

车辆导航系统的设计与实现

Design and Implement of Vehicle Automatic Navigation System

汪永红 罗军宏 张有为 (信息工程大学 河南郑州 450052)

摘要:本文主要介绍了一个基于 PDA 的在 Windows CE 上实现的车辆导航系统,它可以实时的为移动对象提供基于地理空间位置的综合信息服务。文章着重讨论了它的数据模型、空间数据结构、系统集成方式、系统主要功能、地图匹配和最短路径的实现方法。

关键词:数据模型 地图匹配 最短路径

利用 GPS 接收机的实时定位技术和移动终端处理空间地理信息的能力,可以组成 GPS + GIS 的各种电子导航系统,给车辆、轮船等交通工具的导航定位提供了实时的定位能力。为此,我们开发了一套基于 PDA 的车载导航系统,它是在 Windows CE 上实现的嵌入式 GIS,它能够在移动计算环境下,利用空间定位技术、嵌入式技术和地理信息系统技术,以数字地图为显示背景,实时的为移动用户提供基于地理空间位置的信息服务的综合性数字化信息系统。它可以在很大程度上提高交通的安全水平及道路网的通行能力,对社会和部队的现代化建设都具有重要意义。

1 系统数据的组织方式

为了使系统取得合理的响应时间,空间数据应根据应用目的、数据特点进行合理组织。由于嵌入式设备任意时刻屏幕显示的图形数据只是读入数据的一部分,因此适当减少非屏幕显示区域的数据,并不影响屏幕图形数据的显示。在嵌入式系统中一般采用矢量数据分块的方式对数据进行存储和管理,这种数据组织方式将空间矢量数据分为 N 份(在数据分块时要根据实际的屏幕显示尺寸和比例尺综合考虑块的大小),任意时刻屏幕显示图形数据时,只是读取部分数据以满足快速显示图形的要求。此外,系统应采用一些数据压缩方法以提高访问速度。

1.1 数据模型

该系统的数据由若干个数据卷组成,每个数据卷又包含多个数据集,一个数据集就是某个比例尺的交通地图数据。经过分析比较,我们确定了五种比例尺:1:1万、1:5万、1:25万、1:100万和1:400万。其中1:1万和1:5万的数字地图存储的是某个城市数据,1:25万的存储的是某个省的数据,1:100万和1:400万存储的是全国范围的数据,这样就可以组成一个从全国范围到某个城市区域内的各种详细程度的交通地图数据,经过试验,这种结构能够满足车载导航系统的需求。

一个数据集又按矩形分块方式划分成若干数据块,每个数据块包含若干地理要素层,每一要素层包括一组在地理意义上相关的地理要素。每个要素层之间在数据组织和结构上相对独立,数据更新、查询、分析和显示等操作以要素层为基本单位来执行。

数据集、数据块、要素层和地理目标构成了一个层次地理数据模型框架,如图1所示。在每一个数据块上建立自己独立的拓扑关系,数据块之间通过经纬度或矩形分块建立邻接相关关系。

1.2 要素分类

由于该系统主要用于车载导航,因此系统主要给出了与道路环境有关的地理实体的定义。根据系统的特点将地理要素分为以下六类:道路和渡口、行政管理区域、居民地和命名区域、土地覆盖和利用、铁路和服

务机构(设施)。系统还进一步规定:在地图比例尺大于或等于 1:10000 时,保留服务机构(设施)要素层;小于这个比例,该要素类别就不保留。

1.3 空间数据结构

该系统采用矢量数据结构。由于嵌入式系统存储器容量有限,而 GIS 系统数据量很大,为提高系统性能,采用了如下一些数据压缩方法。

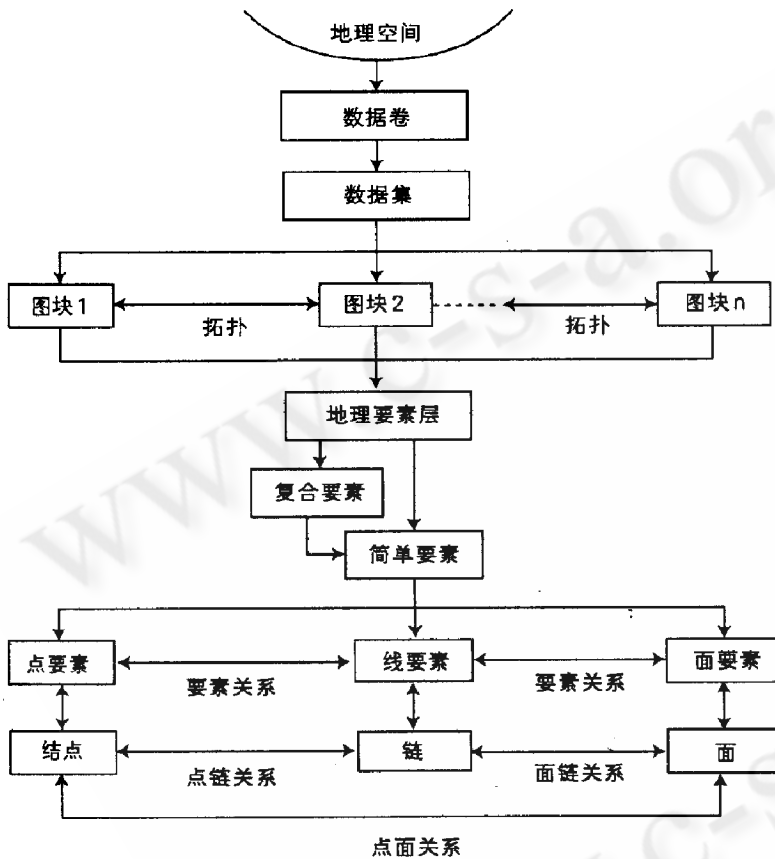


图 1 空间数据模型框架

(1) 将双精度型或浮点型的坐标映