y' = b_1 x + b_2 y + b_3 \end{cases}$$

式中,只需知道不在同一直线上的3对控制点的坐标及其理论值,就可求得待定系数。

仿射变换只考虑 x 和 y 方向上的变形,它的特点是：

- 直线变换后仍为直线；
- 平行线变换后仍为平行线,且长度比不变；
- 不同方向上的长度比发生变化。

在对仿射变换的实际使用中,为提高变换的精度,往往利用4个以上的点进行改正,利用最小二乘法求解其待定系数。方法如下。

设仿射变换方程的误差方程为：

$$\begin{cases} Q_x = X - (a_1 x + a_2 y + a_3) \\ Q_y = Y - (b_1 x + b_2 y + b_3) \end{cases}$$

式中 X, Y 为已知的理论坐标。

根据 Q_x^2 最小和 Q_y^2 最小的条件,可得到两个法方程组：

$$\begin{cases} a_1 \sum x + a_2 \sum y + a_3 n = \sum X \\ a_1 \sum x^2 + a_2 \sum xy + a_3 \sum x = \sum xX \\ a_1 \sum xy + a_2 \sum y^2 + a_3 \sum y = \sum yX \end{cases}$$

$$\begin{cases} b_1 \sum x + b_2 \sum y + b_3 n = \sum Y \\ b_1 \sum x^2 + b_2 \sum xy + b_3 \sum x = \sum xY \\ b_1 \sum xy + b_2 \sum y^2 + b_3 \sum y = \sum yY \end{cases}$$

式中 x, y 为控制点坐标, X, Y 为控制点的理论值, n 为控制点个数, a, b 为待定系数。

解上述法方程组即可求得仿射变换的各待定系数。

(2) 投影变换

地图投影变换的实质是建立两地图平面点之间的一一对应关系,投影变换的关键就是要找出该关系式。常用的方法有解析法和数值法,简介如下。

①解析法。解析法就是找出两投影间坐标变换的解析计算公式。解析法可以直接求出两种投影点的直角坐标关系式：

$$\begin{cases} X = f_1(x, y) \\ Y = f_2(x, y) \end{cases}$$

式中 x, y 为原图上点的坐标, X, Y 为新图上点的坐标。

也可以用间接的方法求出两种投影点的直角坐标关系式,即先解出原地图投影点的地理坐标 (φ, λ) , 对于 x, y 的解析关系式,再将其代入新图的投影公式中求得其坐标。即先求出：

$$\begin{cases} \varphi = f_1(x, y) \\ \lambda = f_2(x, y) \end{cases}$$

再求出：

$$\begin{cases} X = h_1(\varphi, \lambda) \\ Y = h_2(\varphi, \lambda) \end{cases}$$

②数值法。在不易求出两投影坐标之间的直接关系时,可采用多项式逼近的数值方法来建立两投影间的变换关系。以二元三次多项式为例：

$$\begin{cases} X = a_{00} + a_{10}x + a_{01}y + a_{20}x^2 + a_{11}xy + a_{02}y^2 + a_{30}x^3 + a_{21}x^2y + a_{12}xy^2 + a_{03}y^3 \\ Y = b_{00} + b_{10}x + b_{01}y + b_{20}x^2 + b_{11}xy + b_{02}y^2 + b_{30}x^3 + b_{21}x^2y + b_{12}xy^2 + b_{03}y^3 \end{cases}$$

选择 10 个以上的两种投影之间的共同点,利用最小二乘法求解各系数的值即可。

另外,也可将上述两种方法混合应用。例如,在已知新投影的公式,但不知原投影的公式时,可先通过数值变换求出原投影点的地理坐标(φ, λ),然后再代入新投影公式中,求出新投影点的坐标。

2. 数据压缩

数据压缩是数据精练的过程,其目的是删除冗余数据,节省存储空间,以利后续处理。在计算机地图制图中,主要是对矢量数据中线(或多边形的弧)的中间坐标点进行压缩,而其两个端点(节点)通常予以保留。常用的矢量数据压缩方法介绍如下。

(1)垂距法

垂距法的基本算法是:从任一个端点起,每次顺序取曲线上的三个点,计算中间点与其他两点连线的垂线距离 D,并与限差 d 比较。若 $D < d$ 则中间点去掉;若 $D \geq d$ 则中间点保留。然后顺序取下三个点继续处理,直到这条线结束(如图 4-1 所示)。

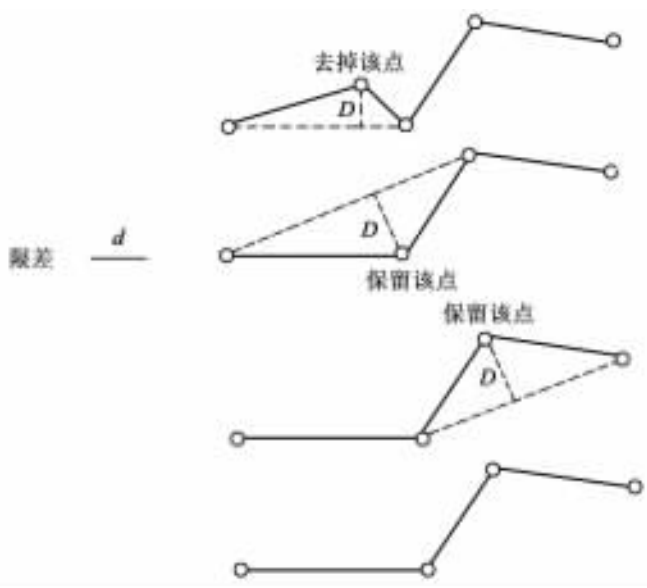


图 4-1 垂距法示意图

(2)偏角法

偏角法的基本算法是:从任一个端点起,每次顺序取曲线上的三个点,计算中间点与其

他两点连线的夹角 α 并与限差 β 比较。若 $\alpha < \beta$,则中间点去掉 ;若 $\alpha \geq \beta$,则中间点保留。然后顺序取下三个点继续处理 ,直到这条线结束(如图 4-2 所示)。

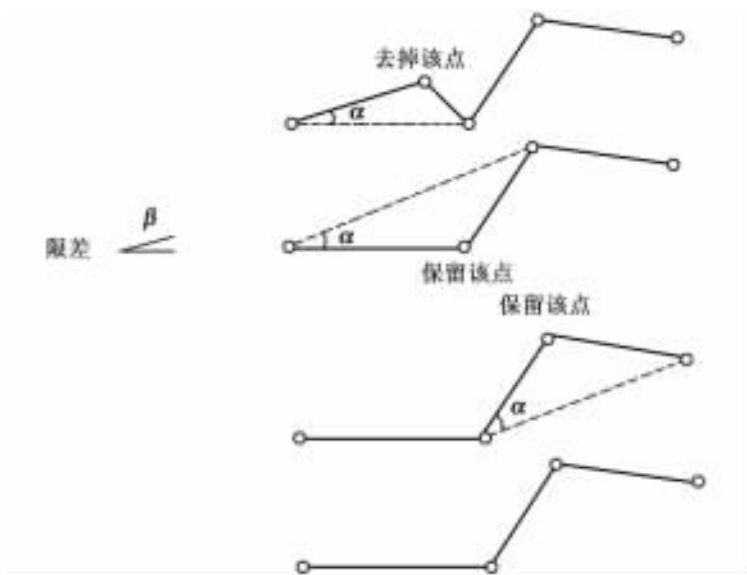


图 4-2 偏角法示意

(3)道格拉斯-普克法

道格拉斯-普克法可以看成是垂距法的推广 ,其基本算法是将一条曲线的首末端点虚连一条直线 ,求其余所有点与直线的距离 ,并找出最大距离值 D_{\max} ,用 D_{\max} 与限差 d 相比较。若 $D_{\max} < d$,则这条曲线上的中间点全部舍去 ;反之 ,保留 D_{\max} 对应的坐标点 ,并以该点为界 ,把曲线分为两部分 ,对这两部分重复使用该方法(如图 4-3 所示)。

显然 ,图中点号为 4 的点应该保留。然后 ,将已知点列分成两部分处理。计算 2、3 点到 1、4 点连线的距离 ,选距离大者与限差比较 ,结果 2、3 点均应舍去。再计算 5、6 点到 4、7 点连线的距离 ,经比较 ,点 6 应保留。依此类推 ,最后保留下来的点在原数据中的编号为 1、4、6、7 点。当然 ,也将压缩后的数据重新排序为点列 1、2、3、4。

(4)光栏法

光栏法可以看成是偏角法的推广 ,其基本思想是 :以当前点为顶点 ,在后续点的方向上定义一个光栏区域 ,通过判断曲线上的点在光栏外还是在光栏内 ,确定该点是保留还是舍去(如图 4-4 所示)。

设有曲线上的点列 $\{P_i\}$ $i=1, 2, \dots, n$,光栏口径为 d (可根据需要自己定义大小) ,则光栏法的实施步骤为 :

①连接 P_1 和 P_2 点 ,过 P_2 点作一条垂直于 P_1P_2 的直线 ,在该垂线上取两点 a_1 和 a_2 ,使 $a_1P_2 = a_2P_2 = d/2$,这里 a_1 和 a_2 为“光栏”边界点 , P_1 与 a_1 、 P_1 与 a_2 的连线为以 P_1 为顶点的光栏的两条边 ,这就定义了一个光栏(该光栏的口朝向曲线的前进方向 ,边长是任意的)。通过 P_1 并在光栏内的所有直线都具有这种性质 ,即 P_1P_2 上各点到这些直线的垂距都不大于 $d/2$ 。

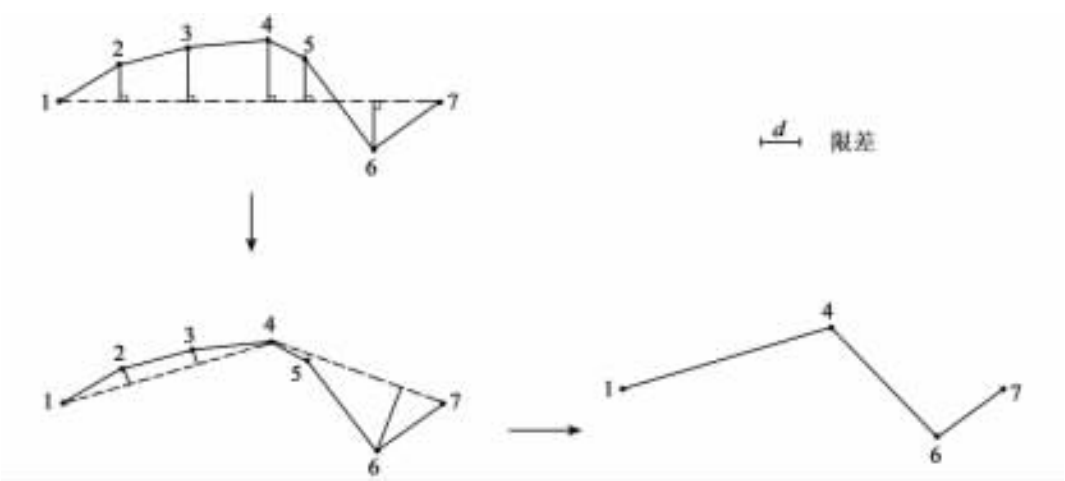


图 4-3 道格拉斯-普克法示意图

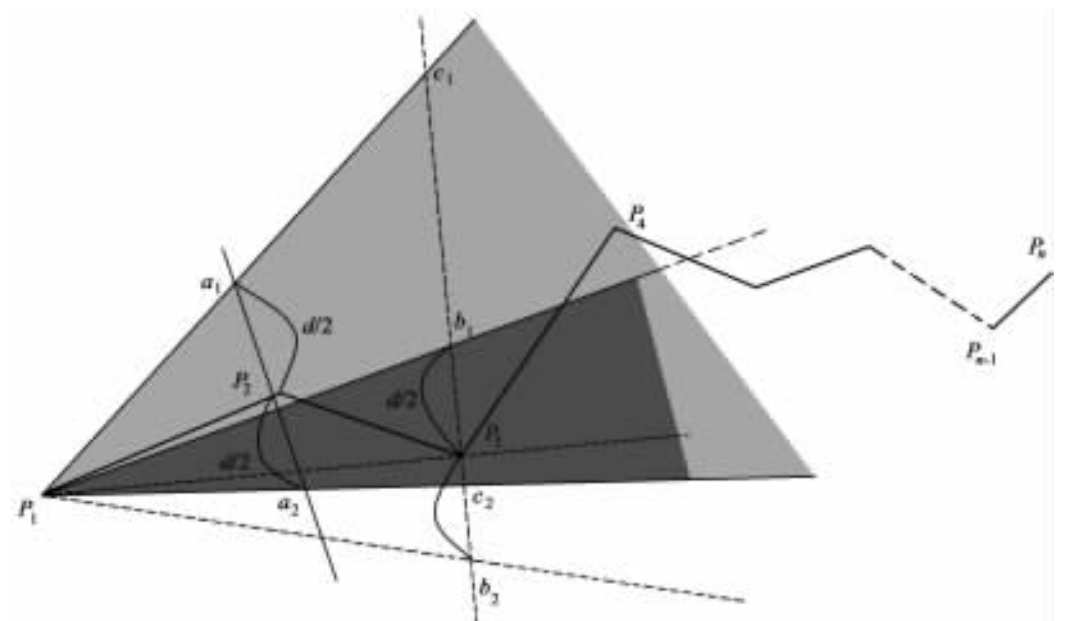


图 4-4 光栏法示意图

②若 P_3 点在光栏内, 则舍去 P_2 点。然后连接 P_1 和 P_3 , 过 P_3 作 P_1P_3 的垂线, 该垂线与前面定义的光栏边交于 c_1 和 c_2 。在垂线上找到 b_1 和 b_2 点, 使 $P_3b_1 = P_3b_2 = d/2$ 。若 b_1 或 b_2 点落在原光栏外面, 则用 c_1 或 c_2 取代(图 4-4 中由 c_2 取代 b_2)。此时用 P_1b_1 和 P_1c_2 定义了一个新的光栏, 一个口径(b_1c_2)缩小了的“光栏”。

③检查曲线的下一曲线点, 若该点在新光栏内, 则重复第二步, 直到发现有一个点在最新定义的光栏外为止。

④当发现在光栏外的节点,如图中的 P_4 ,此时保留 P_3 点,以 P_3 作为新起点,重复第一步至第三步。如此继续下去,直到整个点列检测完毕为止。所有被保留的曲线点(含首点、末点)顺序地构成了简化后的新点列。

上述几种矢量数据的压缩方法各有优劣。

一般来说,如果某种矢量数据的压缩方法既能精确地表示数据,又能最大限度地淘汰不必要的点,那就是一种好的算法。通常可以依据压缩前后曲线的总长度、总面积、坐标平均值等来判别算法的优劣。

通过对上述几种方法压缩前后曲线的总长度、总面积变化对比分析可以发现,大多数情况下道格拉斯-普克法的压缩算法较好,但必须在对整条曲线同时进行计算,且计算量较大;光栏法的压缩算法也很好,并且可在数字化时实时处理,每次判断下一个数字化的点,且计算量较小,垂距法和偏角法简单,速度快,但有时会将曲线的特征点去掉而导致曲线形态失真。

4.1.2 图形编辑

在计算机地图制图系统中,图形编辑功能是实施几何数据及其有关的属性数据修改和更新的基础。

图形编辑除在数据采集过程中用来纠正错误外,在数据处理的全过程中都起着很重要的作用。图形编辑功能的强弱直接影响系统性能的好坏。

图形编辑功能应具有良好的人机交互界面、较快的响应速度、操作灵活、易于理解等,并具有对几何数据和属性编码的修改和更新功能,如点、线、面数据的增加、删除、变更、移动、分割、合并等,还应具有分层显示和窗口功能。

图形编辑的功能与数据组织有关,必须得到地图数据库的支持,其关键是点、线、面要素的数字化定位,即如何根据光标的位置找到需要编辑的目标。分别简要介绍如下。

1. 数据组织

计算机地图制图系统中的数据通常可以是按要素类型存取的,例如居民地、道路、境界、水系、地貌、植被等。在进行图形编辑时,为提高效率,可先确定在什么要素类型中进行操作,以便对选定的数据进行编辑。

地图数据所涉及的数据量很大,若每次编辑都针对全部地图数据进行,其定位查询和编辑操作所需的时间是难以忍受的,因此一般需要采用建索引文件的方法。常用的方法有格网索引和四叉树索引等。

建立索引文件后的图形编辑,不仅要修改原始的地图数据,而且要修改相关的索引文件。在对空间数据进行删除操作时,通常不直接删除地图数据库中的相关数据,而只在相应的索引文件中作标志;只有在重新整理数据库时,才进行真正的删除。在对建立了拓扑关系的地图矢量数据进行编辑时,可能会破坏原有的拓扑关系,这时还必须进行拓扑重构。

2. 点的定位

在计算机地图制图系统中,图形编辑是在计算机屏幕上进行的,因此首先应把地图数据库中图幅的坐标转换为当前屏幕状态的坐标系和比例尺。设屏幕上光标点为 $C(x,$

y),图幅上某一点状要素的坐标为 P(X,Y),则可设一定位半径为 D 的搜索圆。若 C 和 P 的距离 d 小于 D,则认为定位成功,即认为找到的点是 P;否则认为定位失败,继续搜索其他点。

d 值的计算公式为：

$$d=\sqrt{(X-x)^2+(Y-y)^2}$$

应用上式计算 d 时需进行乘方运算,影响了搜索的速度,因此可以把距离 d 值的计算改为：

$$d=\max(|X-x|,|Y-y|)$$

也即用矩形定位范围取代圆,以加快搜索速度。

3. 线的定位

设光标点坐标为 C(x,y),D 为定位搜索半径,线的坐标为 (x₁,y₁),(x₂,y₂),...,(x_n,y_n)。通过计算 C 到该线的每条直线段的距离 d_i(如图 4-5 所示),若 min(d₁,d₂,...d_{n-1})<D,则认为光标 C 找到了该条线;反之为未找到。在实际的定位过程中,可每计算一个距离 d_i 就进行一次比较,若 d_i<D,则定位成功,无须再计算后续直线段到点 C 的距离。

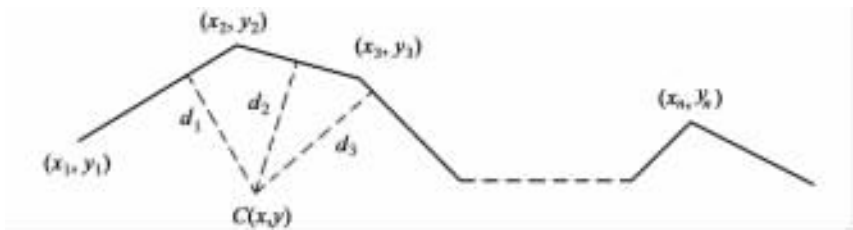


图 4-5 线的定位

为了加快定位速度,在对一条线段进行距离计算前,应先检查该线段是否可能在光标点的定位搜索范围内。即对由线段两端点组成的矩形再往外扩 D 的距离,构成新的矩形。若 C 落在该矩形内,才计算点到该直线段的距离;否则应放弃该直线段,而取下一直线段继续搜索(如图 4-6 所示)。类似地,对整条线也可求出其最大、最小坐标值 X_{min},Y_{min},X_{max},Y_{max},对由此构成的矩形再向外扩 D 的距离,并以此作定位搜索范围,只有当光标点 C 落在该外扩后的矩形范围内,才可能找到该条线,因此可以去除大量的不可能搜索到的情况,从而提高了定位速度。

d 的计算可以应用点到直线段的距离公式：

$$d=\frac{|(x-x_1)(y_2-y_1)-(y-y_1)(x_2-x_1)|}{\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}}$$

但是,直接使用该公式计算量较大,速度较慢,因此可从点 C(x,y)分别作水平和垂直方向的射线与线段(x₁,y₁)(x₂,y₂)相交得 d_x,d_y(见图 4-6),取 d_x,d_y 的最小值作为 C 点到该线段的近似距离 d'。由此可减小计算量,加快搜索速度。

近似距离 d' 的计算方法为：

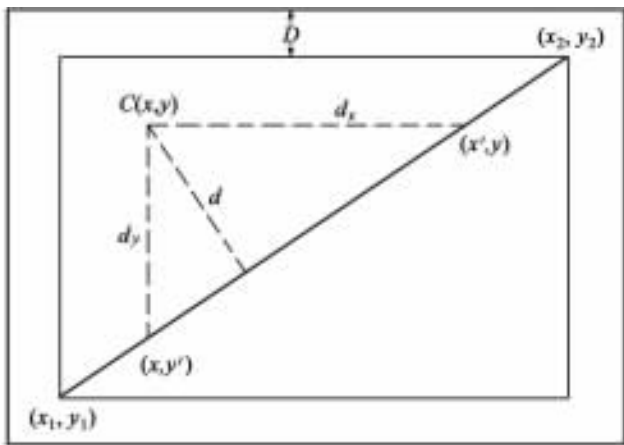


图 4-6 线段定位搜索范围

$$\begin{cases} x' = \frac{(x_2 - x_1)(y - y_1)}{y_2 - y_1} + x_1 \\ y' = \frac{(y_2 - y_1)(x - x_1)}{x_2 - x_1} + y_1 \\ d_x = |x' - x| \\ d_y = |y' - y| \\ d' = \min(d_x, d_y) \end{cases}$$

4. 面的定位

可以通过判断光标点 $C(x, y)$ 是否在目标多边形内来进行面的定位。若 C 在多边形内, 则说明定位成功; 反之亦然。判断点是否在多边形内最常用的算法有铅垂线法, 介绍如下。

铅垂线法根据的是拓扑学原理, 即一条直线与一条封闭的曲线相交, 其交点的个数是偶数。该法的基本思想是从光标点引垂线, 计算与多边形的交点个数。若交点个数为奇数, 则说明该点在多边形内; 若交点个数为偶数, 则说明该点在多边形外 (如图 4-7 所示)。

为了加快系统的搜索速度, 可以先搜索由该多边形的最大、最小坐标值构成的矩形范围。只有当光标点落在该矩形范围内, 才有可能找到该面; 否则放弃对该多边形的进一步计算和判断。由此可大大减少运算量, 提高定位速度。

在计算垂线与多边形的交点个数时, 并不需要每次都对多边形的每一条线段进行交点坐标的具体计算。对不可能有交点的线段可通过简单的坐标比较迅速去除。方法为: 设直线段为 $(x_1, y_1)(x_2, y_2)$, 光标点为 $C(x, y)$, 只有当 $x_1 \leq x \leq x_2$ 或 $x_2 \leq x \leq x_1$ 时才有可能与垂线相交, 才有必要继续进行进一步的交点计算和判断, 判断方法如下:

① 当 $y > y_1$ 且 $y > y_2$ 时, 直线段必然与 C 点所作的垂线相交; 而当 $y < y_1$ 且 $y < y_2$ 时, 必然不与 C 点所作的垂线相交。这样, 就可不必进行交点坐标的计算而判断出是否有交点了。

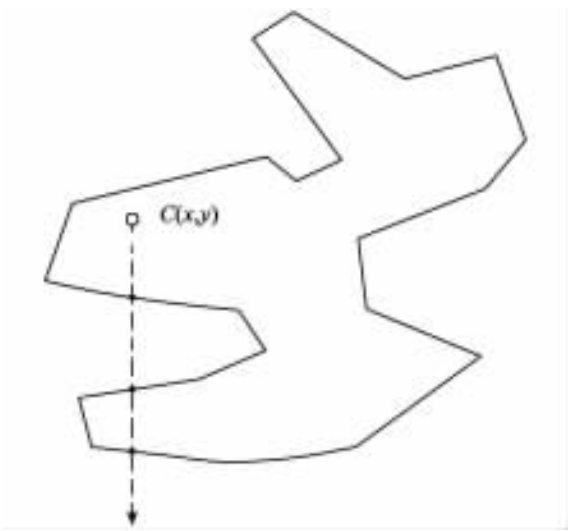


图 4-7 铅垂线法示意图

②当 $y_1 \leq y \leq y_2$ 或 $y_2 \leq y \leq y_1$ 时,可求出铅垂线与直线段的交点 (x, y') ,若 $y' < y$,则该点是真交点 若 $y' > y$,则不是真交点 若 $y' = y$,则交点在线上 ,即光标点在多边形的边上。

4. 1. 3 数据变换

数据变换是数据处理的形式之一 ,内容包括几何变换和数据结构、格式等的变换。在此主要介绍几何变换中的二维图形变换。

1. 二维图形变换的概念

二维图形变换是指对平面图形经过几何变换后产生新的平面图形。它提供了在计算机地图制图中建立和改变图形的方法。二维图形变换除可使图形的位置改变外 ,还可以将图形放大、缩小、拉伸以及扭曲变形。

对二维图形而言 ,一个点可以用一对坐标 (x, y) 来表示 ,这个坐标值可以作为一个一行两列的矩阵 (x, y) 的元素 ,也可以作为一个两行一列的矩阵 $\begin{pmatrix} x \\ y \end{pmatrix}$ 的元素。

二维图形是点的集合 ,其图形变换的基本思想是利用变换矩阵作为一个算子 ,将该算子作用到二维图形的每一个点的位置向量 ,以得到点在几何变换后的位置。

为了便于统一表达二维图形的变换矩阵 ,可以引入点的齐次坐标。所谓齐次坐标 ,就是用 $n + 1$ 维向量表示一个 n 维向量。如二维点 (x, y) 的齐次坐标可表示为 (sx, sy, s) ($s \neq 0$)。齐次坐标的表示不惟一 ,仅当 $s = 1$ 时 (此时称为正常化齐次坐标) ,前两维保持原值不变。引入齐次坐标的好处是可以把多种变换用统一的矩阵算子形式表达出来。例如 ,二维图形变换矩阵算子可表示为：

$$T = \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}$$

其中 a, b, d, e 可对图形进行缩放、旋转、错移等变换 ; g, h 可对图形进行平移变换 ; c, f 可对图形进行透视变换 ; i 可对图形进行整体伸缩变换。当 $i < 1$ 时 ,图形被放大 ;当 $i > 1$ 时 ,图形被缩小 ;当 $i = 1$ 时 ,图形不变。

下面介绍几种常用的二维图形变换。

2. 平移变换

平移变换是将图形中的每个点进行平行于坐标轴的移动。将一个点 (x, y) 沿水平方向移动 T_x 单位 ,垂直方向移动 T_y 单位 ,平移到一个新位置 (x', y') ,其数学表达式为 :

$$\begin{cases} x' = x + T_x \\ y' = y + T_y \end{cases}$$

如果 T_x 是正值 ,则点向右移动 ;如果 T_x 是负值 ,则点向左移动。类似地 ,如果 T_y 是正值 ,则点向上移动 ;反之 ,点向下移动。

平移变换的矩阵算子为 :

$$T_t(T_x, T_y) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ T_x & T_y & 1 \end{pmatrix}$$

齐次坐标平移变换为 :

$$(x' \ y' \ 1) = (x \ y \ 1) \times T_t(T_x, T_y) = (x + T_x \ y + T_y \ 1)$$

3. 比例变换

比例变换是用比例因子(S_x 及 S_y)乘以图形的点集 ,使图形放大或缩小的变换。点的比例变换的数学表达式为 :

$$\begin{cases} x' = x \times S_x \\ y' = y \times S_y \end{cases}$$

比例变换的矩阵算子为 :

$$T_s(S_x, S_y) = \begin{pmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

齐次坐标比例变换为 :

$$(x' \ y' \ 1) = (x \ y \ 1) \times T_s(S_x, S_y) = (x \times S_x \ y \times S_y \ 1)$$

式中 :

- ①当 $S_x = S_y = 1$ 时 ,为恒等变换 ;
- ②当 $S_x = S_y \neq 1$ 时 ,为相似变换 ;
- ③当 $S_x = S_y > 1$ 时 ,为等比例放大 ;
- ④当 $S_x = S_y < 1$ 时 ,为等比例缩小 ;
- ⑤当 $S_x \neq S_y$ 时 ,图形沿两个坐标轴方向进行非等比例变换。

特别地 ,

- ⑥当 $S_x = 1, S_y = -1$ 时 ,为关于 x 坐标轴的对称变换 ;
- ⑦当 $S_x = -1, S_y = 1$ 时 ,为关于 y 坐标轴的对称变换 ;
- ⑧当 $S_x = S_y = -1$ 时 ,为关于坐标原点的对称变换。

4. 旋转变换

旋转变换是将图形绕一固定点顺时针或逆时针方向进行旋转。一般规定,逆时针方向为正,顺时针方向为负。

设点 (x, y) 沿逆时针旋转 θ 角,变换后的点为 (x', y') ,数学表达式为:

$$\begin{cases} x' = x\cos\theta - y\sin\theta \\ y' = x\sin\theta + y\cos\theta \end{cases}$$

旋转变换的矩阵算子为:

$$T_r(\theta) = \begin{pmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

齐次坐标旋转变换为:

$$(x' \ y' \ 1) = (x \ y \ 1) \times T_r(\theta) = (x\cos\theta - y\sin\theta \ x\sin\theta + y\cos\theta \ 1)$$

5. 错移变换

错移变换是使图形产生一个切变。设 x 方向及 y 方向的错移系数分别为 C_x 及 C_y ,其数学表达式为:

$$\begin{cases} x' = x + C_x \times y \\ y' = y + C_y \times x \end{cases}$$

错移变换的矩阵算子为:

$$T_c(C_x, C_y) = \begin{pmatrix} 1 & C_y & 0 \\ C_x & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

齐次坐标错移变换为:

$$(x' \ y' \ 1) = (x \ y \ 1) \times T_c(C_x, C_y) = (x + C_x \times y \ y + C_y \times x \ 1)$$

错移变换中,沿 x 方向或 y 方向错移较为常用,即:

- ①当 $C_x \neq 0, C_y = 0$ 时,图形沿 x 方向错移变换;
- ②当 $C_x = 0, C_y \neq 0$ 时,图形沿 y 方向错移变换。

6. 复合变换

上面所讨论的图形变换都是基本的二维图形变换,实际应用中可以将多个基本变换组合起来使用。例如,设一个矢量箭头符号的齐次坐标为:

$$A = \begin{pmatrix} 1 & 2 & 1 \\ 2 & 2 & 1 \\ 2 & 1 & 1 \\ 2 & 2 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

可以对其进行比例、旋转、平移等连续变换,如可令

$$A' = A \times T_s(0.5 \ 0.5) \times T_r(-90^\circ) \times T_t(3 \ 3)$$

其结果如图4-8所示。

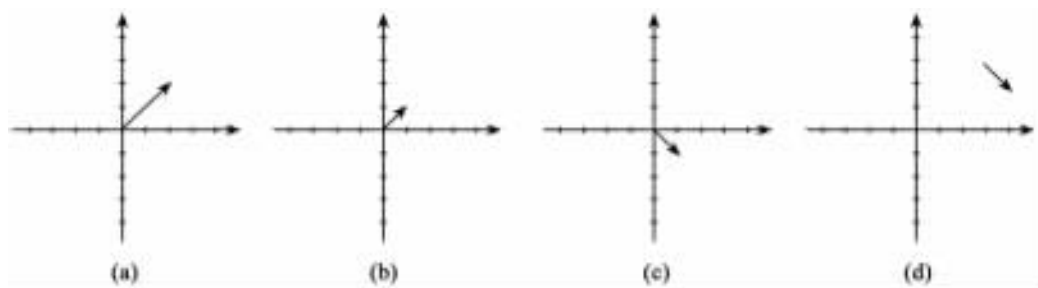


图 4-8 基本变换的组合

图 4-8 中 (a) 为原图 (b)、(c)、(d) 分别为在原图基础上连续进行的比例、旋转和平移变换。也可以先将几个变换矩阵的乘积求出后再作用于原图, 其结果是一样的, 且速度还会更快。

要注意的是, 上述二维图形的基本变换是相对于坐标原点进行的。在具体应用中, 有时需要相对于任意点来进行变换。例如, 若要相对任意一点 (x_0, y_0) 作比例变换, 其变换过程为: 先把坐标原点平移到 (x_0, y_0) 处, 然后在新的坐标系下作比例变换, 再将坐标原点平移回原坐标系的原点。其组合变换矩阵为:

$$T = T_t(-x_0, -y_0) \times T_s(S_x, S_y) \times T_t(x_0, y_0) = \begin{pmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ x_0(1 - S_x) & y_0(1 - S_y) & 1 \end{pmatrix}$$

类似地, 若要相对任意一点 (x_0, y_0) 作旋转 θ 角变换, 其组合变换矩阵如下:

$$T = T_t(-x_0, -y_0) \times T_r(\theta) \times T_t(x_0, y_0) = \begin{pmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ x_0(1 - \cos\theta) + y_0\sin\theta & y_0(1 - \cos\theta) - x_0\sin\theta & 1 \end{pmatrix}$$

4.1.4 曲线光滑

在地图数据采集过程中获取的曲线数据是离散的点列数据, 有时由于其点列距离过大 (尤其是经过压缩后), 要将其图形绘制出来就必须经过曲线光滑处理。

所谓曲线光滑, 就是根据已知离散点列用曲线插值或拟合的方法建立符合某种要求的连续光滑曲线函数, 并按该函数计算加密点列来完成曲线的光滑连接。曲线光滑的数学方法有多种, 下面介绍一些常用的方法。

1. 正轴抛物线加权平均法

该方法的基本思想是按点列数据的顺序, 以每相邻的三点为基础作一条抛物线, 将每相邻两点间前后两条抛物线重叠部分用加权平均曲线作为最终的插值曲线 (如图 4-9 所示)。该插值曲线的基本算法如下:

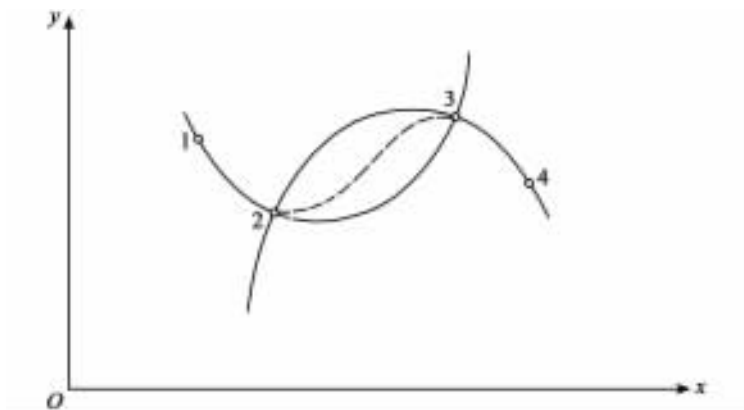


图 4-9 正轴抛物线加权平均法示意图

地图上的曲线较为复杂,一般为多值函数,故设该插值曲线的参数方程为:

$$\begin{cases} x = X(t) \\ y = Y(t) \end{cases}$$

其中,参数 t 是相对弦长参数,相应的正轴抛物线方程为:

$$\begin{cases} X(t) = a + bt + ct^2 \\ Y(t) = d + et + ft^2 \end{cases}$$

在过 $i-1$ 、 i 、 $i+1$ 三点建立抛物线参数方程时,令:

- ①当通过 $i-1$ 点时 $t_{i-1} = 0$;
- ②当通过 i 点时 $t_i = 0.5$;
- ③当通过 $i+1$ 点时 $t_{i+1} = 1$ 。

将 x_{i-1} 、 x_i 、 x_{i+1} 、 y_{i-1} 、 y_i 、 y_{i+1} 和 t_{i-1} 、 t_i 、 t_{i+1} 代入上述抛物线方程可得:

$$\begin{pmatrix} 1 & t_{i-1} & t_{i-1}^2 \\ 1 & t_i & t_i^2 \\ 1 & t_{i+1} & t_{i+1}^2 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} x_{i-1} \\ x_i \\ x_{i+1} \end{pmatrix}$$

和

$$\begin{pmatrix} 1 & t_{i-1} & t_{i-1}^2 \\ 1 & t_i & t_i^2 \\ 1 & t_{i+1} & t_{i+1}^2 \end{pmatrix} \begin{pmatrix} d \\ e \\ f \end{pmatrix} = \begin{pmatrix} y_{i-1} \\ y_i \\ y_{i+1} \end{pmatrix}$$

解该方程,可求出系数 a 、 b 、 c 、 d 、 e 、 f ,也即确定了过 $i-1$ 、 i 、 $i+1$ 三点建立的抛物线参数方程:

$$X_L(t) = (1 \quad t \quad t^2) \begin{pmatrix} 1 & 0 & 0 \\ -3 & 4 & -1 \\ 2 & -4 & 2 \end{pmatrix} \begin{pmatrix} x_{i-1} \\ x_i \\ x_{i+1} \end{pmatrix}$$

和

$$Y_L(t) = (1 \quad t \quad t^2) \begin{pmatrix} 1 & 0 & 0 \\ -3 & 4 & -1 \\ 2 & -4 & 2 \end{pmatrix} \begin{pmatrix} y_{i-1} \\ y_i \\ y_{i+1} \end{pmatrix}$$

式中 $X_L(t)$ 及 $Y_L(t)$ 为左分支, 且 $0 \leq t \leq 1$ 。

同理也可求出后续过 $i, i+1, i+2$ 三点建立的抛物线参数方程, 即右分支:

$$X_R(t) = (1 \quad t \quad t^2) \begin{pmatrix} 1 & 0 & 0 \\ -3 & 4 & -1 \\ 2 & -4 & 2 \end{pmatrix} \begin{pmatrix} x_i \\ x_{i+1} \\ x_{i+2} \end{pmatrix}$$

和

$$Y_R(t) = (1 \quad t \quad t^2) \begin{pmatrix} 1 & 0 & 0 \\ -3 & 4 & -1 \\ 2 & -4 & 2 \end{pmatrix} \begin{pmatrix} y_i \\ y_{i+1} \\ y_{i+2} \end{pmatrix}$$

式中 $X_R(t)$ 及 $Y_R(t)$ 为右分支, 且 $0 \leq t \leq 1$ 。

考虑到应在点列 i 及 $i+1$ 左右两抛物线间取加权平均值作最终插值曲线, 设左、右两抛物线的权函数分别为 $W_L(t)$ 和 $W_R(t)$, 并将参数 t 的原点统一到 i 点, 即

$$\begin{cases} W_L(t) = 1 - 2t \\ W_R(t) = 2t \end{cases} \quad (0 \leq t \leq 0.5)$$

点 $i, i+1$ 间加权平均插值曲线为:

$$\begin{cases} X_{i \sim i+1}(t) = W_L(t)X_L(t) + W_R(t)X_R(t) \\ Y_{i \sim i+1}(t) = W_L(t)Y_L(t) + W_R(t)Y_R(t) \end{cases}$$

即

$$\begin{cases} X_{i \sim i+1}(t) = x_i - (x_{i-1} - x_{i+1})t + 2(2x_{i-1} - 5x_i + 4x_{i+1} - x_{i+2})t^2 - 4(x_{i-1} - 3x_i + 3x_{i+1} - x_{i+2})t^3 \\ Y_{i \sim i+1}(t) = y_i - (y_{i-1} - y_{i+1})t + 2(2y_{i-1} - 5y_i + 4y_{i+1} - y_{i+2})t^2 - 4(y_{i-1} - 3y_i + 3y_{i+1} - y_{i+2})t^3 \end{cases} \quad (0 \leq t \leq 0.5 \quad j=2, 3, 4, \dots, n-2)$$

在曲线的两端点处应各补一点, 以使有效的加权平均插值曲线能从曲线的第一点开始并到达最后一点。

2. 三次参数样条曲线

为了获得一条通过各个点列的光滑曲线, 人们常常使用一根有弹性的均匀细木条(称为样条)将它压在各个点列处, 并强迫它通过这些点, 然后沿这根样条画出曲线, 这条曲线就是样条曲线。

样条曲线有多种, 其中, 三次样条曲线次数低, 便于计算, 既能保证曲线上各点的切线斜率连续变化, 又能保证曲率连续变化, 得到了广泛的应用。

各段三次样条曲线都是由其起始点位置矢量 P_0, P_1 和起始点处的切线矢量 P'_0, P'_1 来确定的。设弦长为参数, 就可以得到三次参数样条曲线, 其矩阵表达形式为:

$$P(t) = \begin{pmatrix} t^3 & t^2 & t & 1 \end{pmatrix} \begin{pmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} P_0 \\ P_1 \\ P'_0 \\ P'_1 \end{pmatrix} \quad (0 \leq t \leq 1)$$

具体计算时,可以分量形式表示:

$$\begin{cases} P(t) = (X(t), Y(t)) \\ P_0 = (x_0, y_0) \\ P_1 = (x_1, y_1) \\ P'_0 = (x'_0, y'_0) \\ P'_1 = (x'_1, y'_1) \end{cases}$$

即有

$$X(t) = \begin{pmatrix} t^3 & t^2 & t & 1 \end{pmatrix} \begin{pmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x'_0 \\ x'_1 \end{pmatrix}$$

和

$$Y(t) = \begin{pmatrix} t^3 & t^2 & t & 1 \end{pmatrix} \begin{pmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_0 \\ y_1 \\ y'_0 \\ y'_1 \end{pmatrix}$$

将曲线首末两点的位置矢量 P_0, P_1 和切线矢量 P'_0, P'_1 的 x, y 分量代入上式,即可求出 t 在 $0 \sim 1$ 之间变化时相应点的 $X(t), Y(t)$,从而可画出一段三次参数样条曲线。

3. Bezier 曲线

曲线光滑除插值外也可用拟合曲线的方法,其中的一种方法是将已知的离散点列数据先连成一个折线多边形(又叫特征多边形),然后用光滑的参数曲线段去逼近这个多边形。Bezier 曲线就是这样的一种参数曲线。

n 次 Bezier 曲线的矢量式为:

$$P(t) = \sum_{i=0}^n P_i B_{i,n}(t) \quad (0 \leq t \leq 1)$$

式中:

$$B_{i,n}(t) = C_n^i t^i (1-t)^{n-i};$$

$$C_n^i = \frac{n!}{i!(n-i)!}$$

式中规定 $0^0 = 1$ $i = 0, 1, 2, \dots, n$ 。向量 P_i 可以看成是特征多边形的各顶点。在 Bezier 曲线中,最常用的有二次和三次 Bezier 曲线,简介如下。

(1) 二次 Bezier 曲线

在上述 n 次 Bezier 曲线的矢量式中,令 $n=2$,即可得到二次 Bezier 曲线的矩阵表达式:

$$P(t) = (t^2 \quad t \quad 1) \begin{pmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} P_0 \\ P_1 \\ P_2 \end{pmatrix} \quad (0 \leq t \leq 1)$$

式中 $P_i(x_i, y_i) (i=0, 1, 2, \dots, n)$ 是特征多边形的三个顶点, 其分量形式为:

$$X(t) = (t^2 \quad t \quad 1) \begin{pmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \end{pmatrix}$$

和

$$Y(t) = (t^2 \quad t \quad 1) \begin{pmatrix} 1 & -2 & 1 \\ -2 & 2 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_0 \\ y_1 \\ y_2 \end{pmatrix} \quad (0 \leq t \leq 1)$$

这是一段抛物线, 它以 P_0, P_2 为端点, 抛物线的顶点位于中线的中点处 (如图 4-10 所示)。

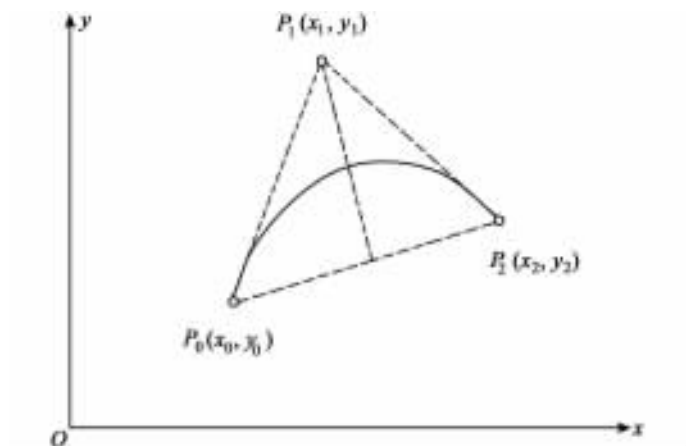


图 4-10 二次 Bezier 曲线

(2) 三次 Bezier 曲线

当在上述 n 次 Bezier 曲线的矢量式中, 令 $n=3$ 时, 即可得到三次 Bezier 曲线的矩阵表达式:

$$P(t) = (t^3 \quad t^2 \quad t \quad 1) \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{pmatrix} \quad (0 \leq t \leq 1)$$

式中 $P_i(x_i, y_i) (i=0, 1, 2, \dots, n)$ 是特征多边形的四个顶点, 其分量形式为:

$$X(t) = (t^3 \quad t^2 \quad t \quad 1) \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

和

$$Y(t) = \begin{pmatrix} t^3 & t^2 & t & 1 \end{pmatrix} \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{pmatrix} \quad (0 \leq t \leq 1)$$

图 4-11 是三次 Bezier 曲线及其特征多边形示意。

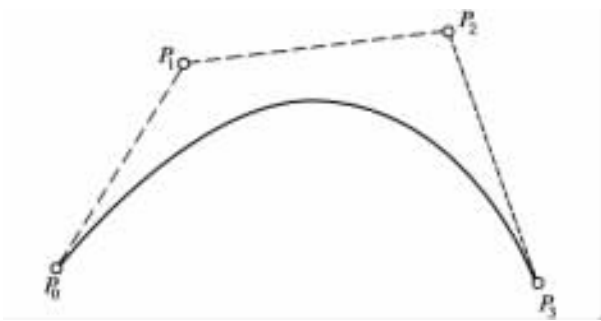


图 4-11 三次 Bezier 曲线

4. 1. 5 矢量符号的生成

地图符号根据几何形状可分为点状、线状和面状三类 ,其中大多数符号比较容易用矢量形式的坐标来表示。符号空间平面内这些点的坐标及绘或不绘指令编码的有序集合称为矢量符号数据。例如 ,一条直线可由其从起点到终点的绘图指令以及相应的两端点坐标构成。

总的来说 ,可以采用三种方法生成矢量符号 ,即信息块法、编程法以及将信息块法和编程法综合起来的方法。本节仅讨论编程方法。该方法对每一类地图符号编写一个绘图子程序 ,并把这些绘制符号的子程序组成程序库。在需要绘图时 ,按符号的编号调用库中相应子程序 ,输入有关参数 ,该程序根据参数及已知数据计算绘图矢量 ,从而完成地图符号的绘制。在使用该方法时 ,关键是应对绘图要素进行精心的分类、编程和选择合适的参数。

下面分别介绍编程方法在点状、线状及面状符号生成中的应用。

1. 点状符号

点状符号是定位于一点的个体符号 ,通常可用直线段和圆弧组合而成。现以绘制圆为例说明其算法。

任何圆都可用正多边形来逼近 ,其边数越多 ,圆弧越光滑 ,但是边数太多会浪费绘图时间。为此 ,可以通过计算的方法来确定适当的边数。选取圆心角 θ (如图 4-12 所示) θ 相对应的正多边形与圆弧之间拱高为 d ,这样 ,圆心角 θ 与圆半径 r 之间的关系为 :

$$d = r \left(1 - \cos \frac{\theta}{2} \right)$$

由此可解出圆心角

$$\theta = 2 \arccos \left(1 - \frac{d}{r} \right) \approx 2.8 (d/r)^{1/2}$$

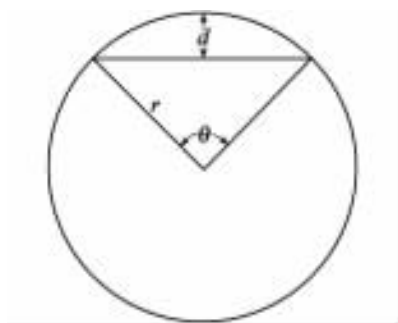


图 4-12 圆的绘制

和多边形的边数

$$n = [2\pi/\theta]$$

式中 $[]$ 是指对括号内数据取整。因此,只要给定了限差 d (一般取 $0.05 \sim 0.1 \text{ mm}$) 和半径 r 就可算出 n 和 θ 。这样,半径为 r 的圆可以角增量 θ ,按逆时针连续旋转计算出各点坐标并顺次连接的正 n 边形取代。各点坐标按下式计算:

$$\begin{cases} x_i = r \cos(i \times \theta) + x_s \\ y_i = r \sin(i \times \theta) + y_s \end{cases} \quad (i = 0, 1, 2, \dots, n)$$

式中 x_s, y_s 为圆心坐标。画圆从 (x_0, y_0) 开始,顺序连至 (x_n, y_n) ,再继续连至 (x_0, y_0) ,以使圆周完全闭合。

在上述算法的基础上,当要绘制一段圆弧时,只需设计好起始角和终止角,即可绘制任一圆弧。

椭圆的绘制与正圆的绘制相类似,各点坐标按下式计算:

$$\begin{cases} x_i = a \cos(i \times \theta) + x_s \\ y_i = b \sin(i \times \theta) + y_s \end{cases} \quad (i = 0, 1, 2, \dots, n)$$

式中 a, b 分别为椭圆的横半轴和纵半轴。

根据上述算法,再配合以绘制某些直线段的功能,即可方便地编制出各种绘制点状符号的子程序。

2. 线状符号

线状符号表示的是线状分布的制图要素符号。由于线状符号的长度不固定,其绘制过程实质上是线状符号的结构单元沿中轴线配置绘图的过程,已知条件是中心轴线及所需配置的线状符号结构尺寸。

下面以土堤符号的绘制过程为例进行介绍(如图 4-13 所示,图中左下角为土堤符号单元的结构尺寸)。

土堤符号的绘制过程中需要解决的关键是如何确定短横线在中心轴线上的坐标位置及短横线两端点的坐标。

这里,中心轴线是从指定起点开始按顺序排列的直线段衔接而成的折线。任取其中一直线段,它与前一线段连接点为第一节点,坐标为 (x_1, y_1) ;它与后一线段连接点称为第二节

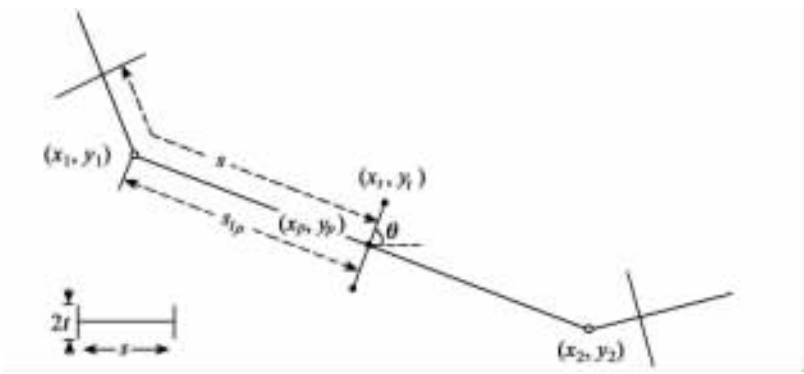


图 4-13 土堤符号绘制过程示意

点 坐标为 (x_2, y_2) 则该两点间直线段长为：

$$s_{12} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

离第一节点距离为 s_{1p} 的短横线位置 (x_p, y_p) 可用下式计算：

$$\begin{cases} x_p = x_1 + (x_2 - x_1) s_{1p} / s_{12} \\ y_p = y_1 + (y_2 - y_1) s_{1p} / s_{12} \end{cases}$$

设短横线定向角为 θ 则

$$\begin{cases} \cos\theta = (y_1 - y_2) / s_{12} \\ \sin\theta = (x_2 - x_1) / s_{12} \end{cases}$$

短横线两端点坐标为：

$$\begin{cases} x_t = x_p \pm t \cos\theta \\ y_t = y_p \pm t \sin\theta \end{cases}$$

计算出当前短横线的坐标后 就可接着计算下一短横线的坐标。

在离第一节点 (x_1, y_1) 距离 s_{1p} 的基础上加上 s 得到新的 s'_{1p} 即

$$s'_{1p} = s_{1p} + s$$

将 s'_{1p} 与 s_{12} 进行比较 若 $s'_{1p} \leq s_{12}$ 则令 s'_{1p} 为新的 s_{1p} 即

$$s_{1p} = s'_{1p}$$

将新的 s_{1p} 代入上述公式 计算下一短横线在折线 $(x_1, y_1)(x_2, y_2)$ 上的位置和新的短横线端点坐标 反复进行该步骤直到 $s'_{1p} > s_{12}$ 这时说明折线 $(x_1, y_1)(x_2, y_2)$ 上已容不下下一个短横线 令

$$s'_{1p} = s'_{1p} - s_{12}$$

并把 2 点作为 1 点 且把下一个节点作为 2 点 重新计算 s_{12} 再对 s'_{1p} 与 s_{12} 进行比较后决定运算流向。

如此循环 即可把中心轴线都绘上短横线 再把中心轴线绘成土堤中心线 就完成了土堤符号的绘制。

为使绘出的土堤符号图形美观 应注意两点：

①若中心轴线长不为短横线间隔的整倍数 可适当调整间隔 s 以使得中心轴线长为短

横线间隔的整倍数。

②如果短横线与中心轴线交点在节点(x_2, y_2)上,或接近该点,当此节点是最后一点时,短横线照常绘制,否则应绘在过该点的角平分线上。

上述土堤符号绘制的算法经扩展可获得离要素中心轴线等距离的两条平行实线(或虚线),如双线公路、街道等,同样还可产生陡坎、长城、境界线、大车路、地类界等一类沿中心轴线保持一定规律配置的点和短线组成的地图符号。

3. 面状符号

面状符号是用来表示面状分布的地图要素的符号,面状地图要素的面域通常由若干封闭多边形组成。面状符号的共同特点就是在面域内填绘不同方向、不同间隔、不同粗细的“晕线”,或规则分布的个体符号、花纹或颜色。

面状符号多种多样,其中“晕线”是最为普遍和基础的。所谓晕线,就是一组平行的等间距的线,确定晕线结构的主要参数是平行线的间距和倾角。有多种绘制晕线的算法,举例介绍如下。

设晕线的平行线间距为 d ,晕线与 x 轴倾角为 θ (如图 4-14 所示),在多边形内填绘晕线的主要算法步骤为:

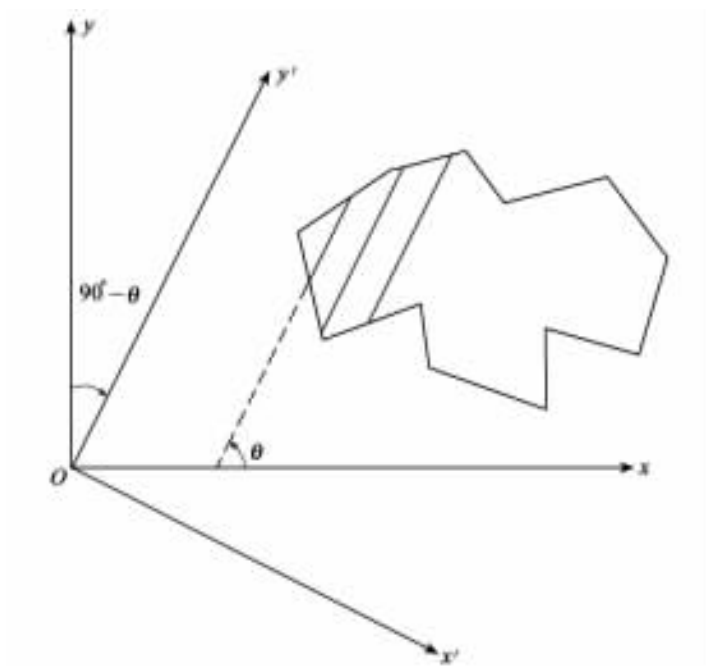


图 4-14 多边形内填绘晕线算法示意

(1) 坐标系变换

将原多边形顶点的坐标系 xOy 顺时针旋转 $90^\circ - \theta$,使新坐标轴 y' 与晕线平行,并计算在新坐标系 $x'O'y'$ 下多边形各顶点的坐标。

(2) 求新坐标系下第一条晕线的位置

计算新坐标系下多边形轮廓各顶点坐标的最小值 x'_{\min} 和最大值 x'_{\max} , 确定第一条晕线在 x' 坐标轴上的值

$$s = [x'_{\min}/d] \times d$$

式中 $[]$ 表示取整 d 为平行线的间距。

若

$$s \leq x'_{\min}$$

则令

$$s = s + d$$

(3) 求晕线与多边形各边交点

首先应判别晕线与某一边是否有交点, 判别式如下:

若

$$(x'_i - s)(x'_{i+1} - s) < 0$$

则有交点

$$\begin{cases} x_{sl} = s \\ y_{sl} = y'_i + [(y'_{i+1} - y'_i)(s - x'_i)/(x'_{i+1} - x'_i)] \end{cases}$$

若

$$(x'_i - s) = 0 \text{ 且 } (x'_{i+1} - s)(x'_{i+1} - s) < 0$$

则交点为

$$\begin{cases} x_{sl} = x'_i \\ y_{sl} = y'_i \end{cases}$$

(4) 排队和配对输出

一条晕线与多边形的交点可能有多个, 按 y' 值排队, 并顺序记录排队后的各点坐标。将交点坐标作坐标系反时针旋转 $90^\circ - \theta$ 的变换, 其序不变, 并按照 1-2, 3-4, ... 的点序配对绘线输出。

(5) 循环判断

一条晕线绘好后, 计算下一条晕线的位置:

$$s = s + d$$

当 $s < x'_{\max}$ 时返回到 (3)、(4)、(5) 进行循环; 当 $s \geq x'_{\max}$ 时停止运算。

上述多边形内填绘晕线的方法适当改进后可绘平行或垂直点线、虚线以及另一组晕线, 也可利用点、实线、虚线组合, 进行各种图案符号绘制, 还可在多边形内以绘晕线方法为基础配置各种规则分布的个体符号。

4.1.6 数字地形模型

数字地形模型 (DTM) 是一种对空间起伏变化的连续表示方法, 常用来模拟表示地面高程的起伏, 也可以用于模拟其他二维表面的连续高度变化, 如气温、降水量等。对于一些不具有三维空间连续分布特征的地理现象, 如人口密度等, 从宏观上讲, 也可以用 DTM 来表示, 进行计算和定量分析。

数字地形模型有许多用途。例如, 在土石工程规划中计算挖填土石方量, 计算坡度和坡向, 分析透视情况, 绘制坡度图、晕渲图等; 用于地貌分析, 计算侵蚀和径流等; 当用专题数据 (如气温、土壤等) 代替高程后, 可进行各种组合分析 (如资源调查、土地规划等), 还可为军事目的显

示地形景观,为武器精确制导进行地形匹配等。另外,在必要时还可用来引绘等值线。

数字地形模型有三种主要的表示方法,即规则格网 DTM、不规则三角网 DTM 和等值线,分别介绍如下。

1. 规则格网 DTM

规则格网 DTM 是数字地形模型中常用的形式,其数据的组织类似于栅格数据,只是每个栅格单元的值是高程值。即规则格网 DTM 是一种高程矩阵。其高程数据可直接由解析立体测图仪获取,也可由规则或不规则的离散数据内插产生。这里主要介绍用离散数据点来建立规则格网 DTM 的方法。

由离散数据点建立规则格网 DTM 是在原始数据呈离散分布,或原有的规则格网 DTM 密度不够时需使用的方法。其基本思路是选择适当的数学函数模型,利用已知点上的信息求出函数的待定系数,然后求算规则格网点上的高程值。离散点构建规则格网 DTM 多采用的是各种内插方法。内插方法对 DTM 精度的影响取决于原始采样点的密度和分布。较为常用的内插方法有距离加权平均法、多项式内插法等,简介如下。

(1) 距离加权平均法

设所需计算格网点 p 的平面坐标为 (x_p, y_p) , 高程为 z_p , 则

$$z_p = \sum_{i=1}^n w_i \cdot z_i / \sum_{i=1}^n w_i$$

式中 $i=1, 2, \dots, n$ 表示取 p 点周围的 n 个点, z_i 为第 i 点的高程, w_i 为第 i 点的权函数。

$$w_i = 1/d_i^t$$

令 $t=2$, 则 i 点 (x_i, y_i) 到格网点 (x_p, y_p) 的距离为:

$$d_i = \sqrt{(x_i - x_p)^2 + (y_i - y_p)^2}$$

n 的取值一般为 $4 \sim 10$, 即在构建格网的过程中,经常需要选取与插值点距离最近的 $4 \sim 10$ 个离散点进行计算。通常可以 p 点为圆心, r 为半径进行搜索。半径 r 可适当变化,使圆内离散点数控制在 $4 \sim 10$ 个即可。初始半径的计算公式为:

$$r = \sqrt{\frac{7S}{\pi N}}$$

式中 S 为制图区域的面积, N 为制图区域内总的离散点数。

也可以在插值点 p 处建立一正方形的选取框,通过坐标值比较就可找出落入框内的数据点。当落入框内的数据点较多时,可缩小框的尺寸;反之,增大框的尺寸。选取框初始边长的计算公式为:

$$L = \sqrt{\frac{nS}{N}}$$

式中 n 常取值 $(4+10)/2=7$, S 为制图区域的面积, N 为制图区域内总的离散点数。

(2) 多项式内插法

多项式内插法中较为简单实用的是线性内插法和双线性多项式内插法。线性内插法的数学模型为:

$$z = a_0 + a_1x + a_2y$$

双线性多项式内插法的数学模型为:

$$z = a_0 + a_1x + a_2y + a_3xy$$

式中 x, y 为平面坐标 z 为高程 a_0, a_1, a_2, a_3 为待定系数。

内插过程中,只需将与插值点 p 距离最近的 3 个点或 4 个点的坐标值和高程值代入方程,即可解出全部系数,然后将插值点的坐标代入方程,即可计算出该点的高程值。在具体取数据点时,可采用上述搜索圆或搜索框的方法。

规则格网 DTM 的优点是 结构简单,应用方便,易于管理。其缺点是 高程点的内插会损失精度,格网过大时将难以表示局部地形特征(如山峰等),地形简单地区存在大量冗余数据。

2. 不规则三角网 DTM

不规则三角网 DTM 直接利用原始采样点进行地形表面的重建,将地形表面剖分为连续的相互连接的三角面,三角面的形状和大小取决于不规则分布的数据点的密度和位置。

建立不规则三角网 DTM 就是由已知的离散数据点构建三角网,即确定哪三个数据点构成一个三角形,也称为自动连接三角网。对于平面上 n 个离散点,其平面坐标为 (x_i, y_i) , $i = 1, 2, \dots, n$, 将其中相近的三个点构成最佳三角形,使每个离散点都成为三角形的顶点(如图 4-15 所示)。

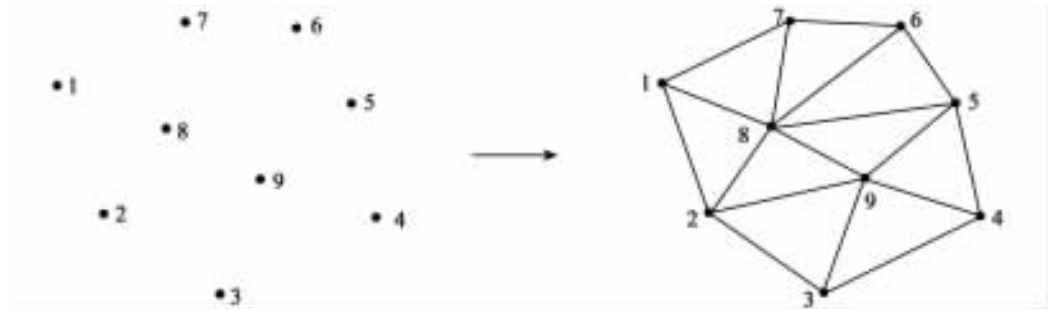


图 4-15 不规则三角网的建立

自动连接三角网的结果是产生所有三角形的三个顶点的标号组合集:

$$\{1, 8, 2, 2, 8, 9, 2, 3, 9, \dots\}$$

构建三角网的方法有多种,近年来较为常用的主要有 Delaunay 三角网。

Delaunay 三角网的构建即不规则三角网的构建。Delaunay 三角网要求为了获得最佳三角形,在构建三角网时,应尽可能使三角形的三个内角均成锐角,即符合 Delaunay 三角形产生的准则:

①任何一个 Delaunay 三角形的外接圆内不能包含任何其他离散点。

②相邻两个 Delaunay 三角形构成凸四边形,在交换凸四边形的对角线之后,6 个内角的最小者不再增大。该性质即为最小角最大准则。

下面介绍构造 Delaunay 三角网的凸包插值算法(Tsai, 1993)。

(1) 凸包生成

①求出离散点集中满足 $\min(x - y)$, $\min(x + y)$, $\max(x - y)$, $\max(x + y)$ 的 4 个点,并按逆时针方向组成一个点的链表。这 4 个点是离散点中与包含离散点的外接矩形的 4 个角点最近的点。这 4 个点构成的多边形作为初始凸包。

②对于每个凸包上的点 i ,设它的后续点为 j ,计算矢量线段 ij 右侧的所有点到 ij 的距

离, 求出距离最大的点 k 。

③将 k 插入 i j 之间, 并将 k 赋给 j 。

④重复②、③, 直到点集中没有在线段 ij 右侧的点为止。

⑤将 j 赋给 i ; 取其后续点, 重复②、③、④。

⑥当凸包中任意相邻两点连线的右侧不存在离散点时, 结束离散点集凸包求取过程。

完成这一步后, 即形成了包含所有离散点的凸包(多边形), 如图 4-16 所示。

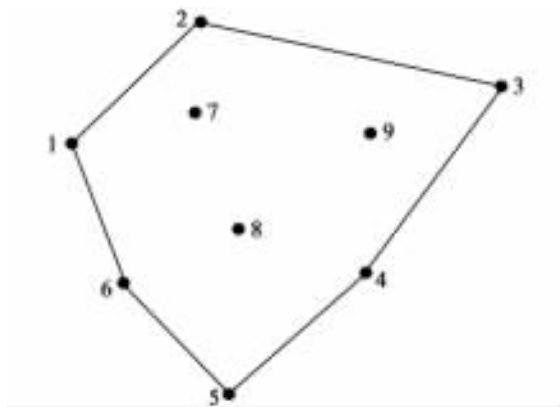


图 4-16 凸包的生成

(2) 凸包三角剖分

用环切边界法进行凸包的三角剖分。在凸包链表中每次寻找一个点, 由相邻两条凸包边组成的三角形, 在该三角形的内部和边界上都不包含凸包上的任何其他点。将这个点去掉后得到新的凸包链表。重复这个过程, 直到凸包链表中只剩 3 个离散点为止。将凸包链表中的最后 3 个离散点构成一个三角形, 结束凸包三角剖分过程。

完成这一步后, 凸包中的点构成了若干 Delaunay 三角形, 如图 4-17 所示。

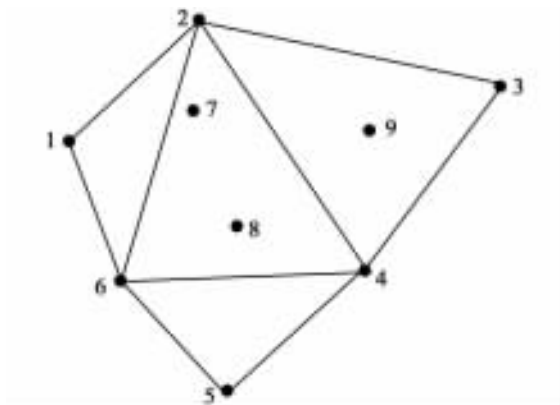


图 4-17 凸包的三角剖分

(3) 离散点内插

对不在凸包上的其余离散点,采用逐点内插的方法进行剖分。基本步骤为:

- ①找出外接圆包含待插入点的所有三角形,构成插入区域。
- ②删除插入区域内的三角形公共边,形成由影响三角形顶点构成的多边形。
- ③将插入点与多边形所有顶点相连,构成新的 Delaunay 三角形。
- ④重复①、②、③,直到所有非凸包离散点都插入为止。完成这一步后,就完成了 Delaunay 三角网的构建,如图 4-18 所示。

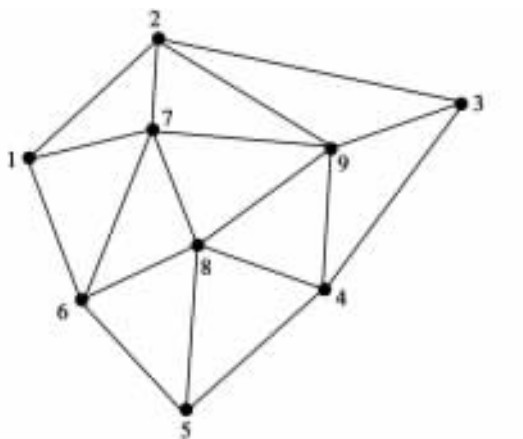


图 4-18 Delaunay 三角网

不规则三角网 DTM 的优点是:能较好地表示复杂地形;可根据不同地形,选取合适的采样点数;应用也很方便。其缺点是:数据结构复杂;不便于规范化管理;难以与矢量数据和栅格数据进行联合分析。

3. 等值线

等值线是地图上一种特殊的面状符号,也是表示 DTM 的最常用方法。

等值线的绘制方法有很多,大致可分为规则网格法和不规则网格法两类。主要过程为在上述构建网格的基础上,寻找各网格边上的等值点,跟踪等值点及光滑连接等值点即可。

等值线的优点是:直观形象,极易与其他地图符号配合使用。其缺点是:不便于坡度计算等地形分析工作,也不适用于制作晕渲图、立体图等。

4.2 栅格数据处理

栅格数据是几何图形(图像)数据的一种重要表现形式,在不同的领域中得到了广泛的应用,已发展出了多种多样的处理方法。下面介绍一些在计算机地图制图系统中常用的栅格数据处理方法。

4.2.1 栅格数据运算

1. 灰度级变换

灰度级变换即用某种方法改变图像的灰度,以提高图像的质量,从而改善图像的效果,以更适应人眼的观察或计算机的处理。

设原栅格图像中像元 (x, y) 处的灰度级为 $g(x, y)$,通过传递函数 T ,生成的新像元灰度级为 $g'(x, y)$ 。即:

$$g'(x, y) = T[g(x, y)]$$

根据不同的传递函数,可得到不同的变换(如图4-19所示)。其中,较为常用的有线性灰度级变换、非线性灰度级变换、切片变换及二值变换等。

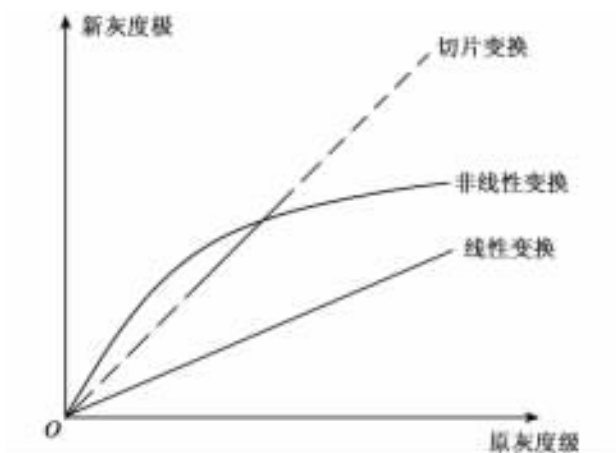


图4-19 灰度级变换

(1) 线性灰度级变换

当原栅格图像的灰度级对比度不理想时,将对比度较差的栅格图像的灰度级进行线性扩展,常常能显著改善图像的观察效果。设原栅格图像 $g(x, y)$ 的灰度级范围为 $[a, b]$,变换后的图像 $g'(x, y)$ 的灰度级范围为 $[c, d]$,则线性灰度级变换公式为:

$$g'(x, y) = (d - c)[g(x, y) - a] / (b - a) + c$$

(2) 非线性灰度级变换

当需要对图像的不同灰度区域进行不同程度的扩展时,可用某些非线性函数(如対数、指数函数等)作为传递函数,以实现图像灰度级的非线性变换。

例如,当希望对图像的低灰度区进行较大的扩展,而对高灰度区进行压缩时,可采用对数变换,其公式为:

$$g'(x, y) = a + \ln[g(x, y) + 1] / (\ln c)$$

式中 a, b, c 为可调参数。

当希望对图像的高灰度区进行较大的扩展时,可采用指数变换,其公式为:

$$g'(x, y) = b^{[g(x, y) - a] - 1}$$

(3)切片变换

有时需要将图像中的某一段灰度范围抽取出来 ,而其他范围的灰度值则变为零 ,以产生“切片”的效果。这时可使用切片变换：

$$g'(x,y)=\begin{cases} g(x,y) & a < g(x,y) < b \\ 0 & \text{其他} \end{cases}$$

(4)二值变换

在栅格图像中 ,二值图像占有非常重要的地位。一般在对地图、文字进行扫描数字化和识别时 ,都将扫描的结果当做二值图像来处理。在二值图像中 ,通常 1 表示图形 ,0 表示背景。

二值变换是为了从图像中分离出所需的对象 ,即把图形和背景作为二值图像对待。图像二值变换可以采用阈值处理方法：

$$g'(x,y)=\begin{cases} 1 & g(x,y) \geq t \\ 0 & g(x,y) < t \end{cases}$$

式中 t 表示阈值。

2. 栅格数据的组合

(1)栅格数据的算术组合

栅格数据的算术组合是指将不同的栅格图像相互叠置 ,对它们相应像元的灰度值进行各种算术运算 ,如相加、相减和相乘等。

例如 ,在图 4-20 中 ,图(a)和(b)为原图 ,图(c)则为两原图相加的结果。

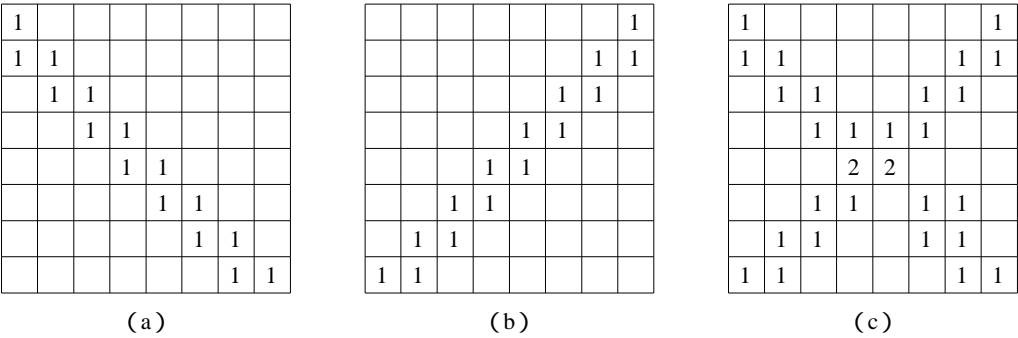


图 4-20 栅格数据的算术组合

(2)栅格数据的逻辑组合

栅格数据的逻辑组合是指将不同的栅格图像相互叠置 ,对它们相应像元的灰度值进行各种逻辑运算 ,如逻辑“或”、逻辑“与”、逻辑“非”等。

在图 4-21 中 ,(a)和(b)为原图 ,图(c)为两原图逻辑“或”的结果 ,图(d)和图(e)分别为两原图进行逻辑“与”和逻辑“非”的结果。

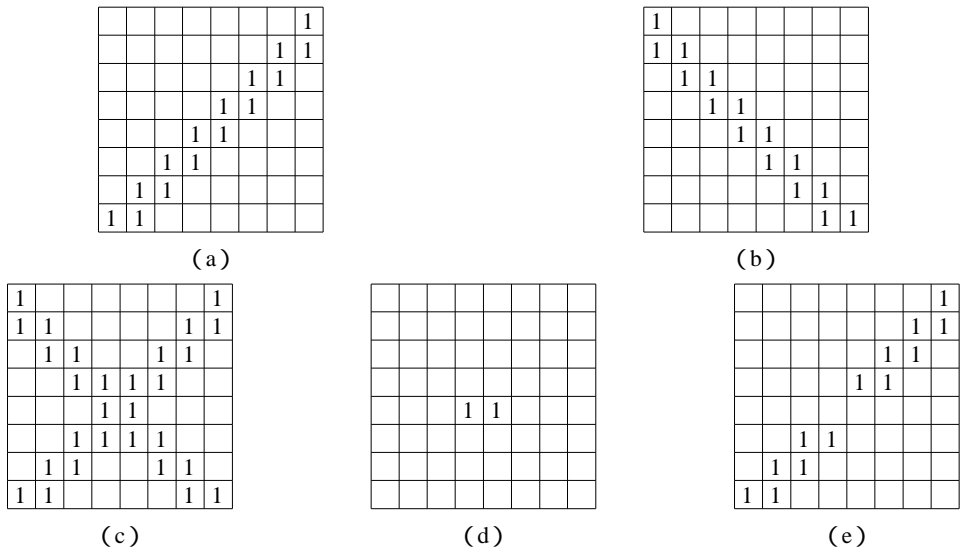


图 4-21 栅格数据的逻辑组合

3. 栅格数据的扩张与侵蚀

(1) 栅格数据的扩张

所谓栅格数据的扩张,是指将栅格数据中同一种属性的物体按事先指定的方向和给定的像元数目进行扩张。

图 4-22 显示了原图向右扩张一个像元的方法。其中:(a)为原图;(b)为原图向右平移一个像元;(c)为(a)与(b)的逻辑“或”,即为原图向右扩张一个像元的结果。类似地也可有向左、向上、向下扩张一个像元或两个像元等方法。

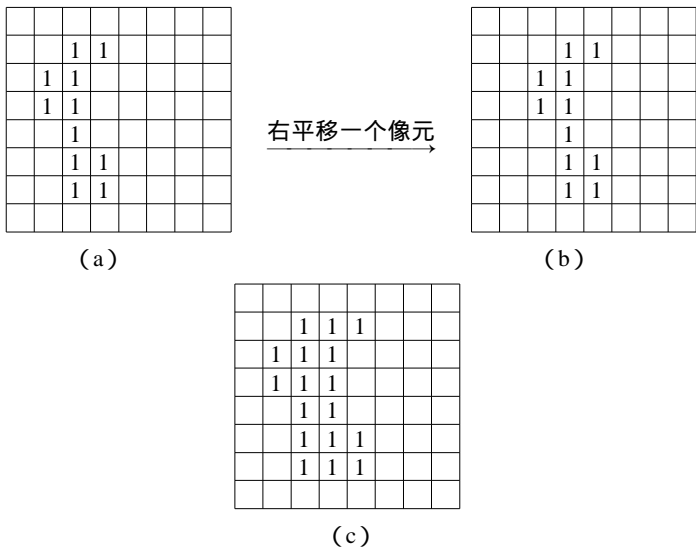


图 4-22 栅格数据的扩张

(2)栅格数据的侵蚀

侵蚀是指在栅格数据中同一种属性的物体在事先给定的像元数目和指定的方向上受到(背景像元)的侵蚀。侵蚀也可看成是背景像元在相反方向上的扩张。

图 4-23 显示了原图左侧被侵蚀一列的情况。

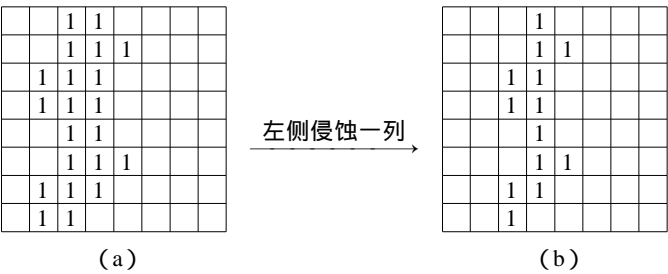


图 4-23 栅格数据的侵蚀

4. 栅格数据的加粗与减细

(1)栅格数据的加粗

栅格数据的加粗是指将栅格数据中同一种属性的物体按事先给定的像元数目进行加粗运算。

加粗可分为“四方向”加粗或“八方向”加粗。图 4-24 表示的是对原图进行四方向加粗一个像元的方法。与此类似,也可作出加粗两个像元的情况以及八方向的加粗等。

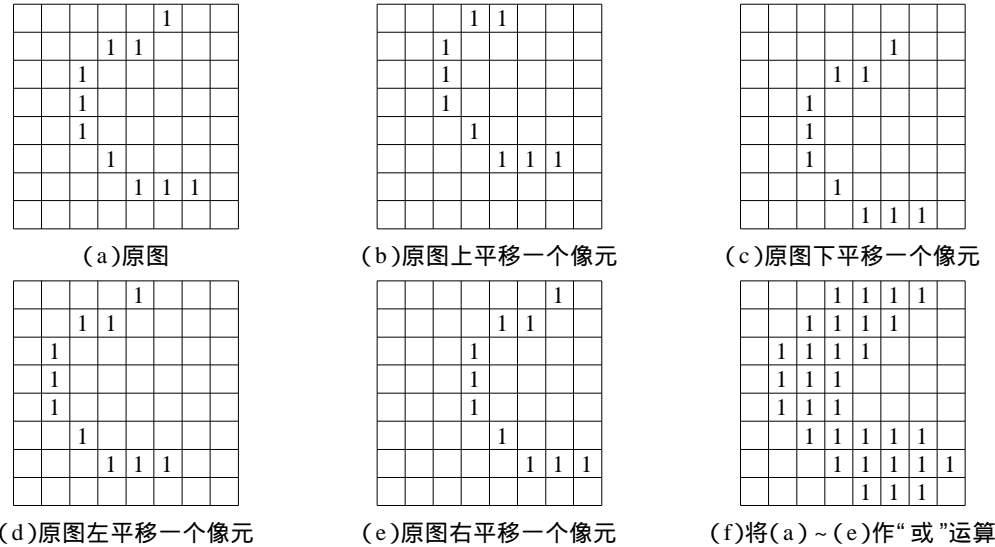


图 4-24 栅格数据的加粗

(2)栅格数据的减细

栅格数据的减细是指将栅格数据中同一种属性的物体按事先给定的像元数目进行

减细运算。减细可以看成是加粗的逆运算,即可看成是背景像元在相应方向上的加粗。

在减细运算中应特别注意减细的像元数目,减细不应导致要素的消失以及连续物体的断裂等。

5. 栅格数据的填充

栅格数据的填充是指在给定的区域范围内,使得一些单个的像元(称为“种子”或“填充胚”)通过某种算法而蔓延,直至充满整个区域范围。填充的算法有多种,常用的有逐步蔓延加粗法和逐行填充法。

(1) 逐步蔓延加粗法

逐步蔓延加粗法的思路如图4-25所示。图(a)中,灰度值为“2”的像元为种子。图(b)为对种子进行一次四方向的加粗,在加粗时要注意种子的4个方向不应覆盖范围线和其他灰度值为“2”的像元,并将新生成的灰度值为“2”的像元作为新的种子放入一个栈。图(c)表示从栈中弹出一个种子,进行一次四方向的加粗,并将新生成的灰度值为“2”的像元作为新的种子放入栈。如此循环,直到生成图(d)。

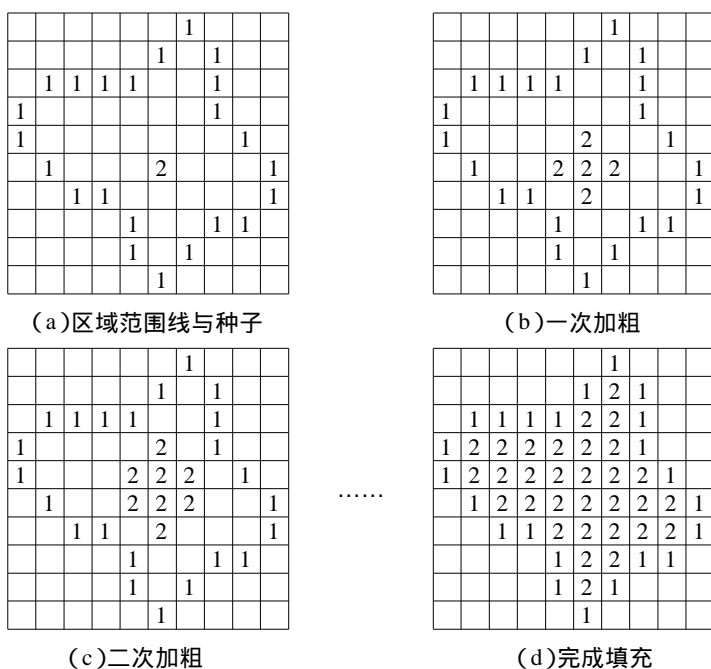


图4-25 逐步蔓延加粗法

(2) 逐行填充法

逐行填充法与逐步蔓延加粗法的思想类似,但不是对种子进行四方向的加粗,而是对种子所在的行进行填充,其基本过程如图4-26所示。其中,在每次填充一行后,应搜索该行的上下两侧以确定新的种子,对于相互连通的一侧,只需一个种子,并将其放入栈中。

						1			
					1		1		
	1	1	1	1			1		
1							1		
1								1	
	1				2				1
		1							1
			1	1			1	1	
				1		1			
					1				

(a)区域范围线与种子

						1			
					1		1		
	1	1	1	1			1		
1							1		
1	2							1	
	1	2	2	2	2	2	2	2	1
		1	2						1
			1	1			1	1	
				1		1			
					1				

(b)一次填充

						1			
					1		1		
	1	1	1	1			1		
1	2						1		
1	2	2	2	2	2	2	2	1	
	1	2	2	2	2	2	2	2	1
		1	2						1
			1	1			1	1	
				1		1			
					1				

(c)二次填充

.....

						1			
					1	2	1		
	1	1	1	1	2	2	1		
1	2	2	2	2	2	2	1		
1	2	2	2	2	2	2	2	1	
	1	2	2	2	2	2	2	2	1
		1	2	2	2	2	2	2	1
			1	1	2	2	1	1	
				1	2	1			
					1				

(d)完成填充

图 4-26 逐行填充法

4.2.2 栅格数据处理应用

栅格数据处理在计算机地图制图中得到了日益广泛的应用,现举例介绍如下。

1. 栅格数据的修涂

对于扫描输入或经过二值变换的地图栅格图像数据,由于原稿不干净或质量较差,常常会出现毛刺、脏点、线画边缘凸凹不平等现象。可以通过对栅格数据的修涂,去掉毛刺、脏点,光滑线画的边缘,填补面状目标内的小孔等,以提高栅格图像数据的质量。

一般的修涂处理方法是,采用 $n \times n$ 的辅助矩阵(n 为奇数,一般取 3 或 5)为模板,逐行、逐列与二值栅格图像匹配,根据辅助矩阵中 0、1 像元的分布,使处于矩阵中心的像元从“0”变成“1”,或从“1”变为“0”。

(1)去毛刺

0	0	0
1	0	1
*	*	*

图 4-27 去毛刺模板

去毛刺通常采用 3×3 的像元矩阵模板(如图 4-27 所示),包括其三次旋转 90° 形成的矩阵模板。当模板在二值图像上移动时,只要二值图像与模板匹配,则把模板中心的“1”变为“0”。

(2)去脏点

脏点(污点)是指其周围的像元都为“0”,而其本身的大小又可被一个 $n \times n$ 的像元矩阵所覆盖。去脏点的方法是建立 $n \times n$ 的像元矩阵模板,使其在图像上移动。移动时计算矩阵周边像元值的和,若其和为 0,则将矩阵中心

$(n-2) \times (n-2)$ 区域的像元值赋为“0”。

(3) 线画光滑与填孔

其方法是采用图 4-28 所示的 3×3 的像元矩阵模板, 包括其三次旋转 90° 形成的矩阵模板。当矩阵模板在二值图像上移动时, 只要二值图像与模板匹配, 则把模板中心的“0”变为“1”。

*	1	*
1	0	1
*	*	*

图 4-28 光滑与填孔模板

2. 栅格数据的边缘特征提取

边缘特征提取是计算机地图制图中栅格数据处理常用的技术, 可以用来对地图扫描数字化后的数据进行面状要素的提取。

边缘特征提取的过程实际上是寻找像元灰度值急剧变化的位置的过程。对二值图像而言, 在这些位置上将像元值置为“1”, 其余位置上的像元值置为“0”, 从而求出目标的边界线。可用如 Prewitt、Sobel 及拉普拉斯等多种数学算子来实现边缘特征提取。

在边缘特征提取中, 使用各种数学算子的区别实际上只是计算差分的方法不同。这些算子都是以一个 3×3 的像元矩阵模板与图像中 3×3 的区域相乘, 得到的结果作为图像中这个区域中心位置的边缘强度。在计算出图像中每一个像元的边缘强度后, 将边缘强度大于一定值的点提取出来, 并赋予像元值“1”, 其余赋予像元值“0”。

设 $g(i, j)$ 是 (i, j) 处的像元值, (i, j) 位置处的边缘强度通常用差分值来表示。常用的差分算法有:

x 方向差分值 $\Delta x_{g(i, j)} = g(i, j) - g(i, j-1)$;

y 方向差分值 $\Delta y_{g(i, j)} = g(i, j) - g(i-1, j)$;

边缘强度值为: $|\Delta x_{g(i, j)}| + |\Delta y_{g(i, j)}|$ 或 $\Delta x_{g(i, j)}^2 + \Delta y_{g(i, j)}^2$ 等。

下面简要介绍几种常用的算子。

(1) Prewitt 算子

计算 Δx_g 和 Δy_g 的模板分别如图 4-29(a)、图 4-29(b) 所示。

1	0	-1
1	0	-1
1	0	-1

(a)

1	1	1
0	0	0
-1	-1	-1

(b)

图 4-29 Prewitt 算子

Δx_g , Δy_g 分别等于相应模板与图像中对应区域元素相乘之和。即:

$$\Delta x_g = [g(i-1, j-1) - g(i-1, j+1)] + [g(i, j-1) - g(i, j+1)] + [g(i+1, j-1) - g(i+1, j+1)]$$

$$\Delta y_g = [g(i-1, j-1) - g(i+1, j-1)] + [g(i-1, j) - g(i+1, j)] + [g(i-1, j+1) - g(i+1, j+1)]$$

边缘强度为 $|\Delta x_g| + |\Delta y_g|$ 或 $\max(|\Delta x_g|, |\Delta y_g|)$ 。

(2) Sobel 算子

计算 Δx_g 的模板如图 4-30(a)所示 ,计算 Δy_g 的模板如图 4-30(b)所示。

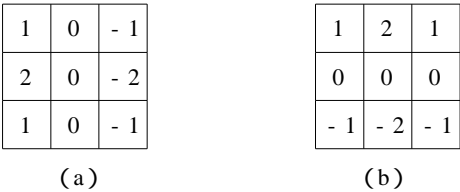


图 4-30 Sobel 算子

(3)拉普拉斯算子

计算 Δx_g 和 Δy_g 的模板如图 4-31 所示。边缘强度 $\Delta^2 g$ 等于模板与图像中对应区域元素相乘之和 ,是一个二次差分 ,具有各向同性 ,即不依赖于边缘线的方向。

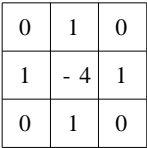


图 4-31 拉普拉斯算子

3. 栅格符号的制作

在计算机地图制图中 ,有很多类型的地图符号用栅格数据处理的方法来绘制会更简洁 ,现举例介绍如下。

(1)公路符号

地图上的公路符号通常用平行于中轴线的两条曲线表示。设图 4-32 中的图(a)为两条相交的公路的中轴线。首先将该两条中轴线加粗到公路符号的内宽(如图(b)所示) ;其次将该两条中轴线加粗到公路符号的外宽(如图(c)所示) ;然后再用图(c)“ 减去 ”图(b) ,即可得到双线公路符号(如图(d)所示)。

用类似的方法还可制作其他一些线状符号。例如图 4-32 中 ,将图(b)用虚线表示 ,再将其叠加到图(d)上 ,即可得到铁路符号。

(2)面状图案符号

地图上有很多面状符号 ,其内填充了各种图案或晕线等 ,很适合用栅格数据处理的方法来绘制。其基本方法介绍如下。

在图 4-33 中 ,设图(a)为指定的面状符号区域的轮廓范围线。图(b)是对该区域轮廓范围线内进行填充。图(c)是在该区域轮廓范围线的外接矩形内选择相应的图案(晕线、图元等)样本。图(d)则表示的是图(b)与图(c)进行逻辑“ 与 ”的结果 ,也即最终所需绘出的面状符号。

由上述绘制过程可知 ,使用不同的样本即可得到不同的面状符号 ,因此该法可绘制出任意多种面状符号。

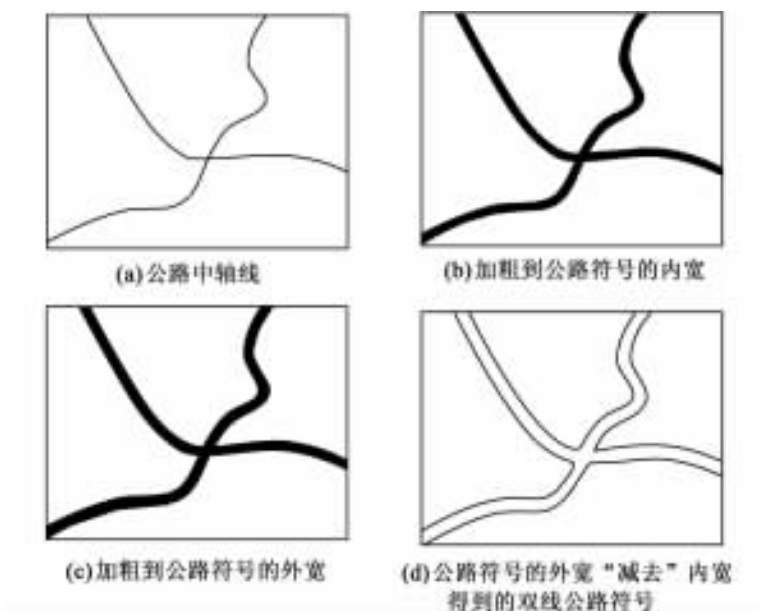


图 4-32 双线公路符号绘制示意

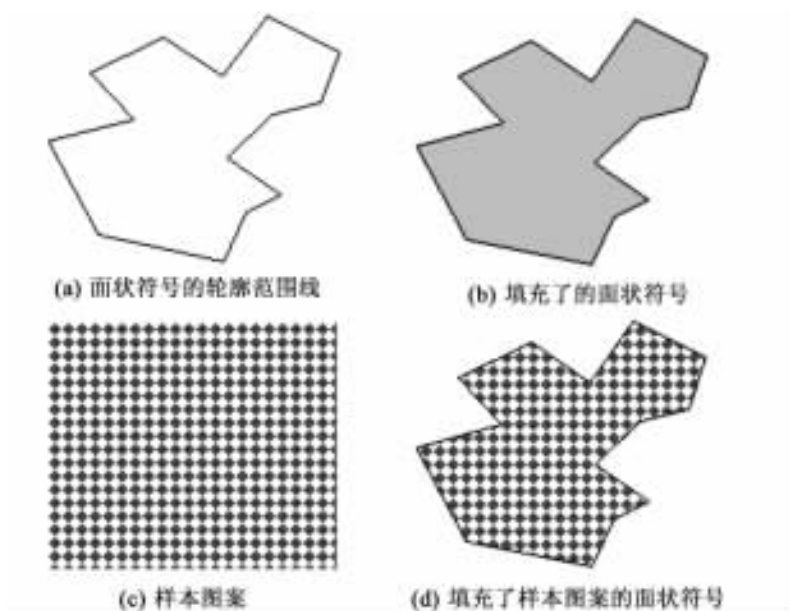


图 4-33 面状符号绘制示意

第5章 地理信息可视化

可视化就是将事物或现象变成可被视觉所感知的过程。

地理信息的可视化是指将原有的地理信息数据转化为直观的图形、图像的技术,它是一个转换过程,其目的是将原始数据转化为可显示的图形、图像,以便为人们视觉所感知。

5.1 可视化的概念与形式

5.1.1 可视化的概念

1. 可视化的概念

科学计算可视化的概念近来频繁地出现在计算机科学和地理信息科学之中,已成为一种技术潮流。其原因是人类信息 80% 以上是通过视觉来获取的,视觉在人们认识世界的过程中有着特殊地位。

可视化是指运用计算机图形学和图像处理技术,将计算过程中产生的数据及计算结果转换为图形和图像显示出来,并进行交互处理的理论、方法和技术。它的主要功能是从复杂的数据中产生图形图像,并可以分析和理解存入计算机的图形图像数据。它使许多抽象的、难以理解的原理、规律和过程变得更加容易理解。可视化涉及的领域包括计算机图形学、图像处理、计算机视觉及人机交互技术等多个方面。

2. 可视化的作用

早期,由于计算机软件、硬件技术水平的限制,科学计算只能以批处理的方式进行,而不能进行交互处理。对于很多的输出数据,只能用人工方式处理,或者用绘图仪输出简单的二维图形。这种处理方式不仅效率低下,而且丢失了大量信息。随着计算机应用的普及和科学技术的迅速发展,来自各个领域的数据量越来越大,但缺乏有效的处理和观察理解手段。再者,随着科学的发展,对传统的实验设备和实验精度的要求也越来越高,这直接导致了传统实验方法的实验费用持续增长。另一方面,由于计算机技术的高速发展,计算成本不断下降,计算精度和速度不断提高,这使得对复杂问题的数值模拟成为一种更直接、更有效的方法。而大规模数值模拟可产生上百兆、上千兆的大量数据,人们已无法用传统的方法来理解大量科学数据中包含的复杂现象和规律。因此,科学计算可视化技术已经成为科学研究中必不可少的手段。它是科学工作者以及工程技术人员洞察数据内含信息,确定内在关系与规律的有效方法。它使人们以直观形象的方式揭示、理解抽象科学数据中包含的客观规律,从而摆脱直接面对大量无法理解的抽象数据的被动局面。

科学计算可视化具有以下作用:

①实现人与机之间的图像通信,而不是目前的文字或数字通信,从而使人们观察到传统方法难以观察到的现象和规律。

②使科学家不再是被动地得到计算结果,而是知道在计算过程中发生了什么现象,并可改变参数,观察其影响,对计算过程实现引导和控制。

③可提供在计算机辅助下的可视化技术手段,从而为在分布环境下的计算机辅助协同设计打下基础。

科学计算可视化在信息世界中具有特殊地位。视觉是信息传输和接收的主要渠道,尤其对于多维信息,可视化具有独特优点。科学计算可视化的范围广泛,覆盖了原子、分子、机器、房屋甚至地球、宇宙。

地理信息可视化是科学计算可视化的一部分,其范围是地学环境空间中具有地理特性的事物,其可视化的过程具有数字化和符号化的特征。

地理信息可视化是指运用计算机地图制图技术以及计算机图形学和图像处理技术,将地理信息输入、存储、处理、查询以及输出的数据结果采用图形符号、图形、图像,结合图表、文字、表格等可视化形式显示并进行交互处理的理论、方法和技术。

5.1.2 可视化的形式

地理信息的可视化正在进入一个崭新的时期。可视化的形式多种多样,并正在不断地发展。地理信息可视化的主要形式有地图、多媒体及虚拟现实。

1. 地图

地图是地理信息可视化最主要的形式。它包括纸质地图、数字地图、电子地图及三维地图等。其中,纸质地图和数字地图仅是计算机上硬、软拷贝的差别。硬拷贝的是纸质地图,软拷贝的是数字地图。电子地图动态性强,形式多样,制作灵活。相对于纸质地图而言,数字地图及电子地图修改制作方便,成图周期短,色彩丰富,查询方便、快捷,方便用户用不同的方式和不同的详细程度来观察地理事物和现象的信息。三维地图是基于计算机三维仿真图形技术而产生的三维仿真地图,具有仿真的形状、光照、纹理等,并可以进行各种三维的分析和量测。

2. 多媒体

各种多媒体形式(如文本、表格、图形、图像、动画、音频、视频等)也是地理信息可视化的重要形式。它们能够从某些特定方面表示地理信息,是全面地表示地理信息的有力手段和补充。

3. 虚拟现实

虚拟现实是地理信息可视化进一步发展的新形式。它是由计算机和其他专用设备(如头盔、数据手套等)组成的高级人机交互系统,以视觉为主,结合触觉、听觉等来虚拟和感知现实世界,使人们犹如进入真实的地理空间环境之中并与之进行交互作用。

5.2 符 号 化

地理信息可视化的最主要形式是地图,它是地学信息的图形符号模型。构成地图的主要内容是图形符号,也可以说地图是图形符号的空间集合,因此符号化在地理信息可视化的

过程中具有重要意义。这里,符号化是指地图符号的符号化。下面分别介绍地图符号及其符号化的方法。

5.2.1 地图符号

1. 地图符号概述

(1) 地图符号的概念

地图工作者为了传递各种地理事物与现象的信息,常采用一些图形符号来作为载体,这些图形符号就是地图符号。地图符号是在地图上用来表示各种地理事物和现象的图形符号。地图符号是地图的主要表达形式,它是地图区别于其他表示地理环境方法的一个重要特征。地图符号不仅能反映地图要素的存在、类别及其数量和质量特征,而且通过它们的组合,还能反映出地图要素的空间分布、结构、联系以及变化等。

一般来说,常用的地图符号大多经过长期的考验,为人们所接受和遵守,在某种程度上具有“法定”的意义。

地图符号采用便于空间定位的形式来表示各种地理事物与现象的性质和相互关系。地图符号是一种用来传递各种自然现象和社会现象的工具,在地图上形成模拟客观实际的空间形象。它因此具有抽象的和客观的意义,并与被表示的对象有一定的关系。地图符号既能反映出地理对象种类及其数量和质量特征,又能表达对象的空间位置和现象的分布。

(2) 地图符号的类型

根据分类依据的不同,地图符号类型的划分也不同。例如,依据地图符号的大小与所表示的对象之间的比例关系可将其分为依比例尺符号、不依比例尺符号和半依比例尺符号;而按符号所表示的对象可将其分为水系符号、道路符号、居民地符号、独立地物符号、境界符号、地貌符号与植被符号等。

在计算机地图制图中,主要是依据地理事物与现象的空间分布特征来划分地图符号的类型。具体可将地图符号区分为点状符号、线状符号和面状符号。

①点状符号。当地图符号所表示的地理对象的空间分布特征在抽象意义下可认为是定位于几何上的点时,称为点状符号。此时,符号的形状大小与地图比例尺无关,并具有定位和方向的特性。例如,控制点、居民点、独立地物及不依比例尺的桥梁等点状符号。

②线状符号。当地图符号所表示的地理对象的空间分布特征在抽象意义下可认为是定位于几何上的线时,称为线状符号。此时,符号沿着线的方向延伸,长度与地图比例尺有关,而宽度与地图比例尺可以没有关系。线状符号的例子如道路、境界、河流、渠道、岸线、航线及管线、垣栅等。

③面状符号。当地图符号所表示的地理对象的空间分布特征在抽象意义下可认为是定位于几何上的面时,称为面状符号。此时,符号所代表的分布范围的大小与地图比例尺有关。面状符号用于表示各种区域范围,如森林的范围、水域的范围、土地利用类型范围、各种资源分布范围及行政区划等。

2. 地图符号的分类分级

各种地理事物与现象众多且复杂,难以在地图上逐一地用符号表示出它们的个性,因此,需要先对各种地理对象进行概括(分类、分级)和抽象,然后用抽象的、具有共性的符号表示某类事物。这种具有共性的、进行了概括和抽象的地图符号集合,构成了地图符号的分

类分级体系。

这里,地图符号的分类分级体系与数据输入时的分类分级相一致。利用地图符号的分类分级体系,解决了逐一描绘地理对象个体比较困难这一问题,而且还能反映群体特征和分布规律。单个地图符号只具备有限的功能,而由符号的集合构成的地图符号体系还能表达地理对象的空间组合和联系,能给出单个符号所不能提供的信息。

例如,用单个灌木林符号表示小面积灌木林,而由单个灌木林符号排列成带状,以表示灌木林带;由单个灌木林符号散列成面状,以表示大面积灌木林。这样,小面积灌木林、灌木林带、大面积灌木林就构成了灌木林符号系列。

地图符号的分类分级体系可以很好地反映所表达现象的层次关系,顾及现象分类分级的可能性。例如,表示行政区划的境界,可以依次被划分为几个层次:第一层次是国界,第二层次是省(自治区、直辖市);第三层次是地区(市、州、盟);第四层次是县(区、市、旗)等。将这些层次依次用相互联系而又相互区别的地图符号表示,则构成了境界符号系列。

3. 地图符号设计的原理

地图符号的质量直接影响地图信息的传递效果。设计地图符号,应在充分考虑地图各要素的分类分级要求的基础上,着重顾及构成地图符号的6个图形变量,即形状、尺寸、色彩、亮度、密度和方向。其中,以符号的形状、尺寸和色彩最为重要,它们被称为地图符号的三要素,分别讨论如下:

(1)形状

在考虑符号的图形形状时,应使设计的符号图案化和系统化,并充分考虑到计算机绘图和屏幕可视化的技术要求。

这里,符号图案化是指符号图形应类似于物体本身的实际形态,或具有象征会意的作用,以便使人们看到符号就能联想出被表示的物体或现象。符号图案化是一个概括抽象和艺术美化的过程。要突出其重要特征,舍去复杂的物体图形中的细部,采用艺术的手法,设计出规则、美观的符号图形。图案化的符号图形应具有形象、简单、便于准确定位等特点。

系统化是指在设计地图符号图形时,应顾及地图内容的分类分级,并考虑到符号图形与符号含义内在的、有机的联系,不能孤立、片面地进行单个符号设计。

(2)尺寸

符号的尺寸大小应与地图用途、比例尺、制图区域特点和屏幕分辨率高低等方面相联系。设计符号的尺寸时,还应充分注意分辨能力、绘图和复制技术能力。一般而言,在清楚显示符号结构的情况下,尺寸应尽量小。若分辨率高,符号尺寸可大一些,结构也可复杂些;反之亦然。

(3)色彩

色彩的使用可以简化地图符号的图形差别,减少符号的数量,增强各要素分类分级之间的联系,美化地图,提高地图的表现力。

设计地图符号的色彩时,要正确利用色彩的象征意义。例如,可用绿色符号表示植被要素,以反映植被的自然色彩;以蓝色符号表示水系等。

对与地图表示的主题或主要现象有关的符号,应使用鲜明、饱和的色彩;而对于基础和次要现象的符号,则宜用浅淡的色彩。通过色彩对比,起到突出主题或主要现象的作用。

5.2.2 符号化的方法

所谓符号化,是指地图符号的符号化过程。所谓符号化过程,就是利用地图数据库得到有关地图要素的分级分类编码及相应符号和要素实体抽象后得到的定位坐标数据,以形成地图空间内的图形符号模型的过程。也可以这样认为,符号化是由对象的定性、定位及时间数据决定的,是将符号从符号空间到地图空间的一个变换过程。

符号化与数字化有紧密联系。数字化是将地图要素或相应图形符号抽象为地图数据库的定性、定位及时间数据,而符号化是将地图数据库内的定性、定位及时间数据变换为图形符号,是数字化的逆过程。

根据所使用绘图数据格式的不同,符号化的方法有矢量符号化和栅格符号化两种。具体又可分为直接信息法和间接信息法两种。

(1)直接信息法

这种方法直接表示符号图形的各个细部,在信息块中直接存储符号图形的矢量数据(图形特征点坐标)或栅格数据(点阵数据)。这种方法信息块占用存储空间较大,但因为该方法只面向图形点,而与符号图形的结构无关,当修改信息块内容时,不必改动绘图程序。也就是用同一绘图程序能绘出不同信息块所表示的各种符号图形,因此有可能使绘图程序统一算法,从而大大减少编程工作量。

(2)间接信息法

该方法的信息块中不直接存储符号图形数据,而只存储符号图形的几何参数(如长、宽、夹角、半径等),绘图时所需的符号图形数据由计算机按相应绘图程序的算法解算出来。该方法信息块占用存储空间较小,但程序量大,图形差异大的符号都需各自编写绘图程序。

上述两种方法各有优缺点。一般而言,在用计算机制作普通地图时,采用直接信息法的较多,而在制作专题地图或绘制比较特殊的符号时,可采用间接信息法。

5.3 地图符号库

计算机地图制图系统中的几何数据和属性数据须转换成各种地图符号,才能完成由数据向图形的转换(即可视化)。系统中的地图空间数据的检索功能只有与地图符号库有机地结合起来,才能使空间数据符号化。

5.3.1 符号库设计的原则

通常以信息块为地图符号库中的基本单位,每个信息块中包含着一个与符号有关的信息(如图形信息、颜色码等)。地图符号库的一般设计原则可归纳如下:

(1)应尽量参照现有的地图图式

在国家基本比例尺地图符号库中,符号信息块表示的图形、颜色、含义及适用的比例尺等,应与国家规定的地图图式一致。个别不适合计算机绘制的符号图形,在经主管部门同意后可作适当修改。

(2)在新设计地图符号时应遵循图案化、精确性、对比性、统一性、象征性及自动绘制适应性等一般原则。

(3)选择适当的符号信息块结构

构成符号信息块的主要方法有直接信息法和间接信息法两种 ,应根据具体情况选择适当的符号信息块结构。

(4)符号库的可扩充性

允许用户自行修改任意一种符号图形 ,删除过时的符号 ,增加新设计的符号。

根据所使用计算机地图制图系统中绘图数据格式的不同 ,地图符号库的设计又可分为矢量符号库和栅格符号库两类。

5.3.2 矢量符号库

大多数点、线、面状的地图符号都比较容易用矢量形式的坐标表示 ,由平面内这些点的坐标和绘图机抬落笔动作编码组合的有序集合便能产生各种点状符号、线状符号和面状符号。

下面以直接信息法为例 ,按点、线、面三类分别介绍矢量符号库中信息块的结构。

1. 点状符号信息块

点状符号信息块中记录图形的颜色码、图形特征点坐标及绘图机抬落笔编码。通常 ,信息块采用以符号定位点为原点的局部坐标系。

以图 5-1 为例 ,这是一个水塔符号的放大表示 ,可按图中点号顺序在信息块中记录(p_i , x_i , y_i)。其中 $i=1 \ 2 \ \dots \ 13$, p_i 为点 i 的抬落笔编码 ;(x_i , y_i)为点在局部坐标系中的坐标值。

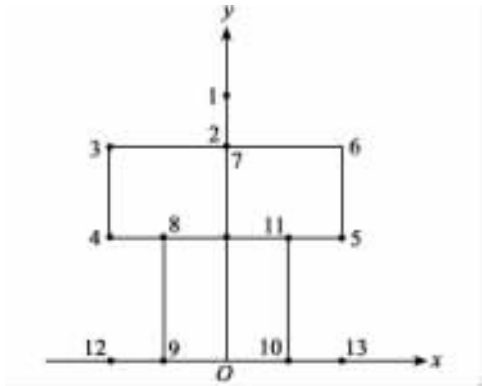


图 5-1 点状符号的局部坐标系及定位点

点状符号信息块的一般结构形式如图 5-2 所示。

符号定位点数 n	$(p_1 \ x_1 \ y_1)$...	$(p_n \ x_n \ y_n)$
------------	---------------------	-----	---------------------

图 5-2 点状符号信息块结构

绘图时 ,读出信息块 ,按地图数据指定的位置和方向对信息块中坐标数据先后进行旋转和平移 ,然后调用基本绘图语句直接连线绘出即可。

2. 线状符号信息块

地图上各种线状符号通常可由某一图元(即线状符号的基本单元)沿线状要素的中轴线串接而成。图元坐标系的 x 轴与线状要素中轴线重合,在转弯处也随着弯曲变形。图元坐标系的原点在图元的首端。

图 5-3 是地形图中栅栏符号图元放大表示的例子。



图 5-3 线状符号图元的局部坐标系及定位点

线状符号信息块的一般结构如图 5-4 所示。

图元定位点数 n	图元长度	$(p_1 \ x_1 \ y_1)$...	$(p_n \ x_n \ y_n)$
------------	------	---------------------	-----	---------------------

图 5-4 线状符号信息块结构

3. 面状符号信息块

地图上面状符号通常由填充符号在面域内按一定方式配置组合而成。多数情况下,填充符号在面域内是按一定方向、间隔逐行配置的。晕线是面状符号的形式之一。其他以呈一定规律分布的个体符号形式表现的面状符号也可像计算晕线端点那样事先算出各行与轮廓边的交点,然后在每对交点间配置相应的填充符号,其过程相当于在每对交点间的连线上绘制线状符号(如图 5-5 所示)。

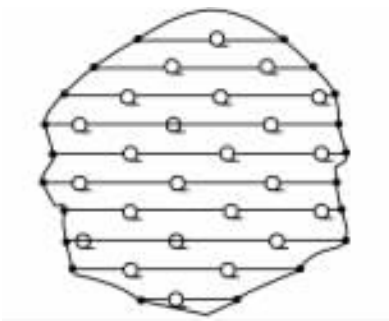


图 5-5 面状符号

在面状符号信息块中,存储的是填充符号的图元信息。其结构类似于线状符号信息块,但还应增加行距、行向倾角及排列方式三种信息。这里,行向倾角指行方向与 x 轴的夹角,有单向和双向之分(如两组相交的晕线)。排列方式常用的有井字形、品字形及散列式等,可在信息块中用不同代码表示。

图 5-6 显示了面状符号信息块的结构。

图元点数 n	图元长度	排列方式	倾角 1	倾角 2	行距	(p_1, x_1, y_1)	...	(p_n, x_n, y_n)
----------	------	------	------	------	----	-------------------	-----	-------------------

图 5-6 面状符号信息块结构

如果填充符号是单向排列的,只需记录行向倾角 1,倾角 2 空缺。当有多种填充符号混合表示时,应按先主后次的顺序填绘各类符号,这时有关的算法应避免符号的重叠。

5.3.3 栅格符号库

栅格符号库用于栅格方式的图形输出,也可按点、线、面三类建库。其中点状符号信息块和线状符号信息块可由矢量符号信息块转换得到,或对符号的标准样式直接进行扫描获得。在栅格符号库中,点状、线状两种信息块内栅格坐标系的确定应便于符号的定位。

栅格符号库中面状信息块的构成不同于矢量符号库中的面状信息块。地图上规则分布的面状符号,在平面上总可以划分成等大的图案块,每个图案块的图形相同。因此面状符号可由这样的图案块在区域内拼接而成,在轮廓边处要裁去超出轮廓的部分。一个图案块的点阵数据就组成面状符号的一个信息块。

图 5-7 是用点阵表示符号图案块的示意图,在面状符号信息块中记录相应的栅格数据。



图 5-7 面状符号及图案块栅格数据示意图

5.3.4 符号库的引用

引用符号库中的符号信息来自动绘图的一般过程是:从地图数据库中检索出地图要素的特征代码以及定位信息,再从符号库中查得相应符号信息块的入口地址,在符号库中读取信息块,然后调用相关的绘图子程序(子函数)完成绘图。

地图数据库与符号库可分别进行设计,它们之间可通过一个对照文件建立联系。其中,地图要素的特征代码与符号信息块地址的对照关系是在建立符号信息块的同时确定并记入

对照文件的。对于算法相同的各信息块调用同一绘图子程序(子函数)。

根据符号信息块结构上的不同,用符号信息块绘制符号的方法分为间接信息法与直接信息法两种。

1. 间接信息法

用间接信息法建立的符号信息块,存储的图元数据主要是一组表示符号形状的类型和几何参数(参见本书第6章)。在绘图时,需用相应的绘图程序将它们转换为矢量数据(包括图元点坐标和抬落笔信息)或栅格数据,然后按给定顺序连接各图元点或输出栅格点阵,绘成符号。

间接信息法绘制地图符号的一般过程如图5-8所示。

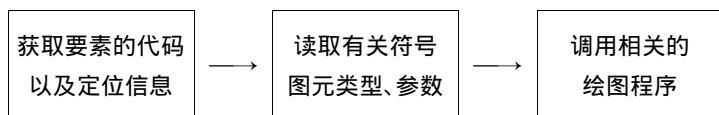


图 5-8 间接信息法绘制地图符号的一般过程

2. 直接信息法

直接信息法直接从符号信息块中获取绘制符号图形的数据,省略了由几何参数到符号图形的数据的转换。

与间接信息法相比,直接信息法的算法比较统一,与此相关的绘图程序通用性强,能尽可能独立于符号图形。

(1) 用矢量符号信息块绘制符号

用矢量符号信息块绘制符号的过程实质上是符号图元坐标系中的特征图元点坐标 (x, y) 变换成地图坐标系中的 (x, y) ,并按给定顺序连线(如图5-9所示)。

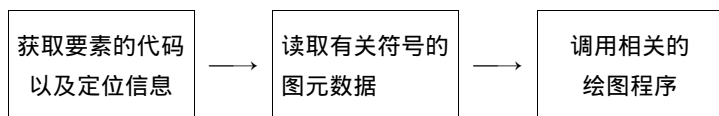


图 5-9 直接信息法绘制地图符号的一般过程

就一般点状符号而言,上述坐标变换就是一个平移变换,而对于有方向性的点状符号则需先旋转,再平移。

线状符号的绘制相对要复杂一些。图元要在线状要素中轴线上分段串接并在拐弯处作变形处理。当中轴线为曲线时,经光滑插值给出的仍可看做是由许多短直线组成的折线。图元坐标系的 x 轴与中轴线重合且指向画线方向,按图元长度在中轴线上分段截取,若图元超出前方拐弯点,则将图元截去超出部分,截去部分转至下一折线段内处理。在拐弯处图元要作变形处理,使图元随中轴线弯曲,不出现裂隙或交叠现象。

下面以双线路符号为例,进一步说明绘制线状符号的过程。

在图5-10(a)中,用1、2、3、4四点坐标及抬落笔编码表示其图元点,图元长 L 为一正

数。绘制双线路符号的过程如图 5-10(b)所示。

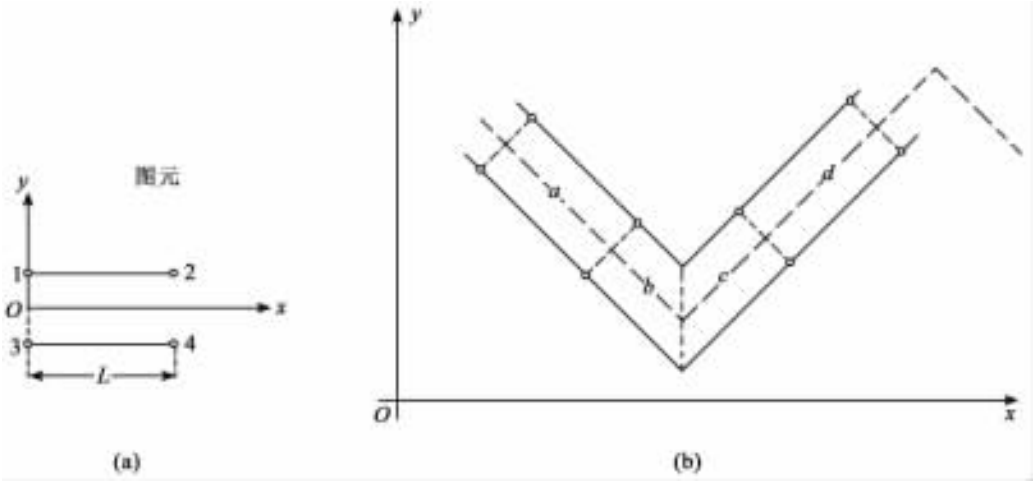


图 5-10 双线路符号绘制过程示意

在图 5-10(b)中,图元 a 是完整的,没有变形,在中轴线拐弯的地方,图元被分割,b 是分割后图元的前段,c 是该图元的后段,b 和 c 在拐弯处都要作变形处理,使之能正确衔接,d 也是完整的图元。重复上述过程,便可将图元串接至终点。

线状符号绘制过程的算法是直接面向各图形特征点的,因此绘制过程便是图元点的变换和连线过程。

有关线状符号绘制中图元点的变换算法说明如下:

设线状要素中轴线为 i j k l(如图 5-11 所示),地图坐标系为 xOy,图元所在局部坐标系为 x'O'y'(局部坐标系原点 O'与节点 i 重合),图元范围为 abcd。因图元长超出前方节点 k,故将图元切成两段 abfe 和 efcd。断面线 ef 过 k 垂直于 x' 轴。两条角平分线和断面线将图元划分成三个点集,需作不同处理。

①在局部坐标系 x'O'y'中,若图元点 $p(x_p, y_p)$ 满足 $0 < x_p < x_k$,且 p 点在两角平分线之间,则该图元点不作移位处理,只按一般方法从局部坐标系变换至地图坐标系即可。

②在局部坐标系 x'O'y'中,若图元点 $p(x_p, y_p)$ 满足以下条件:

$x_p = 0$,

或 $x_p = x_k$,

或 $0 < x_p < x_k$ 且 p 点不在两角平分线之间,

则将该图元点沿 x' 轴方向(正方向或反方向)平移至附近的角平分线上,然后变换成地图坐标系。属于这种情况的有图 5-11 中 $\triangle ajg$, $\triangle kfh$ 内的点和线段 jb ke 上的点。

③图元超出前方节点 k 的部分,即四边形 efcd 内的图元点 $p(x_p, y_p)$ 。若 p 点在局部坐标系中满足 $x_p \geq x_k$,则将其转至中轴线的下节处理,在节点 k 处建立新的局部坐标系,处理方法同上。这段处理完后,便再串接一个完整的图元,仍用上述方法进行判别和处理。如此反复进行串接,直至该线状符号全部绘出。

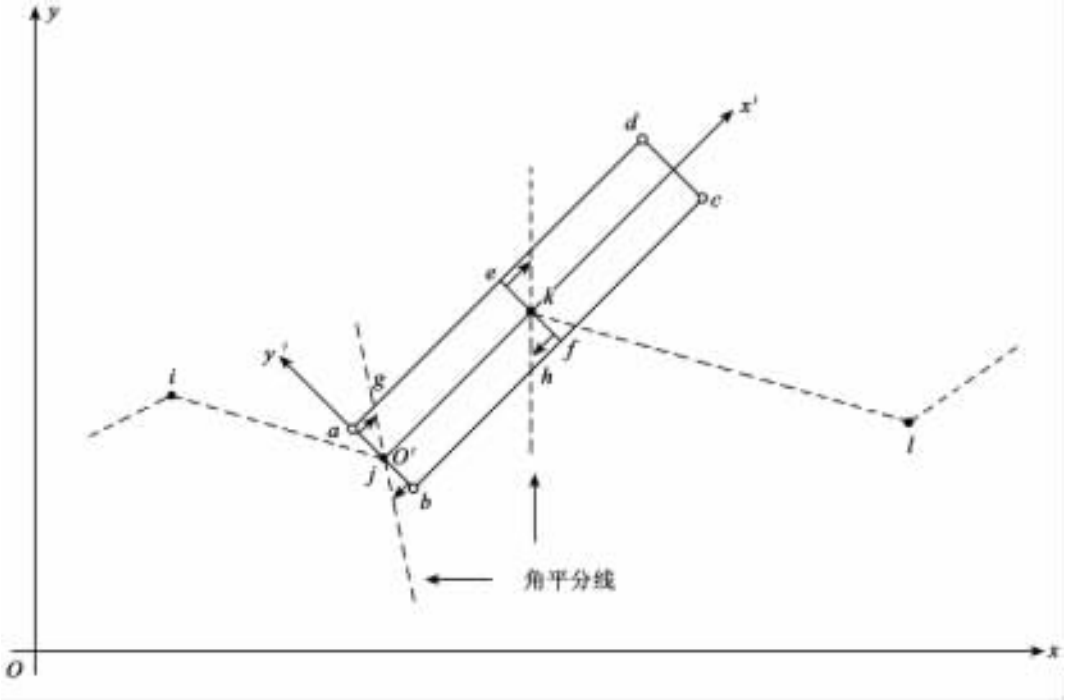


图 5-11 线状符号绘制中图元点的变换

另外还应注意的 在线状符号中有些图形元素是不能变形的 在转弯处只能变向而不能改变形状(如境界符号中的圆点等)。这时 需要调整图元长度 以便在中轴线上均匀分布图元段 且使线状要素端点或节点处保持实部。在绘图程序中应作相应的判别并给出有关的算法。

面状符号绘制时首先要计算晕线端点(见图 5-5) 然后逐行配置填充符号。其算法和线状符号绘制的算法相似 不需作弯曲移位处理 但需考虑填充符号的排列方式 并合理确定各行起点对应图元的位置。

(2)用栅格符号信息块绘制符号

用栅格符号信息块绘制符号的算法较为简单。

一般点状符号的绘制用平移即可 对于有向点状符号也是先旋转后平移至指定位置输出。

线状符号绘制时要对信息块作逐列处理 这是因线状符号走向多变 故不便对信息块进行整体操作。“轮移法”是一种产生线状符号的常用方法 简介如下。

首先从符号库中读出所需信息块的像元矩阵 接着从左至右逐列取出点阵信息 按中轴线走向进行旋转变换(列向与中轴线垂直) 然后平移至指定位置输出。位于图元中轴线上的像元应与线状要素中轴线重合 各列像元在中轴线上的配置间隔在直线段处为一个像元 在转弯处为了不产生裂隙 需缩小间隔(这可由绘图程序自动计算 随曲率变化而调节)。图 5-12 是境界的栅格图元及逐列输出结果的放大图形。



图 5-12 “轮移法”产生线状符号示意

在对面状符号进行绘制时,首先对轮廓内进行填充,然后将填充后的区域分块与图元点阵进行逻辑“与”运算,结果便能在轮廓内形成规则配置的面状符号。

5.4 电子地图

电子地图(Electronic Map)诞生于20世纪80年代中期,是地图学与计算机系统、信息技术等相结合而发展起来的一种新的地图形式。迄今为止,国内外已相继推出了众多的电子地图产品,如1997年美国出版的3D电子地图,1998年美国微软公司制作的虚拟地球,2001年中国地图出版社出版的世界电子地图集,以及中国、加拿大等许多国家先后发行的国家电子地图集(如图5-13所示)。电子地图越来越多地成为人们喜闻乐见的、应用方便和内容丰富的大众化地理信息产品。

5.4.1 电子地图的基本特征

目前,电子地图还没有一个统一的定义,但一般的观点认为,电子地图是以数字地图为基础,并以计算机屏幕为媒介实时显示的地图数据的可视化产品,所以有人又将电子地图称为“瞬时地图”、“屏幕地图”或“无纸地图”。电子地图不同于数字地图,是因为电子地图需要依赖于地图的表达,而数字地图则更多地倾向于地图数据的组织与存储。电子地图是数字地图在屏幕上的符号化显示。

电子地图与传统的纸质地图相比,具有以下几个特点:

(1) 数据与软件的集成性

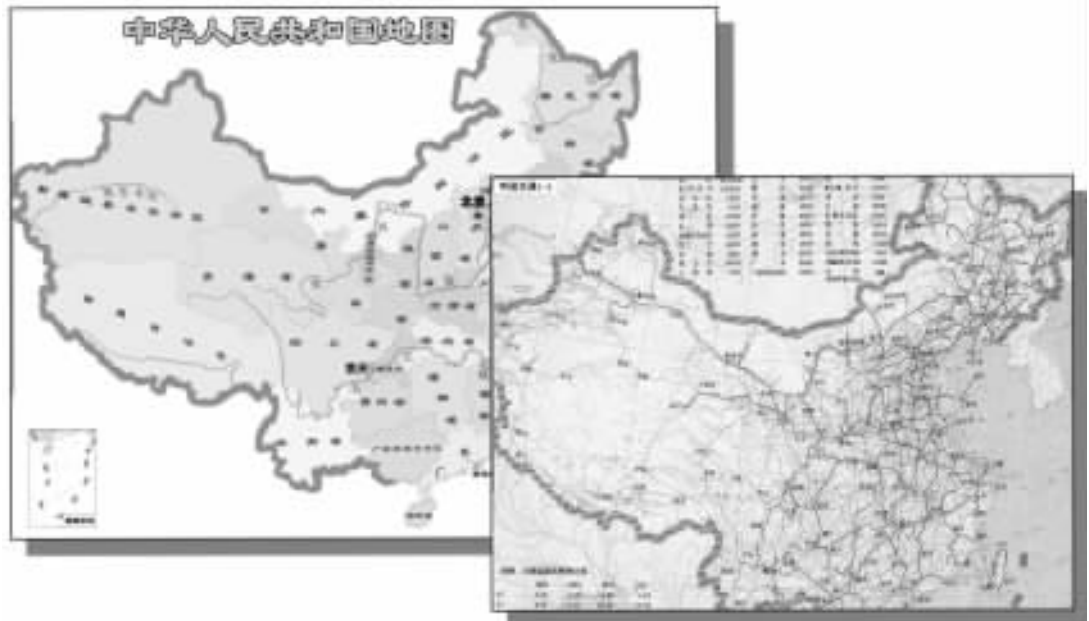
在产品形式上,纸质地图表现为单一的地图数据输出,而电子地图是地图数据与软件系统的集成,缺一不可。电子地图的软件系统又可称为电子地图浏览系统,它是地图由数据形式到模拟表达形式的翻译器和转换器,负责将地图数据库中的内容显示在屏幕上,并支持一系列对地图的操作功能。

(2) 交互性

纸质地图一旦印刷完成就成为定型产品,幅面、内容、形式都不会再发生改变。电子地图保存在计算机的存储设备中,电子地图系统的浏览软件允许用户对表达的地图内容进行选择,并通过缩放、漫游对地图表达区域进行调整,经过用户的交互操作在屏幕上形成不同的地图。



(a) 美国 Microsoft公司的 1997 世界电子地图集



(b) 中国地图出版社出版的中国电子地图集

图 5-13 几种电子地图集的界面

(3)表达的多样性

由于受到比例尺、幅面和媒介的制约 纸质地图能反映的信息量有限 ,只能通过地图符

号的结构、色彩、大小来反映地理对象的信息。而电子地图的交互功能(如放大、漫游功能)以及地图数据库的支持使得地图的载负量可以得到极大的扩展,同时,计算机系统的多窗口、多线程技术可以应用于视频、声音、图像、文字、动画等多媒体信息的表达,丰富了电子地图的内容,最大限度地发挥了电子地图的阅读功效。

(4)多尺度数据的结合

一幅纸质地图通常只能表达某一个固定比例尺的地图内容。而电子地图可以同时载负多个比例尺数据,并通过设定的显示条件动态地调整地图表达的内容,如随着地图的逐步放大,更大比例尺的地图被显示出来。

(5)快速、高精度的信息检索与地图分析

在纸质地图上搜索地图目标需要用户独立完成,并且只能进行一些比较简单的量算和分析,不仅费时,精度也不容易得到保证。电子地图利用地图数据库的查询、检索功能和GIS的空间分析功能,很容易实现用户对地图目标的高效、高精度查询和分析的需要。

(6)多维与动态可视化

在纸质地图中,制图人员通常将地理对象的空间分布形态通过制图综合转换为二维平面形式表现出来,即使三维、连续分布的地图信息,也间接地转化为等值线形式来表达。纸质地图表达的地图目标都是静态的、不变化的,要在图上反映动态变化的地图现象,往往通过几个时间段的静态地图组合来实现。而在电子地图上,不仅可以进行地图的三维显示、空中飞行、虚拟环境漫游等,而且可以描述地图现象的动态变化过程。

(7)共享性

与纸质地图相比,电子地图依托于计算机技术,网络通信技术和容量大、便于携带的存储设备,如光盘等,更容易实现地图的复制、传播和共享。目前在Internet上已建立了众多的电子地图网站,人们可以很方便地从地图上查询城市交通、地名、旅游景点、商业服务业信息等,极大地提高了电子地图的利用率。

(8)低成本性

电子地图的内容以数据的形式保存在地图数据库中,可以方便地进行无损失复制和数据的编辑、修改,从而很容易更新再版,制作周期短,成本低。

5.4.2 电子地图的设计与制作

电子地图的设计是指以计算机为基础,研究电子地图的数据采集、存储管理和输出显示等技术方案的一种过程。在这一设计基础上,利用电子地图制作系统进行电子地图的加工与制作,最终建立用户可使用的电子地图产品。

电子地图的分类形式多种多样。例如,根据媒介和用户使用方式的不同,电子地图可以划分为单机电子地图、光盘电子地图、触摸屏电子地图、网络电子地图和移动电子地图等;根据表达的专题信息的不同,电子地图可以划分为多种类型,如城市电子地图、交通旅游电子地图、商业电子地图等;根据数据形式的不同,电子地图又可以划分为矢量电子地图和像素(栅格)电子地图等。

尽管不同类型的电子地图在设计上存在着差别,但总体上电子地图的设计仍然应遵循一些基本原则。

1. 电子地图的设计原则

(1) 数据设计

为了方便管理和表达,电子地图的数据根据不同性质划分为基础地理数据和专题数据两大部分。基础地理数据是对地图中一般地理要素信息的数字化描述,如地貌、水系、居民地等,在电子地图中属于背景信息,起到基础的地理环境信息说明的作用。专题数据是反映电子地图的主题要素信息,如城市交通电子地图的专题数据有道路、公交站点、公交线路、出租车停靠站、加油站等。专题数据按照不同的属性又划分为不同的图层。图层将具有相同类型的地图目标集中在一起,便于统一、协调地管理,如一级道路图层、公交站点图层、加油站图层、公园图层等。

由于基础地理数据起辅助作用,数据既可以采用矢量数据形式,也可以采用图像(栅格)数据形式。矢量数据形式对基础地理数据以目标为单位进行了进一步的细分,可以描述到更具体的地图目标实体,这对于需要利用基础地理数据进行信息分析的电子地图来说特别适合。图像数据形式是将整个基础地理数据转化为图像保存,无法实现对其中单个地图目标或地图目标集合的查询与分析。因此在一般情况下,专题数据采用矢量数据形式,包括点、线和面目标,从而满足系统对专题地图目标处理的需要。

多媒体数据在电子地图中可视为专题属性数据,它依托于专题要素的空间数据,可以更直观、形象地反映地物的真实面貌。但是视频、图像、声音等多媒体数据具有信息量大的特点,需要考虑是否对它们进行压缩处理。

(2) 界面设计

界面是电子地图的外表,一个专业、友好、美观的界面对电子地图是非常重要的,它能激发用户强烈的使用兴趣。电子地图的界面设计包括界面的形式设计、显示设计和布局设计。

①界面的形式设计。电子地图在功能上需要考虑一组地图的浏览、查询和分析的操作命令。与此相适应,界面形式应符合电子地图强调美观的特点。传统的 Windows 菜单表现形式单一,因此,电子地图通常采用专门定义的菜单,或者直接采用工具栏按钮。

②界面的显示设计。对于电子地图,应尽可能多地表示地图丰富的内容,因此,地图显示区应尽可能设计得大一些。通常整个屏幕都是地图,其界面包括工具条、查询区、地图相关位置显示等。点击地图上的任一位置,则通过与之相对应的其他链接地图,来进一步显示该位置的详细信息。

③界面的布局设计。电子地图布局设计是指界面上各功能区的排列位置。一般情况下,为了便于电子地图的操作,工具栏宜布置在地图显示区的上方或下方,查询区和图层控制区宜布置在显示区的两侧。为了让地图有尽可能大的显示区域,可以采用隐藏不常用工具栏的方法。

(3) 符号和注记设计

电子地图和纸质地图在符号设计上既有一致性,又存在差别。一致性体现在电子地图的符号设计应符合地图符号设计的一般原则,如综合、形象、逻辑性和协调性,其中基础地理底图的符号应尽可能符合纸质地图符号设计的规范。另一方面,由于在屏幕上显示和具有交互功能,电子地图中专题要素的符号设计和传统纸质地图相比存在一定的差别,体现在电

子地图的符号设计具有更大的灵活性,地图符号要求更加突出醒目,通常可以通过立体、闪烁和敏感提示(即鼠标跟踪显示)的方式来设计。

电子地图的注记大小应保持固定,一般不随着地图比例尺的变化而改变大小。路名注记往往沿街道方向配置,如果表示了行政区,一般还要有行政区名称注记,通常用较浅的色彩表示,字体要大一些。

(4)色彩设计

电子地图的色彩设计主要考虑色彩的整体协调性。在一般情况下,符号的设色应尽量参照习惯用色,地图内容的设色与界面的设色应形成反差,以突出地图显示区;点状符号、线状符号应选择相对较饱和和醒目的色彩,使之与面状符号或背景色有清晰的对比。

2. 电子地图的制作过程

电子地图的制作是设计的具体实施过程,不过两者之间通常没有明显的界线和区别,表现在制作者在电子地图的制作过程中需要随时调整设计方案,以达到最佳的制图效果。

电子地图制作的过程为:生产计划→搜集资料→制作底图、文字、图像、声音、动画等→专题信息数据的制作→各种图表的制作→各种数据合成→出版。

5.4.3 电子地图集的设计与制作

电子地图集是围绕某一主题,由若干电子地图或者专题图片(图像)数据按照一定结构组织起来的集合形式。电子地图的设计与制作在上一节中已作了介绍,在电子地图集中主要讨论电子地图和专题图片之间如何建立有机的联系。

1. 电子地图集的基本单元

电子地图集的基本单元可以是单幅电子地图,也可以是非地图的图片、视频等。根据所在位置的不同,电子地图集中包含了9种基本单元,包括片头、目录、图组、主图、图幅、插图、片尾、背景音乐、专题目标。

片头是电子地图集的开始部分,往往通过视频,配以背景音乐来说明地图集的主题。目录是地图集全部图组的索引,表现为层次结构形式,每一个图组都反映主题条件下的一个专题部分,包括一幅主图和若干图幅或插图;主图是一个图组的开始页,它通常用于反映图组的主要内容,并建立与本图组中其余图幅的索引联系;图幅是反映某一专题的相关信息的页面,是地图集的主要组成部分;插图是一种特殊的图幅,只有通过图幅链接才能被浏览,用于进一步反映所选地图目标的图形图像信息;背景音乐和片尾都是附属信息,前者可以定义在目录、图组、主图、图幅或插图之上;专题目标主要是指在图幅背景底图之上添加的属于某一个专题要素的地图目标,根据几何属性又划分为热点、热线和热面,如城市电子地图中“商业网点”图幅的各个具体商业网点目标表现为一个个热点,专题目标可以具有相对应的目标属性数据,包括多媒体数据。

2. 电子地图集的一般结构

在图5-14中描述了电子地图集的一般结构,其中目录、图组、主图和图幅之间通过指定关联区域范围实现相互间的连接,在电子地图图幅和插图之间通过用户定义的具有专题要素性质的热点、热线和热面三种节点形式与专题属性数据或多媒体数据相连。

关联区域是指在专题图片上划分出的一个矩形或多边形坐标范围,制作者给该范围指

定一个相关联的电子地图或者专题图片的名称,在电子地图浏览系统中,一旦用户点击该区域,则系统自动调用和显示关联的电子地图或专题图片。

热点、热线和热面是在目录、图组、主图、图幅、插图等背景数据之上根据专题信息查询的需要另外增加的点、线、面目标,是连接专题属性和多媒体信息的关键。

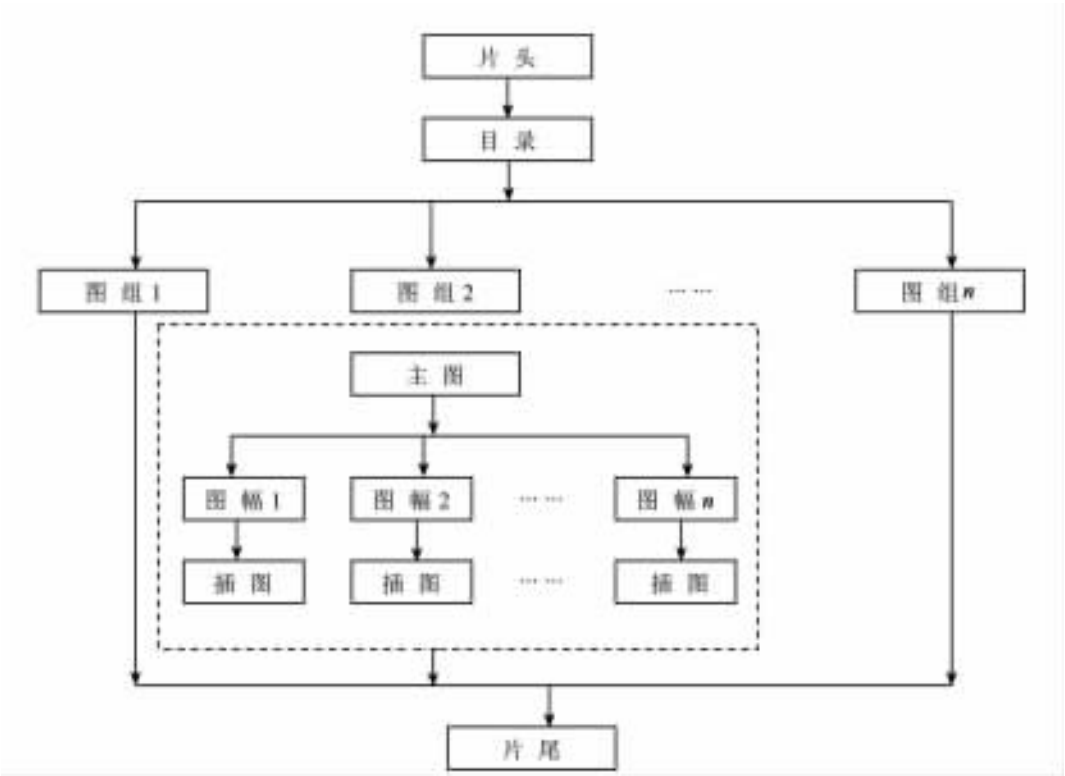


图 5-14 电子地图集的结构框图

3. 电子地图集的制作过程

电子地图集的制作一般包含如下过程：

- ①电子地图集的选题和目标分析；
- ②电子地图集的结构与风格设计；
- ③电子地图集数据(包括电子地图、专题数据与多媒体数据)的加工与制作；
- ④电子地图集的数据集成；
- ⑤成果检查与电子地图浏览系统的应用测试；
- ⑥出版发行。

5.4.4 电子地图集系统的开发

电子地图集是电子地图集数据与电子地图集软件的结合体,缺少其中任何一项都不能称之为完整的电子地图集产品。

电子地图集软件包括设计制作系统和浏览系统两个部分。考虑到在实际过程中应尽可能保持制作与应用的相互独立性,两个系统各自开发为最佳。这就要求在电子地图和电子地图集的设计基础上,应用软件平台分别进行电子地图设计系统和浏览系统的开发,以实现电子地图的设计、制作、浏览、查询和分析功能。

1. 系统的总体框架

设计系统与浏览系统虽然各自独立开发,但两者之间仍然是紧密联系的,主要体现在两个系统之间共享电子地图数据库文件,即浏览系统所使用的电子地图数据正是来自于设计系统。图 5-15 反映了两个系统之间的这种联系。

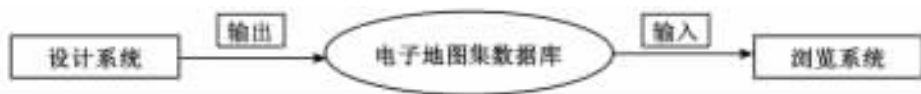


图 5-15 电子地图集设计系统与浏览系统的关系

因此,针对相同的数据,设计系统与浏览系统应具有相同的电子地图集数据库 I/O 接口。这就要求在系统开发前首先确定电子地图集数据库的结构,建立电子地图库数据库的检索、查询、修改和输入输出等功能函数。

2. 电子地图集设计系统的开发

电子地图集设计系统的任务是根据电子地图集的设计要求,将电子地图、专题图片、专题空间目标、专题属性数据、多媒体数据等集成起来,建立相互之间的关联与被关联关系,并保存在电子地图集数据库中。

电子地图集设计系统应具备的功能包括:

- ①电子地图集数据库的读写功能;
- ②对电子地图集的素材(包括电子地图、专题图片、视频、音频、图像、动画等)进行分门别类的管理,以便在需要时可以快速检索得到;
- ③可以构建 9 种类型的基本单元,不同的单元类型具有自己独立的参数形式;
- ④建立各基本单元之间的关联方法,包括在专题图片上通过指定关联区域与其他素材关联,在电子地图上通过添加热点、热线和热面与专题属性数据关联;
- ⑤可以构建电子地图集的符号库系统,实现电子地图的显示。

其中,第三、四项功能是电子地图集设计系统的核心,它将电子地图集中的不同图组、图幅和插图集成为一个完整的系统。在实现上,系统将界面分为 5 个部分,即素材管理目录区、基本单元显示/编辑区、当前单元参数设置区、电子地图集树结构显示区和菜单工具栏区(如图 5-16 所示)。从片头开始构建,系统支持每一个基本单元的初始化显示、素材的调入、布局设置、关联设定和参数保存。

3. 电子地图集浏览系统的开发

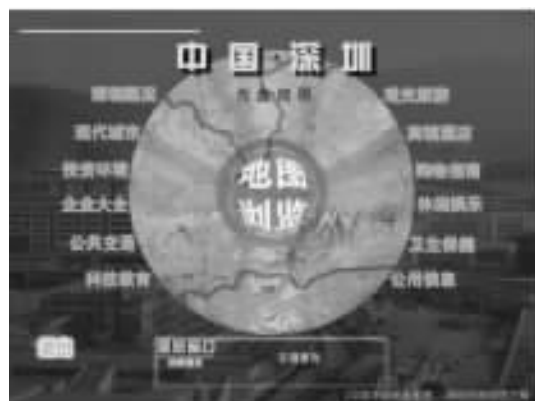
电子地图集浏览系统是用户阅读电子地图的工具,它以电子地图集数据库为数据源,其功能包括:

- ①电子地图集数据库的读写功能;



图 5-16 电子地图设计系统界面

- ②电子地图集中各种基本单元的显示功能；
 - ③电子地图的缩放、漫游功能；
 - ④多媒体(视频、音频、图片、动画等)的播放功能；
 - ⑤当鼠标在一个专题图片单元的关联区域或电子地图的热点、热线、热面上点击时,关联到另一个单元或专题属性信息的快速响应功能；
 - ⑥电子地图的查询、检索功能；
 - ⑦电子地图的空间分析功能,主要表现为地图量算、缓冲区分析和最短路径分析功能。
- 电子地图集浏览系统首先运行片头和目录,等待用户从目录中选择感兴趣的图组。一旦用户选定了图组,则显示该图组的主图,通过选择主图的关联区域浏览图幅。对于电子地图+专题目标的图幅,执行缩放、漫游、查询和分析功能,对于显示专题图片的图幅,则继续通过关联区域浏览其他图幅信息。在每个主图、图幅中设置“返回目录”、“返回主图”和“退出系统”的菜单或工具按钮,方便用户快速地切换到新的内容。图 5-17 为武汉大学资源与环境科学学院设计开发的“深圳光盘电子地图”中的部分界面。



(a) 目录



(b) 电子地图 (左上角为查询后显示的視頻)



(c) 公交车辆换乘方案



(d) 专题图片图幅

图 5-17 深圳光盘电子地图

第 6 章 计算机地图制图系统的软件开发

计算机地图制图(Computer Aided Cartography , CAC)系统是计算机地图制图软件与硬件的集成系统。随着计算机硬件技术和设备 ,包括图形图像输入输出设备的不断发展 ,CAC 系统的性能在另一方面也越来越依赖于软件系统开发功能的完善程度。在前面的 5 章中详细介绍了计算机地图制图各个环节的原理、方法与技术 ,本章将这些知识结合起来 ,介绍如何设计开发一个计算机地图制图软件系统。

6.1 CAC 系统开发的基础知识

20 世纪 60 年代末提出了软件工程 ,即采用工程化的思想进行软件开发与维护。CAC 系统的软件开发既要遵循软件工程的一般理论与方法 ,同时也要考虑 CAC 系统的自身特点 ,即以地图数据采集、处理与表达为核心。

6.1.1 系统开发的基础条件

CAC 系统开发需要软硬件环境支持和系统设计、开发人员的创造性劳动。如图 6-1 所示 ,开发 CAC 系统必须具备以下三个基础条件：

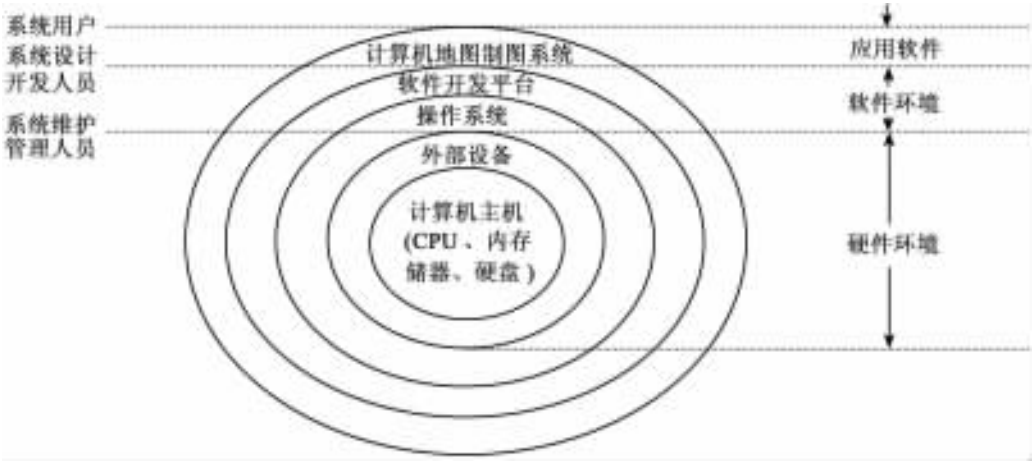


图 6-1 计算机地图制图系统架构图

1. 可支撑的硬件环境

硬件环境是软件开发的物质基础 ,即开发所需要的设备条件。如果没有相应的设备支

持,软件开发就成为无米之炊。对于 CAC 软件系统而言,往往需要对大数据量的图形图像数据进行存取、运算、处理和分析,因此需要计算机设备具有大容量的存储器(主存储器和外存储器)和快速、高效的中央处理器(CPU)。

在 20 世纪 90 年代以前,主要依靠图形工作站,而目前高性能微机的内存容量已能达到 1G 以上,外存储器容量可达到数百个 G,甚至更大,CPU 更是不断推出新的产品,已经完全能够取代图形工作站,且在软件系统上具有更大的灵活性。此外,大幅面、高精度的图形图像输入输出设备,如数字化仪、扫描仪、屏幕、激光或喷墨打印机等也成为系统的支撑条件。

2. 适合的软件环境

应用软件系统的开发需要软件环境的支持,包括操作系统和软件开发平台。目前应用较普遍的操作系统有 Microsoft Windows 系统、UNIX 系统和开放源代码的 Linux 系统等,它们均支持处理器管理、存储管理、设备管理、文件系统和作业控制五大功能。其中,Windows 系统应用最广,具有良好的图形用户界面和网络服务支持等。软件开发平台是应用软件开发的前端环境和加工车间,它包括开发所需的工具和软件支持,如程序设计语言、数据库系统以及支持图形数据处理的各种动态库、组件,等等。选择什么样的软件开发平台,决定了软件开发的效率、目标和质量。因此,CAC 系统的软件开发平台应同时兼顾到地图图形数据和属性数据的组织、管理与运算。

目前基于 Windows 环境的程序开发语言,如 Visual C++、Visual BASIC、Borland C、Delphi 等,都具备了丰富的图形图像处理与表达功能。对于数据库系统,一种方式是将地图图形数据交给专门开发的地图数据库系统来组织、管理,并实现图形数据与属性数据的相互连接,另一种方式是运用商业数据库系统,如 Dbase、FoxPro、Access、Sybase、Informix、Oracle 等,它们通常都用于非空间数据的管理,其中 Oracle 推出的 Spatial 组件已经实现了在关系型数据库中存储、管理空间数据,其他数据库如 FoxPro、Access 等经过特定的软件开发也可以应用于管理图形数据。

3. 全面的专业知识与技能

CAC 系统作为一个专业软件系统,要求系统设计开发人员首先全面地了解甚至掌握传统地图制图的专业知识与专业技能,包括地图(制图)学的理论、方法、技术与应用,特别是地图的分类、投影变换、制图流程(含设计、绘制、整饰、输出)、制图规范等几个关键性环节;其次是熟悉与计算机地图制图相关的学科,如数学、图形学、测绘学、计算机科学的理论与方法,熟练掌握软件开发平台的应用。

6.1.2 系统开发过程

按照软件工程化方法定义,CAC 系统开发包括系统的需求分析、设计、软件实现、软件测试与运行维护 5 个阶段。

1. 需求分析

需求分析是系统开发的前期阶段,主要明确系统要实现的目标与任务要求。只有确定了系统需要做什么,才能进一步研究系统必须具备哪些功能,以及如何实现这些功能。

CAC 系统的目标是利用计算机系统取代传统手工地图制图流程,生产不同地区、不同类型的地图产品。在实际应用中,通用性强、适合于任何类型地图制作的计算机地图制图系统往往存在针对性差、效率低下等问题,它在某种程度上可能更像一个通用的图形处理软

件。而在实际系统开发中,不同的地图类型(如普通地图、专题地图)、不同的比例尺(如大、中、小比例尺地图)以及应用需要(是否包含地图分析功能以及包含哪些分析功能)等都会产生不同的系统要求,从而直接影响到 CAC 系统的数据结构和软件结构。

在这种情况下,进行系统的需求分析就要求系统分析员和地图制图人员密切配合,充分了解相关类型地图的制图流程与产品形式,从中分析系统要求的功能。

2. 系统设计

在通过需求分析,明确系统任务之后,就进入系统设计阶段。在这一阶段,主要解决系统如何做的问题,以实现系统既定的目标。

系统设计又包括总体设计、数据库设计与详细设计三个阶段。

(1) 总体设计

该阶段确定系统的总体结构框架,主要将前面的各项功能需求转化为统一的体系结构,包括各个模块的功能、组成以及模块之间的相互关系。在 CAC 系统中,地图数字化采集模块、数据处理模块和制图输出模块可以进一步划分。例如,地图采集可以划分为空间数据的采集与非空间数据(包括属性数据与地理统计数据)的采集。其中,空间数据采集又可划分为点、线、面以及复合目标的采集等;地图数据的编辑处理则可以划分为增加、删除、修改等功能;在地图表达中,点、线、面目标的符号化实现又各不相同。

(2) 数据库设计

数据库作为存储在计算机内的有结构的数据集合,主要提供系统必需的数据支持。数据库设计的目的在于针对系统需求,建立最优化的数据模型,包括概念模型、逻辑模型与物理(存储)模型。地图数据库与一般数据库的不同之处在于它需要考虑空间及空间关系数据。如何在海量的、多要素的、关系复杂的地图数据基础上建立高效、科学的数据模型是关系到 CAC 系统性能的重要研究内容。

(3) 详细设计

这一阶段要对每个模块完成的功能进行具体的、结构化的过程描述,包括数据结构与算法、实施的功能和使用数据之间的关系等。为了描述数据和算法,通常采用流程图、N-S 图、伪码等表示方法。

3. 软件实现

软件实现是系统开发的核心环节,它直接表现为程序编制,即将系统设计的结果采用特定的程序设计语言,如 BASIC、C++、PASCAL 等编写出来。具体选择何种程序设计语言作为 CAC 系统的开发语言,需要考虑多个因素,包括语言是否有丰富的、调用方便的图形图像处理函数,开发人员对语言的熟悉程度,语言的可移植性,编译程序的效率,等等。不过在此之前,软件开发平台还要首先考虑数据的存储与管理问题,即根据数据库设计的数据模型选择或者开发数据库系统,同时数据库系统和程序设计语言之间应具有良好的接口。在程序编制过程中,为了保证程序质量,要求编写的程序源代码具有正确性、可读性、可移植性、结构性与高效性等。

4. 软件测试

软件测试的目的是尽可能找出在总体设计、详细设计、软件编码中的错误,加以纠正,从而确保得到高质量的软件。

软件测试包括单元测试、组装测试、确认测试、系统测试等。其中,单元测试早在软件编

码时就已经开始,组装测试是在将单元模块集成起来时进行的测试,解决单元之间不能正确结合的问题。确认测试是检验软件是否符合软件规格说明的技术标准,而系统测试主要是测试软件和硬件、数据等其他支持环境一起,在实际运行情况下与用户需求的匹配程度。

在各个测试环节,实施测试的方法有分析方法(静态分析法和白盒法)与非分析方法(黑盒法)两种。白盒法也称为逻辑驱动方法,它通过分析程序内部的逻辑与执行路线来进行测试。黑盒法是功能驱动方法,仅根据 I/O 数据条件来设计测试用例,不管程序的内部结构与路径如何。

5. 运行维护

测试通过的系统经过一定时期(一般为几个月)的试运行并得到进一步完善后,可以正式投入运行使用。但是,软件中仍然可能存在暂未发现的问题隐患,系统的数据、功能在实践过程中需要修改、调整,需要适应新的系统工作环境、设备等,都会提出系统维护的要求。在 CAC 系统中,地图数据采集设备的不断更新、采集方式的不断进步、地图制图规范的进一步完善,都会导致系统软件的升级维护。

6.1.3 软件开发模型

软件开发模型是软件工程思想的具体化,是实施于过程模型中的软件开发方法和工具,是在软件开发实践中总结出来的软件开发方法和步骤。总之,软件开发模型是跨越整个软件开发过程、工作与任务的结构框架。

目前,软件开发模型有瀑布模型、原型模型、增量模型、螺旋模型和喷泉模型等。

1. 瀑布模型

瀑布模型又称为生存周期模型,它将软件开发过程从需求分析开始,到系统设计、软件编码、测试与维护,各个环节由前至后,相互衔接,从而按照线性顺序连接起来,如同瀑布流水,自上而下,逐级下落(如图 6-2 所示)。

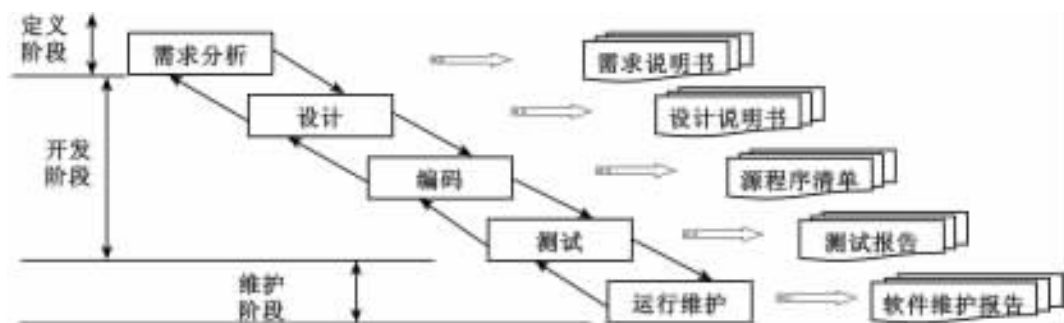


图 6-2 瀑布模型

瀑布模型强调了生存周期中各个阶段的严格性,且前一个阶段的成果总是决定着后一个阶段的任务实施,特别是开发前期的良好需求说明和系统设计,避免了盲目进行软件程序编写易造成的巨大浪费。但是该模型是一种理想的线性开发模型,缺乏灵活性,尤其是无法解决软件需求不明确或不准确的问题,在开发效率上也存在着明显的不足。一旦需求分析和系统分析存在隐性问题,在开发完成后才能发现,但这时已为时太晚。因此,瀑布模型适

合于功能和性能明确、完整、无重大变化的软件开发。

2. 增量模型

增量模型主要针对用户需求不明确或者不完整的情况 ,由用户首先给出核心需求 ,开发人员按照需求开发出一个原型系统 ,实现部分主要功能 ,提交给用户并征求他们的反馈意见 ,然后逐步完善 ,直至整个系统的最终完成。

增量模型弥补了传统瀑布模型在每个阶段都要求严格、完整的不足 ,建立了一种更加灵活、有效的软件开发机制 ,可以通过局部功能的实现来促进用户对整个软件需求的认识 ,从而达到实现用户需要的最终目标。

3. 螺旋模型

螺旋模型是在瀑布模型与增量模型结合的基础上 ,加入风险分析所建立的模型。在软件开发中 ,有各种各样的风险 ,且风险有大有小。开发项目越大、越复杂 ,其设计方案、资源、成本和进度等因素的不确定性越大 ,风险也越大。为了保证软件开发的顺利实施 ,人们将开发过程划分为几个螺旋周期 ,每个螺旋周期都包含计划制定、风险分析、实施工程和用户评估几个步骤 ,且比其前一周期有着更为完善的软件新版本(如图 6-3 所示)。

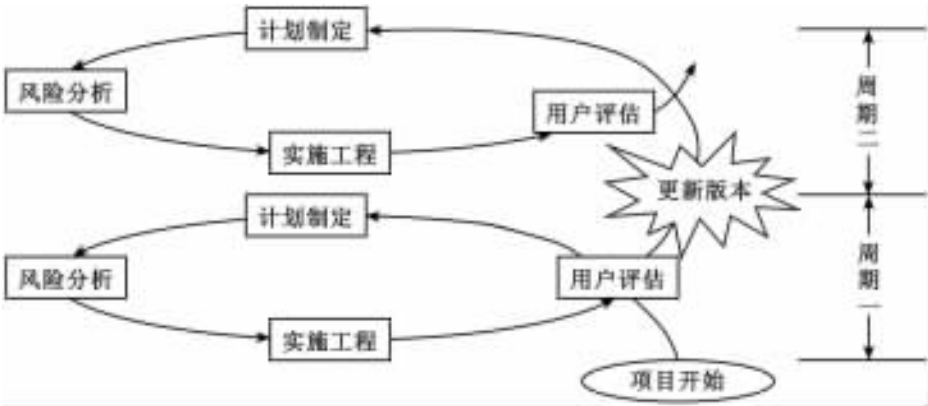


图 6-3 螺旋模型

螺旋模型适合于复杂的大型软件的开发 ,它使得开发人员和用户对每个螺旋周期出现的风险有所了解 ,并因此做出正确的反应。假如风险过大 ,开发人员和用户都无法承受 ,项目就有可能中止。

4. 喷泉模型

瀑布模型、增量模型和螺旋模型在开发过程中都有明显的层次性、阶段性 ,一个阶段完成才能进入下一个阶段。在面向对象程序设计中基于每个对象的分析、设计、编码、测试等活动不断重复进行。喷泉模型将这些活动统一起来 ,且在各活动之间不存在明确的界线 ,并能够反复进行 ,体现了迭代和无间歇的特征 ,因此 ,主要用于支持面向对象的软件开发。

6.1.4 软件设计与实现方法

软件开发模型表现了整个软件开发活动的过程特性 ,其中软件的设计与实现是软件系统开发的关键环节 ,在软件生存周期中占有重要的地位。这就要求开发人员根据系统需求

进行软件的系统框架、数据结构、功能模块的合理设计,并将其成功地转变为程序代码。目前经常采用的两种方法是结构化方法和面向对象方法。

1. 结构化方法

结构化方法是软件工程产生以后首先提出来的软件开发方法,来源于模块化的思想,它通过自上而下、逐层细化的方法将系统划分为一个个模块,其采取的策略是分解和抽象(如图6-4所示)。一方面,通过分解将复杂的问题划分成若干个相对简单的小问题,然后分别予以解决;另一方面,不同的分解层次又具有不同的抽象程度,在顶层首先关注问题最本质的方面,其抽象程度最高,随着层次降低,越来越多地关注问题的具体方面,即问题的细节。这一方法贯穿于分析、设计和程序设计(即软件实现)的过程中,分别称为结构化分析(Structured Analysis, SA)、结构化设计(Structured Design, SD)和结构化程序设计(Structured Programming, SP)。

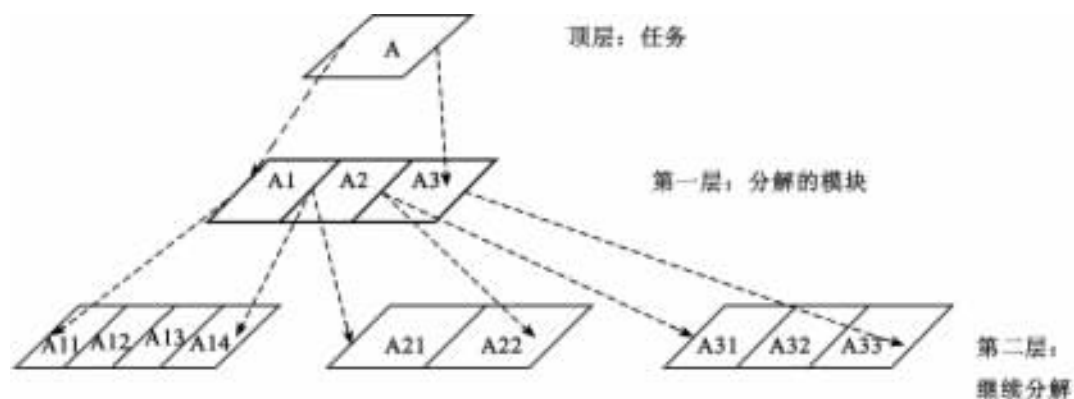


图 6-4 结构化方法对任务的逐层分解过程

结构化方法使用的工具有数据流图(Data Flow Diagram, DFD)、数据字典、判定表、判定树和结构化语言等。数据流图包括5个部分,即加工、数据流、数据存储、数据源及汇点,它通过图形方式描述数据流从输入到输出的移动变换过程。数据字典说明数据流图中各个部分的具体成分及其含义,是对数据流图的补充说明。判定表、判定树和结构化语言详细描述数据流图中不能被再分解的每一个基本加工的处理逻辑。

结构化程序设计是20世纪60年代中期发展起来的一种新的程序设计方法,它规定了一套应该如何进行程序设计的准则,采用了自上而下分而治之的分割技术以及模块化的组织结构形式。不论程序的复杂程度如何,其功能结构可以仅用顺序、选择和循环三种基本控制结构来描述,并且每个结构只有一个入口和一个出口。这使得设计的程序具有合理的结构,易读,易调试,并且容易保证其正确性。

结构化方法作为使用最早、使用时间最长、发展最为成熟的软件设计与实现方法,具有简单、实用的特点,开发者容易掌握和使用,特别适合于瀑布模型。

2. 面向对象方法

面向对象(Object-Oriented, OO)的概念起源于20世纪60年代的程序设计语言,在其后的30年时间里发展成为成熟的、系统的面向对象程序设计方法(OOP),并被进一步应用于软件分析与设计过程中,产生了面向对象的系统分析(OOA)和面向对象的系统设计(OOD)。

面向对象方法是一种围绕客观世界的概念来组织模型的全新的思考方式 ,它是一种运用对象、类、继承、封装、聚合、消息传送和多态性等概念来构造系统的软件开发方法。在面向对象方法中 ,对象作为客观世界实体的抽象描述 ,具有封装性 ,它同时包括数据和对数据的操作 ,对外部只提供一个抽象接口而隐藏具体实现细节。类是对相似对象的共同特性的描述。类具有继承的特性 ,它使得某类对象可以继承另一类对象的特征和能力。消息是对象之间通信的手段 ,是对象之间相互请求或相互协作的途径 ,它用来指示对象的操作。方法是对象接收到信息后应采取的动作序列的描述。实例是由一特定类描述的具体对象。类和实例之间的关系是抽象和具体的关系。

图 6-5 描述了类和对象(实例)之间的关系。

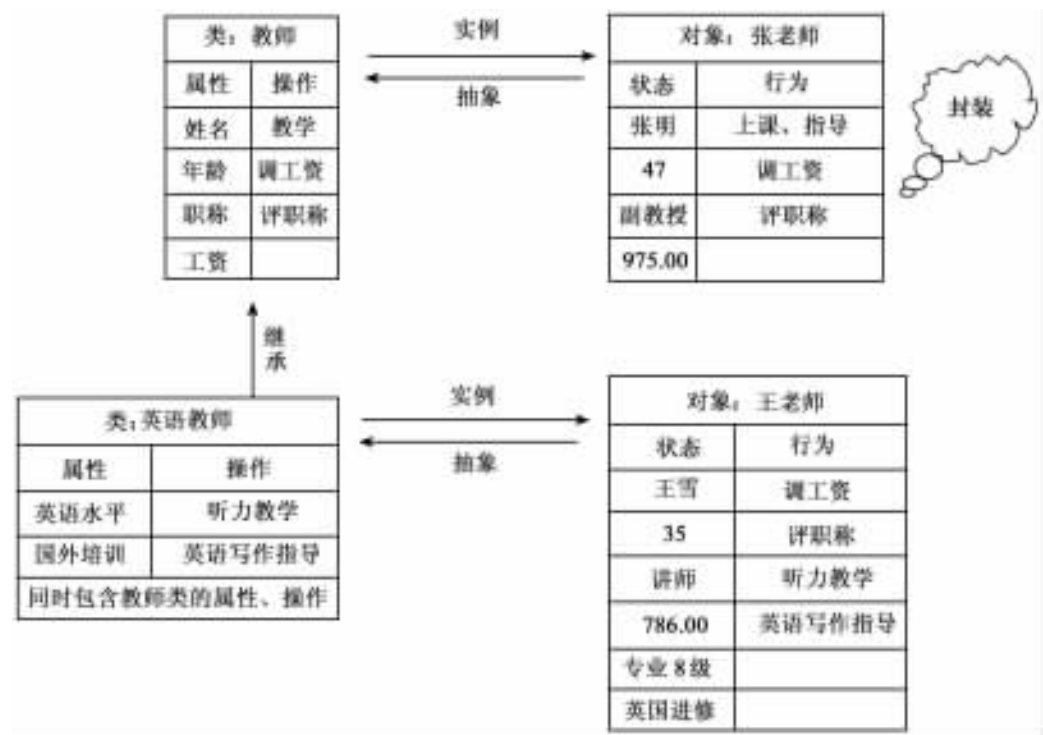


图 6-5 类和实例的关系示意图

OOA 采用从特殊到一般的归纳方法 ,对客观世界中的实体进行分类 ,区分对象及其属性 ,整理对象及其组成部分 ,并划分成不同的对象类 ,从而得到现实系统中对象及其关系 ,进而分析并掌握系统运行的规律。OOA 的重点是使用面向对象的方法解决现实世界模型的建立问题。OOA 是利用从问题领域中抽象出的类和对象来研究系统实际需求的一种分析方法。

OOD 采用从一般到特殊的演绎方法 ,它包含了两个重要的方面 :一是面向对象的分解过程 ;二是使用面向对象表示方法描述所设计系统的逻辑模型(类和对象结构)和物理模型(模块和过程体系结构)以及描述系统的静态和动态模型。OOD 提供的面向对象分解是使得面向对象设计方法在本质上区别于结构化设计方法的重要方面。

OOP 是一种系统实现方法 ,在这种方法中 ,程序由一组相互协作的对象构成 ,其中每个

对象都是某一个类的实例,而这些类都是由继承关系建立起来的类层次结构中的一员。简单地说,OOP是指在软件开发中,把OOD的结果代码化的方法,即编程方法。

6.2 结构化方法在 CAC 系统开发中的应用

结构化方法的关键在于自上而下逐层进行模块划分,并保证各模块之间的连接关系。在 CAC 系统的实际开发过程中,需要根据系统的具体需求,进行系统设计与开发,包括数据与功能模块的各个方面。考虑到实例的普遍性,下面主要介绍一个应用结构化方法进行通用 CAC 系统开发的基本思路与过程。

6.2.1 系统分析

地图种类众多,按内容划分有普通地图和专题地图。普通地图以同等的详细程度,综合地表示自然地理和社会经济的一般特征,包括地形、水系、居民地、交通网、境界线、植被与土质等(如图 6-6(a)所示)。专题地图则在基础地理底图的基础上着重表现一种或几种专题要素(如图 6-6(b)所示),如地貌图、植被图、经济地图、交通地图、旅游地图等。



(a)普通地图(地形图)



(b)专题地图(城市交通旅游图,圈内为专题要素)

图 6-6 普通地图和专题地图

尽管两种地图在表达的内容上各有侧重,但是两种地图都有地图特定的数学法则、要素对象和符号化表达过程。在数学法则上主要通过地图投影和比例尺,建立地图平面与地球球面之间相对应的数学关系,表达的要素对象都是自然地理要素和社会人文要素的空间实体,它们通过制图综合被抽象为点、线、面等几何对象,符号化表达是地理信息可视化的必然途径,包含了定点符号法、线状符号法、范围法、质底法、等值线法等众多方法。

因此,无论是普通地图,还是专题地图,地图制图的技术方法是基本一致的。要建立通用的 CAC 系统,只有从两种地图的共同之处出发,才能将它们的需求最终结合起来考虑。

概括起来,CAC 系统需要完成以下基本功能:

- ① 地图数据(包含空间数据和非空间数据)的采集与入库;
- ② 地图数据的维护和更新;
- ③ 地图数据的制图输出。

6.2.2 系统设计

1. 系统的基本功能结构

对于普通地图或专题地图,我们从计算机地图制图的基本过程可以看出,其系统都必然包括地图数字化采集、地图数据处理与地图制图输出三个主要的功能模块。此外,系统还需要两个子系统提供的支持:一个是地图数据库子系统,它提供地图空间数据和非空间数据的组织、管理、存储与输入输出功能;另一个是地图符号库子系统,它提供各种地图符号的建立、组织、管理、存储与符号化功能。计算机地图制图系统结构如图 6-7 所示。

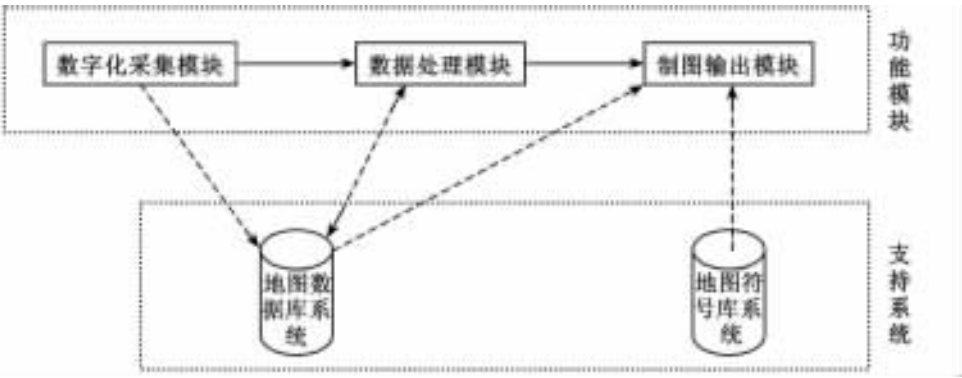


图 6-7 计算机地图制图系统结构图

采集模块和处理模块是系统的基础功能部分,前者通过数字化手段将地图信息转化为计算机能够识别和处理的地图数据(包括空间数据与非空间数据);后者则将得到的数据通过投影变换、编辑修改、制图综合、图幅接边等处理转换为正确、有效的地图数据,并存储在地图数据库中。但多数情况下,坐标变换、编辑修改也穿插在数据采集的过程中,因此两个模块之间的界线并不是非常明确。

制图输出模块是系统的关键功能部分,关系到地图的制图质量。它根据制图需要从地图数据库中提取数据,从地图符号库系统中得到表达方式,在屏幕上显示或者在其他打印输出设备上输出符合制图规范与要求的普通地图或专题地图。

地图数据库子系统与地图符号库子系统是计算机地图制图系统的重要组成部分,给系统提供地图数据存取和地图符号表达等方面的支持功能,与系统既紧密联系,又相对独立。数据库系统包括数据库(Data Base, DB)与数据库管理系统(Data Base Management System, DBMS)两个部分。DBMS 是数据库系统运行的核心,起着应用模块与数据库之间的接口作用,在系统开发中往往作为支持模块首先设计开发。

2. 地图数据库的数据结构

(1) 图层的概念及其分类

在目前常用的制图软件和 GIS 软件中,地图数据的组织通常采用图层的概念,在 Arc-Info 中称为覆盖层(Coverage),在 MapInfo 中称为一个表(Table),一个表可以对应一个以上的表达图层(Layer)。因此,地图图层是指处在同一个表达层面上的地图目标集合,这些目标一般具有空间或属性上的联系性,在存储、分析、显示等诸多操作中紧密结合。每个图层都拥有特定的对象集、属性和显示性质。如果图层中所有地图目标都具有完全相同的要素类型(分类码)和几何特性(点、线或面),该图层称为单一型图层,如图层中只有惟一类型的等高线或高速公路,否则称为混合型图层。混合型图层通常将具有一定联系的不同类型和性质的目标存放在同一个图层中,通过目标分类码进行类型的区分,如将等高线、高程点、陡坎(土质与石质)等放在同一地貌图层中。但是考虑到这种图层对于每一个目标的分类都要求单独输入,增加了数据采集的难度,因此,地图图层一般可以采用一个类型对应一个图层的单一型模式,以利于统一地图符号化和进行专题分析。在本节后面的内容中如果没有专门提及,图层均采用单一类型。

图层按照层内目标集的基本类型可以划分为基础图层、专题图层、注记图层和栅格图层 4 类。

基础图层表示普通地图所表达的要素,如铁路图层、湖泊图层、政区图层、等高线图层等,其地图目标强调空间数据,而非空间数据较为简单,主要表现为分类信息,其次如等高线的高程、道路的等级等简单数量与质量指标信息。

注记作为一种特殊的目标形式可以独立地保存在注记图层中,其空间数据仅用于注记的定位,可表现为点、线形式,又称为伪空间数据(如图 6-8 所示)。注记本身表示为非空间数据。同一个注记层具有相同的字体、字大、颜色和字型(正体、左斜、左耸、右耸等)。

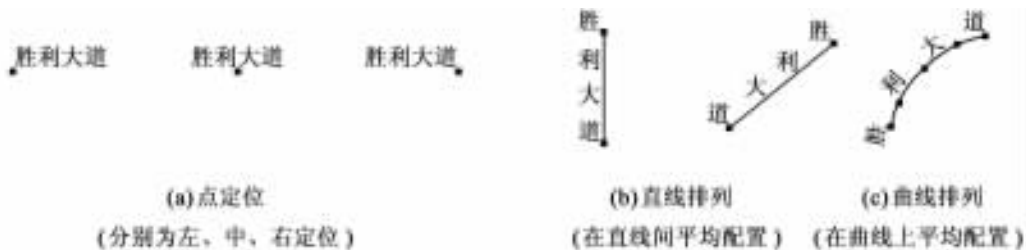


图 6-8 注记的空间数据及其定位

用于表达专题地图信息的专题图层,其空间数据形式与基础图层相同。但是专题图层更强调非空间数据,非空间数据中不仅有分类信息(属性),还有大量的统计观测数据,如降

雨量、人口数、工农业总产值等,可以用于各种专题信息的表示。

此外,为了提高地图数据来源及表达的灵活性,系统还有必要增加一种特殊的图层类型——栅格图层。栅格图层可以用于存放图像数据。

除栅格图层外,在其他图层中均采用空间数据与非空间数据分离建模的方式,两种数据通过目标标志码被统一在同一地图对象中。目标标志码又称为地图目标的关键字,在每个图层单元中具有惟一标志性。图 6-9 为地图数据库的逻辑结构。

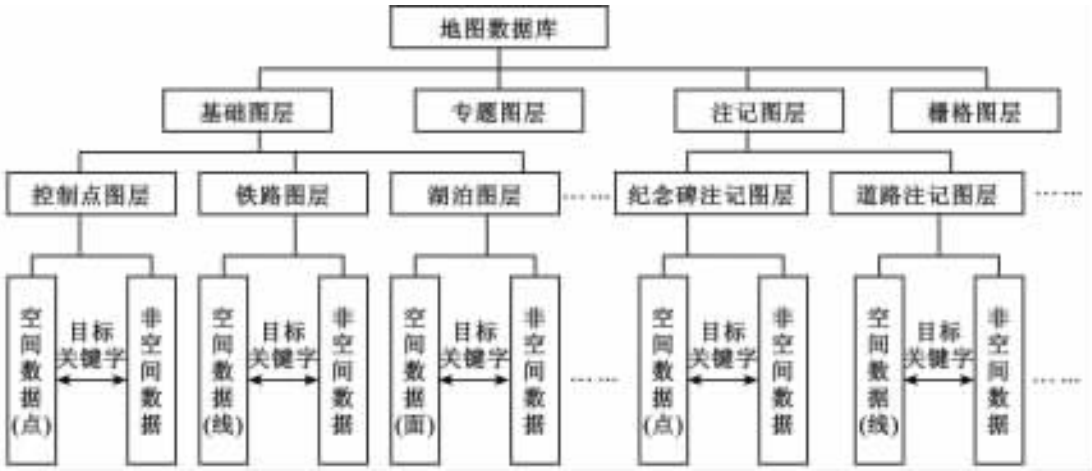


图 6-9 地图数据库的逻辑结构图

根据所存放的空间数据的几何类型,图层又可分为点目标图层、线目标图层和面目标图层。在普通地图和专题地图的基础地理底图中,地图目标是基于矢量形式的,在几何上表现为点、线、面等图形元素,其特点是面向实体,即一个逻辑记录对应于一个地理实体,如河流、湖泊、道路等。专题地图的专题要素主要是面向区域的,可以表示为网格系统或多边形系统,其中网格系统表现为栅格形式,而多边形系统与面目标的图形表达方式相同。因此,专题图层和基础图层在数据结构上是基本一致的。

(2)空间数据结构

点目标图层的空间数据结构最为简单,表现为点群集合,各目标由目标标志码和空间坐标(X,Y)构成。在线目标和面目标图层中,目标之间存在着更直接的空间关系,因此需要通过节点、弧段和面域之间的显式拓扑结构来组织空间数据。图 6-10 表示了三种图层的空间数据结构形式。

由于每个图层内部具有相同的类型,因此在图层内各目标的非空间数据项保持一致,可以采用关系数据模型建模,并通过已商业化的关系数据库来组织管理。

尽管空间数据和非空间数据已经反映了地理实体的特征,成为地图数据库的主体数据,但是海量的空间数据和非空间数据导致系统无法在数据查询、检索等方面快速响应,需要借助目标本身性质及目标之间的基本关系来建立地图数据库的辅助文件,包括目标索引文件、栅格索引文件等。目标索引文件是关于地图目标标志码 ID 与该目标记录在主体文件中的起始位置偏移量的对照索引表,可采用稠密索引方式。栅格索引反映地图目标之间的空间

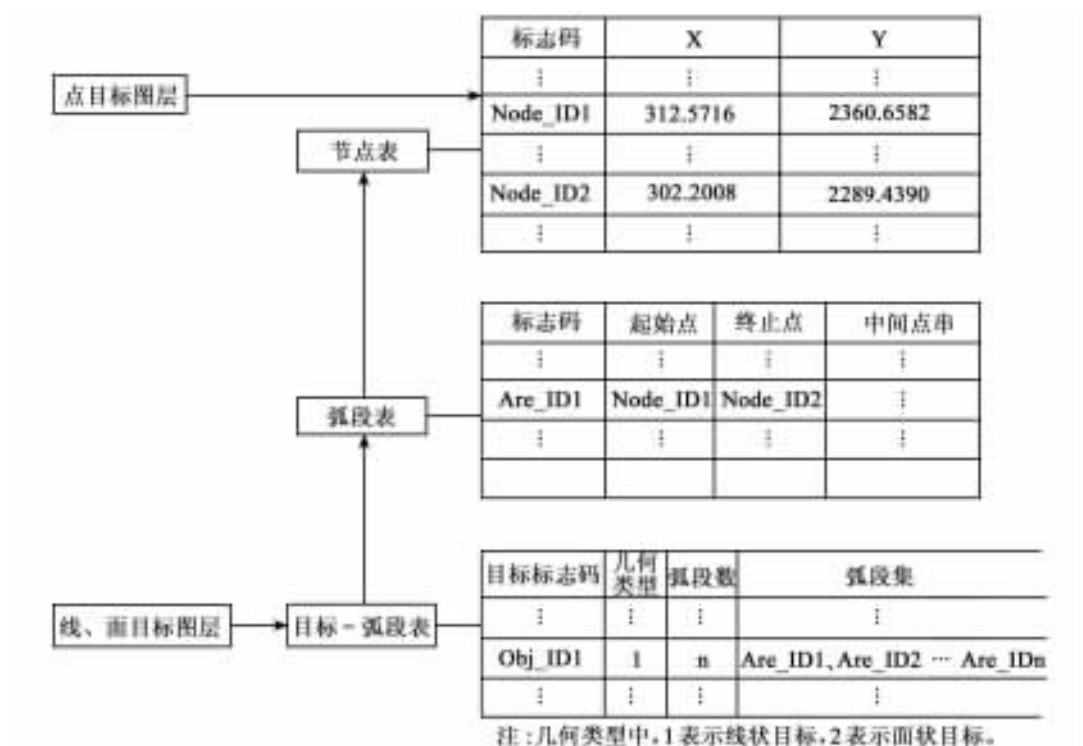


图 6-10 空间数据结构

定位关系。其过程是用一个平行于坐标轴的正方形数学格网覆盖在整个地图空间上，记录每个格网中的地图目标集合。

由于管理非空间数据的关系数据库系统已经提供了索引、排序等诸项功能，因此目标索引文件和栅格索引文件仅仅是针对空间数据建立的（如图 6-11 所示）。

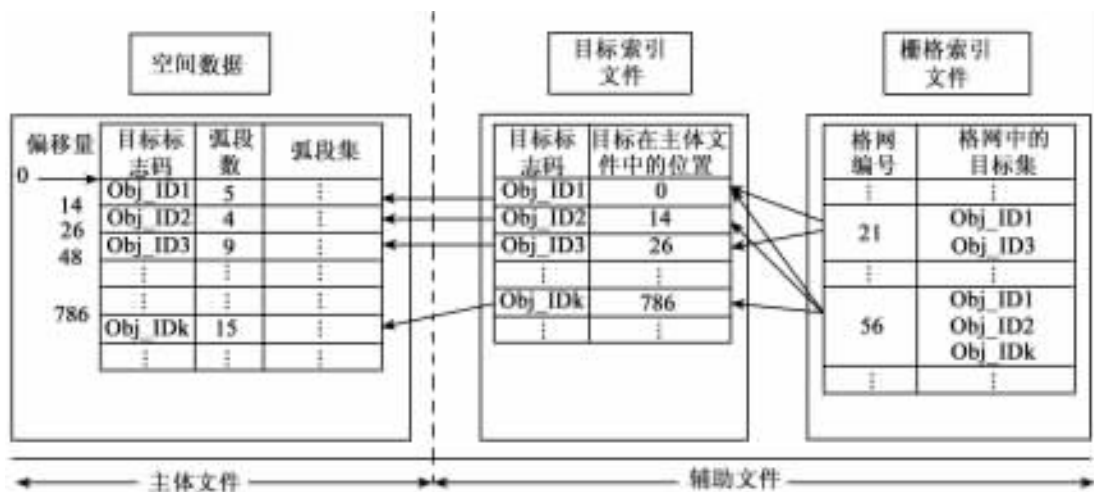


图 6-11 辅助文件结构

3. 地图数据库管理系统的功能模块

地图数据库管理系统的作用是以地图数据库及其结构为基础 ,为 CAC 系统的其他功能模块提供地图数据的组织管理和存取接口 ,是地图数据库子系统的软件系统部分(如图6-12所示)。

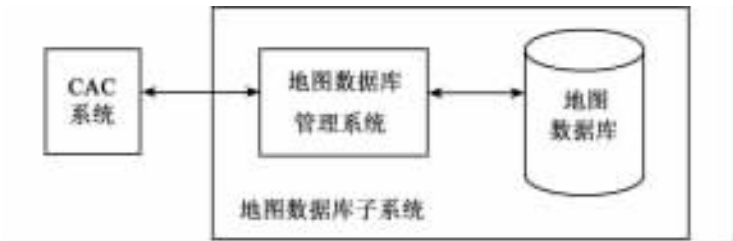


图 6-12 地图数据库管理系统的作用

在设计上 地图数据库管理系统的功能模块包括三个层次：

- ① 逻辑功能层。该层是面向地图目标的 ,包括点、线、面目标图层的建立与删除模块、打开与关闭模块 ,点、线、面目标的读写模块、修改模块和删除模块等。
- ② 编辑功能层。该层是对逻辑功能层模块在操作上的进一步分解。包括图层中目标的索引建立 ,查找模块 ,点、线、面目标空间数据与非空间数据各自的读写、修改与删除功能等。
- ③ 物理功能层。这一层包括各种类型的物理记录的读写、插入、修改、删除等模块 ,是编辑功能层模块在系统软硬件平台上的具体实现。这一层的模块与设备密切相关 ,应集中起来 ,使之与其他通用模块分开。

4. 地图符号库子系统的结构与功能

与地图数据库子系统的结构相同 ,地图符号库子系统也包括了地图符号库及其管理系统。由于一个图层总是对应一类地图符号 ,根据图层的类型 ,地图符号库中的符号又分为两类 ,即基础地图符号和专题地图符号。两类符号具有各自不同的描述结构 ,其基本构成如图6-13所示。

符号编码	符号类型	符号的具体描述单元
------	------	-----------

图 6-13 地图符号库的基本符号结构

其中 ,符号类型用于区别基础地图符号与专题地图符号 ,符号的具体描述单元视符号类型的不同而定。

在两类符号中 ,基础地图符号一般用于基础图层的符号化表达 ,主要为传统意义上的点、线、面符号 ,如纪念碑、塔、铁路、土堤和不同类型的植被。基础地图符号的特点表现为符号的形状、颜色、大小只取决于目标的分类类型。

专题地图符号则主要应用于专题图层的符号化表达。与基础符号不同 ,专题地图的符号不仅依赖于目标类型 ,而且也直接受到相关的专题数据及其统计值的影响。例如同样采

用饼图表示人口密度,不同政区的人口密度相差悬殊,其饼图大小也就不同。因此,专题地图符号主要考虑专题数据变化对其形态的影响。此外,专题地图的表达不仅仅表现为符号,也表现为某些表示方法,如等值线法、点数法等。

(1) 基础地图符号

基础地图符号反映地图目标表达的可重复单元,包括点、线、面符号。其中地图符号的内部组成的几何要素称为图元,基础地图符号可视为一组图元的有序集合。图 6-14 举出了几个基础地图符号分解为图元的例子。
















基础地图符号	图元分解
	   
	 
	  
	 

图 6-14 基础地图符号与图元关系示例

图元可分为点图元、线图元和面图元三种基本类型。其中,点图元表示具有规则几何形态的图形单元,包括椭圆形、三角形(含等腰三角形、等边三角形)、矩形(含正方形)、菱形、扇形、星形、正多边形,点图元在符号中不允许发生变形,但可以通过旋转角度改变点图元的方向;线图元包括折线、曲线(Bezier 曲线)两类;面图元表示用户自定义的任意多边形,包括折线型多边形和曲线型多边形两类。不同的图元类型具有不同的几何参数与属性参数,各图元的参数见表 6-1。

其中,在线图元的基本属性项中,线型可分为实线、虚线、点虚线三种基本类型。点、面图元由轮廓线及其围成的内部区域构成,轮廓线的线型与线图元相同,填充方式分为颜色填充、晕线填充和图像填充三种形式。颜色填充和图像填充相对简单,而晕线填充又包括了基于点图元和线图元的填充,需要考虑晕线的排列方式,如倾斜角度、间距等。

为了便于基础地图符号的绘制,符号库通常规定基础地图符号的定位点或起始点为符号坐标系原点,点符号的向上方向为 Y 轴,线符号的中心延长线方向为 X 轴。

基础地图符号的基本结构如下:

地图符号编码 ID,图元数 n

图元序号 1,图元类型,图元几何参数,基本属性值





图元序号 2,图元类型,图元几何参数,基本属性值

...

图元序号 n,图元类型,图元几何参数,基本属性值

表 6-1

图元类型及其参数

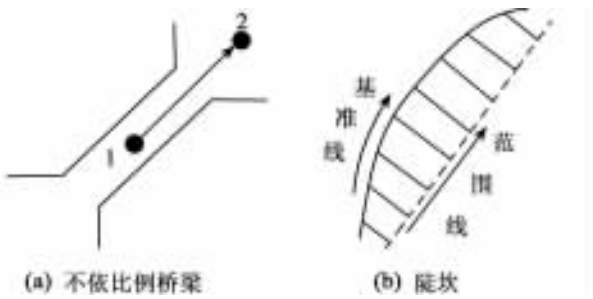
几何类型	图元类型		图元图形	图元几何参数	基本属性项
点图元	椭圆形	椭圆形	椭圆、圆	中心点坐标、长轴半径、短轴半径	旋转角度、线型、线画颜色、线粗、图元填充方式
		椭圆弧	椭圆弧、圆弧	中心点坐标、长轴半径、短轴半径、起始角度、终止角度	
	三角形	等腰	等腰三角形 等边三角形	顶点坐标、腰长、顶角角度	
		直角	直角三角形	顶点坐标、左直角边长、右直角边长	
	矩形		长方形、正方形	中心点坐标、长度、宽度	
	菱形		◇	中心点坐标、边长、顶角	
	扇形		扇形	中心点坐标、半径、起始角、终止角	
	星形		☆	中心点坐标、长半径、短半径、边长数	
	正多边形		正多边形	中心点坐标、半径、边长数	
线图元	折线			点数、坐标串	线型、线画颜色、线粗
	曲线			点数、坐标串、曲线类型(贝塞尔曲线)	
面图元	折线型多边形			点数、坐标串	线型、线画颜色、线粗、图元填充方式
	曲线型多边形			点数、坐标串、曲线类型(Bezier 曲线)	

基础地图符号编码采用“几何类型 + 分类码”的方式构成,其中几何类型占符号编码的第 1 位,其值为 0、1、2 时分别表示该符号为点、线、面符号。不过需要补充一点,由于基础地图符号表达的复杂性,系统有必要对地图符号类型进行进一步划分。如除了标准点、线、面符号外,在点符号中增加有向点符号;在线符号中增加有向线符号、范围线符号和渐变线符号;在面符号中增加散列面符号等(如图 6-15 所示)。

这种细分的意义体现在两个方面。一是在地图目标数据采集时采用不同的方式。例如不依比例桥梁符号属于有向点符号,采集时和标准点目标不同,需要采集两个点:第一个点为目标定位点,第二个点和第一个点的连线代表目标的长轴方向。再如标准线目标,只需要采集一条中心控制线(又可称为进行符号配置的基准线),就可以实现目标符号化;而范围线目标(如陡坎)由于符号配置并不均匀,除了基准线外还需要增加一条范围线来控制(如图 6-16 所示)。有向线符号和渐变线符号与标准线采集相同,只是需要顾及数字化的方向。



图 6-15 特殊的符号类型示例



注 箭头方向为采集方向和顺序

图 6-16 有向点和范围线目标的采集方式

如规定城墙采集时将符号保持在数字化方向的右侧 ;单线河流采集时总是从上游到下游 而在符号表示时必须与此一致。二是在地图表达时更符合传统的地图制图规范。

(2)专题地图符号

由于专题地图符号不仅反映目标类型 ,同时还反映目标的专题数据及其统计值大小 ,因此其表现形式比基础地图符号要复杂 ,主要体现在符号的大小、颜色等随欲表达的专题数据及其统计值大小而变化 ,需要在符号参数表中增加控制这种变化的参数。具体方法是在基础地图符号的基础上 ,增加两个参数 ,即符号控制参数数量和符号控制参数序列(如图 6-17 (a)所示)。符号控制参数为专题图层的专题数据项表达式。

为了专题地图表达的一致性 ,需要将一些特殊的专题地图表示方法(如等值线法、点数法)也纳入符号库管理范畴。考虑它们没有特定的符号构成 ,系统给每个常用的方法一个惟一的编号 ,以此取代符号中的图元序列 ,其后面的参数主要决定该方法应用中的各种可选项 ,也包括方法需要用到的专题数据及其表达式形式(如图 6-17(b)所示)。

图元序号	图元信息	控制参数数量 n	控制参数 1	控制参数 2	...	控制参数 n
------	------	----------	--------	--------	-----	--------

(a)一个图元结构

方法编号	专题数据表达式	控制参数数量 n	控制参数 1	控制参数 2	...	控制参数 n
------	---------	----------	--------	--------	-----	--------

(b)表示方法的结构描述

图 6-17 专题地图符号的结构示意图

地图符号库管理系统提供了基础与专题两类地图符号的建立、组织、管理功能与应用接口。这包括利用各类图元的设置与组合进行点、线、面符号的创建、存取、修改、删除等处理功能,以及面向各种类型地图目标的符号化调用接口。基础地图符号的符号化调用函数的参数包括地图目标分类码和目标空间数据。专题地图符号的符号化调用函数的参数在分类码和空间数据的基础上,增加目标的专题数据表达式,以确定专题表达的内容。

5. 地图数字化采集模块

地图数字化采集模块主要提供用户应用数字化输入设备进行地图目标数据跟踪采集的软件功能。地图数据建立的途径可以是多方面的,如利用手扶跟踪数字化仪数字化和利用图像跟踪数字化等。目前,越来越多地采用后一种方法,即采用在屏幕显示的图像背景上跟踪数字化,其数据源主要有纸质地图、遥感影像等。纸质地图通过扫描仪扫描为图像数据;专题属性数据与地理统计数据可以通过键盘或转存的方法输入计算机。

空间数据采集的基本过程包括图像准备、图像的坐标匹配、图层建立、目标输入和目标保存,如图 6-18 所示。

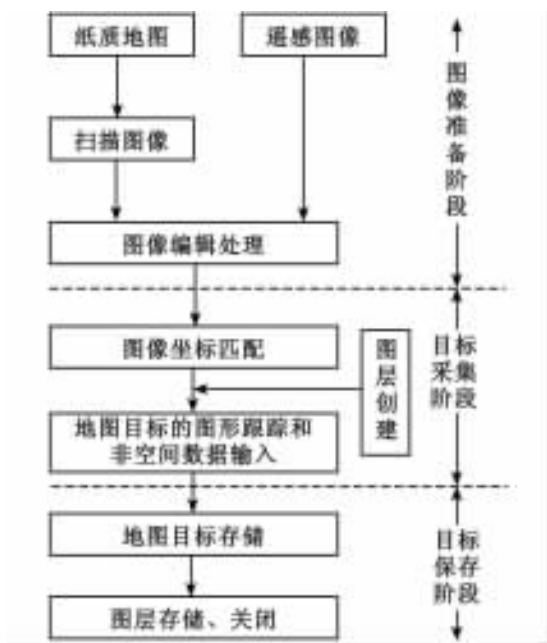


图 6-18 地图数字化采集过程

(1) 图像准备

图像准备是指和图像文件建立有关的一系列操作,主要有图像扫描、图像处理(主要指图像的色度、亮度、明暗度、大小、角度变换等常规处理)、图幅拼接等,但考虑到目前扫描仪都自带高质量的图像扫描和纠正处理程序,系统不需考虑另行开发扫描模块,而是提供多种图像格式的读入功能。

(2) 图像的坐标匹配

图像的坐标匹配是建立图像的行列定位形式与地理坐标系统的转换关系,它主要通过

在图像中确定三个以上的定位控制点经过仿射变换得到。定位控制点是由用户自己选择的图像上的一种特殊点 ,它同时具有图像的像素点坐标和地理坐标两种数据。定位控制点的地理坐标可以通过地形图量算或实地测量获取。

(3) 图层建立

图层是地图目标存放的逻辑载体 ,因为它和地图数据库直接关联 ,因此又间接成为地图数据的物理载体。采集一个地图目标的首要工作是将其归入相应的图层 ,如果这个图层不存在 ,用户就要事先创建该图层。建立图层需要确定图层名、图层类型、非空间数据字段表。图层名称可以自由指定 ,但最好采用目标的分类编码命名。图层创建和目标存储的功能模块主要由地图数据库子系统支持完成。

(4) 目标输入

目标图形的数字化跟踪和非空间数据输入是采集模块的核心功能。它需要根据用户的操作判断当前采集状态 ,实时获取采集设备的信息 ,并及时通过目标存储模块建立与地图数据库的联系。该模块对系统界面要求很高 :一方面 ,要求各图层(含栅格图层)具有快速、稳定的显示、缩放、漫游等功能 ,这些功能被统称为地图的表达功能 ,由地图输出模块中的屏幕输出功能提供 ;另一方面 ,要求能够实现图层切换、目标标志、目标节点的添加和删除等目标编辑功能 ,由地图数据处理模块提供。由此可以看出 ,三个模块之间相互融合 ,相互支持 ,并不能简单地从过程上将它们分离开。

目标的采集方法取决于目标的特定类型 ,图 6-16 表示了有向点和范围线目标的采集过程。表 6-2 为几种主要目标类型的采集方法 ,其中基准线代表目标符号化时的配置控制线。更多的类型在此不再一一介绍。

表 6-2 地图目标采集方法

几何类型	目标类型	目标采集方法
点目标	标准点目标	定位点
	有向点目标	定位点 + 方向点
线目标	标准线目标	中心线点集
	有向线目标	基准线点集 ,顾及采集方向
	范围线目标	基准线 + 范围线
面目标	标准面目标	轮廓线
	散列型面目标	轮廓线

目标空间数据采集的具体流程如图 6-19 所示。

(5) 目标保存

图层中各目标并非是孤立存在的 ,如线目标和面目标 ,需要通过拓扑关系将各目标结合起来。因此 ,采集获取的地图目标空间数据被临时存放到地图数据库中 ,直到整个图层目标都采集结束。然后将图层中的所有目标进行综合处理 ,建立节点表、弧段表、目标-弧段表以及各种辅助文件 ,实现目标间的数据关联。一旦图层中的目标或目标的空间数据发生变化 ,

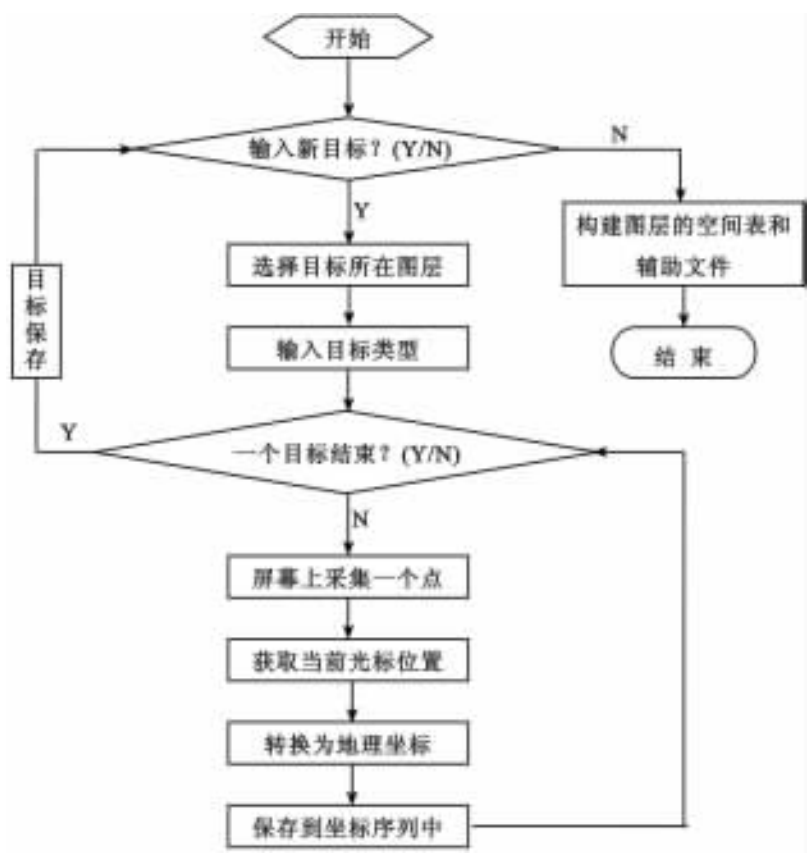


图 6-19 地图目标空间数据采集流程图

就需要重新生成各空间数据表和辅助文件。

采集的非空间数据被直接保存到关系数据库中。

6. 地图数据处理模块

采集并保存的地图数据常常需要修改、更新或再应用。对于地图数据,新的应用可能提出新的要求,从而需要对原有地图数据进行相应的变换处理,如坐标变换、投影变换、制图综合、地图图幅的合并与分割,等等。地图数据处理的功能模块包括坐标变换模块、投影变换模块、编辑修改模块、制图综合模块和地图接边模块等。

以坐标变换模块为例。坐标变换是将原图形的坐标 (x, y) 转换为新的图形坐标 (x', y') 的过程,例如把数字化采集的地图目标的设备坐标转换为地理坐标,或者对地图数据进行坐标纠正,以获得更准确的地理数据。该过程步骤如下:

(1)控制点的建立

允许用户通过屏幕确定三个以上的控制点,获得其当前坐标值,并给出控制点的新坐标,即理论坐标值。控制点可以选择地图的内图廓点、方里网或经纬网的交点,以及通过实地测量建立的点。

(2)坐标变换关系的建立

可以利用控制点的已知新、旧坐标通过仿射变换建立坐标转换公式。对于仿射变换,只需知道不在同一条直线上的三个控制点就可以计算待定系数。因此,如果控制点仅为三个点,经过误差分析,达到精度要求,可以直接建立转换关系。但在实际应用中,为了提高变换精度,往往允许采集的控制点数 n 超过 3。于是就有两种建立方式。一是通过最小二乘法计算仿射变换参数,仿射变换方程为:

$$\begin{cases} x' = a_1 x + b_1 y + c_1 \\ y' = a_2 x + b_2 y + c_2 \end{cases}$$

设转换前坐标为 (x, y) , 经仿射变换后的坐标为 (x', y') , 则误差方程为:

$$\begin{cases} P_x = x' - (a_1 x + b_1 y + c_1) \\ P_y = y' - (a_2 x + b_2 y + c_2) \end{cases}$$

由 P_x^2 最小和 P_y^2 最小的条件可得到两组方程:

$$\begin{cases} a_1 \sum x + b_1 \sum y + c_1 n = \sum x' \\ a_1 \sum x^2 + b_1 \sum xy + c_1 \sum x = \sum xx' \\ a_1 \sum xy + b_1 \sum y^2 + c_1 \sum y = \sum yx' \end{cases} \quad \begin{cases} a_2 \sum x + b_2 \sum y + c_2 n = \sum y' \\ a_2 \sum x^2 + b_2 \sum xy + c_2 \sum x = \sum xy' \\ a_2 \sum xy + b_2 \sum y^2 + c_2 \sum y = \sum yy' \end{cases}$$

式中 n 为控制点个数。解上述方程就可求得仿射变换的待定系数 $a_1, b_1, c_1, a_2, b_2, c_2$ 。

二是假定控制点的采集精度能够得到保证的前提下,可以采用分块仿射变换方法,将误差控制在分块范围内。图 6-20(a)表示了一种最简单的分块仿射变换方法——双仿射 4 点变换,该方法能够保证边界的拓扑一致性。由此推广,分块仿射变换方法的关键之处在于利用控制点构建三角网(TIN),对每一个三角形建立一个仿射变换关系,并计算出各自独立的变换系数(如图 6-20(b)所示)。

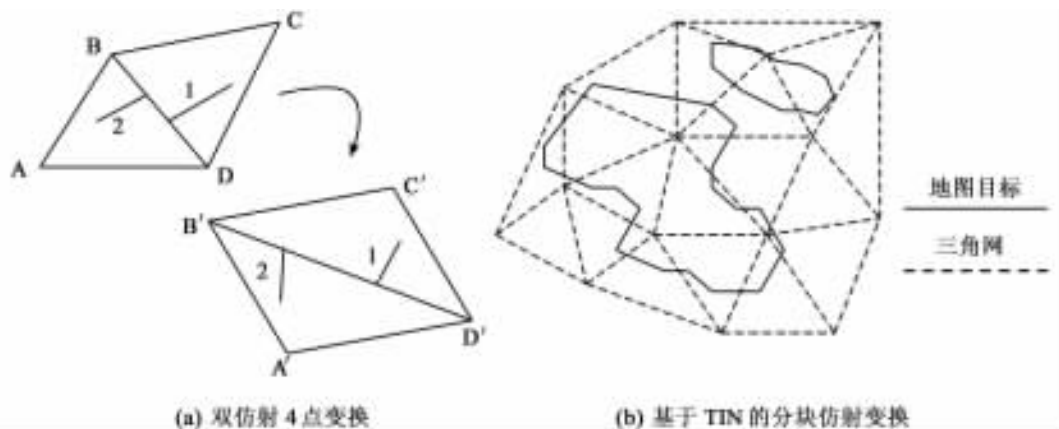


图 6-20 分块仿射变换方法示例

(3) 坐标变换的实施

在这一过程中,主要应用建立的仿射变换公式对所有采集的地图目标进行坐标转换。在分块仿射变换中,需要先判断地图目标内部每一个构成点所在的三角形,然后再应用相应的变换系数对该点实施变换。

从图 6-20 中可以看出,三角形内的点位在变换后仍然保持在该三角形区域的对应位置,从而保证了转换精度。这种方法适用于扫描底图出现非均匀变形,如局部折皱的情况。

7. 地图制图输出模块的设计

地图制图输出主要依赖于地图符号库子系统的支持。该模块首先从地图数据库中依次取出一个图层,判断图层类型,然后在图层内再依次取出每一个地图目标,判断目标的分类码,将图层类型以及地图目标的分类码、空间数据和相关专题数据(如果为专题地图符号)一起传递给地图符号化模块。在符号化模块中,根据图层类型确定符号应该取基础地图符号还是专题地图符号,利用目标分类码搜索地图符号库中的相应符号信息,将符号数据和地图目标数据一起作为参数传递给不同类型的地图符号化函数,实施符号化表达。地图符号化流程如图 6-21 所示。

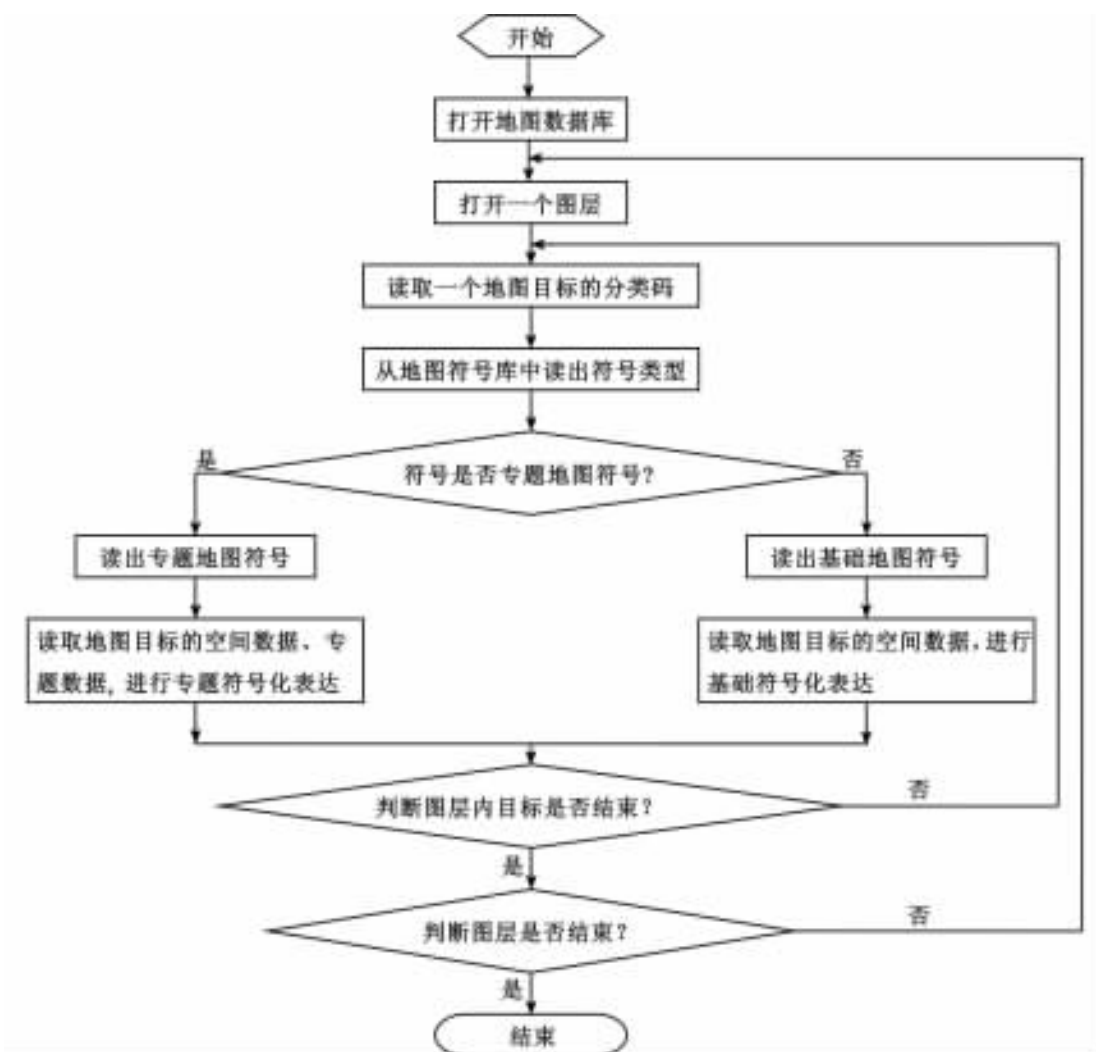


图 6-21 地图符号化流程图

6.2.3 系统的软件实现

CAC 系统软件必须通过程序编制实现。程序编制简称编程,是软件开发的核心工作,其任务就是应用编程语言把已设计的各个模块转变为计算机能够接受的代码。

1. 编程的前期工作

在编写程序前首先应完成系统的详细设计,也称为“过程设计”。详细设计本来属于系统设计过程,但在这里另行提出是因为详细设计是直接面对更具体的程序流程而进行的设计,它们之间的联系更加紧密,甚至在很多时候它们的过程相互渗透。详细设计并不是具体地编程序,而是将各个模块细化成很容易从中产生程序的“图纸”。结构化程序设计(SP)是其中的关键技术之一,其基本思想是按照自上而下逐步细化的方式用三种基本结构(顺序结构、选择结构和循环结构)反复嵌套来实现一个程序。这是一种基于过程的程序设计方法,产生的程序简单清晰。结构化程序由许多块组成,每块只有一个入口和一个出口,程序中一般没有 goto 语句。结构化程序易于阅读、测试、修改和验证。

系统界面设计也是详细设计的重要内容之一。作为人机交互的接口,系统界面帮助用户操作软件系统,帮助系统向用户反馈运行信息、结果。一个设计良好的系统界面将有利于用户更好地掌握软件的应用方法,最大限度地发挥软件的效能。要设计一个成功的用户界面,必须以用户为中心,充分了解系统流程,保证系统应用的灵活性、稳定性和集成性,避免过于花哨的界面、模棱两可的提示、迟缓的信息反馈和与界面信息不相符的操作。系统界面设计的过程包括用户特性分析、用户工作分析、记录用户有关系统的概念和术语、确定用户界面类型 4 个步骤。其中前两项工作是为了充分了解系统用户的特点和 workflows,以确保系统界面真正方便用户的使用,提高系统应用效率。

2. 软件开发平台的选择

软件开发平台不仅仅指软件的编程语言,还包括相关的开发环境,如编程语言的版本、支持的库函数和资源、数据存储模式等。选择软件开发平台要考虑到系统开发的任务和需求。CAC 系统对软件开发平台的基本要求包括:

- 具有灵活方便的编程环境。CAC 系统的开发涉及数据组织、处理、表达与用户界面构建等诸多方面,一个灵活、应用方便的编程环境可以大大提高编程效率。
- 支持大数据量数据的处理。CAC 系统处理的数据具有类型多、数量大的特点,考虑的数据对象往往跨数十、数百甚至更多的图幅,并且在单个图幅也拥有众多的地图目标,因此要求编程语言能够允许大数据量的存储与管理。
- 具有构建复杂数据结构的能力。地图目标及目标之间关系的复杂性,需要编程语言环境能够表示和处理复杂的数据结构。
- 支持图形数据的运算、表达。图形运算与表达是 CAC 系统的基本要求,这是选择开发平台的主要指标之一。
- 具有良好的软件可移植性、可维护性和可扩展性。

目前人们使用的各种编程语言都有自身的不同特征。FORTRAN、Pascal、C、PL/1 拥有大量的标准库函数,有助于进行各种复杂的数值计算;COBOL、SQL 能够方便地进行数据处理与数据库应用;汇编语言、C、Pascal、Ada 适用于系统软件的开发;Lisp 和 Prolog 在知识库系统、专家系统、推理工程、模式识别等人工智能应用领域具有很好的应用前景。此外,随着

计算机软硬件技术的发展,编程语言也不断推出新的版本,功能不断完善。

在这些编程语言之中,C语言是较适合CAC系统开发的编程语言之一。它的优势主要表现在:

- C是一门介于高级语言和低级语言之间的编程语言,同时具有两者的优势。
- C是结构式语言,它将代码与数据分隔开。
- C语言简洁、紧凑,使用方便、灵活。
- 运算符丰富。C语言共有34种运算符。
- 数据结构丰富,可以实现结构类型(struct)、联合类型(union)和链式结构,便于数据的描述与存储。
- C语言允许直接访问物理地址和进行位操作。
- 生成目标代码质量高,程序执行效率高。
- 用C语言写的程序可移植性好,适合于多种操作系统,如DOS、UNIX,也适用于多种机型。

其中作为C语言开发平台之一的Turbo C是美国Borland公司开发的基于DOS环境的产品,也是目前GIS相关专业学习计算机图形学、数据结构等课程的主要编程语言。Turbo C进一步拓展了强大的图形库和文本窗口函数库,并和汇编语言以及BASIC、FORTRAN、Pascal等高级语言都有相互接口,可以实现对任意*.dbf文件的访问。因此,下面以Turbo C 2.0为例来进行进一步的讨论。

3. 程序编写的基本要求

在CAC系统的开发中,由于系统功能多,内容复杂,一般需要安排若干人同时参加程序编写,所以需要更加强调程序的可读性和可重用性。一般认为好的程序应具备的特点是:能够完成系统设计目标;调试代价低;易于维护;易于修改;设计不复杂;效率高等。

4. 系统界面的实现

(1) CAC系统界面的一般形式

目前系统界面主要有以下三种基本形式:

① 基于命令的界面。这种界面下,用户通过键盘输入命令语句驱动系统运行,早期的Arc/Info系统就属于这种类型。这种界面简单易行,容易实现,但需要用户非常熟悉各个命令语句的格式和参数,因此对于操作种类多、过程复杂的系统应用较为不便。

② 基于菜单驱动的界面。这是目前最普遍的一种界面类型。它按照层次,列出了系统提供的所有操作,用户可以通过键盘或鼠标来选择并执行一个操作。这种界面友好,用户容易掌握,开发较易实现,但对于一个需要进行批量数据处理或者过程复杂的运行任务,反复进行菜单操作就变得不可忍受。

③ 基于数据流图的图形用户界面。在图形数据流界面中,系统将工作流程转化为数据流图,数据、设备和操作都用图形符号(又称为“图标”)来表示,通过鼠标点击或者拖放图标来执行对应的操作。一组有序图标完成一项完整的工作任务。这种界面适合于数据流清晰、过程相对单一的系统。一旦流程确定下来,用户必须严格按照步骤执行(如图6-22所示)。

由于CAC系统的处理对象类型多,数量大,操作过程针对不同的目标并不能完全统一,其用户界面通常采用菜单方式。菜单项是菜单的最小单元。若干具有相近功能或者一定联

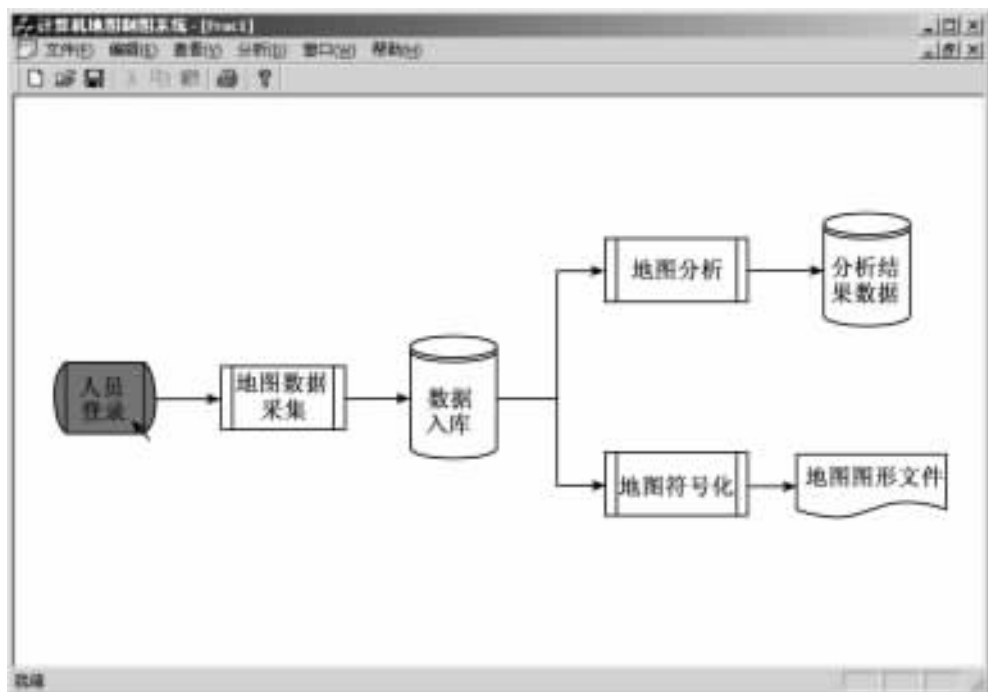


图 6-22 采用数据流图的图形用户界面

系的菜单项被归并在一起,称为一个菜单项组。当一个菜单项组同时显示在屏幕上的一个矩形区域内时,就称为菜单栏。菜单由各级菜单项组构成,最高一级菜单项组称为主菜单,主菜单的子菜单统称为一级子菜单,一级子菜单的子菜单统称为二级子菜单,依此类推。其结构表现为典型的树结构形式(如图 6-23 所示)。

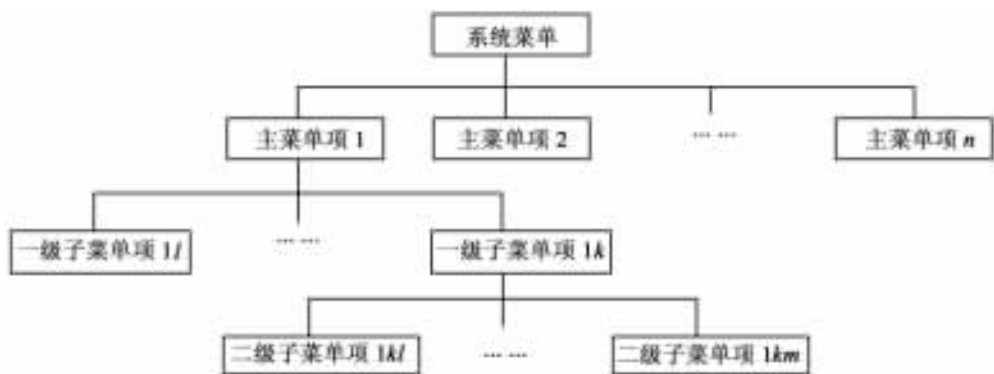


图 6-23 菜单结构层次示意图

没有子菜单的菜单项称为叶子菜单项。由于叶子菜单项直接关联到某一个可执行的功能,又可称为功能菜单项。

(2) 鼠标的应用

菜单驱动的界面往往需要鼠标支持。在 Turbo C 环境下,系统可以自行开发鼠标信息函数,主要利用 33H 中断的 3 号子功能调用,不需要任何参数,便可以直接获取鼠标信息,包括当前鼠标的按键(左、中、右键)状态和位置(X,Y),位置坐标以像素为单位计算。在目前出版的大量 Turbo C 编程方面的书籍中均有详细的应用说明。

(3) 菜单库的结构与应用

由于 CAC 系统的菜单项层次可达 3~4 层,且数量多,采用通过程序固定实现的方法,往往存在程序代码冗长且重复性大的问题。为了增加菜单的灵活性,系统宜采用菜单自动生成的方法。针对这一要求,菜单的界面实现除了图形上的表达外,还需要菜单库的支持。菜单库存储各个菜单项的位置关系,如果菜单项为功能菜单项,则还需要存储要执行的功能编码。其结构体如下:

```
#define MAXMENUNAMESIZE 20
typedef struct MenuItem
{
    int Menu_ID;           //菜单项标志码
    char Menu_name[ MAXMENUNAMESIZE ];    //菜单项名称
    int Topmenu_ID;        //父菜单项标志码
    int FirstSubmenu_ID;   //第一个子菜单项标志码
    int Submenu_num;       //子菜单项数
    int Func_ID;           //执行功能编码
} MenuItem;
```

注:非功能菜单项的执行功能编码为-1,而功能菜单项的标志为 Submenu_num,其值为 0,主菜单项的父菜单项标志码为 0。

菜单项标志码 ID 从 1 开始,采用顺序编码。在菜单库中通过父菜单项 ID 和第一个子菜单项 ID 可以实现各菜单项的上下关联。然而对菜单栏来说,在屏幕上的显示区域是通过计算得到的。为了提高显示效率,在系统运行过程中需要临时保存这些信息,因此应针对菜单栏专门开辟一个结构体,其结构如下:

```
#define MAXMENUITEMNUM 50           //一个菜单栏允许的最大菜单项数
typedef struct MenuBar
{
    int MenuBarType;                //菜单栏类型:0 主菜单(横向),1 子菜单(纵向)
    MenuItem curMenuItem[ MAXMENUITEMNUM ];    //所有菜单项信息
    int left, right, top, bottom;    //菜单栏显示的左、右、上、下边界
    int maxlength;                  //菜单栏中菜单项字符的最大个数
} MenuBar;
```

菜单库的应用包括菜单的表达、菜单响应与菜单功能执行三个部分。

菜单表达需要系统首先确定主菜单在屏幕上的显示区域,以及单个菜单项显示的尺寸控制参数、上下左右菜单项之间的控制间距,并且开辟一个 MenuBar 类型数组 CurMenuBar

和计数器 numBar(初始化为 0),专门存放处于打开状态的菜单栏信息。每激活一个新的菜单栏,就将菜单栏信息存入 CurMenuBar 数组中,计数器 numBar 加 1;同样,当返回到上一级菜单时,只需要计数器减 1,就等同于将该菜单栏从数组中清除掉了。显然,保存在 CurMenuBar 中的菜单栏应互为父子关系,且层次越高越在数组的底部。

在通常情况下,主菜单栏是一个向水平方向延伸的菜单栏。系统从菜单库中搜索所有的主菜单项,将其各项属性值保存在已开辟的菜单栏数组元素 CurMenuBar[0]中,进行主菜单的布局运算,保证主菜单栏在规定的显示区域内居中、对称,各菜单项及其文字均匀、吻合。在已知主菜单栏范围和单个菜单项尺寸的条件下,通过菜单项在主菜单中的顺序可以推算出其显示范围;反之,也可以通过指定的屏幕位置,推算其所在菜单项。

系统运行过程中,当鼠标在某一个主菜单项上点击时,系统可以通过鼠标信息函数获取鼠标的按键状态和屏幕位置信息,从而推算出主菜单项序号 seq,由 CurMenuBar[0]的菜单项信息可以得到其标志码:

```
Menu_ID = CurMenuBar[ numBar ]. CurMenuItem[ seq ]. Menu_ID ;           //主菜单栏时
numBar = 0
```

然后得到该菜单项显示区域 rect,判断其子菜单项数 Submenu_num 是否为 0。如果为 0,说明该菜单项为功能菜单项,根据其执行功能编码 Func_ID 在程序中执行相应操作,同时关闭除主菜单栏的所有子菜单栏,并将 numBar 清 0;如果不为 0,由标志码 Menu_ID 可以从符号库中搜索其所有的子菜单项,保存在 CurMenuBar 数组中,numBar 加 1。子菜单栏是一个纵向延伸的菜单栏。根据前面父菜单项的显示区域 rect 计算当前激活的子菜单栏的起始位置,根据已知的子菜单项数计算子菜单栏的高度,根据子菜单项数组的最大字符数计算子菜单栏的宽度,由此可以显示子菜单栏。与主菜单栏一样,子菜单栏也可以建立各子菜单项与其显示区域的联系。进一步继续上述过程,直到选择某一个功能菜单项为止。

如果鼠标点击的屏幕位置不在当前激活的子菜单栏区域,则关闭子菜单栏,numBar 减 1,同时判断是否落在上一级父菜单栏中。如果不是,则重复该清除过程,直到回到主菜单栏。

(4) 菜单的显示

菜单的显示实际上可分解为菜单边框绘制、填充与字符显示三个部分,这是 Turbo C 的基础知识,在此不再重复。只是在键盘光标或鼠标移动到某一个菜单项时,需要恢复前一个选择菜单项的填充和字符颜色,同时对当前选择菜单项做改变填充颜色和字符颜色的处理,增加界面的用户响应效果。

图 6-24 为一个 CAC 系统菜单界面的例子。

5. 图层和地图工程的结构

图层是构建地图的基本层次单元,它包含一组地图目标及其相关属性。图层的相关属性在图层建立时设置。在前面已经详细分析了图层的分类性质,考虑到单一类型与混合类型的统一,其基本结构可以如下所示:

```
#define MAXLAYERNAME_SIZE 50
typedef struct MapLayer
{
    char LayerName[ MAXLAYERNAME_SIZE ];           //图层名
```



(a)



(b)

图 6-24 CAC 系统的菜单界面(Turbo C 环境开发)

```
unsigned char LayerType ;  
long LayerCode ;
```

```
//图层类型 具体内容见后  
//图层编码 只有当图层类型为  
单一型时有效 否则为 - 1
```

```

long ObjectNum ;                //图层现有地图目标数
int pointcolor , linecolor , fillcolor ;    //在制图表达前图层默认的临时点、线画和填充颜色 0~255

} MapLayer ;

```

图层的类型划分有多种方式,可以分别定义成员变量类型来保存,但也可以尽量控制在一个变量中。在上面的结构体类型 MapLayer 中,图层类型 LayerType 为单字节无符号整数,取值范围 0~255。其中百位数最大值为 2,用于表示图层的目标类型组合情况,0 和 1 分别表示单一型和混合型。十位数表示图层的基本性质,用 0、1、2、3 分别表示基础图层、专题图层、注记图层和栅格图层。个位数表示图层中目标所具有的几何性质。由于混合型图层中可以同时存放不同几何属性的目标,要能够既保存单一型目标,又可以保存混合型目标,只能采用位平面方式,用 $1 (=2^0)$ 、 $2 (=2^1)$ 和 $4 (=2^2)$ 分别代表图层中存在点、线、面目标,从而可以利用简单相加表示混合的目标类型。如个位数为 6,说明该图层中有线目标和面目标,因为 $6 = 2^1 + 2^2$,指数 1 和 2 分别代表线和面。个位数最大值不超过 7,等于 7 时表示一个图层中同时包含了点、线、面目标。

只用一个字节,就保存了多种图层类型信息,使得数据结构紧凑,节省空间。但在应用上需要对其进行分解。例如,类型值 LayerType 为 105,则

```

int i , j , k , k1 , k2 , k3
i = LayerType / 100 ;
j = ( LayerType - i * 100 ) / 10 ;
k = ( LayerType - i * 100 - j * 10 ) ;
k1 = k % 2 ;    k2 = ( k / 2 ) % 2 ;    k3 = ( k / 4 ) % 2 ;

```

i、j、k 分别表示变量 LayerType 分解得到的百位数、十位数和个位数的值,结果为 1、0、5,k1、k2、k3 是将个位数值 k 转换为二进制的低三位值,结果为 1、0、1,说明该图层为混合型的基础图层,图层中存放了点目标和面目标。

从广义上讲,地图工程是指一幅地图从采集到制作的全部过程。但从内容上,一幅地图通常由一组图层按照一定次序叠加生成。因此,在系统中我们采用一个地图工程 (Map-Project) 记录一幅地图绘制所需要的图层构成,包括这些图层的名称、属性和顺序等信息。

```

#define MAXLAYERNUM 200
#define MAXNAMELEN 100
typedef struct MapProject    //地图工程结构类型
{
    char MapName[ MAXNAMELEN ] ;    //图名
    MapLayer mapLayer[ MAXLAYERNUM ] ;    //图层数组
    int LayerNum ;    //图层数
    unsigned char mapAttr[ MAXLAYERNUM ] ;    //图层属性数组
    int CurLayerNo ;    //当前图层序号
} MapProject ;

```

无符号字符型数组 mapAttr 存放各地图图层的属性,即可编辑性和可显示性。其中,十位数为 0、1,分别表示可编辑和不可编辑;个位数为 0、1,分别表示可显示和不可显示;另一个成员变量 CurLayerNo 记录当前处理图层的序号。

在一个计算机地图制图任务中,地图工程成为重要的入口信息,由它可以进一步调用图层,再由图层可以调用地图目标,从而实现地图分析与表达的目的。

6. 扫描图像数据的导入功能实现

地图扫描后获取的图像保存为 bmp、pcx、tiff 等图像文件形式,可以作为特殊的栅格图层导入到一个地图工程中,并作为地图数据采集的参考底图。

图像文件一般都包括图像文件头和图像数据两个部分。文件头是一个约定好的数据结构,它包含图像的长、宽以及解压缩程序可识别这类图像的特定标志,帮助程序实现对图像数据的读取。每一种图像文件都有自己特定的图像格式,掌握了这些格式就能利用 Turbo C 很方便地读取并显示图像文件。

对于大的图像文件,由于受到 DOS 环境和 Turbo C 内存管理模式的限制,需要对图像数据进行分块处理。分块处理有两种方式:一是硬分块方式,即利用通用的图像处理软件将一个大的图像文件分割成若干小的图像文件,并以原文件名加上行列编号对得到的一组小图像文件重新命名,以便系统每次只是处理少量的图像数据;二是软分块方式,主要是在系统中直接编写程序对图像数据进行分块读写、处理和显示,不破坏图像文件的完整性。此外,利用硬盘作为虚拟内存可以一次性对整个大的图像文件进行运算,但缺点是处理效率低,速度慢。

读入的地图图像数据要成为参考底图,必须经过坐标匹配,建立栅格的行列坐标与地理坐标之间的转换关系。采用的步骤见前面的仿射变换和分块仿射变换方法(见图 6-20)。

7. 地图目标的采集与保存功能实现

(1) 地图目标的采集模块

地图目标采集模块的功能主要包括点、线、面目标和注记的采集(如图 6-25 所示)。采集前首先指定当前工作图层,将该图层序号存放在地图工程的 CurLayerNo 中。图层从已建立和打开的图层中选择,采集的目标将存放在当前工作图层中。

为了保证地图采集的质量和精度,采集过程首先需要地图浏览功能的支持。地图浏览功能包括对地图的缩放、漫游等操作,主要起到调整所显示地图的区域、尺寸的作用,以便于用户能够用最佳的视野观察感兴趣的地图目标。缩放和漫游主要通过不断调整地图坐标与屏幕坐标的对照关系来实现(如图 6-26 所示)。

地图采集的基本功能分为空间数据采集和非空间(属性)数据采集两个部分。空间数据采集主要通过捕捉键盘和鼠标的按键信息及其所控制的光标位置来获取屏幕坐标,再转换为地图的坐标。一般情况下需要规定不同的键盘、鼠标操作功能,例如规定鼠标左键单击一次采集一个点;Alt 键加上鼠标左键单击可以直接咬合距离最近的已建立的空间点,并以此为新增加点,右键单击则结束当前目标采集等。如果目标为面目标,右键单击后应自动将起始点复制作为终止点,以确保目标弧段封闭。非空间数据采集可以利用 C 语言直接访问 dbf 数据库文件,首先为当前图层建立一个包含多个属性项的关系表,然后针对图层内的每个地图目标通过键盘输入属性值,也可以通过其他数据文件转换到目标数据库中。

(2) 地图目标的存储模块

每个图层都对应着一系列文件,包括节点表(*.nod)、弧段表(*.arc)、点目标文件(*.pob)、线面目标文件(*.lpb)、目标属性文件(*.att)、目标索引文件(*.ind)和目标栅格索引(*.gid)等。这些不同类型的文件除了扩展名不同外,都采用其图层名作为统一



图 6-25 地图目标采集功能菜单

的文件名。不过,各种类型的文件结构在前面章节中已经较详细地阐述了,在此不再赘述,仅以点目标文件和线面目标文件为例进行说明。

点目标的类型包括标准点和有向点。标准点不考虑地图目标的方向,只需要一个定位点就能满足要求,而有向点则要采集两个点(第一点为定位点,第二点为方向指示点)。因此,点目标的信息存放在以下结构体类型中:

```
typedef struct NodeXY{ double x ;double y ;} NodeXY ;      //节点坐标类型
typedef struct PointObject      //点目标结构类型
{
    long ObjectID ;           //目标标志码
    long ObjectCode ;         //目标分类码
    unsigned char ObjectType ; //目标类型 0 为标准点 1 为有向点
    NodeXY node[2] ;          //目标节点坐标 :ObjectType == 0 ,node[0]有效 ;
                                //ObjectType == 1 ,两个点都有
    效
} PointObject ;
```

由于一个点目标保存的空间坐标数量随目标类型(标准点或有向点)而变化,但最大不超过两个,因此点目标文件既可以保存为变长记录的文件,也可以保存为固定长度记录的文件。

线目标和面目标均由若干弧段构成,弧段数量差异较大,且范围线目标需要表示为基准线和范围线两个部分。其结构类型如下:



注 图中粗黑线为采集的地图目标。

图 6-26 地图的缩放、漫游功能

```
#define MAXARCNUM 500 //一个目标的最大弧段数
typedef struct LinePolyObject //线、面目标结构类型
{
    long ObjectID ; //目标标志码
    long ObjectCode ; //目标分类码
    unsigned char ObjectType ; //目标类型 ,具体内容见表 6-3
    int ArcNum ; //弧段数
    long Arc_ID[ MAXARCNUM ] ; //弧段的标志码数组 ,对应到弧段表中的相应弧段
} PointObject ;
```

与其他类型不同 ,当目标类型为范围线目标时 ,基准线的 Arc_ID 表示为正值 ,范围线的 Arc_ID 自动在前面添加一个负号 ,改为负值 ,以区别范围线目标的不同线段类型。图6-27中陡崖 A 为范围线目标 ,其弧段编号依次为 {a , b , -c , -d}。其他类型目标的 Arc_ID 均为正值。

线、面目标文件可保存为如下格式：

ObjectID ,ObjectCode

ObjectType ,ArcNum //假设 ArcNum 为 n
Arc_ID1
Arc_ID2
...
Arc_IDn

表 6-3 目标类型 ObjectType 的意义

ObjectType (1 byte)	十位数		个位数	
	1	线目标	1	标准线目标
			2	有向线目标
			3	范围线目标
			4	渐变线目标
	2	面目标	1	标准面目标
			2	散列型面目标

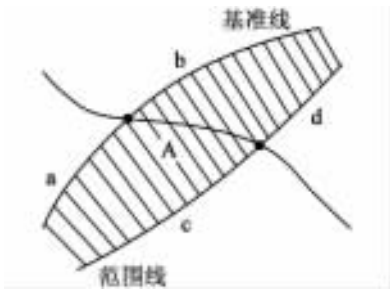


图 6-27 范围线目标的弧段构成

8. 地图目标的编辑功能实现

地图目标采集过程中 ,由于硬件设备、软件系统和人为的诸多因素 ,可能会出现各种各样的采集错误 ,修改这些错误需要地图目标的编辑功能支持。目标编辑主要是在地图浏览的基础上对地图目标的空间数据和属性数据进行增加、修改、删除处理 ,包括目标处理和节点处理两个方面。

(1)目标处理

目标处理是针对一个完整的地图目标或目标集合而进行的操作 ,如选择目标(集合) ,删除目标(集合) ,修改目标(集合)的属性 ,对目标(集合)进行移位、旋转、缩放、合并、拆分、复制等处理。在处理目标之前 ,首先要选定欲编辑的目标 ,其步骤如下：

第一步 ,判断是否单击了鼠标左键 ,是则获取当前光标位置 ,转换为地图坐标。

第二步 ,计算光标所在的栅格 ,从当前图层的栅格索引文件(*. gid)中取出该栅格中所有的地图目标集合{ ID1 , ID2 , ... , IDn }。

第三步 根据目标 ID 可以快速从目标索引文件(*. ind)中找到目标在主体文件(点目标文件或线、面目标文件)中的起始位置 ,从而读出目标的空间数据。

第四步 ,从选择的地图目标集合中搜索离该光标点距离最近的目标 ,返回结果目标的 ID 值。

然后选定的目标被高亮显示 接着进行编辑处理。编辑处理的内容不同 ,需要得到的参数也不同。如目标的旋转需要获取旋转角度 ,目标的移位和复制需要获取目标在 X,Y 方向上的移动距离 ,目标的缩放需要获取缩放的倍率 ,等等。这些参数都可以由鼠标或键盘输入得到。最后在地图数据库中保存修改确认后的地图目标数据。图 6-28 是地图目标选择、旋转的例子。

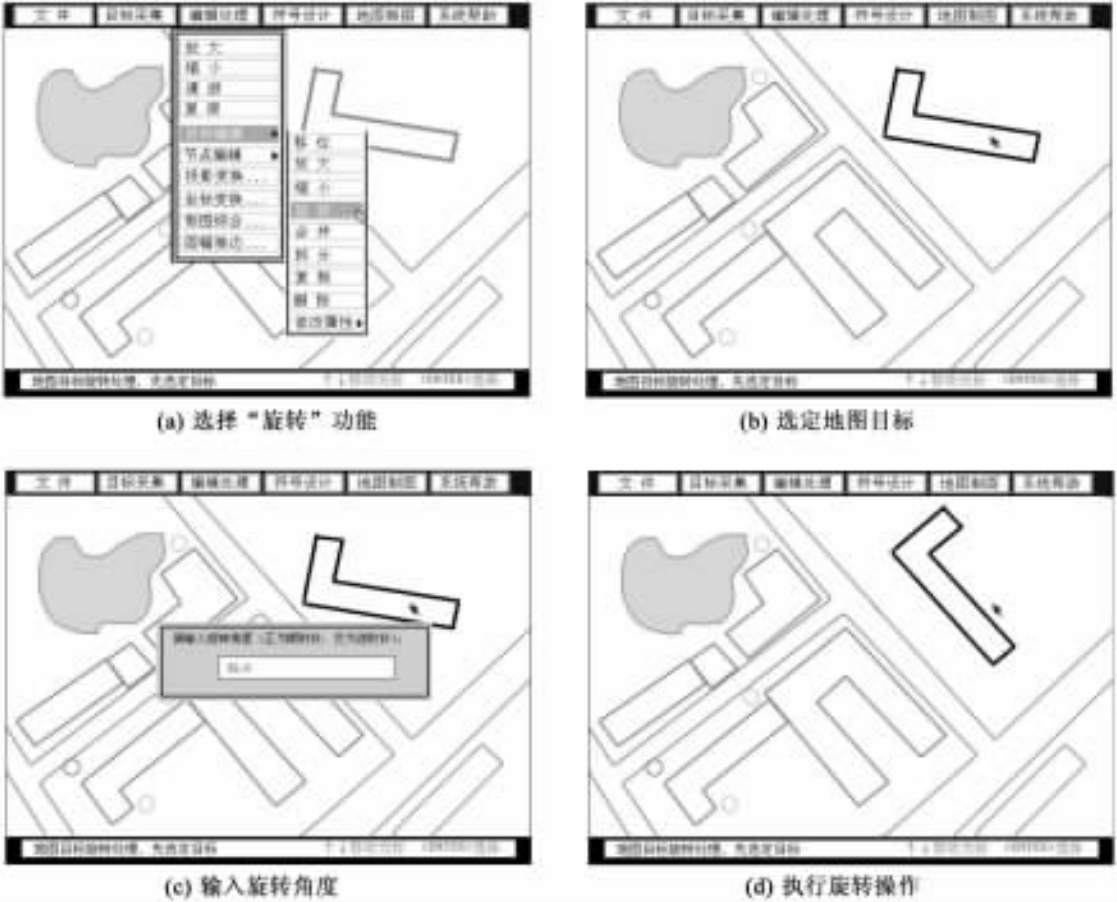


图 6-28 地图目标的旋转处理

(2)节点处理

节点处理是针对一个地图目标内部的构成节点进行的操作 ,其目的是修改地图目标的空间形态 ,包括节点的增加、删除、移位等。处理步骤如下：

- 第一步 通过鼠标选择欲编辑的地图目标 ,获取目标 ID 值。
- 第二步 ,由目标 ID 值 ,通过目标索引文件和主体文件 ,读出该目标的空间数据 ,即各节

点的地图坐标,将地图坐标转换为屏幕坐标,并显示所有的目标节点。

第三步 通过鼠标选择欲编辑的目标节点,进行节点处理。

第四步 保存修改后的地图目标数据到地图数据库中。

在上述的第三步操作中,目标节点的选定是通过将得到的鼠标光标位置的屏幕坐标转换为地图坐标后,和所选地图目标的所有节点进行比较,寻找距离值最小的节点来实现的。与目标处理一样,不同的节点处理需要得到的参数不同,如节点的移位需要新的位置点。增删节点往往导致目标节点的顺序发生变化,需要后移或前移来重新排列。图 6-29 为一个地图目标节点处理的例子。

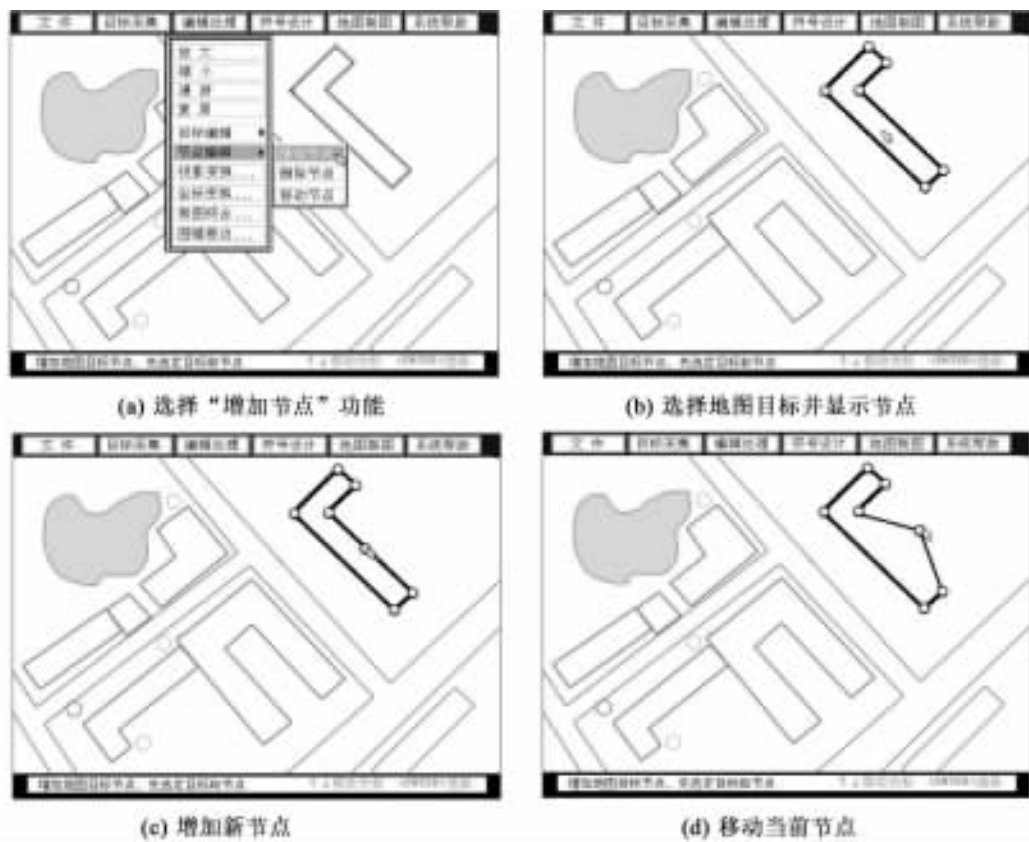


图 6-29 地图目标节点处理示例

9. 地图符号的设计实现

地图符号库文件的扩展名为 sym,其主名与对应的地图工程文件名相同。在一个地图工程中,对地图符号的任何处理均是在该地图符号库中进行的。地图符号库的建立需要符号设计系统的支持,包括普通地图符号的设计和专题地图符号的设计。在当前地图工程中,一个地图目标分类的类型都对应一个地图符号。其中普通地图符号仅由目标分类码来确定,而专题地图符号由专题图层编码、图层属性表中的若干数据项表达式分别对照其符号控制参数来建立。

(1)普通地图符号设计

普通地图符号的设计主要包含新建和打开地图符号、地图符号的删除和复制、图元的增加和删除、图元参数的设置、图元位置和顺序的调整等。其实现步骤如下:

第一步 设定当前欲设计的地图符号分类码。

第二步 选择图元,设置该图元的所有参数(如长度、宽度、半径、颜色、线型、线粗等),并将该图元显示在符号中心区域,可进一步通过鼠标修改该图元的形状、位置。

第三步 继续第二步,直到所有图元都建立完毕为止。

第四步 将已建立的符号保存到地图符号库中。

符号的空间定位采用各自独立的平面坐标体系。为了将地图数据库中符号的空间坐标都保持为整数形式,其内部存储时尺寸大小以 0.01mm 为单位。针对点、线、面符号的具体坐标系构建方法在前面已经论述过,这里不再赘述。此外,图元的修改首先必须进行图元选取。和地图目标的选取一样,该过程通过搜索离当前光标位置最近的图元来得到,以便对被选图元进行删除、修改等操作。图 6-30 是铁路符号的设计与建立过程。

(2)专题地图符号设计

专题地图符号的设计需要将符号参数与专题属性表的属性项关联起来。在设计专题地图符号之前,首先建立专题图层和图层属性表(包含若干属性项),然后设定该图层对应的专题符号类型、图元、图元的控制参数及其所对应的属性项表达式。其中,专题符号类型反映了专题符号表达的基本特性,包括定点专题符号、线状专题符号、等值线、点密度、定位图表、分区统计图表、分级统计图和特殊专题符号等类型。图元和普通地图符号的图元基本相同,由点、线、面三类图元构成,其中点图元种类最多,包括圆、椭圆、正多边形、矩形、菱形、三角形、扇形等。图元控制参数是专题符号中可以随着专题数据变化而变化的可变量。一般情况下,定量数据通过图元的几何控制参数来区分,如长度、半径;定性数据通过图元的颜色、线型等来表示。不同类型的图元具有不同的控制参数,如:矩形图元为长和宽;圆为半径;扇形为半径、起始角度和终止角度;线为线宽、填充颜色、填充图案、填充方式则是面图元的控制参数。

在此基础上建立图元控制参数与图层的属性项之间的对应关系是专题符号设计的关键。属性项表达式由属性项名和数学运算符(加+、减-、乘*、除/、幂[^]和左右括号())构成,并确定符号图元的控制参数随属性项表达式变化的对应关系,包括属性项值的分级数量与分级界线、变化的相对比率等。此外,对于一些特定的专题表示方法,可以采用特定的参数,如点密度法中需要设定每点代表的统计数量值和点的大小、颜色,等值线需要设定间距、线宽与线画颜色,等等。

图 6-31 为专题地图符号设计的示例。例中 popu78、popu98 分别为属性表中 1978 年和 1998 年的城市人口数,图元(圆)的半径与 1978~1998 年的 20 年间人口平均增长数成正比。当然,随着图元类型的变化,矩形区域内的参数内容也将相应发生改变。

10. 地图制图模块的实现

CAC 系统的最终目标是地图的表达与输出。地图制图模块将地图数据库与地图符号库结合起来,通过调用符号库系统中的符号化函数实现地图显示。

(1)普通地图图层的显示

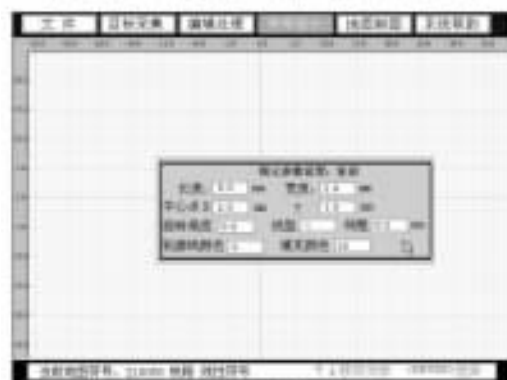
在前面的小节中已经从地图符号化表达的角度分析了普通地图要素的类型,即在传统



(a) 选择“新建符号”功能，输入符号名称等



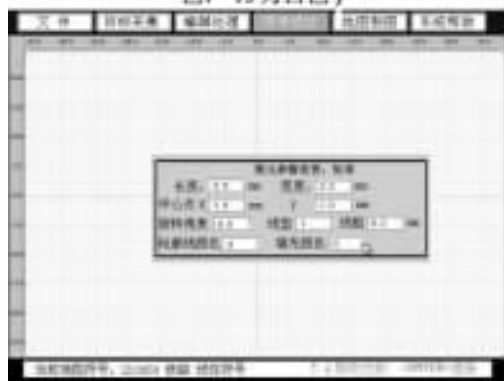
(b) 选择“增加图元”功能



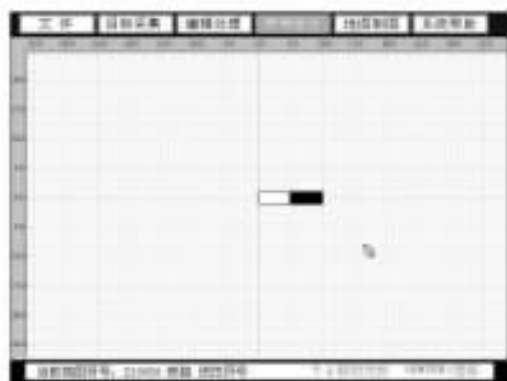
(c) 输入图元参数(颜色值 0 为黑色, 15 为白色)



(d) 建立和显示图元



(e) 输入下一个图元参数



(f) 符号设计完成

图 6-30 普通地图符号——铁路符号设计过程示例

的点、线、面目标基础上进一步进行了细分,如有向点、有向线、范围线等(如图 6-15 所示)。由于不同的类型具有不同的采集方式,因此在其符号化显示时需要传递不同的目标参数。如属于标准点符号的地图目标类型,其符号化的参数最简单,只需要目标定位点和目标分类码;有向点目标类型需要传递两个点坐标,一个是定位点,另一个是方向点;标准线、有向线目标和面目标需要传递一组空间点坐标序列,而范围线需要传递两组点坐标序列,一组为基



图 6-31 专题地图符号的图元参数设计界面

准线,另一组为范围线。这导致了符号化函数调用的复杂化。因此,要将普通地图符号的符号化函数统一起来,可以通过一个地址参数来传递。地址参数可以传递变量的存放地址或数组的起始位置,因此一次可以输入类型相同而数量不同的一组数据,同时并不需要额外开辟更多的存储区间。其函数形式为:

BOOL Symbolize(long ObjectCode,NodeXY * Node,int n,double zx,double zy,NodeXY startpoint)

函数中目标分类码 ObjectCode 是从地图符号库中获取符号信息的桥梁,Node 是目标空间坐标变量的地址,它既可以用来表示一个点,也可以用来表示一组点(坐标数组),具体情况由其后的点数 n 来确定。对于范围线目标,将范围线上的所有坐标自动增加一个负号,转换为负数,从而可以将基准线和范围线两组坐标数据合并在一起传递给函数。zx,zy 分别反映了在 X 和 Y 方向上地图坐标和屏幕坐标的距离比值,即

$zx = dx / DX$; //dx,DX 分别表示在 X 方向上两点间的图上距离与对应的屏幕距离
 $zy = dy / DY$; //dy,DY 分别表示在 Y 方向上两点间的图上距离与对应的屏幕距离
startpoint 表示当前屏幕上地图显示区左下角点的地图坐标,三个值可以帮助实现由地图坐标到屏幕坐标的转换,其转换公式为:

$X = SCREENMINX + (int)((x - startpoint.x) / zx + 0.5)$;
 $Y = SCREENMAXY - (int)((y - startpoint.y) / zy + 0.5)$;

其中,SCREENMINX、SCREENMAXY 分别表示地图显示区左下角的屏幕坐标 X,Y 的值,(x,y)、(X,Y)分别为一个空间点的地图和屏幕坐标。在符号化函数 Symbolize()中传入这三个参数是为了能够直接在函数中实现制图显示,避免因符号化后数据再从函数中传出而造成表达的复杂化。

(2)专题地图图层的显示

与普通地图图层相同,专题地图图层中的地图目标可以是定位点、线或多边形,但同一个专题图层中目标的几何特性应严格保持一致。在普通地图中,属于相同类型的地图目标往往具有基本一致的符号化表示,如相同的颜色、尺寸、填充模式等,而专题地图目标显示过程中,符号的诸多形态因子取决于该目标的属性项表达式值。例如,在如图 6-32 所示的人口专题图层中,不同政区的人口数差异导致了符号——圆的半径变化。

除了顾及由于专题属性值导致的符号变化外,还要考虑符号配置的图面位置。专题的点、线目标的定位与普通地图要素一致,对于表示区域信息的多边形,往往有几种形式,在点

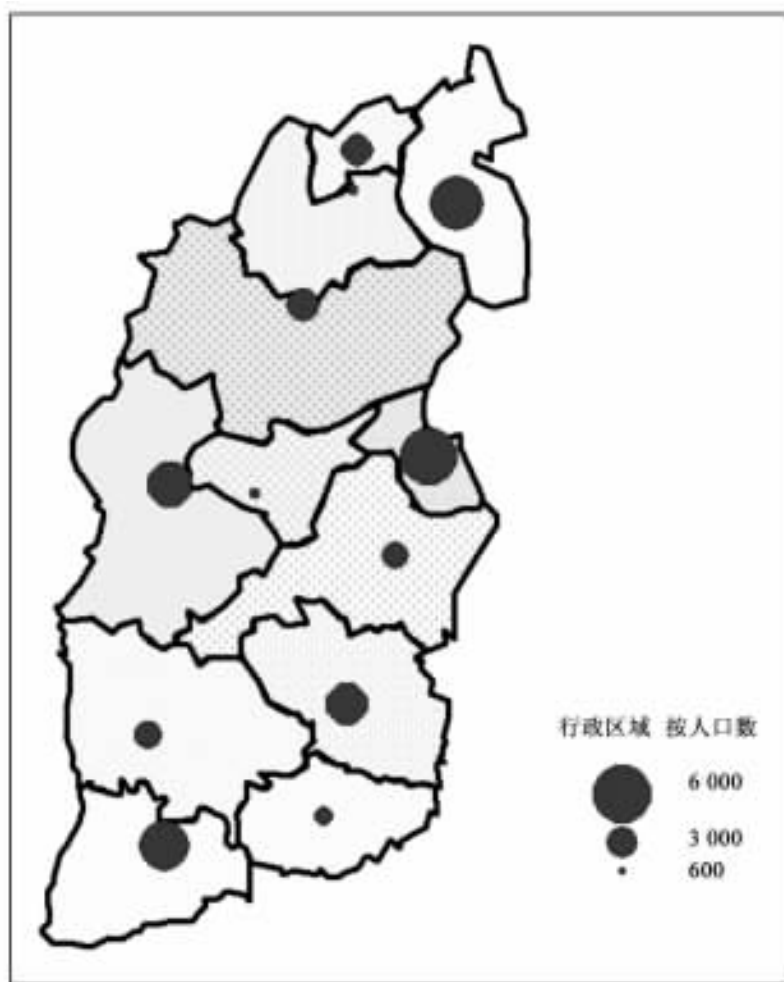


图 6-32 政区人口专题示意图

密度、分级统计图中作为多边形本身来处理。在定位图表法、分区统计图表法中，往往取其中（重）心或局部的中心区域作为符号配置位置。这一过程都由专题地图的符号化函数来自动实现，它根据专题符号的类型来分别处理。

在专题地图的符号化函数中，既要考虑空间坐标信息的输入，也要考虑属性信息的输入。不同的专题符号，其图元控制参数各不相同。因此，要将这些符号统一起来，需要采用和空间数据同样的方法，通过地址参数来实现。由于在专题符号中保存了图元的不同控制参数的属性项表达式，因此在传入属性项值序列的同时也需要对应传入相关的属性项名称字符串序列。其函数形式为：

```
#define MAXITEMNAMELEN 40 //定义属性项名称字符串的最大长度
```

```
BOOL TSymbolize ( long LayerCode , NodeXY * Node , int numNode , char ItemName[ ]  
[ MAXITEMNAMELEN ] , double * ItemVal , int numItem , double zx , double zy , NodeXY start-  
point )
```

其中 ItemName 是属性项名称字符串数组 ,ItemVal 是属性项值数组 ,numItem 是实际的属性项数 ,其余参数意义同普通地图目标的符号化函数。图 6-33 是应用人口和人口密度数据调用该函数得到的直方图显示(注 :人口与人口密度数据为实验数据 ,无实际意义)。



图 6-33

(3)地图的图面配置与图外要素的表示

地图的图面配置是指将与本图相关的各类统计图表、图例乃至图片在一个页面上合理安排布局的过程。这个过程需要 CAC 系统提供进一步的针对整个图层的编辑修改功能 ,可以实现对图层整体的缩放、移位 ,在这里就不再作更详细的阐述。

图廓外要素是指地图上内图廓以外的各种附图、图表、标记和文字说明等 ,包括图廓线、经纬度注记、分度带、方里网、比例尺、图例、坡度尺、地图投影、成图时间、地图编制与出版单位、平面与高程坐标系等。这些也是地图制图不可缺少的重要信息。这些要素中的一部分 ,如图廓线、方里网、经纬度注记等可以直接从多个控制点的坐标中通过计算得到 ,另一部分则需要从键盘中人工录入。

当完成了上述功能后 ,CAC 系统就基本具有了地图的数据建库与制图输出功能 ,可以直接将显示的地图提供给系统支持的打印机打印成图 ,也可以转换为地图成果文件 ,如 TIFF 图像 ,利用激光照排系统制版、印刷。

6.3 面向对象方法在 CAC 系统开发中的应用

对象(Object)是一个属性(数据)集及其操作(行为)的封装体。一个对象就是一个实

际问题论域,一个物理的或逻辑的实体。在计算机程序中,一个对象可视为一个“基本程序模块”,它包含了数据结构及其提供的相关操作功能。因此,面向对象(Object Oriented, OO)方法不是单纯以解决问题的执行过程、功能模块和数据结构为研究的出发点,而是以对象作为解决问题的中心环节,通过对研究系统中涉及的各种事物的分析,将它们抽象为各类对象,构建对象内部的属性与行为,以及对象间的相互联系,并运用消息驱动机制,协同完成系统任务所需的各项功能,使计算机程序的分析、设计和实现的过程与方法有机地结合。随着 MS Windows 系统平台及其支持下的面向对象程序设计语言(如 VC++、Delphi、Java 等)的普及推广,面向对象方法已成为目前软件系统分析、设计和开发的主流,对 CAC 系统的建设同样起到了至关重要的作用。在本节中,主要以 VC++ 为基础来阐述 CAC 系统的开发思路与过程。

6.3.1 CAC 系统的面向对象分析设计

面向对象分析(Object Oriented Analysis, OOA)是面向对象软件工程方法中的第一个环节,它采用面向对象方法,把对问题论域和系统的认识理解,正确地抽象为规范的对象(包括类、继承层次)和消息传递联系,形成面向对象模型,为后续的面向对象设计和面向对象编程提供指导。

面向对象设计(Object Oriented Design, OOD)则是把分析阶段得到的需求转变成符合要求的、抽象的系统实现方案的过程,或者说,它是用面向对象观点建立求解域模型的过程。OOD 是面向对象方法在软件设计阶段应用与扩展的结果。

尽管分析和设计的定义有明显区别,但是在实际的软件开发过程中两者的界线是模糊的。在一般情况下,许多分析结果可以直接映射成设计结果,而在设计过程中又往往会加深和补充对系统需求的理解,从而进一步完善分析结果。因此,分析和设计活动是一个反复迭代的过程。

1. 面向对象分析设计的基本内容与方法

OOA 总是从所面对的应用问题出发,分析系统基本构成的对象以及系统所必须遵从的、由应用环境所决定的规则和约束,从而明确规定构成系统的对象之间的协同合作,以帮助完成指定的系统功能。因此,OOA 的最终目标是构建以对象概念为中心的系统概念模型(也称为形式模型),它被划分为 5 个顺序层次:①类-对象层;②属性层;③服务层;④结构层;⑤主题层。这 5 个层次不是构成软件系统的层次,而是分析过程中的层次。其中,类-对象层用以引入对象和从对象中抽象出来的类;属性层和服务层分别定义类-对象的属性与操作;结构层表征对象之间的联系;主题层是一些类和对象的特定组合表示,可用于构建更高一级的层次单元。

OOD 的代表性方法则是采用扩展 OOA 模型的方法,将系统划分为 4 个部分:问题论域部分、人机交互部分、任务管理部分、数据管理部分。其中,问题论域部分是设计构造一组为底层应用建立模型的类和对象,细化分析结果;人机交互部分是设计一组有关类接口视图的用户模型的类和对象,设计用户界面;任务管理部分通过确定系统资源的分配,设计用于系统中类的行为控制的对象(类);数据管理部分设计将对象转换成数据库记录或表格。

2. 对象模型的表示方法

经过面向对象分析设计建立的概念模型的表达需要定义一组图形符号,并且规定一组

组织这些符号以表示特定语义的规则。一些对面向对象方法学的建立和发展做出过重大贡献的著名专家,如 Crady Booch ,Peter Coad & Ed Yourdon ,James Rumbaugh ,Sally Shlaer 等人 ,都提出了自己的表示方法。这些表示方法虽然形式各异,但它们的共同之处在于,都独立于具体的开发模式,都包含下列符号:表示类的符号,表示对象(类实例)的符号,表示继承关系的符号,表示类、对象之间其他关系的符号。目前,对象模型尚没有统一的、标准的表示方法。张海藩、牟永敏在《面向对象程序设计实用教程》(清华大学出版社,2001)一书中,对 Peter Coad 和 Ed Yourdon 提出的表示方法进行了少量的改进与简化。这里简略介绍一下他们的表示方法。

(1)表示类-对象的图形符号

类-对象的含义是“一个类及属于该类的对象”。图 6-34(a)是表示类-对象的图形符号,图 6-34(b)是一个单独表示类的图形符号。类符号是类-对象符号的变化形式,通常用来代表一个没有实例的抽象类。类符号是一个矩形框,被分为三个部分:类名、类的属性和服务。在表示一个类符号时,除了类名必须表示外,如果属性或服务不需要详细描述,也可以不表示。类-对象符号由类符号外加一个虚线框构成,虚线框代表属于该类的对象。

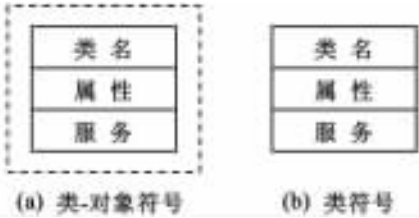


图 6-34 类-对象的图形表示

类的命名应该使用在应用领域中人们习惯的标准术语,其含义应明确、无歧义,必要时可以使用形容词加名词或其他形式的名词短语作名字,如“主干河流”、“道路”、“最短的路径”等。

(2)表示结构的图形符号

在面向对象分析与设计中,结构反映了问题域中的复杂关系,是对客观世界实体相互间关系的抽象。结构与目标系统的任务直接相关,目标系统的任务决定了系统的结构。类-对象之间的关系可以概括为归纳关系、组合关系和关联关系。

① 归纳关系。归纳关系就是一般与特殊的关系,体现了从一组类中抽取它们的共同特性构成高一个层次的类的形式。低层类(即派生类)继承了高层类(即基类)的属性和服务。

归纳关系采用一个半圆形标记来表示,其中从半圆弧中引出的直线指向基类。图 6-35(a)表示了归纳关系的图形符号。注意直线端点的位置,这个端点直接连接到里面的实线矩形框,表明这是类(而不是对象)之间的关系。

② 组合关系。组合关系就是整体与部分的关系,反映了对象之间的构成关系,因此也常常称为聚集关系。组合关系采用三角形标记来表示,从三角形顶角引出的直线指向整体对象。图 6-35(b)是表示组合关系的图形符号。注意,整体与部分结构线的端点位置表明,这是对象(而不是类)之间的关系。结构线每端标出的数值(或值的范围)表示在此结构中该端对象的数量(当值为 1 时无须标出)。

组合关系具有的最重要的性质是传递性。也就是说,如果 A 是 B 的一部分,B 是 C 的一部分,则 A 也是 C 的一部分。

当组合关系有多个层次时,可以用一棵简单的聚集树来表示它。聚集树是多级组合关系的一种简化表示形式。图 6-36 以一个地图的线目标图层为例,表明一个线图层由目标索引、栅格索引和若干线状目标构成,而线状目标又包括一个分类信息、若干条空间弧段。

③ 关联关系。关联关系反映对象之间相互依赖、相互作用的关系。通常把两类对象之

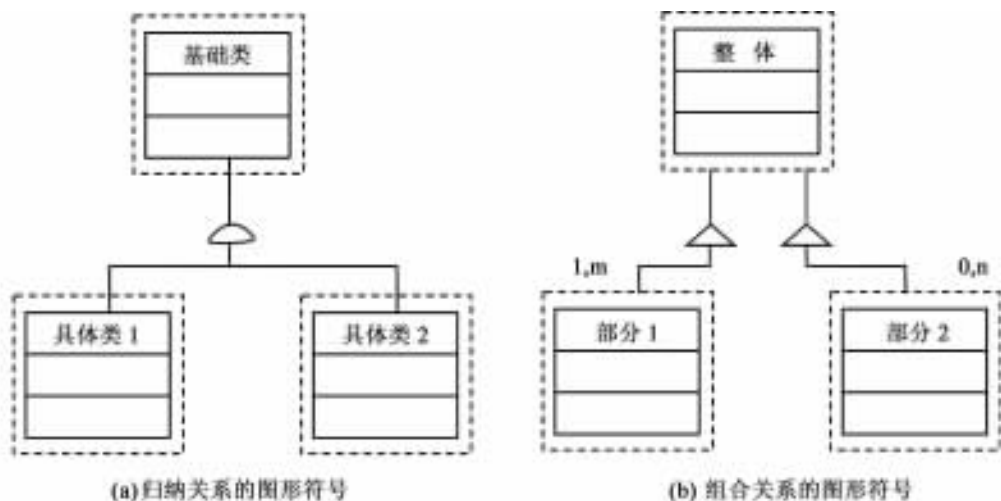


图 6-35 结构的图形符号表示

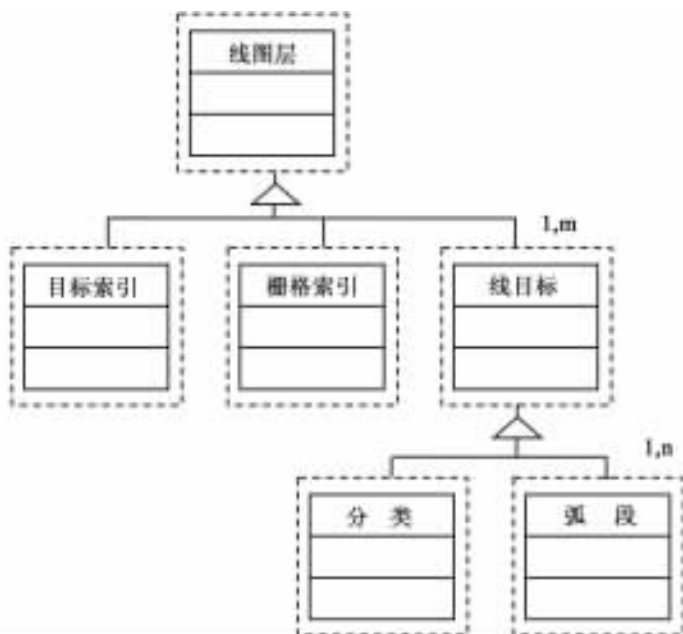


图 6-36 描述地图线目标图层的聚集树

间的二元关系再细分为一对一(1:1)、一对多(1:M)和多对多(M:N)三种基本类型。

关联关系用相互关联的两个对象之间的连线表示。图 6-37 描述了端点与弧段的关系，一个弧段与两个端点相关联，而一个端点则与至少一个弧段有关系。连线端点的所在位置表明这是对象(而不是类)之间的映射关系。

下面介绍关联关系中经常使用到的几个概念及其表示方法。



图 6-37 弧段和端点之间的关联关系表示

• 阶

阶是参与关联的对象的个数。与组合关系的表示方法类似,阶数用标在连线端点的单个数字或数值区间表示。例如,“1+”表示1个或多个,“3,5”表示3~5,等等。通常在软件开发的早期阶段不必仔细考虑阶值,应该先确定类、对象和关联,然后再确定阶值。

• 链属性

链属性就是关联链的性质。例如,计算机中的文件与文件的用户之间,存在着多对多的关系。但是,每个用户对不同文件具有不同访问权限,访问权限就是文件与用户之间的关联链的属性。图 6-38 是链属性的图形表示方法示例。

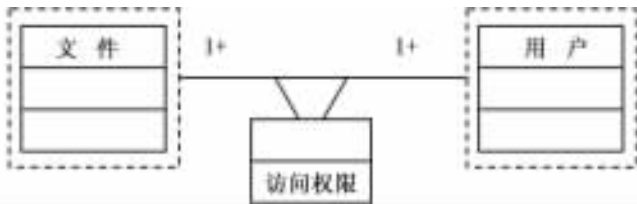


图 6-38 链属性的图形表示方法示例

• 限定

一个受限的关联由两个对象及一个限定词组成。可以把限定词看做是一种特殊的链属性。利用限定词通常能有效地减少关联的阶数。例如,一个地图符号库中存储了若干地图符号,一个地图符号仅属于一个地图符号库,并且每个地图符号都有自己惟一的符号编码,因此,借助于符号编码,可以将地图符号库与地图符号之间的一对多关系简化为一对一关系(如图 6-39 所示)。



图 6-39 限定关联的表示方法示例

• 消息连接

消息连接反映了一个对象对另一个对象的处理依赖性。这种关系表明,一个对象为了完成自己的任务,需要借助于另一个对象提供的服务。消息连接的表示符号,是从消息发送者指向消息接收者的箭头线(如图 6-40 所示)。



图 6-40 消息连接的图形表示示例

3. 面向对象的 CAC 系统分析设计

在前面讨论结构化方法应用时已经介绍了 CAC 系统建立的过程、任务和目标,在这里主要探讨如何应用 OOA 方法来构建一个 CAC 系统的对象模型。

(1) 对象层次分析

对象层次反映对象之间的包容关系,更高级的层次对象总是由许多低层次的对象组合而成,对高层次对象的操作也可以分解为对低层次对象的一组操作过程。

CAC 系统处理的对象呈现明显的层次关系,表现为从地图工程到地图图层,再到地图目标,最后到构成目标的空间数据和非空间数据。应用面向对象的方法,CAC 系统的处理对象可以分别概括为地图工程类、图层类、目标类、目标图形类和目标属性类。图 6-41 描述了 CAC 系统的这种层次关系。例如,一个地图图层对象包含了若干点、线、面目标对象,地图图层的显示就体现为目标对象的有序显示。



图 6-41 CAC 系统的对象层次

(2) 对象-类的内部结构

① 地图工程类的结构。地图工程对象是针对一个具体实施的地图项目,它的主要作用是对地图自身的环境参数以及构成地图的各个图层进行组织与管理。CAC 系统执行处理时总是首先调用地图工程对象,从中获取进一步的图层对象信息。

地图工程类的属性层包括两方面内容:一是目标地图的相关信息,如图名、比例尺、投影类型等;二是构成目标地图的图层信息,主要是图层数和各图层的索引。地图工程类的操作层主要包括地图图层的增删处理、地图的显示、缩放、漫游、查询、打印输出等功能,这些功能实际上都通过对更低一个层次的图层类调用来实现。表 6-4 中列出了 CAC 系统对象的类结构。

表 6-4 CAC 系统对象的类结构

类	地图工程	地图图层	地图目标	目标图形(弧段)	目标属性
属性	图名 比例尺 投影类型 图层数 图层对象数组 当前选择图层	图层名 图层类型 图层编码 目标数 目标对象数组 目标索引 栅格索引 当前选择目标	目标关键字 目标分类码 目标类型 目标弧段数 目标弧段数组 目标属性索引 目标名称 当前选择弧段	弧段关键字 弧段起始节点对象 弧段终止节点对象 弧段中间点数 弧段中间点对象数组 当前选择节点	目标关键字 目标属性项数 目标属性值表 当前选择属性项
操作	增删图层 图层选择 地图显示 地图缩放、漫游 地图查询 地图输出	增删目标 图层显示 图层缩放、漫游 目标查询 目标选择 图层输出 目标索引建立 栅格索引建立	增删目标弧段 编辑目标弧段 输入目标属性 修改目标属性 目标显示 目标缩放、漫游 目标弧段选择 目标属性项选择	增删点 编辑点 点选择 弧段显示 弧段缩放、漫游	增删目标属性值 修改目标属性值 属性值运算 属性值显示
派生类		基础图层类 专题图层类	线目标类 面目标类	弧段类	
特殊类		注记图层类 栅格图层类	点目标类	节点类	

② 地图图层类的结构。地图图层对象既是按照几何或属性类型划分的地图目标的集合单元,又包含了对它的各项操作,主要用于图层内地图目标及目标之间关系的建立、组织、管理和表达,如目标选择、增删目标、建立目标索引、栅格索引等。

地图图层类根据图层的基本性质又分为基础图层、专题图层两个派生类。派生类继承基本类的属性和操作,并在此基础上可增加派生类所独有的属性与操作,如在专题图层类中增加“图层属性项表”(属性)以及对它的操作,如属性项表达式的建立、编辑与运算,以实现专题地图显示的需要。

注记图层和栅格图层属于特殊的图层形式,其内部结构应另行定义。

③ 地图目标类的结构。地图目标对象是关于地图目标的空间数据(图形)、属性数据及其相关操作的封装体。除点目标外,线、面目标都是弧段对象的有序组合,对目标空间数据的处理实质上就是对目标弧段实施的选择、增删、编辑等操作。目标属性数据包含了对应一系列图层属性项的属性值集合,因此在目标类中对属性的管理表现为对所有属性项的处理,即对属性项的输入、删除和修改等。对具体属性项值的操作则是更底层的属性对象的任务。

④ 目标图形类与属性类的结构。地图目标的图形对象和属性对象同属于 CAC 系统的

两个底层对象,分别包含了对地图目标的最小图形单元和属性单元的操作。

图形对象又分为弧段和节点两类。点目标对象直接由节点对象构成。一条弧段中包含起始点和终止点两个节点对象及其之间的一组中间点,中间点同样可以用节点类描述。对弧段的操作主要通过对其上的节点(含中间点)的增删、移位来实现。

属性对象关注的目标是属性项组中某一个具体属性项的值。

(3)类-对象之间的关系模型

前面分析了 CAC 系统类-对象的属性与操作的封装结构,但对于系统的面向对象分析而言,还需要进一步确定类-对象之间的相互关系,包括继承与关联的关系。

图 6-42 描述了 CAC 系统的类-对象的关系模型。

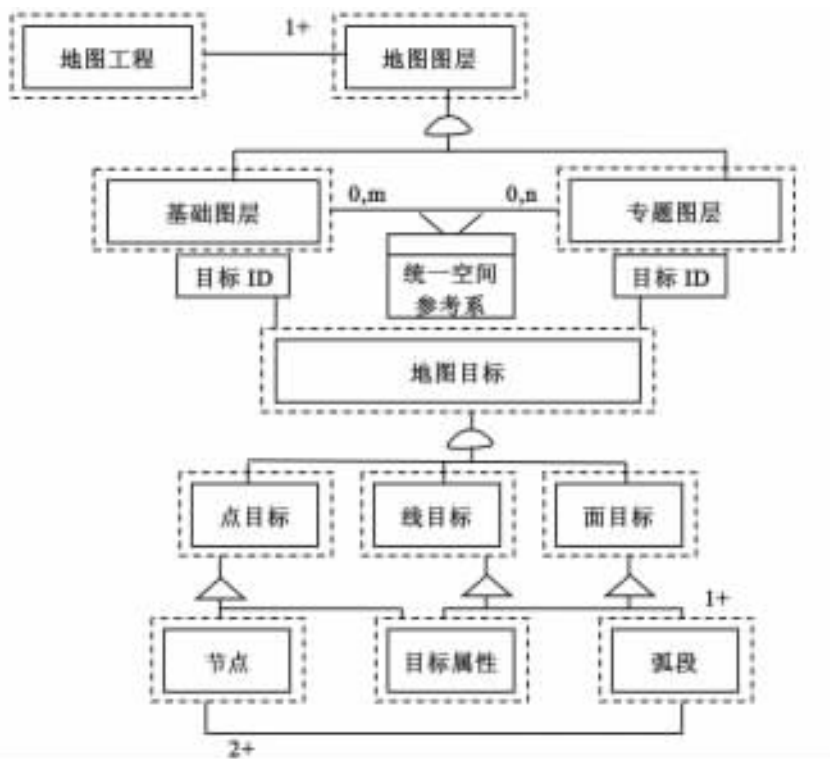


图 6-42 CAC 系统的类-对象的关系模型

在地图工程—图层—目标—图形与属性的层次结构中,以地图图层类为基类,派生出基础图层类和专题图层类,地图目标对象都分别存放在这两种派生类创建的图层对象中,并且通过目标关键字(ID)在图层中实现惟一的标志。地图目标类派生出的点、线、面目标类都是由图形类和属性类构成的,其中点目标的图形由一个节点对象构成,线、面目标由至少一个以上的弧段对象构成,而弧段又可表示为节点对象的集合。

4. 地图符号库子系统的面向对象分析设计

正如前面所述,根据地图符号所反映的地物与地理现象的分布特征,可将地图符号分为点状符号、线状符号、面状符号三类。按照面向对象的思想,由地图符号基类可派生出点符号类、线符号类和面符号类。地图注记因为与点状符号在诸多方面具有相似性,故归于点状

符号类。

按照地图符号的制图特性,点状符号又可进一步划分为标准点符号类、有向点符号类和注记类;线状符号划分为标准线符号类、有向线符号类、范围线符号类等;面状符号划分为标准面符号类、散列型面符号类等。

图元是构建地图符号的低一级图形单元,图元类根据它的几何特性和可变化性划分为点图元类、线图元类和面图元类。其中点图元类包括圆(椭圆)、矩形、三角形、五角星、正多边形、扇形、菱形等。

图 6-43 所示为地图符号库子系统的对象模型。

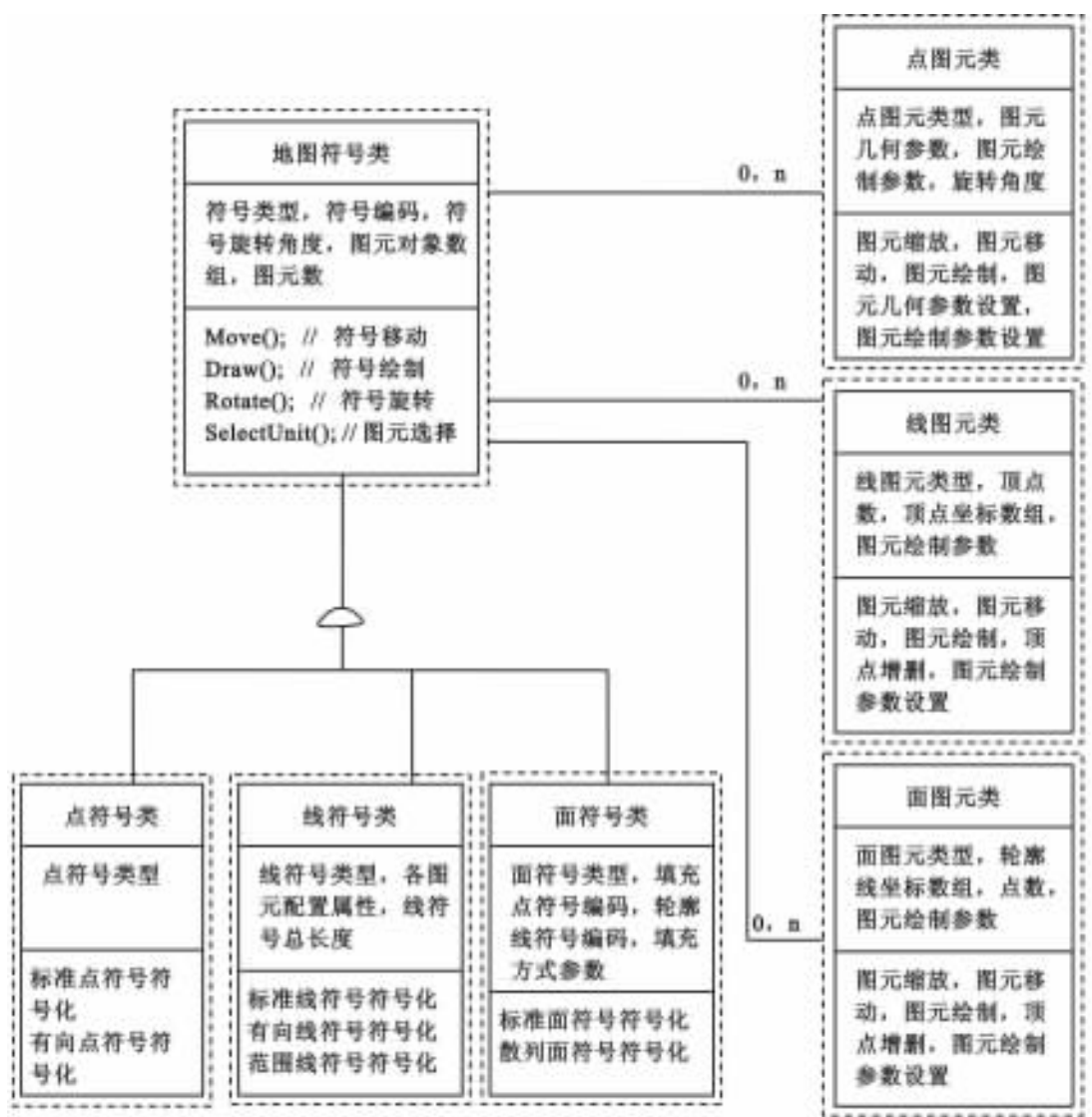


图 6-43 地图符号库子系统的对象模型图

5. CAC 系统的用户界面设计

用户界面是人机接口的重要部分,因此用户界面设计起着越来越重要的作用,其好坏直接影响到 CAC 系统的质量。一个好的用户界面能够最佳地发挥系统功能,从而容易被用户所接受。

整个 CAC 系统界面划分为 5 个区,即菜单区、工具栏区、状态栏区、图层管理区和工作区。菜单区中设置一个菜单树,最高一层菜单项为主菜单,菜单树中每一个叶子菜单项对应一个系统的功能。工具栏区建立最常用的叶子菜单项所对应的各个按钮,方便用户的点击驱动。状态栏区提示菜单项以及和操作过程、结果有关的信息,帮助用户了解操作步骤和运行结果。图层管理区专门负责图层的建立、编辑、属性设置等,并可以直接指定当前图层。

在 CAC 系统中,除了地图制图功能外,地图符号库子系统的功能也是一个重要的组成部分。在界面设计上,将两个部分统一起来,通过菜单项和工具栏的按钮实现两者的切换。

6.3.2 CAC 系统的面向对象程序实现

1. CAC 系统的 Windows 开发平台——Visual C++

VC++ 一直是 Windows 环境下最主要的应用开发系统。在 Windows 中,程序的基本单位不是过程和函数,而是窗口。一个窗口是一组数据的集合和处理这些数据的方法和窗口函数。从面向对象的角度来看,窗口本身就是一个对象,Windows 程序的执行过程本身就是窗口和其他对象的创建、处理和消亡过程。Windows 中消息的发送可以理解为一个窗口对象向别的窗口对象请求对象的服务过程。因此,用面向对象方法来进行 Windows 程序的设计与开发是极其方便和自然的。

VC++ 提供了一个支持可视化编程的集成开发环境:Visual Studio(又名 Developer Studio)。Developer Studio 是一个通用的应用程序集成开发环境,它包含了一个文本编辑器、资源编辑器、工程编译工具、一个增量连接器、源代码浏览器、集成调试工具,以及一套联机文档(如图 6-44 所示)。使用 Developer Studio,可以完成创建、调试、修改应用程序等各种操作。

2. 类的建立

在 VC++ 环境下,利用其 ClassWizard 可以方便地进行类代码的编写工作。如果想添加新的类、新的虚拟函数或新的 Windows 消息控制函数,ClassWizard 能够给出原型、函数体以及将消息同应用程序框架相联系的相应代码。ClassWizard 还可以用来对用户自己编写的类的代码进行修改。

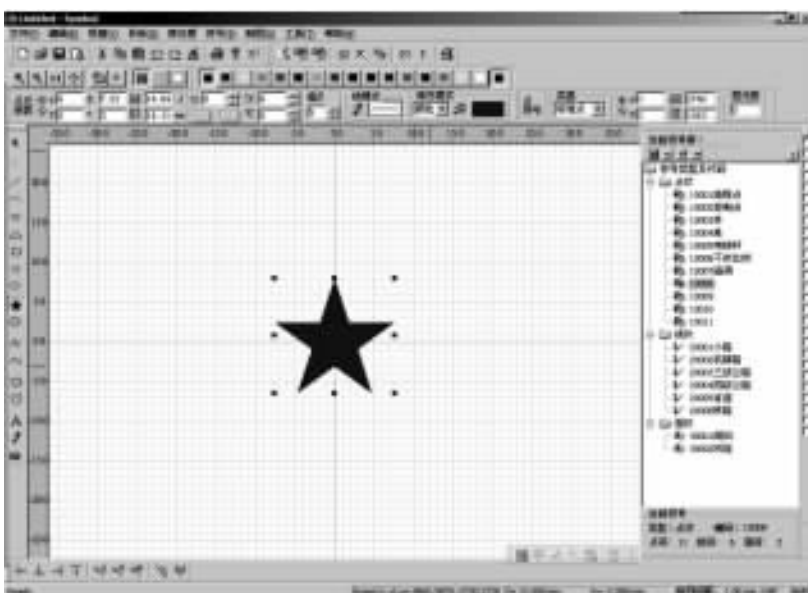
3. 文档类与视类的作用

在文档视结构里,文档是一个应用程序数据基本元素的集合,它构成应用程序所使用的的数据单元;另外,它还提供了管理和维护数据的手段。文档对象是由 CDocument 派生出来的。CAC 系统的文档类 CCacDoc 中要处理两类数据:一类是地图数据库;另一类是地图符号库。

```
class CCacDoc : public CDocument
{
    ...
public :
```

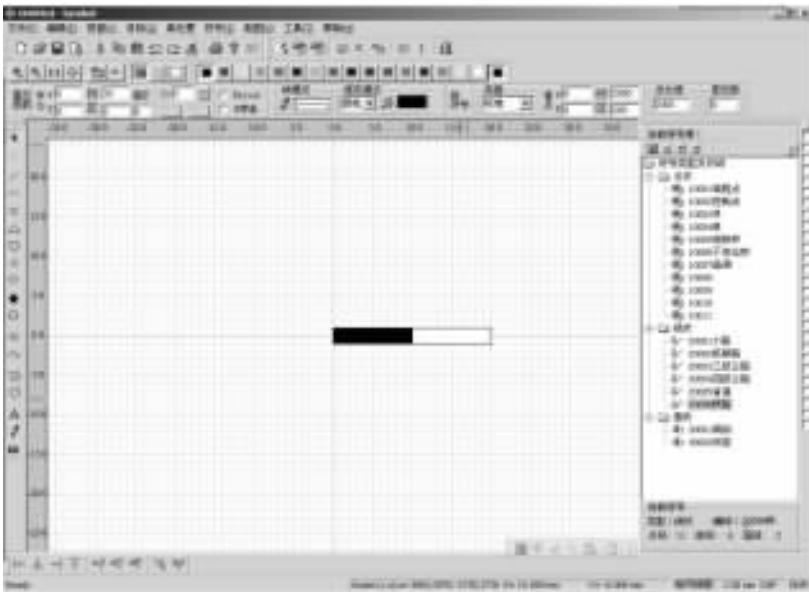



(a) 地图数据的采集

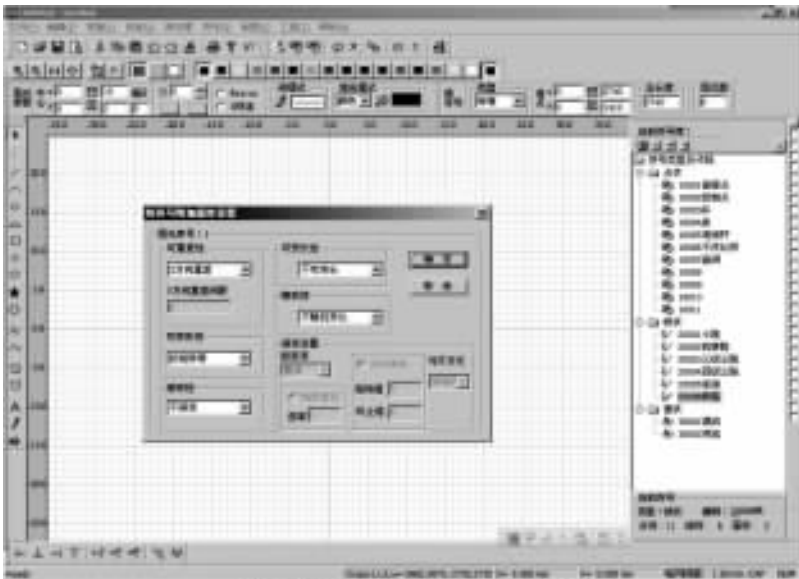


(b) 点符号设计 (由点图元构成)

图 6-45(a)(b) VC++ 环境下开发的 CAC 系统示例

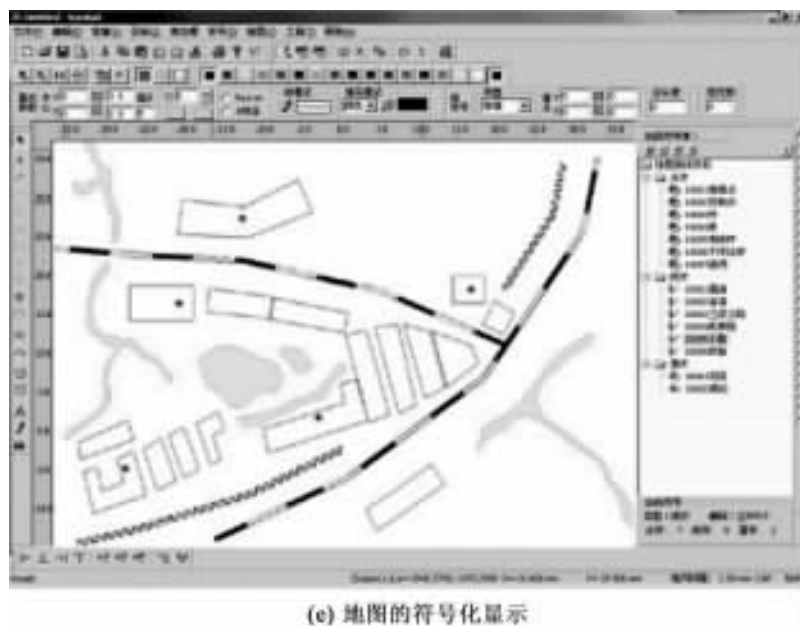


(c)线符号设计 (由两个面图元构成)



(d) 线符号设计中图元配置属性的输入

图 6-45(c)(d) VC++ 环境下开发的 CAC 系统示例



第7章 计算机地图制图系统的应用

计算机地图制图(CAC)系统为地图的设计、加工、制作提供了新的工具和手段,可以帮助制图人员采集、处理、存储和输出地图数据,增加了数据的可重复应用性,极大地缩短了地图的更新周期,提高了地图编制的效率。因此,CAC系统的应用并不仅仅是一个简单的软件操作和数据处理过程,其设计制作的地图产品还受到地图的数据来源、用途、设计方案、编制流程等诸多因素的影响。一个高质量的地图产品应同时集科学性、综合性、艺术性和实用性于一体。

7.1 地图制图的一般过程

计算机地图制图与常规的传统地图制图相比,虽然在技术上有了很大的发展,但仍然具有相当紧密的联系。分析传统地图制图的一般过程,其目的是要充分了解传统地图制图的经验和特色,为进一步指导计算机地图制图服务。

尽管不同的地图往往在制图区域、比例尺、表现主题和制图方法等方面存在差异,但就地图常规编制的总体过程而言,仍然可以归纳出其一般的形式,包括从总体设计、编辑计划的制定、编稿图的完成,到地图清绘、地图整饰和制印。这一过程是通过长期以来大量的地图制图实践而总结出来的、行之有效的规范化和程序化方法。我们把这个过程划分为三个阶段,即地图设计和编辑准备阶段、地图编稿和编绘阶段、地图整饰和制印阶段。下面简单地介绍一下各个阶段的任务。

7.1.1 地图设计和编辑准备阶段

该阶段主要完成地图设计和地图正式编绘前的各种准备工作。地图设计是关于地图的内容、形式和编绘方式的设计活动,它是地图制图各种活动的基础和关键环节。尽管地图设计贯穿了整个制图过程,但在地图正式编绘前首先需要通过地图设计来建立地图编制设计书或编图大纲,以指导其后的制图活动。

地图设计的任务包括:根据制图的目标、任务和用途,确定地图的选题、内容、指标、比例尺和地图投影;搜集、选择、分析和加工制图资料;选择表示方法和设计图例符号;确定制图综合原则和综合指标;对于专题地图,还需要将地理底图和专题内容分别设计,最后形成完整、协调、统一的地图编制设计书,并制定出地图编制的具体工作计划。

7.1.2 地图编稿和编绘阶段

该阶段主要按照地图编制设计书规定的技术要求完成地图的编稿和编绘工作。这是地图编制的主要实施过程,其任务主要包括:资料处理;展绘数学基础;进行地图内容的转绘与

编绘,其中包括对地图内容的综合(概括)处理。地图编绘是一种创造性的工作,编绘阶段的最终成果是编绘原图。编绘原图就是依据地图的设计思想,按规定的地图符号、约定的精确程度和制图综合程度,把制图区域的地理内容表达在图面上的地图,它是制作出版原图的唯一依据,直接决定着最后的成图质量和形式。

在专题地图的编绘中,地理底图的编绘方法和普通地图的编绘方法相一致,而专题要素的编绘是在地理底图的基础上编绘专题内容要素,从而得到专题地图的编绘原图。

在编绘图内容复杂、资料种类繁多的情况下,特别是对于专题地图,为全面、正确地使用资料,提高成图质量,加快编图速度,往往在地图正式编绘前由专业人员制作编稿草图或作者原图,再由制图人员编辑加工,完成合乎要求的编绘原图。

7.1.3 地图整饰和制印阶段

该阶段主要完成地图印刷前的准备工作(地图整饰)及最后的制版印刷工作。

地图整饰泛指地图设计与生产中美化地图外貌及规格化的各种技术工作,这里主要指地图印刷前依据编绘原图进行的地图清绘(或刻绘),制成印刷用的出版原图,以及制作与出版有关的分色参考图和彩色样图等。

地图制印包括出版原图的复照或翻版,分版分涂,制版打样,以及印刷、装帧等。

7.2 CAC 系统的应用流程

计算机地图制图是地图制图在技术和方法上的发展,其任务是在计算机软硬件环境支持下,应用 CAC 系统实现地图制图的整个过程。回顾地图生产的历史不难发现,在传统的地图生产过程中,地图设计、编绘、整饰和制印的每道工序都由不同的专业人员用手工方式完成,且在不同的工序之间往往需要专门的沟通与协调,周期长,效率低,修改和更新复杂。和传统地图生产过程相比,计算机地图制图可以将设计、编绘、整饰和制印功能集成在一起,在统一的 CAC 系统中予以实现,交互性强,可以快速实施、调整各种设计方案,包括图面设计、符号设计、色彩设计等,并直接实现和打印机、激光照排系统甚至印刷系统的连接与驱动。但是,值得强调的一点是,CAC 系统虽然可以提供地图设计、地图综合的功能模块,究其实质仍是通过地图内容的编辑、修改的交互功能来间接实现的,其本身包含的高智能性活动仍然需要专业人员的思维来完成。

因此可以说,在目前条件下,CAC 系统虽然不是一个可以取代制图人员的全自动系统,但其应用贯穿了地图制图的全过程。CAC 系统的应用主要包括编辑准备、数据采集、数据处理和图形输出 4 个过程(如图 7-1 所示)。

7.2.1 编辑准备

这一阶段的任务与传统的制图过程基本相同,包括对编图资料的搜集、分析和处理,根据编图要求确定地图投影、比例尺、地图内容、表示方法等。除此之外,为了计算机地图制图的需要,在充分了解作为制图平台的 CAC 系统的数据组织、功能和操作的基础上,还必须考虑以下几方面的内容:

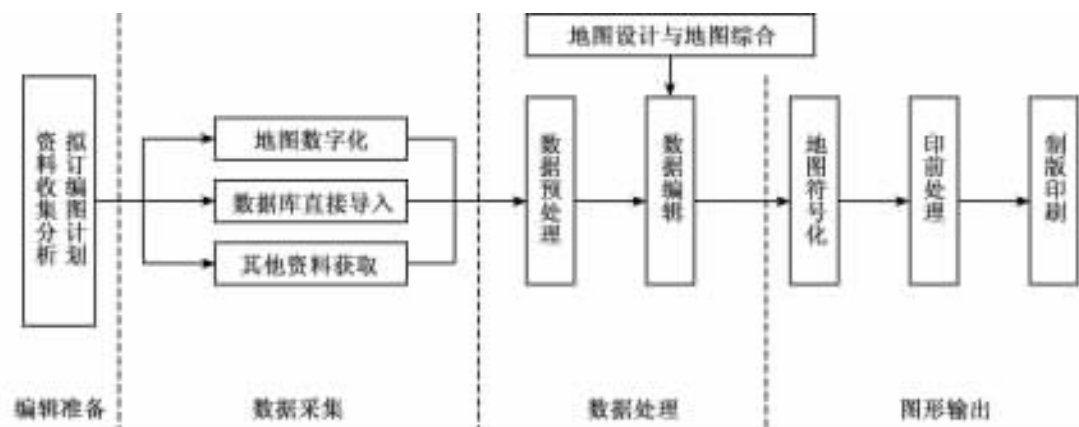


图 7-1 CAC 系统的应用流程

1. 地图资料的数字化采集方式

针对一项编图任务,为了保证编制的地图内容全面,现势性强,所采用的地图资料往往类型多样,如不同比例尺、不同时段的地图、遥感影像、原有的地图数据库或地图数据文件、文字资料、报表数据等。在采集方法上应区别处理。在一般情况下,经过坐标纠正的遥感影像数据可以经过图像自动识别方式,或者人工解译判读加屏幕矢量化跟踪两种方法来采集;纸质地图可以直接采用数字化仪数字化,也可以采用扫描转换成图像数据然后在屏幕上进行矢量化跟踪;已有的地图数据可以采用经格式化转换直接导入地图数据库的方式;文字、报表则可以根据其内容容量的大小采用文字报表自动识别软件,或者直接采用手工键盘录入。

2. 地图的图层设置方案

在计算机地图制图中,地图主要是通过图层来组织的。不同的类型、主题、比例尺和制图区域的地图往往在图层划分、结构和属性上各不相同。例如,比例尺分别为 1:50 万和 1:5 000 的基础地形图由于图式规定的分类类型以及制图综合程度的不同,在图层的命名和定义上有着较大的差异。其中,图层的性质反映了图层内地图目标的公共特性,图层的顺序决定了各类型地图目标的绘制顺序。可以说,选择、设置正确的数字化图层方案直接关系到地图编绘的效果与质量。

对于普通地图的图层设置,一般以相关地图图式的要素和类型划分规定为基础,结合制图区域的目标类型特点,在原则上按照面状目标图层、线状目标图层和点状目标图层的顺序自下而上设置。对于专题地图,基础地理图层设置在下面,专题图层设置在上面,以突出专题要素信息。

图层的性质是关于该图层的编辑、显示、绘制等一系列操作的通用参数设定,如图层默认的目标类型、线型、线画颜色、填充颜色、填充晕线等。

表 7-1 是某城市 1:50 000 城区交通旅游图数字化时的图层设置的一个示例,其中图层序号越小,表示图层居于越下层。

表 7-1 图层设置示例

图层 序号	图层名	目标类型	线型	线画颜色	填充颜色
1	街区	面 状	实 线	黄 色	浅黄色	
2	双线河流	面 状	实 线	深 蓝	浅 蓝	
3	湖 泊	面 状	实 线	深 蓝	浅 蓝	
4	植被、绿化带	面 状	点虚线	深 绿	绿 色	
5	市 界	线 状	长线画一点一点	浅褐色	—	
6	区 界	线 状	长线画一点	浅褐色	—	
7	街道界	线 状	短线画一点	浅褐色	—	
8	单线河流	线 状	实 线	蓝 色	—	
9	桥 梁	线 状	实 线	紫 色	—	
10	铁 路	线 状	(组合线)	黑 色	黑(白)色	
11	政府部门	符号注记	—	大 红	大 红	
12	街道办事处	符号注记	—	大 红	大 红	
13	医 院	符号注记	—	橙 色	橙 色	
14	学 校	符号注记	—	橙 色	橙 色	
15	旅游景点	符号注记	—	橙 色	橙 色	
16	区 名	文本注记	—	—	大 红	
17	街道名	文本注记	—	—	大 红	
18	医院名	文本注记	—	—	黄 色	
19	学校名	文本注记	—	—	黄 色	
20	一级路名	文本注记	—	—	棕 色	
21	二级路名	文本注记	—	—	棕 色	

3. 地图目标的数字化方法

在地图图层设置方案确定之后 ,需要确定对图层中的目标如何进行数字化 ,这是和目标采集直接联系的更具体的实施方案。

不同的目标类型具有不同的数字化采集方法。地图目标包括点、线、面目标 ,但是在考虑数字化方法时需要进一步、更详细地进行划分。在上一章中我们已经分析过这种划分 ,如点目标又包含了标准点目标和有向点目标两种类型 ,两种类型虽然都为点目标 ,但数字化方法却存在着显著的差异 ,线目标、面目标也是同样的情况。这需要对每一类地图目标的表达方式进行深入的分析 ,归纳它的表达特点和规律 ,在此基础上进行类型的详细划分。

考虑到地图应用的其他方面 ,如地图量算、地图空间分析等的需要 ,地图目标在数字化时也要采取相应的措施。例如 ,河流、道路遇到桥梁、村庄产生目标的分割 ,不利于长度、面

积的计算和空间连通性分析,需要在采集时增加辅助线或辅助面加以解决,辅助线或辅助面在地图显示时不绘制(如图 7-2 所示)。

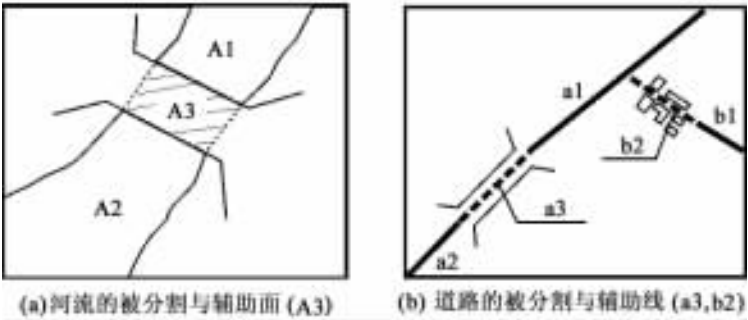


图 7-2 地图目标被分割情况下数字化采集示例

4. 地图符号与地图符号库

地图符号应该与地图目标类型相一致,即每一个目标类型均对应于一个地图符号。在 CAC 系统的应用中,符号的设计一方面决定着地图的表达,同时又直接与数字化方法紧密联系。因此,这一过程成为计算机地图绘制的重要环节。

地图符号的设计一般有两种方式。一种是采用地图制图过程中经过长期经验积累总结出来的符号约定形式,最典型的是在基本比例尺地形图图式中规定的各种符号。这些基本约定的符号在一定范围内为大家所公认与遵守,例如铁路符号、境界线符号、城墙符号等。另一种是由专业制图人员进行自定义设计,设计时应遵循地图符号设计的一般原则,既能够突出地图的主题内容,又体现地图目标表达的合理性、艺术性和直观性。

对于计算机地图制图,其地图符号由若干图元的有序组合构成。当数据来源为纸质地图时,符号设计有时可以直接参考纸质地图,这就需要对某一个类型的地图目标的表现形态进行分析,从中提取其最小的重复单元,结合地图符号库子系统中对符号图元的划分,建立对应的图元组合形式,并确定各图元的属性参数,其中包括可变参数与不可变参数。

地图符号库是对符号的集中组织与管理。需要结合 CAC 系统的地图符号库子系统功能进行地图符号的分类,明确欲建立的不同地图符号类型及其具体数量,完成对地图符号库的基本结构设计。

5. 地图属性数据项的定义

CAC 系统中对于地图数据库所采用的数据结构已有明确的规定,这包括地图空间数据、属性数据、空间或属性的关系数据以及它们相互之间的联系信息。但是从定义的内容上看,对于一个明确的 CAC 系统,地图目标的几何特征及其空间数据构成在形式上一般是关联和固定的,如点、线、面目标的空间坐标点数分别为 1 点(如果为有向点,则坐标点数为 2)、 n 点(首末点一般不重合, $n \geq 2$)和 m 点(首末点重合, $m \geq 3$)。一旦定义了目标的类型(分类编码),则目标的空间数据由几何特征、坐标点数和坐标集合三个部分构成,这三项为系统内部所规定,通常不允许随便更改。

而属性数据由属性数据库直接管理,属性数据库一般采用已商品化的关系数据库来建立,制图人员可以根据地图绘制的需要为某一类目标添加属性数据项。各个目标类型均具有自己特定的属性数据项,如道路可以有“分类分级编码”、“道路编号”、“道路等级”、“道

路宽度”、“路面底质”、“所属政区”、“道路名称”等属性项,而等高线则主要有“分类分级编码”、“高程值”属性等。因此,在进行地图数据采集之前,应根据编绘需要将每个目标类型的属性数据的属性项确定下来。

7.2.2 数据采集

数据采集是按照编辑准备阶段确定的地图资料的数字化采集方式和地图目标的数字化方法来具体实施的。在上一章中已经介绍了 CAC 系统中地图数据采集的基本步骤与相应的功能模块,它主要包含了空间(几何)数据采集与属性数据采集两个部分。

1. 空间数据采集

(1) 采集的多样性

空间数据采集的过程实际上就是应用 CAC 系统的数据采集功能将地图图形资料转化为地图数据库中图形数据的过程。但是在实际应用中,这一过程并非单一方式的,主要存在以下两方面原因:

① 地图编绘可能同时采用不同类型和来源的地图图形资料,因此需要进行多种数据的采集。众所周知,地图图形资料是多种多样的。一方面在类型上可以是地形图、专题地图、遥感图像、外业测量的数据、已有的地图(空间)数据库或地图数据文件等;另一方面,即使数据类型相同,也存在来源上的差异,如不同的时间、不同的比例尺、不同的波段和不同的主题内容。

② 为了提高数据采集效率,对同一幅地图资料的不同要素采用不同的采集方式。例如,在小比例尺地形图数字化过程中,居民地、道路和水系可以采用手扶跟踪数字化仪采集或者地图扫描后手工在屏幕上数字化的方法,而等高线密集、数量大的地图资料,手工采集速度慢,考虑到等高线的形式单一,彼此间影响非常小,宜采用对扫描后提取的等高线分版图进行全自动矢量化的方法。

(2) 采集方法

在这种情况下可以考虑采用两种数据采集的方法模式——全采集模式和间接采集模式,如图 7-3 所示。

全采集模式是当 CAC 系统可以同时提供需要的各种地图图形数据采集功能时,可以按照一定的顺序依次采集不同的地图图形资料,获取的图形数据分别存放在不同的命名图层中,从而直接集成到 CAC 系统支持下的一个统一的地图数据库中。

间接采集模式是在 CAC 系统不具备所有需要的采集功能的情况下,可以首先应用其他数字化软件采集某些系统不能接收的地图数据,然后转换为系统能够接收的图形数据库或数据文件形式,通过系统具备的数据库或文件读取功能导入到统一的地图(空间)数据库中。

但无论是全采集模式,还是间接采集模式,要将不同的地图资料经过数字化采集后真正集成到统一的地图(空间)数据库中,就必须保证多源地图数据在空间定位和空间关系上的协调一致性,这需要通过 CAC 系统的数据编辑处理功能模块来完成。

2. 属性数据采集

属性数据的采集一般通过键盘输入或数据文件转换得到。在计算机普通地图制图中,地图目标的属性数据主要为分类分级编码,数量或质量特征数据相对单一,可以采用逐个目

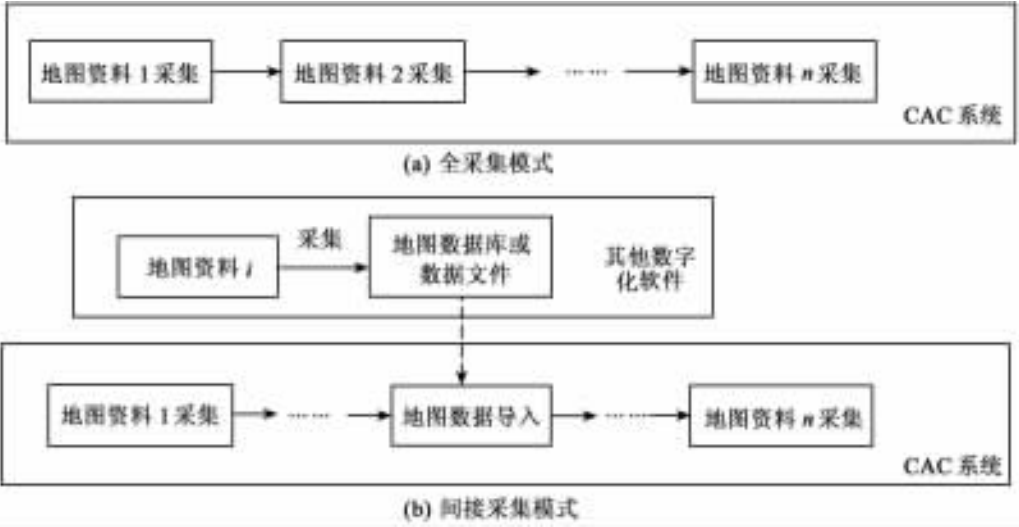


图 7-3 CAC 系统的地图图形数据采集模式

标输入的方法。该方法直接将地图目标的图形与属性关联起来,其一般步骤为:

- ① 选择基础地图图层,新建一个属性项表;
- ② 在属性项表中添加若干属性项,确定每一个属性项的参数,如字段类型、字段长度等;
- ③ 保存属性项表结构;
- ④ 选择属性输入功能;
- ⑤ 使用鼠标选择一个图形目标,在相应的属性栏中输入正确的属性值;
- ⑥ 重复⑤,直到该图层目标全部输入完毕为止;
- ⑦ 保存属性项表。

如果此后在某一个图层中增加了新的目标,只需要选择该图层,执行④、⑤、⑥、⑦就可以增加新目标的属性。

但在计算机专题地图制图中,目标的属性包含了专题属性数据和地理统计数据两大部分,数据量大,仅依靠上述方法进行逐个目标的手工输入,效率低,准确性差。因此,可以采取文件输入的方式,即先将属性数据直接输入到数据库或数据文件中,然后将属性数据库与地图(空间)数据库自动连接起来,如果是数据文件,则通过转换将属性数据导入到属性数据库中。为了保证地图目标的空间数据与属性数据的正确匹配,在属性数据库和地图(空间)数据库中均建立统一的“目标标志码”(Object Identification)。

7.2.3 数据处理

数据处理作为计算机地图制图的核心工作,是联系数据采集和制图输出的桥梁。它的主要作用是检核所采集的空间数据和属性数据的错误,从而修正错误,保证数据质量,并在此基础上构建目标在空间及属性上的联系,等等。

由于采集的地图数据可能具有多种来源,在比例尺、主题内容等方面也并不一致,要将

其集成到同一个地图数据库中,往往存在一定的矛盾和冲突,需要分别进行地图数据的相关处理。这些矛盾和冲突主要体现在以下方面:

(1) 采集数据的空间参考系不一致

由于采集所选择的各类地图资料在投影类型、空间坐标系、高程基准等方面采用的标准可能不同,有些类型的专题地图甚至没有明确的空间参考系,导致地图数据之间无法在空间上统一起来。这就需要对不同的地图资料进行空间参考系的分析,分别建立正确的定位控制点,通过 CAC 系统进行投影变换以及对扫描图像进行几何纠正、坐标匹配和矢量数据的坐标变换,从而保证空间参考系的一致性,为多源地图数据的融合奠定基础(如图7-4所示)。



图7-4 多源地图数据空间参考系不一致的解决方法

(2) 数据的详略程度不匹配

地图资料采用的比例尺和它们要反映的专题内容的不同,使得地图目标表达的信息详细程度存在差异,这种差异可能表现在空间信息上,也可能表现在属性信息,如分类分级上。众所周知,大比例尺地形图和中小比例尺地形图相比,目标在几何形态上具有更复杂、更详细的细节信息。同样,为了突出专题内容,专题地图中专题要素的类型或等级划分相对于普通地图而言也总是更加详尽。这必然造成空间数据和属性数据在信息详细程度上的差异,最终影响到地图数据的融合。

要解决这个问题,一方面是根据地图制图的比例尺设计要求选用具有合适比例尺的地图资料,一般选择具有较大比例尺的地图资料,因为它可以提供更加丰富的细节信息。另一方面主要是结合 CAC 系统提供的制图综合功能模块进行地图概括处理,将不同详细程度的地图数据分别转变为具有相同比例尺的数据形式。

(3) 目标之间的空间和属性关系矛盾

正因为各类地图资料的数学基础可能存在差别,加上大多数地图资料都是独立进行制图综合处理的,各类地图数据尽管经过一系列处理后在数学基础和综合程度上趋于协调一致,但在更具体的地图目标之间,空间和属性上的联系依然会存在矛盾。如两个相邻目标的边界线互相压盖、交叉,以致无法进一步构建完整的目标之间的关系数据。这将主要依赖于 CAC 系统的地图编辑功能模块来进行处理。

因此,数据处理包含了投影变换、比例尺变换、坐标变换、数据合并归类、制图综合以及地图目标和节点的编辑修改等众多的功能模块,其过程实际上就是分析数据中存在的问题

并解决问题的过程。由于具体方法及步骤在前面的章节中已有较详细的介绍,这里不再赘述。

7.2.4 图形输出

图形输出是 CAC 系统的目标功能。地图数据的采集、编辑、处理最终是为图形输出服务,它将采集、处理以后的正确的地图数据转换为地图图形,可以直接通过屏幕显示或打印机、绘图仪输出,也可以将输出的图形转换为图像后进行分版处理,印刷成图。

但是,在 CAC 系统中,图形输出的关键是为该地图数据提供高质量的地图符号库支持。地图符号库子系统为符号库建立提供了功能平台,制图人员需要在此平台上将符号设计方案转化为实际的符号库。符号有两种来源:一种是直接在符号库中利用设计、编辑功能建立符号;另一种是从其他已建立的符号库中选择符号导入到当前符号库中,允许进行适当的编辑和修改,并用输入的新符号编码保存。

图形在屏幕上显示后,针对少量地图目标、注记的位置冲突,选择相关图层,应用地图编辑功能进行修改处理,以保证制图质量。

7.3 常用 CAC 软件介绍

目前可以用于计算机地图制图的软件非常多,在功能上各具特色,下面主要介绍几种可以应用于计算机地图制图的软件。

7.3.1 CorelDraw 软件

CorelDraw 是加拿大 Corel 公司推出的集矢量图形绘制、文字编辑和印刷排版于一体的图形软件。CorelDraw 具有强大的平面设计功能,而且还提供了 3D 的效果设计,自推出以来,一直受到广大美术人员和平面设计人员的欢迎。鉴于其功能完善,操作设计简便,也越来越多地应用于地图的编绘工作之中。

它的主要特点有:

① 灵活的位图图像导入功能。CorelDraw 虽然是一个矢量图形设计软件,但仍然提供了各种格式的图像导入功能,从而可以导入扫描的地图图像作为基础图层,在其上进行数字化跟踪、编辑和符号化表达。

② 强大的图层和对象管理功能。CorelDraw 建立了对象管理器和对象数据管理器,可以方便增加、删除图层和设置、修改图层属性,并进一步对图层中各对象进行管理。通过对图层的“可见”、“可编辑”和“可打印”等特征项的设置,可以方便地控制地图的编辑和显示、打印输出。

③ 强大的绘线功能。尤其是绘制曲线,可以说是 CorelDraw 软件的一大特点。它拥有一组各具特色的绘线工具,可以根据用户的具体需要选择。其中有一种“贝塞尔工具”,可用于绘制平滑曲线,且节点很少。

④ 强大的节点编辑功能。节点编辑实际上就是线编辑,用节点编辑功能可以很方便地进行形状改变、线段连接和分割、曲线光滑和锐化等一系列的曲线编辑工作。

⑤ 强大的面填充功能。CorelDraw 在面填充方式中除了平铺色以外,还有图案花纹填

充、PostScript 填充、渐变色填充等,大大丰富了地图的表现能力。

⑥ 丰富的文字注记和编辑功能。在一些计算机制图软件中,道路名和水系的注记不一定呈直线均匀排列,往往需要一个字一个字地输入、定位。而 CorelDraw 只要一个路名或几个路名以字符串的形式一次输入,执行一个“文本适合路径”的命令就可以沿着道路或河流自动注记,而且随着道路或河流的方向变化而自动改变其文字的方向。

⑦ 丰富的图形和文字效果功能。CorelDraw 软件能提供丰富的图形和文字效果,如立体的、阴影的、变形的,等等。这对于提高地图艺术效果是必不可少的,也是一般矢量制图软件所难以实现的。

⑧ 方便的图例符号建库和调用功能。CorelDraw 提供了图例符号库建立和调用的功能,可以有针对性地开发地图符号库,不管是点符号、线型符号还是面填充符号,调用都十分方便。符号库还有一个十分明显的特点是,符号可以无级缩放,并且只要事先设置好轮廓线的缺省值,符号再小,仍能保持它的精致程度。

⑨ 图形文件数据量小。同一个图形文件,从 AutoCAD 的 dxf 格式转为 CorelDraw 的 cdr 格式,其数据量减少了 7~10 倍,这对于大数据量的地图制图作业是十分有利的。

⑩ 可以输出众多格式的图形图像文件。包括常用的 AI、dxf、dwg、emf、svg、pct 等图形文件和 bmp、jpg、tif、gif、psd 等图像格式文件。可以方便地与其他软件系统实现数据共享,也有利于解决二次开发过程中的数据读取问题。

尽管 CorelDraw 软件在图形编辑、设计和表达上功能强大,但在地图制图方面仍然存在着局限性。由于 CorelDraw 并非专业的 CAC 系统,无法建立目标之间有效的空间关系,通常按照图形对象来组织地图目标的结构,缺乏更有效的描述和管理机制,在数据采集、编辑过程中往往存在步骤重复、操作烦琐的情况,难以提高地图制图的自动化程度。

图 7-5 是 CorelDraw 用于计算机地图制图的一个界面示例。

7.3.2 AutoCAD 软件

AutoCAD 是 Autodesk 公司推出的工程设计制图软件,是目前计算机辅助设计领域最流行的 CAD 软件包。它广泛应用于建筑、机械设计和制造、服装设计、土木工程等领域。从最早的 2.0 版到以后的 R13、R14、2000、2002 直到如今的 2005 版,从最早期的 DOS 操作命令到现在的 Windows 窗口式的操作界面,软件系统在功能、用户友好性和应用上已经有了显著的提高。

AutoCAD 软件最早是针对二维设计绘图而开发的,随着其产品的日益成熟,在二维绘图领域该软件已经比较完善,而且随着产品设计的发展需要,越来越多的产品设计已经不再停留在二维的设计领域,正在越来越多地朝着三维的产品设计发展,因此在 AutoCAD R12、R13 的版本中已经加入了三维设计的部分,而且随着版本的不断更新,三维设计的部分也在不断发展。

由于该软件开发中的自身原因,该软件也存在一些不足之处。例如,该软件在二维设计中无法做到参数化的全相关的尺寸处理,并且和 CorelDraw 一样,早期的 AutoCAD 也主要适用于图形设计,虽然价格较便宜,设计精度高,对所使用的微机要求较低,应用也比较简单,但并不能完全适应地图制图的需要。

目前 Autodesk 公司发布的 Autodesk Map Series 2004 是一个综合性地图绘制软件包,包



图 7-5 CorelDraw 用户界面

括 Autodesk Map 和 Autodesk Raster Design。Autodesk Map 2004 是精确的地图绘制和 GIS 软件 ,可用来创建、维护和分析地图及地理数据 ,它在 AutoCAD 环境中运行 ,带有一些特殊开发的工具用于创建和处理地图和地理信息数据 ,并添加了所有 AutoCAD 软件的基本功能。Autodesk Raster Design 2004 是一个图像解决方案 ,用来显示、操作和输出具有矢量数据的光栅影像。

图 7-6 为 AutoCAD 的用户界面。

7.3.3 ArcGIS 软件

ArcGIS 是美国 ESRI 公司(Environment System Research Institute Inc.)开发的 GIS 软件 ,是世界上应用广泛的 GIS 软件之一 ,也是我国 GIS 领域常用的软件。从 1987 年第一个 Arc/ Info 产品诞生以来 ,随着计算机技术的飞速发展以及 GIS 技术的不断成熟 ,ERSI 的 GIS 产品不断更新扩展 ,形成适应各种用户和机型使用的系列产品。1999 年推出的 ArcInfo 8 以及 2000 年推出的 ArcGIS 8 是 ESRI 在继承已有成熟技术的基础上 ,整合了 GIS 与数据库、软件工程、人工智能、网络技术及其他多个方面的计算机主流技术 ,成功地开发的新一代 GIS 平台。

ArcGIS 8 包含了三个组成部分 :桌面软件 Desktop、数据通路 ArcSDE 和网络软件 Arc-IMS。ArcSDE 是 ArcGIS 的空间数据引擎 ,负责应用关系数据库系统存储、管理多用户空间数据库 ,提供数据接口支持。ArcIMS 主要提供基于 Internet 的分布式 GIS 数据管理和服务。ArcGIS 8 Desktop 是 ArcGIS 中一组桌面 GIS 软件的总称 ,它包括 ArcView 8、ArcEditor 8 和 ArcInfo 8 三个等级 ,功能由简单到全面。其中 ,ArcView 8 是最低一级的软件 ,主要提供 GIS



图 7-6 AutoCAD 的用户界面

数据的使用、地图的显示以及分析,它的数据编辑功能有限;ArcEditor 8 更高一级,在 ArcView 8 的基础上增加了 GIS 数据管理与编辑功能,具有创建和管理关系、子类、几何网络和尺寸要素等作用;ArcInfo 8 是桌面 ArcGIS 的高端软件,除了包括 ArcView 8 和 ArcInfo 8 的所有功能外,还增加了空间处理能力,是一个完整的 GIS 数据建立、更新、编辑、查询、管理、分析与制图的集成系统。除此之外,ArcGIS 8 Desktop 还有若干可选的扩展模块,如 Spatial Analyst 3D Analyst Geostatistical Analyst ArcPress Publisher 等。

三级桌面 ArcGIS 软件都由一组相同的应用环境构成,即 ArcMap ,ArcCatalog ,ArcToolbox。ArcMap 提供数据的显示、查询与分析(如图 7-7 所示);ArcCatalog 提供空间的、非空间的数据管理、生成和组织;ArcToolbox 提供基本的数据转换(如图 7-8 所示)。通过这三种应用环境的协调工作,可以完成任何从简单到复杂的 GIS 任务,包括数据采集、数据编辑、数据管理、地理分析和地图制图。

在 ArcGIS 系统中,三级桌面 ArcGIS 软件都具有地图制图、表达的功能。其中 ArcView 是新一代桌面地理信息系统的代表,其使用方便、灵活、操作简单、通用性强,已成为最常用的 GIS 软件之一(如图 7-9 所示)。ArcView 采用了可扩充的结构设计,整个系统由基本模块和可扩充功能模块构成。其中,基本模块包括对视图(Views)、表格(Tables)、图表(Charts)、图版(Layouts)和脚本(Scripts)的管理,扩充模块主要集中在空间分析功能上,例如网络分析、三维分析、影像分析、追踪分析等。

7.3.4 MapInfo Professional 软件

MapInfo 是美国 MapInfo 公司开发的桌面地理信息系统软件,其中 MapInfo Professional 是一套业界中领先的 Windows 平台的地图化解决方案。MapInfo Professional 可以提供地图绘制、编辑、地理分析、网格影像等功能。由于其简单易学、功能强大、二次开发能力强,并且可以方便地与通用的关系数据库连接,因此用户数量增长十分迅速。随着每次版本的升级和新产品的推出,MapInfo Professional 都有很大的改进,功能越来越强,界面也越来越完善和



图 7-7 ArcMap 用户界面

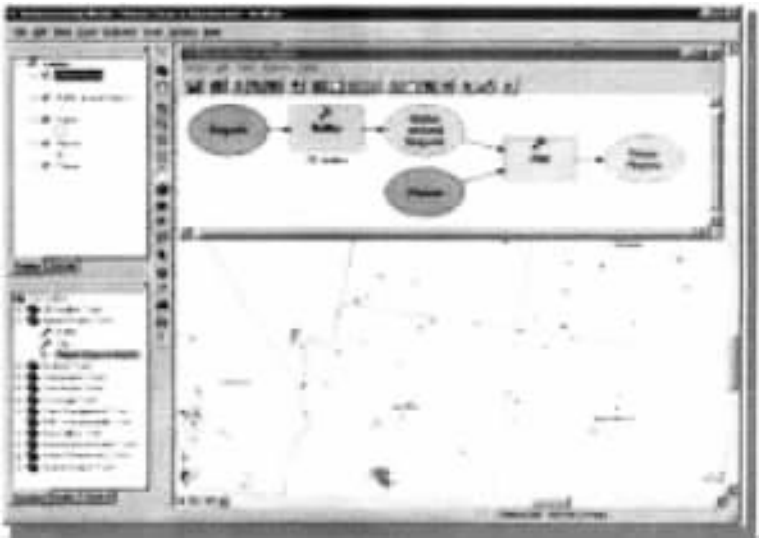


图 7-8 嵌入到 ArcGIS Desktop 环境中的 ArcToolbox

实用。现在的最新版本为 MapInfo Professional 8.0 ,它为新老用户提供了更强大的数据维护、可视化、数据展现、输出能力和更好的可用性。

① MapInfo Professional 提供一整套功能强大的工具来实现复杂的商业地图化、数据可视化和 GIS 功能。通过 MapInfo Professional 可连接本地及服务器端的数据库 ,创建地图和图表以揭示数据背后的真正含义。也可以定制 MapInfo Professional ,以满足用户的特定需

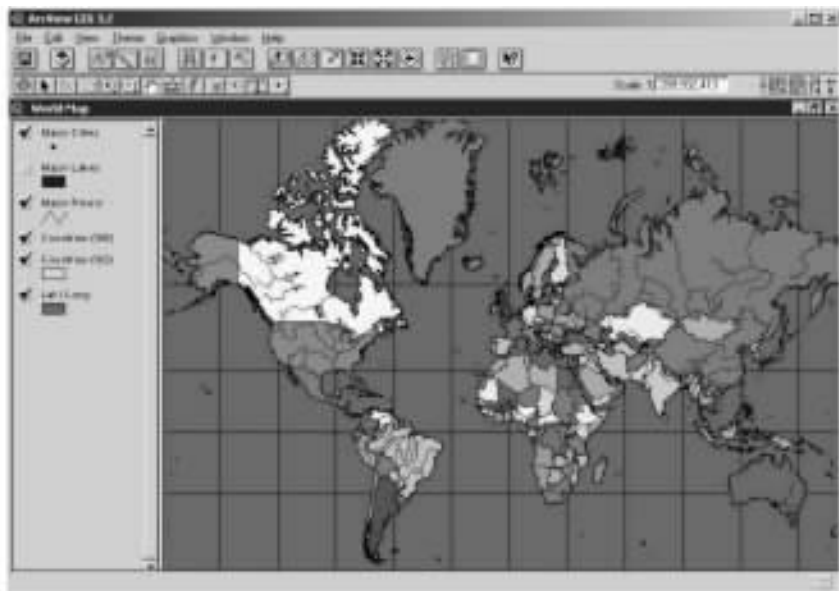


图 7-9 ArcView GIS 用户界面

要(如图 7-10 所示)。

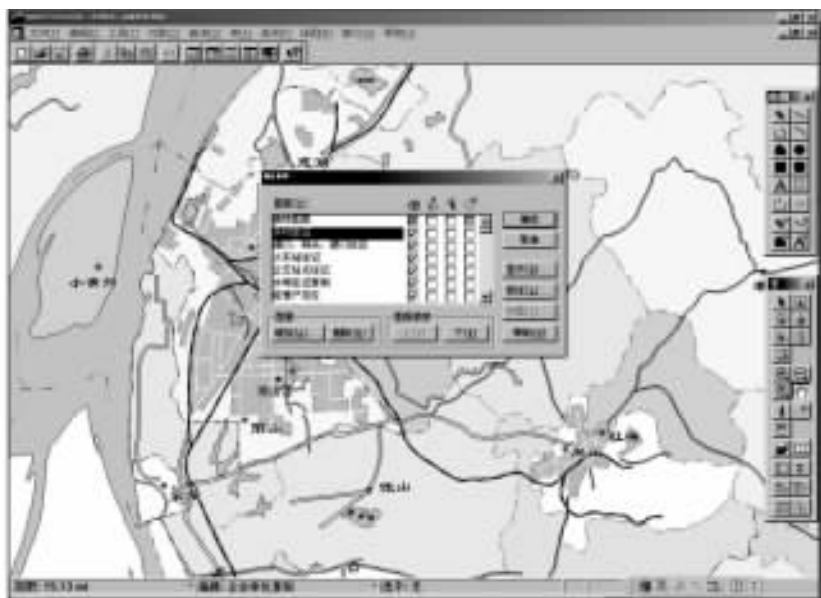


图 7-10 MapInfo Professional 用户界面

② MapBasic 是为在 MapInfo 平台上开发用户定制程序的编程语言,它为所有新功能提供了可编程界面,在创建图表、目标处理和控制、投影等很多方面都有了很大程度的增强。通过 MapBasic,用户可以使商业进程自动化,自定义菜单和界面来适应工业、商业的不同需

要。

③ 运行平台 Runtime 与 MapInfo Professional 使用同样的内核,功能与其完全一样,与 MapBasic 同样兼容。但没有菜单、工具条等用户界面,也没有内置的 MapBasic 窗口,没有二次开发能力,为大量使用开发好的 MapInfo 应用程序提供了经济的运行平台。

④ MapInfo Professional for SQL Server 是为那些支持 Microsoft SQL Server 共享空间文件的用户设计的。它利用 MapInfo Professional 的界面和功能来访问和分析 SQL Server 内部数据的单独产品。系统扩展了 SQL Server 在处理、安全和可测量性特性上的共享,这些特性为 MapInfo Professional 的桌面用户提供了更多的附加价值。MapInfo Professional for SQL Server 使最终用户能够感受在数据库支持方面的简单易用。并不是所有用户都必须通过 SpatialWare 将空间信息和非空间信息一同存储在 SQL Server 中来共享他们的数据,MapInfo Professional for SQL Server 是一个简便而经济的解决方案。

7.3.5 Genamap 软件

Genamap 是澳大利亚 GENASYS 公司开发的 GIS 软件产品,功能较强,具有较好的一致性、开放性和易操作性。与其他 GIS 专业软件包有很大不同,Genamap 被它的厂家称为数据库管理系统(DBMS)的“GIS 外壳”,因为这个软件是建立在使用流行的相关 DBMS 如 Oracle、Sybase、Informix 和 Ingres 的客户/服务器结构上的。可应用于自然资源调查和管理、地质勘测、城市规划和管理、环境监测和管理、交通管理,以及社会经济分析和管理等许多方面的计算机地图制图任务。Genamap 系统的主要功能模块包括:

(1) Genamap 模块

Genamap 模块是一个完整的矢量式 GIS 系统,是整个系统的核心模块,包含有空间型关系数据库,用于坐标存储、空间图形管理、投影变换、检索查询和图形显示。该数据库可自动建立库中每一图层内点、线、面以及各图层之间的拓扑逻辑关系,实现库内图形连续式的无缝拼接,通过各图形与属性数据库中的属性数据建立关系,可生成动态空间图形视图。其关系型属性数据库用做属性数据管理,以及与各层图形建立相关关系;其图示图例符号库用于存储和管理各类专用符号、图形,用户还可自造符号;各种图形编辑工具可用于数字化、校正、修改、拼接和更新各类图形。此外,还有多种空间分析工具,如网络分析、叠加分析、缓冲区分析、重新赋值分类分析、相邻分析等。使用其过程化程序设计语言 Script,可将 Genamap 系统中 10 个模块的所有功能组合成一个可执行程序,满足用户的各种需求。

(2) Genacell 模块

Genacell 模块是实现 Genamap 栅格图形数据处理、管理和分析的模块,既可独立使用,也可与 Genamap 组合使用。

(3) Genaref 模块

Genaref 模块是从 Genamap 向其他系统或其他系统向 Genamap 输送图形信息的格式转换和接口模块。它可以转换并接受 DXF、DLG、DGN、ARC Export、TIGER、Etak、SPOT/Landsat、USGSDEM、VPF、IGES 等系统的不同格式的图形数据文件。

(4) Genacomn 模块

Genacomn 模块是 Genamap 向外部商用关系数据库输送数据或调用外部数据库的接口模块。它可以连接 Oracle、Ingres、Informix、Sybase 和 IBM'S SQL/400 等关系型数据库系统。

(5)Genius 模块

Genius 模块是用户菜单生成和管理模块,它将 Genamap 系统中的图形库、命令过程和数据库组织在统一的菜单管理之下,具有下拉、弹出、覆盖、滚动等功能菜单。

(6)Genacarto 模块

Genacarto 模块是一个具有图形用户界面的综合制图工具模块,可快速生成、放置、编辑地图的各种元素和组合成分(图形的或非图形的),并作硬拷贝输出。

(7)Genavive 模块

Genavive 模块是一个矢量图、影像图集成显示和处理模块,能提供仿真图像显示环境,可显示和操作多种图像数据,并可以影像作为背景底图,在屏幕上用鼠标跟踪数字化,形成矢量图。影像显示格式直接支持 TIFF、JIFF 和 Genamap 栅格图。

(8)Genadoc 模块

Genadoc 模块是文档图像处理和管理模块。

(9)Genarave 模块

Genarave 模块是栅格图像数据自动矢量化模块。它既可与 Genamap 构成完整的系统,也可在 UNIX、Windows 平台上独立使用,是一个图形自动矢量化模块。它有许多功能特点,例如 细化图形线、等高线自动生成、点、线拓扑关系生成、线型识别、地图定位管理等。

(10)Genacivil 模块

Genacivil 模块是土木工程辅助设计和测量数据处理模块,具有工程制图、测量处理、二维模型建立、管线设计等功能。

7.3.6 SICAD 软件

SICAD 是来自德国的地理信息系统软件,自 1979 年进入 GIS 领域以来,如今已在德国占有 50% 以上的市场份额,是欧洲和世界上领先的地理信息系统之一。SICAD 已在制图、测绘、土地管理和网络设施应用等方面成功应用了多年,具有完善的产品系列,适用范围非常广,如市政(供水、供电、供热、排水、道路)、军事、地方城镇规划及土地管理等。其系统产品包括:

(1)SICAD/open

SICAD/open 是多年来地理信息系统领域的应用实践的结晶,并且融入了当今最先进的 workstation、服务器和数据库技术。SICAD /open 是市场上极少数在同类和不同类计算机网络环境下,支持分布式数据存储和分布式地理数据处理的系统之一。在 SICAD /open 系统中集成了各种各样的图形和工程应用,加上使用了高效的开发环境,该系统成为理想而完整的地理信息系统。

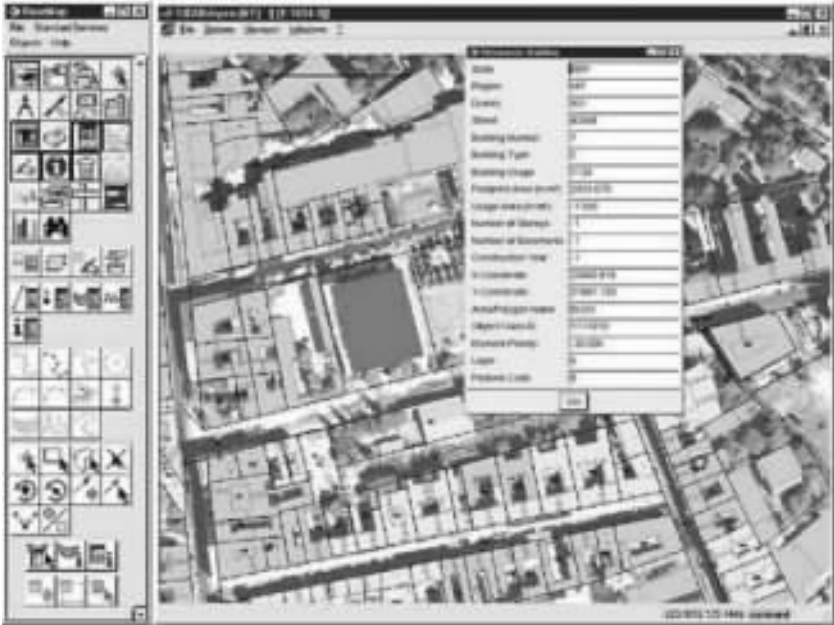
(2)SICAD-UT

SICAD-UT(SICAD-设施管理)是国际化的标准应用,可广泛应用于电力、煤气、自来水供应网络的档案管理、分析计算和规划设计。

(3)SICAD 互联网地图服务器

SICAD Internet Map Server 可以使互联网或企业内部互联网上的各类用户能够访问 SICAD 数据信息。

图 7-11 为 SICAD 的 GIS 界面。



(摘自博达新创网 <http://www.bdx.com/product/020.htm>)

图 7-11 SICAD 的 GIS 界面

7.3.7 GeoStar 软件

GeoStar 是武汉大学(原武汉测绘科技大学)开发的、面向大型数据管理的地理信息系统软件,主要功能包括图形矢量化、图形及属性编辑、空间数据库管理、拓扑关系建立与分析、与其他 GIS 的数据接口、专题图制作、三维模型制作等,涵盖了 GIS 必备的所有功能。其功能模块包括:

(1)GeoStar 模块

该模块是整个系统的基本模块,提供的功能包括空间数据管理、数据采集、图形编辑、空间查询分析、专题制图和符号设计、元数据管理等,从而支持从数据录入到制图输出的整个 GIS 工作流程。

(2)GeoMap 模块

GeoMap 模块由一个 OLE 控件——GeoMap 和一组近 20 个 OLE 自动化对象构成,应用于标准 Windows 开发环境,用户可以根据需要选择合适的开发工具。GeoMap 是基于 Windows NT 4.0 开发的,因此其开发平台也立足于 Windows NT 4.0,Windows NT 4.0 下的 Visual Basic、Delphi、PowerBuilder 等环境均适合利用 GeoMap 进行软件开发。

(3)GeoGrid 模块

该模块将数字高程模型、数字正射影像与常规的矢量数据和各种属性信息集成在一起,建立起一体化的三维数据输入、操作与可视化机制,为数据处理、空间查询与分析和各种三维模型操作提供了更加有力的支持。

(4)GeoTIN 模块

该模块是专门针对大比例尺地形数据的处理而设计开发的,它利用各种随机分布的地形单点(包括随机点和特征点)和特征线数据,快速、高效地建立不规则三角形网格(TIN)模型。TIN DEM数据来源可以是摄影测量、地面测量的数据,也可以是已有地图数字化或扫描数字化后形成的数据。由于顾及特征的 TIN 模型具有高逼真度和灵活方便的特点,可以广泛用于大比例尺地形图制图、土木工程设计、三维分析与仿真以及规则网格 DEM 的生产等领域。

(5) GeoDEM 模块

该模块是基于 OpenGL 1.1 开发的通用三维引擎,是场景浏览器、三维查询、三维分析的核心开发库。它是一套以数字高程模型(DEM)为主,集成数字正射影像(DOM)、数字线画图(DLG)作为综合处理对象的虚拟现实管理的二次开发平台。

(6) GeoImager 模块

该模块主要用于航空与多种卫星影像制图、地形图更新及其相关应用,同时为用户提供更复杂的、多波段的影像处理与分析、专题信息提取以及栅格与矢量数据的相互转换等功能。

(7) GeoSurf 模块

该模块是基于当今最先进的 Internet/Intranet 的分布式计算环境,系统各个模块采用部件化构造、分布式处理形式,同时以 Java/JDBC 构造了多数据源地理信息互操作中间件,不但实现了系统平台与硬件的无关性,而且矢量图形与主数据库的无缝连接,使得用户对地理数据可互操作以及异质数据透明地获取、操作。

(8) GeoImageDB 模块

该模块用于建立多尺度的大型遥感影像数据库系统,以管理省级和国家级多尺度、多数据源、多时相的航空、航天遥感影像数据为目标。

7.3.8 MAPCAD 软件

MAPCAD 是中国地质大学(武汉)开发的彩色地图编辑出版系统,实现了彩色地图的输入、编辑、出版全过程计算机化。其主要特点为:

- 灵活先进的图形输入功能(数字化仪输入、彩色扫描矢量化输入、GPS 输入);
- 强大独特的编辑及处理功能(拓扑处理、误差校正、投影变换、任意裁剪);
- 自由拓展的系统库(线型库、子图符号库、可选汉字库、花纹图案库、颜色库);
- 高质量的彩色输出功能(彩喷输出、PS、EPS 页面格式输出);
- 多种图形数据交换格式(如 ARC/INFO、MAPINFO、AUTOCAD、SPDF、DLG 等)。

MAPCAD 软件由 MAPCAD 桌面版、MAPCAD 测图版和 MAPCAD 工程版三级软件构成,每一级软件都比前一级功能更完善。

(1) MAPCAD 桌面版

MAPCAD 桌面版是 MAPCAD 系统的低级版本,主要提供了地图的输入子系统、处理子系统和实用服务子系统。MAPCAD 系统界面如图 7-12 所示。

在输入子系统中,提供了数字化仪交互采集、扫描矢量化输入、数据转换输入等多种数据采集方式,可以实现地图的分层管理,图形与属性数据的连接,交互式、半自动、全自动等多种矢量化采集,运用人工智能及模式识别技术,可以实现灰度地图、彩色扫描地图的矢量

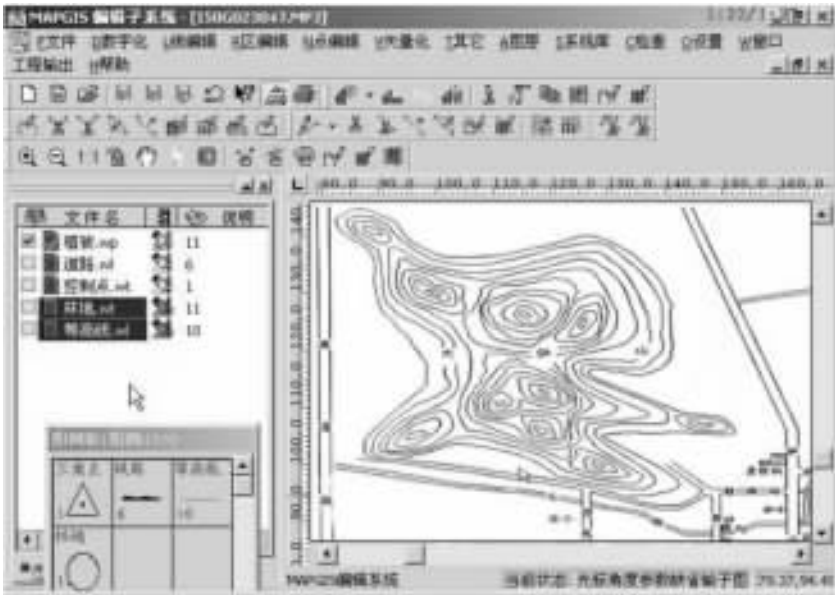


图 7-12 MAPCAD 系统界面

化,能接受 ARC/INFO ,MAPINFO ,SPDF ,AUTOCAD ,DXF ,DLG ,DGN 等多种格式的数据。

在图形处理子系统中,提供了基本 CAD 编辑(图元的创建、修改、删除等)、方便的图形窗口操作、智能查询编辑及图形参数替换、对图元的移动及定位功能,同时提供了模拟及数字操作两种方式,可以实现面元编辑、裁剪、拓扑编辑、地图投影变换、误差校正、显示坐标系设置等多种功能。

实用服务子系统是为地图编绘提供数据和接口服务的功能模块,可以借助强大的编辑功能,方便地建立专用的系统库(子图生僻字库、线型库、图案库、颜色库),增强地图的表现力;可以运用错误检查辅助功能,对线段自相交、图元参数、拓扑关系等进行错误检核,从而节约修编时间,提高数据质量;可以应用数据交换功能,进行 MAPGIS 与其他系统(如 CAD、GIS)的诸多格式数据的交换,如 DXF、MAPINFO、SPDF(国家数据交换格式)、ARC/INFO 公开格式、标准格式、E00 格式及 DLG 文件等。

(2)MAPCAD 测图版 MAPSUV

MAPSUV 在 MAPCAD 桌面版的全部功能基础上,增加了外业测图系统,可以将全站仪采集的数据存储在便携式或掌上式笔记本电脑、全站仪内存中,然后用 MAPSUV 读取,也可以直接读入山维、瑞得等格式的数据。

(3)MAPCAD 工程版

MAPCAD 工程版在 MAPCAD 测图版 MAPSUV 的全部功能基础上,着重增加了输出子系统,包括版面编排、WMF 显示输出、打印输出、印前出版输出、喷绘输出、电子报表输出等,也可以直接转换成指定分辨率的高清晰度的栅格图像。

MAPCAD 系统支持应用开发工具 VC ,VB 等,实施以上功能的二次开发。

7.4 计算机地图制图的应用研究进展

计算机地图制图的发展不仅仅是计算机图形学的问题,同时体现在理论、方法、技术与应用的各个层面,包括地图数据的采集技术、空间分析技术与方法、图形图像处理技术与方法、数据库管理技术与方法以及地图在产品形式上发展所涉及的计算机软硬件技术、通信技术等多方面因素。本节仅从数学理论方法的应用、与相关技术的结合和数字地图产品发展三个方面进行简略的阐述。

7.4.1 数学理论方法在计算机地图制图中的应用研究

现代地图学的研究内容已经清楚地表明,地图学不仅仅是地图制作技术的科学,也是地图理论研究、地图分析应用的科学,忽视任何一方面都会给其发展带来不利影响。地图学研究的定量(数学)化是现代地图学发展的主题之一。陈述彭院士认为,“20世纪的地图学,不仅只是简单的工艺技术,它已经建立在坚强的数学与地理学基础上,并参与到地学的各部门”。和现代地理学一样,定量(数学)方法的进一步应用将有助于现代地图学的系统化、科学化发展。吴忠性(1994)在“数学在我国地图学发展中的作用”一文中指出:“我国地图学各个时期的发展,都是由数学推动的。每前进一步,都离不开数学。一门科学向数学化方向发展,最终将成为一门精密科学。”

由于地图所表达的地理对象的复杂性,地图目标在形态、分布上仅仅依靠传统的经典几何学方法——欧氏几何学是难以描述和表达的,需要不断引入新的数学理论方法来解决计算机地图制图过程中尚没有解决的地图目标描述、分析与表达问题。但需要指出的是,在本小节中并不着重阐述数学理论方法的内容,而是简单介绍它在计算机地图制图中的有关应用,需要进一步了解这方面的知识请查阅相关的专业书籍。

1. 拓扑学与图论的应用研究概述

拓扑学是一门研究图形在连续变形下的不变性质的几何学。与其他较古老的几何学比较起来,它更为灵活和更具有可塑性,所以又被称之为“橡皮板几何学”。拓扑学与欧氏几何学的不同之处在于,拓扑学不关心图形的度量性质,如距离、方位或曲直等,而是在弹性运动(任意伸张、扭曲)中保持不变的拓扑性质。图7-13反映了两组“拓扑等价”的图形。

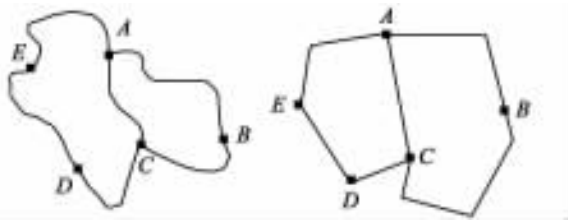


图 7-13 拓扑等价的图形

图论是拓扑学的一个分支,它研究在平面上可用点和线来表示的图形的性质,因此也称其一维拓扑学。作为图论的研究对象,图是一个抽象的代数系统。一个图的诸元素(节

点、弧段、面域)之间存在着两类二元关系:拓扑邻接与拓扑关联。图论的研究内容包括图的构成、表示、运算、连通性分析和存储等。

尽管拓扑学与图论的研究不依赖于地图制图,但是对于拓扑关系的研究成果可以应用于计算机地图制图和 GIS。它们的应用主要体现在地图(空间)数据库的数据模型、结构的建立方面。拓扑关系已成为反映地图目标之间空间联系的最基础、最关键的形式,在相当多数的 CAC 和 GIS 软件中都采用了以它为核心的数据模型。这种模型以节点、弧段、多边形为基本元素,目标之间共有的部分仅被保存一次,可以减少地图存储的数据冗余,以确保目标之间不会出现重叠、压盖、缝隙等冲突情况。

超图的概念是对图的概念的一种扩展。超图数据结构通过类别、物体、属性与关系这 4 种抽象数据类型,描述复杂的复合制图目标,包括与它们有关的空间的和非空间的属性与联系,特别适于专题地图数据的建模。

在地图分析和数据处理方面,拓扑关系可以提供地图目标之间的邻接与关联关系,从而有助于进行目标集范畴内的查询、编辑处理和任务分析。图论中关于网络路径分析的结构、算法在城市管线、交通、物流等专题地图制图和 GIS 中发挥了极其重要的作用。

2. 计算几何的应用研究概述

计算几何(Computational Geometry)诞生于 20 世纪 70 年代,是一门关于计算机中几何问题的算法设计和算法分析的科学。Shamos 和 Hoey(1975)最早利用计算机有效地计算了平面点集的 Voronoi 图,并发表了一篇著名的论文——“Closest-point Problems”(最接近点问题)。从那时开始,计算几何迅速地发展起来,其成果在计算机图形学、统计分析、模式识别、地图学与 GIS 等学科和领域得到了广泛的应用。

计算几何研究的内容主要由几何基元、查找、优化等几个方面构成。

(1) 几何基元

几何基元包括凸壳和 Voronoi 图、多边形三角剖分、划分问题与相交问题。凸壳是计算几何中最普遍、最基本的一种结构,点集 S 的凸壳是包含 S 的最小点集。 E^{d+1} 中点集 S 的下凸壳在 E^d 的投影正好是点集 S 在 E^d 中投影点的 Delaunay 三角剖分,由 Delaunay 三角剖分可以容易地得到 Voronoi 图。因此可以看出,Voronoi 图是凸壳的特例,两者在构造算法上的研究具有一致性。多边形的三角剖分研究如何设计复杂度低,或者三角形最小角最大化,或者分割线段长度之和最小的三角剖分算法。划分问题是多边形三角剖分的扩展,它把几何体划分成若干好的部分,即划分成尽量少的凸部分,或者各凸部分最小角最大化。相交问题主要研究如何对二维平面上的 n 条直线段确定其所有的相交线对,分析三维中两个凸多面体的交和多面体的交。

(2) 几何查找

几何查找包括点定位、可视化、区域查找问题。计算机图形学、数据库中的区域查找、计算机地图制图与 GIS 的点定位等都是典型的几何查找实例。

(3) 几何优化

几何优化包括参数查找和线性规划。参数查找技术是将一个优化问题的检验算法转变为求解过程,它必须满足某些条件,并且具有广泛的应用性。在线性规划方面,目前有确定变元数目的线性规划问题可以通过线性时间算法求解,但对于广义线性规划是否有多项式时间算法,还有待进一步的研究。

正因为计算几何关注几何问题求解的数据结构形式、算法实现与分析,它和以地理空间实体为研究对象的计算机地图制图和 GIS 迅速、紧密地结合起来,在地图数据建立、空间分析、制图综合和地图表达等诸多方面得到了很好的应用。

在地图数据建立方面,反映地形信息的 DEM 数据主要有正方形网格(Grid)和不规则三角网(TIN)两种组织形式,其中 TIN 具有分辨率可变、当地形表面相对单一时存储的数据量小和考虑重要表面数据点的优点,在三维可视化方面有着一定的优势。DEM TIN 建立一般通过把离散的高程点集或成组的等高线数据作为数据源来实现,在所有可能的三角网中,Delaunay 三角网在地形拟合方面的效果更为突出。自从 1977 年 Lawson 提出根据最大最小角度建立局部几何形态最优的三角网,以及 1986 年 Lee 和 Lin 提出了带约束条件的 Delaunay 三角网规则以来,已经先后建立了一系列的 Delaunay 三角网的构网及其改进方法,确保了 DEM TIN 建立的质量和效率。

在计算几何中,几何查找、交与并的运算等都是计算机地图制图和 GIS 中目标编辑和分析的常用功能,其算法效率的高低将直接影响到空间分析的效率。凸壳、Voronoi、TIN 等数据结构为地图目标间的复杂空间关系提供了强大的描述和分析依据,使得空间分析不再仅仅依靠直接邻接和关联的拓扑关系,而是可以进一步探测更复杂、更大范围的地图目标内部及目标之间的联系,分析其影响区域及其作用。通过运动规划中的最短路径分析,即首先建立可视图,然后由可视图寻求最短路径,可以对存在若干离散障碍物(多边形)的路径进行研究。

地图综合是计算机地图制图和 GIS 领域中的一个热点问题,是随比例尺减小而进行的信息化简运算。这种化简并非简单的图形缩小和数据压缩,而是需要顾及地图目标的特征信息和目标之间的空间联系,包括选取、化简、概括与位移的组合过程。这个过程往往依赖于对地图目标及其内部诸要素的重要性指标的分析和定量化。计算几何对空间分析功能的支持为地图综合奠定了基础,目前已取得了一系列的应用研究成果。在点群目标的综合中,对于聚集分布的点群,利用凸壳算法形成多层嵌套,以反映其逐层分布特征,从而将面状(群)目标的综合转化为线状目标的综合,为点群目标的结构化选取提供整体分布控制。建立点群的 Voronoi 图可以为各目标的区域性评价提供补充性的量化依据。在建筑物的几何合并中,首先利用 Delaunay 三角网或 Voronoi 图识别出需要合并的有关目标,特别是彼此分离的目标,然后进行几何合并。在河流自动综合中,双线河流综合涉及可能需要将其转化为单线河流及河间距计算的问题,借助于 Delaunay 三角网中的三角形的边互相连接所具有的空间邻接性特点,首先建立双线河流内部的 Delaunay 三角网,并在此基础上提取三角网的骨架线作为双线河流的中轴线,然后在河流之间构建 Delaunay 三角网,计算其中所有三角形的高,然后以这些高的平均值作为河间距。对于综合后的曲线相交和多边形(间)相交,通过 Delaunay 三角网可以进行冲突区域的判断和确定位移方法。

在地图表达方面,基于 Delaunay 三角网、Voronoi 图、凸壳和矩形几何的原理模型,可以设计完成地图综合中的选取、化简、合并、移位、骨架化等操作算子,通过制图综合建立与比例尺相适应的地图图形输出与显示。

3. 分形几何的应用研究概述

分形理论创立于 20 世纪 70 年代。1975 年 Mandelbrot 出版了法文版《分形对象:形状、机遇和维数》一书,成为分形几何学的第一本专著。目前分形尚没有一个明确的定义,一般

地,如果一个集合具有以下几个典型特征,就可以认为它是分形:①精细的结构;②不规则性;③近似的或统计的自相似性;④维数 $D > D_T$;⑤可以以简单的方法定义,并可能由迭代方法建立。分形维数(简称分维,Fractal Dimension)被定义来描述分形集的形态与结构的复杂性。它以 Hausdorff 维数为基础,从而将欧氏空间的整数维扩展到分数维。

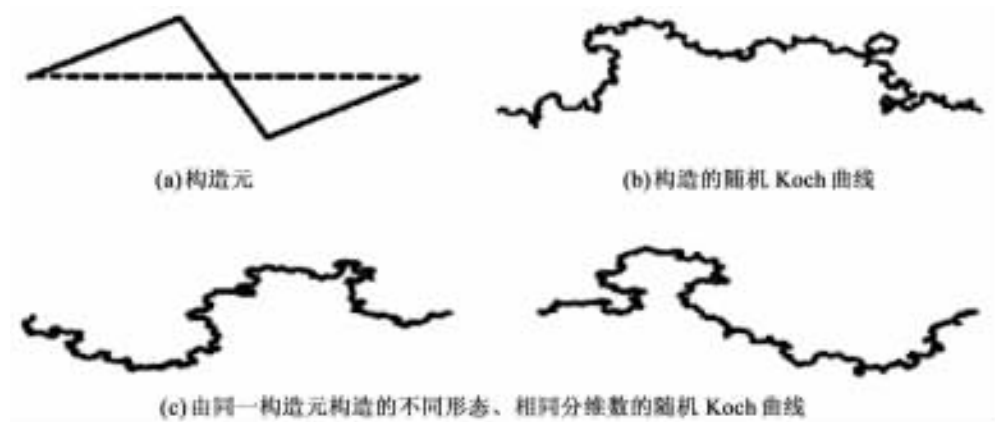
与欧氏几何学相比,分形几何学新的应用意义体现在:①分形学没有回避地理现象的复杂性与不规则性,并试图阐述这些复杂现象内部存在的有序性与规律性;②分形学将地图目标的宏观形态(结构)与微观形态(结构)联系起来,建立了可以从整体到局部,从局部到整体的目标分析方法;③分形学定义了普遍适用的量化指标——分维数,有效地反映了地图目标对空间的填充能力,从而建立了地图目标复杂性程度的评价方法。

分形关注地图学应该说非常早,Mandelbrot 的第一个分形问题——“英国的海岸线有多长?”实质上就是一个地图量测问题。因此,可以说,从分形几何学建立开始,各国地图学家就注意到分形理论在地图学领域的应用价值。早在 1980 年,Goodchild 就讨论了“分形与地图量算的精度”问题,以后人们利用分形理论进行了大量的地图学领域的相关研究工作。

地图目标空间信息的分维计算与分维分析是分形应用最基本的方式。研究表明,由于制图综合的作用,地图目标并非理想的分形对象,地图目标的分维值只有在无标度区间内有效,因此,顾及地图特征的分维计算方法的研究有着十分重要的作用,如应用道格拉斯-普克(Douglas-Peucker)方法来计算曲线的分维数,应用分数布朗运动原理计算地形表面的双参数(H, σ)等,其中 H 为表面复杂度指标, σ 为垂直比例因子,并可以进一步得到分维值。无标度区间的判定方法是分维计算的另一个关键问题,目前的主要方法有拟合误差法、轨迹分析法、总体拟合法、自相似比法、分维函数曲线法和基于遗传算法的误差拟合法等。在此研究基础上,运用多种分维计算方法计算点群目标、线状目标、面状目标的分维值,分析分维的几何性质以及分维变化与目标形态之间的相关规律,为地图分析和进一步的分形应用服务。

在地图表达的应用方面,利用分维的不变性可以建立新的插值方法,为制图物体增加细节信息,提高地图表达的真实性,这同样适用于二维和三维的情况,如海岸线、河流(单线河流或双线河流)、DEM 等。图 7-14 是利用一条直线和同一个构造元加上随机因子(随机取 +1 或者 -1,分别代表了弯曲向上或向下)构造的分维值相同的模拟海岸线。

分形方法对于几何对象空间信息的尺度变化规律的关注使之可应用于地图综合领域,其中主要包括对目标的正确选取和图形化简。一些要素的制图综合,通常首先采用方根规律模型进行数量的选取,其中两个系数——符号尺度修正系数和重要性修正系数的设置主要依靠经验和人为确定,而与要素本身的形状结构、分布特征无直接联系。而利用分形分析方法可得到各数量指标随比例尺变化的规律,从而建立方根规律的分形扩展模型。在地图曲线的分形综合过程中,采用不同的尺度 r 对曲线改变粗视化程度进行重新截取,得到不同的曲线长度 $L(r)$,可以建立 $\lg r - \lg L(r)$ 的 Richardson 曲线。以此为基础,根据综合前的曲线长度和综合前后的比例尺关系在 Richardson 曲线上得到综合后需要的曲线长度 L_1 ,由此反求 r 的取值,以得到的 r 为依据重新实施粗视化处理,最终可以建立长度为 L_1 的综合后的曲线。进一步的研究表明,地图目标并不是理想的分形对象,并不具有单一的分维值,而是分维随着尺度变化而变化,因此,可以把分维数 $D(r)$ 视为尺度 r 的函数,从而从扩展分维的角度研究地图对象的形态特征,并进一步指导地图综合的实践,确保制图综合的质量和自动化程度的不断提高。



$$(D_s = \lg 3 / \lg 2 \approx 1.3652)$$

图 7-14 模拟海岸线的一种随机 Koch 曲线

4. 小波分析的应用研究概述

小波分析(Wavelet Analysis)是 20 世纪 80 年代中期发展起来的新兴的数学理论方法,在数学领域被认为是调和分析,即现代傅立叶分析这一重要学科半个世纪以来的工作结晶;在其他科学技术领域,特别是在信号分析、图像分析、模式识别、计算机视觉和非线性科学等方面都被当做近年来在工具和方法上的重大突破。

小波分析是为了克服傅立叶分析的缺陷而引进的。因为傅立叶变换只能提供信号的整体描述而不能进行局部描述,窗口傅立叶变换虽然有了改进,但它的窗口是处处不变的。而小波分析在时域和频域同时具有良好的局部化性质,而且由于对高频成分采用逐渐精细的时域或空域取样步长,从而可以聚焦到对象的任意细节,因此又被人们称为“数学显微镜”。

小波分析在地图制图中的应用十分广泛,原则上讲,传统上使用傅立叶分析的地方,都可以采用小波分析来替换。借助于小波分析,可以提取多源、多尺度、海量的空间数据集的基本特征,并通过小波系数来表达,再作相应的处理与重构,就可以获得该数据集的优化表示。在解决地图自动采集问题的地图模式识别中,图像特征提取是一个重要环节,首先计算 x 、 y 方向的小波,然后通过基与小波变换的两个分量计算梯度矢量的模及幅角,进而检验出影像边缘特征。小波变换也可应用于影像地图的拼幅接边,即将要拼接的图像首先各自分解为不同频带的小波分量,然后在不同尺度下,选取不同的拼接宽度,把两幅图像按照不同尺度下的小波分量先拼接起来,最后再用恢复程序恢复出整个图像,可以同时兼顾到清晰度和光滑度两个方面。地图矢量数据压缩和制图综合是小波分析技术在地图学领域中应用最广的一个方向,因为小波的多分辨率分析可提供在不同分辨率下分析表达信息的有效途径,使其可应用于地图目标在不同比例尺或者不同详细程度下的数据处理。一般采用二进制、多进制、B 样条小波等对矢量数据进行压缩,如等高线数据综合(简化)。此外,小波在地图的图像变换、纹理分析、数据编码等方面都发挥着重要的作用。

7.4.2 与其他技术结合的应用研究进展

1. CAC 与遥感技术

在通常我们讨论 CAC、GIS 与遥感的结合应用时,都会指出遥感影像数据是地图制图重要、快捷的数据来源之一。经过图像纠正和投影变换的遥感影像可以经过目视判读或计算机自动识别转换为地图目标信息,直接用于地图的数据采集或更新。虽然这种应用模式从遥感诞生开始就一直沿袭至今,但在这一过程中仍然有大量的技术问题并没有彻底解决,如采集效率问题、自动化程度问题、识别正确率问题、原有地图数据库与更新数据的协调问题,等等。大量的研究一直尝试将新的理论技术,包括新的数学理论方法、图形图像处理技术和人工智能技术等应用于这一领域中,以提高地图数据获取的速度与质量。

遥感影像处理技术的发展不仅为 CAC 提供更快速的数据获取手段,也直接为栅格地图提供更强大的处理能力。同时,也有大量的研究探讨将 CAC、GIS 和遥感数据直接结合起来的方法,如通过格式转换实现 GIS 与遥感图像处理系统的结合,以及如何在图像处理系统中应用 GIS 功能或在 GIS 系统中更多地应用图像处理功能。

目前,随着遥感技术的日趋成熟和应用上的推广,遥感数据的种类、数量有了很大的发展,质量和时效性都大大提高。因此,直接利用遥感影像,在其上增加一些专题要素信息和注记,编绘成图,不仅可以极大地缩短制图周期,而且可以提供更翔实的地表信息,例如采用大比例尺的航空影像制作城市交通道路图等。

遥感不仅可以应用于普通地图要素的数据获取上,也可应用于量测各种环境变量,如农作物估产、植被类型与生长情况、降雨或灾害性天气分布等,这些量测得到的数据是专题地图的重要专题内容来源。

2. CAC 与 GPS 技术

GPS 是从 20 世纪 70 年代美国军方开始设计研制的卫星导航系统发展起来的,它依靠专门的设备和技术获取高精度的定位信息,这使得 GPS 可以为 CAC 及 GIS 的数据采集服务。服务有多种方式,既可以通过采集地表若干控制点的地理坐标,进行遥感影像数据的坐标纠正和精度分析,间接地提供地图数据采集的遥感数据源,保证了采集的地图数据的精度;也可以持 GPS 接收机沿目标地物移动,直接实施地图数据的采集。此外,GPS 还可以用于不可及或者有危险的目标测量方面,通过 GPS 和手持激光测距仪的结合,可以获得人们不能达到的目标的位置信息。

由于 GPS 获取定位信息的速度快,设备简单,便于携带,且操作、应用灵活,非常适于动态目标的快速定位,已广泛地应用于个人定位服务、城市交通导航、物流管理与调度、变形监测预报等各个方面,进一步扩大了 CAC 和 GIS 的应用领域。

3. CAC 与网络和移动通信技术

CAC 与网络技术的结合,一方面使得地图数据的传输、共享变得非常简单、迅速,分布式地图数据库系统支持同时来自不同客户端和采集设备终端的数据,并将它们有效地组织起来;另一方面使得直接在网络环境下表达的网络地图成为现实,通过网站、网页可以实现对地图信息的浏览、查询、分析与打印输出。目前,随着网络技术的进一步发展,国际上和国内比较大的网站都纷纷开辟网络地图专栏,也出现了一批专门的、通用的地图网站或专题地图网站,极大地推动了地图应用的发展。

网络技术与移动通信技术的结合,为 CAC 系统的应用提供了动态、可实时更新的信息服务模式。目前,空间定位信息服务(Location Based Service, LBS)已越来越多地为大众服务,它是在移动环境下,利用 GIS(含 CAC)技术、空间定位技术和网络通信技术,为移动对象提供基于空间地理位置的信息服务。如用户在随身携带的移动终端上,通过 LBS 功能,在电子地图上查询周围旅店、餐馆信息。传统的桌面 GIS(含 CAC)已不能适应 LBS 空间信息的动态性特点和为用户提供智能化服务的要求,从而提出了基于空间信息服务的分布式移动地理信息系统。

分布式移动 GIS(Distributed Mobile-GIS, DM-GIS)已成为 GIS(含 CAC)、GPS、无线通信和移动计算技术集成的产物。例如,野外工作队和移动车辆可用 DM-GIS 进行运算来确定最佳行动路线方案。DM-GIS 的应用领域包括紧急任务、公用事业、运输、营救工作、电信、科学研究、环境监测和规划。

4. CAC 与虚拟现实技术

虚拟现实(Virtual Reality, VR)技术是一个运用计算机技术和相关设备构建三维虚拟现实环境及其交互式应用的现代技术。CAC 与虚拟现实技术的结合,产生了虚拟现实地图。虚拟现实地图并非对二维平面地图在空间上的简单拓展,而是依托于对现实环境的认识、描述与表达以及与图形图像处理技术和设备、空间数据结构和关系模型等的结合。它以三维的、动态的地理场景为目标,涉及场景的数据组织、可视化表达、查询与分析等诸多方面。但是,虚拟现实技术的应用受到硬件设备、系统处理能力和空间目标复杂性的制约,还有待于进一步深入的分析研究。目前这方面的应用主要集中在城市仿真、军事活动、区域环境变化等方面。

5. CAC 与人工智能技术和专家系统

人工智能是研究如何运用计算机或智能机器模拟人的思维和行为方法解决问题的一门技术。它涉及知识库、推理机、控制机和人工智能分析与设计的理论、技术和方法的多个方面。专家系统是人工智能技术的一个应用研究领域,它表现为一个计算机程序系统,该系统具有相当于专家的知识 and 经验水平以及解决专门问题的能力。

人工智能技术和专家系统在 CAC 中的应用体现在地图模式识别、自动综合、智能化设计和表达输出的各个方面。地图模式识别是地图数据自动采集和自动理解的关键技术,它是一种智能行为,研究如何利用地图分类及其符号相关的知识和规则,从二维数字扫描图像中提取目标的色彩、形状和语义信息,从而实现了对地图目标的分类判别。在地图自动综合过程中,各要素相互之间的复杂关系形式直接影响到制图综合的实施方案和综合效果,只有在充分结合专家系统和自动综合算子的前提下,才能真正解决多要素相互影响、相互制约条件下的综合质量问题,目前这方面的研究还有待进一步的深化。地图设计和表达直接关系到地图制图的结果,如何得到一个分布合理、主题突出、既具有科学性又具有艺术性的地图产品,涉及众多的专家经验与思维过程,充分利用专家系统的知识与规则,是提高地图制图质量与效率的最佳途径。

7.4.3 数字地图的新产品

1. 三维电子地图

三维电子地图是对传统的二维平面地图的扩展,它既保持了平面地图的比例尺、地理坐

标和地图符号化的特点 ,同时又将表达的地图目标由二维空间发展到三维空间 ,构建结构更加复杂的三维地图模型和数据库 ,通过渲染虚拟再现地理环境 ,增强地图表达的直观性、真实性 ,提高地图应用的能力。

三维电子地图通常以 DEM 数据为基础进行建模 ,在其上建立三维地图目标的几何模型 ,如建筑物、道路、桥梁、植被、堤岸等 ,然后叠加纹理、符号、色彩等 ,以获得逼真的效果。同时 ,为了保证三维地图地理定位的准确性和可量测性 ,目标建模主要以平面地图的相应地理坐标为基础 ,增加第三维的高度信息。此外 ,三维点状地图符号与符号库建立、光照设置和场景渲染、模型的动态显示及航空影像的叠加显示等都是三维地图设计制作需要探讨的问题。

三维电子地图不仅包含了三维数据结构和模型 ,还包含了若干的应用功能 ,包括目标浏览、查询、编辑、模拟飞行、三维空间分析等。三维电子地图的特点体现在 :① 地形信息准确 ;② 地物表示详细 ;③ 地物表现直观 ;④ 浏览方便 ;⑤ 地形信息检索 ;⑥ 属性查询 ;⑦ 可以进行地形分析和量算 ;⑧ 可以模拟显示动态过程。目前 ,三维电子地图在城市、交通、军事、旅游、灾害监测等方面起到了重要的作用。图 7-15 显示了某三维电子地图的界面。

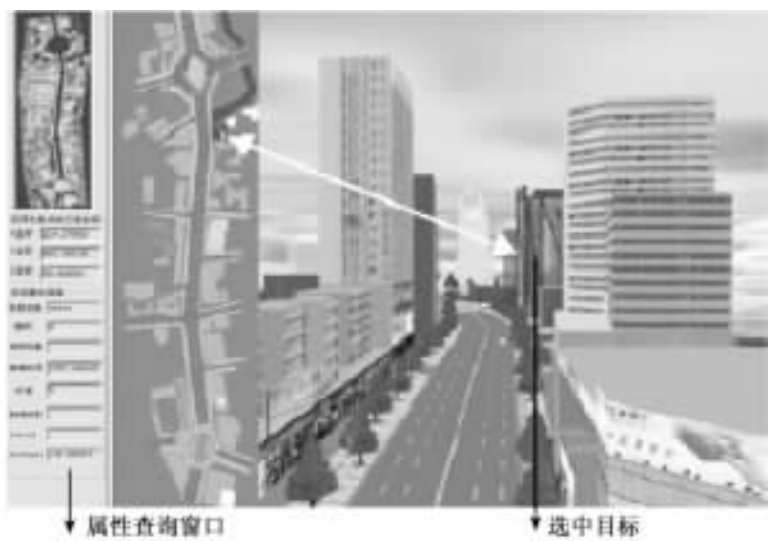


图 7-15 三维电子地图界面示例

2. 网络地图

计算机技术和网络技术的不断发展 ,促进了地图制图学的发展和变革 ,同时影响到了地图制作和出版发行等各个环节。这些发展之一就是地图与网络结合 ,产生了网络电子地图 (Web Map)。网络电子地图是以网络作为传播介质的新型地图 ,它具有传统地图以及单机电子地图无法比拟的优点。借助于计算机和 Internet ,用户可以不受时间和空间的限制 ,进行地图的浏览和查询 ,而且地图的现势性强 ,更新速度快 ,系统的维护、更新方便。图 7-16 显示了某网络电子地图的网页。

网络电子地图的特点包括 :

① 适应性强。由于网络电子地图是基于互联网的 ,因此它是全球性的。地图使用者可



(摘自武汉热线网的数字武汉地图 <http://map.wuhan.net.cn/#>)

图 7-16 网络电子地图网页实例

可以在世界上任意位置用互联网终端浏览、查询任意地区的地图,不受区域的限制。

② 现势性好。传统制图中,地图更新的周期长,费用高,所以不能满足使用者的要求。网络电子地图是在网上发布的,能对地图进行实时更新,因此人们可以通过互联网得到最新的地图。

③ 使用简单。网络电子地图用户可以直接从网上下载地图阅读软件以及获取所需要的各种地图,而不用关心地图阅读软件的开发、维护、更新和管理。

④ 数据共享、费用低。由于不同的生产单位使用的数据处理和制图软件可能不一样,在不同软件所采用的数据交换过程中经常出现产生冗余数据和丢失重要数据的现象,这就导致了数据的重复采集和处理,浪费了大量的人力、物力和财力。而网络电子地图能够避免数据的重复采集和处理,节约了费用,并且能够实现数据共享。

⑤ 利用超链技术。借助于互联网提供的超链技术,用户可以方便地查询空间信息以及相关的各种信息。

3. 移动导航电子地图

人类正进入以计算机、网络、卫星通信为特征的信息社会,人员流动的日益频繁使每个人对高精度空间信息有了更多的依赖,尤其是在户外和移动过程中,对地理信息的需求非常普遍。人们更加关心“当前我在哪里”、“目的地在哪里”、“如何到达”等问题。

移动导航电子地图是一种结合电子地图和实时的移动目标信息,支持导航服务的地图产品形式。它最大的特点是地图中记录和表达了移动目标的位置信息,这些信息通过 GPS 卫星定位、基站定位等方式实时获取并存储在数据库中。

移动导航电子地图主要应用在空间定位信息服务(LBS)和智能交通系统(ITS)等领域。

LBS 指的是在移动环境下,利用 GIS 技术、空间定位技术和网络通信技术,为移动对象提供基于空间地理位置的信息服务。例如,用户在随身携带的手机、PDA 等移动终端上,通过 LBS 功能,查询周围旅馆、餐馆信息;也可以通过 LBS 监测旅行者、儿童、老人的位置,避免危险的发生,等等。

智能交通系统(ITS)将成为今后交通发展的主要方向。作为智能交通系统重要组成部分的车辆导航成为了研究的热点。导航系统帮助司机选择最佳路径,提高效率,为司机提前提供道路信息,例如道路转弯、交通事故易发区的提示,降低交通事故的发生率。

全球定位系统是美国研制的导航、授时和定位系统,现广泛应用于农业、林业、水利、交通、航空、测绘、安全防范、军事、电力、通信、城市管理等部门。电子地图与全球定位系统的结合产生了多种应用,如车辆、船舶的监控、定位和导航等。

移动地图的意义在于将移动计算技术应用到传统的空间信息服务中,将革命性地改变传统的基于位置的服务机制,使作为主体的人、作为客体的真实世界以及经由网络传输的数字世界三者无缝地结合起来,实现不受任何时间和空间局限的互动,最终目的是根本性地改变人与数字世界、人与真实世界的交互方式,能够为任何基于空间的作业系统如导航、数字战场、野外采样、物流管理、智能交通、旅游、娱乐等提供全新的作业模式。

参考文献

- 徐庆荣等. 计算机地图制图原理. 武汉: 武汉测绘科技大学出版社, 1993
- 廖克. 现代地图学. 北京: 科学出版社, 2003
- 毋河海. 地图数据库系统. 北京: 测绘出版社, 1996
- 潘云鹤. 计算机图形学——原理、方法及应用. 北京: 高等教育出版社, 2001
- 毋河海, 龚健雅. 地理信息系统空间数据结构与处理技术. 北京: 测绘出版社, 1997
- 祝国瑞. 地图学. 武汉: 武汉大学出版社, 2003
- 龚健雅. 地理信息系统基础. 北京: 科学出版社, 2002
- 胡鹏, 黄杏元, 华一新. 地理信息系统教程. 武汉: 武汉大学出版社, 2002
- 郭仁忠. 空间分析. 武汉: 武汉测绘科技大学出版社, 1997
- 尹贡白等. 地图概论. 北京: 测绘出版社, 1991
- 邬伦, 刘瑜, 张晶等. 地理信息系统——原理、方法和应用. 北京: 科学出版社, 2001
- 黄仁涛, 庞小平, 马晨燕. 专题地图编制. 武汉: 武汉大学出版社, 2003
- 蔡孟裔等. 新编地图学教程. 北京: 高等教育出版社, 2000
- 孙立铸. 计算机图形学. 哈尔滨: 哈尔滨工业大学出版社, 2000
- 祝国瑞, 张根寿. 地图分析. 北京: 测绘出版社, 1994
- 汤国安, 赵牡丹. 地理信息系统. 北京: 科学出版社, 2000
- 总参谋部测绘局. 数字地图制图与地理信息工程. 北京: 解放军出版社, 2000
- 李志林, 朱庆. 数字高程模型. 武汉: 武汉测绘科技大学出版社, 2000
- 毋河海. 地图综合基础理论与技术方法研究. 北京: 测绘出版社, 2004
- 李德仁, 关泽群. 空间信息系统的集成与实现. 武汉: 武汉测绘科技大学出版社, 2000
- 王汝传. 计算机图形学. 北京: 人民邮电出版社, 2002
- 刘光运, 韩丽斌. 电子地图技术与应用. 北京: 测绘出版社, 1996
- 张文诗. 电子地图及其应用. 解放军测绘学院学报, 1994 年, 11(1)
- 陈毓芬. 电子地图的空间认知进展. 地理科学进展, 2001, 20(增刊)
- [美] David J. Kruglinkski 著. 潘爱民, 王国印译. Visual C++ 技术内幕. 北京: 清华大学出版社, 1999
- 周鸣扬. Visual C++ 界面编程技术. 北京: 北京希望电子出版社, 2003
- 董士海等. 图像格式编程指南. 北京: 清华大学出版社, 1994
- 徐青. 地形三维可视化技术. 北京: 测绘出版社, 2000
- 严桂兰, 刘甲耀. C 语言与图形处理. 上海: 华东化工学院出版社, 1993
- 马启文. 高级 C++ 图形/图像编程技术. 北京: 学苑出版社, 1994
- 毕硕本, 王桥, 徐秀华. 地理信息系统软件工程的原理与方法. 北京: 科学出版社, 2003

- 范玉顺,曹军威. 复杂系统的面向对象建模、分析与设计. 北京:清华大学出版社,2000
- 吴立新等. 地图符号库的面向对象技术与引用接口设计. 矿山测量,1999(1)
- 张海藩,牟永敏. 面向对象程序设计实用教程. 北京:清华大学出版社,2001
- 祝国瑞,苗先荣,陈丽珍. 地图设计. 广州:广东地图出版社,1993
- 党安荣等. ArcGIS 8 Desktop 地理信息系统应用指南. 北京:清华大学出版社,2003
- 郭达志等. 地理信息系统基础与应用. 北京:煤炭工业出版社,1997
- 喻沧. 陈述彭院士的若干地图学学术思想. 中国测绘,1999(6)
- 吴忠性. 数学在我国地图学发展中的作用. 地图,1994(2)
- 周培德. 计算几何——算法分析与设计. 北京:清华大学出版社,2000
- 毋河海. 地图信息的分形描述. 测绘通报,2001(2)
- 毋河海. 分维扩展的数值实验研究. 武汉测绘科技大学学报,1998,23(4)
- 王桥,毋河海. 地图信息的分形描述与自动综合研究. 武汉:武汉测绘科技大学出版社,1998
- 王桥. 分形理论在地图图形数据自动处理中的若干扩展与应用研究. 武汉大学博士学位论文,1996
- Kaye B H 著. 徐新阳等译. 分形漫步. 沈阳:东北大学出版社,1994
- 龙毅. 扩展分维模型在地图图形特征描述中的应用研究. 武汉大学博士学位论文,2002
- 应申,李霖. 计算几何在地图综合中的应用. 2004 年全国博士生学术论坛文集,2004
- 刘贵忠,邸双亮. 小波分析及其应用. 西安:西安电子科技大学出版社,1997
- 朱长青. 小波分析及其在地图制图中的应用. 解放军测绘学院学报,2002,12(1)
- 曾文曲等. 分形理论与分形的计算机模拟. 沈阳:东北大学出版社,1993
- 张济忠. 分形. 北京:清华大学出版社,1995
- 吴纪桃,王桥. 小波分析在 GIS 线状数据图形简化中的应用研究. 测绘学报,2000,29(1)
- 李水根,吴纪桃. 分形与小波. 北京:科学出版社,2002
- Paul A. Longley, Michael F. Goodchild 等著. 唐中实,黄俊峰等译. 地理信息系统. 北京:电子工业出版社,2004
- 奚大平,江文萍. 数字地图的三维可视化研究及其若干应用. 地球科学——中国地质大学学报,2002,27(3)
- Whalley W B, Orford J D, 1989, The Use of Fractals and Pseudofractals in the Analysis of Two-dimensional Outlines: Review and Further Exploration, Computers & Geosciences, 15(2): 185 ~ 197
- <http://www.go2map.com/product/mapinfo-product/professional.asp>
- <http://www.huady.com/cpjs/cpjs.htm#a02>
- <http://www.mapgis.com.cn/product/index1.htm>
- http://www.geostar.com.cn/bbs/forum_posts.asp?TID=426&get=last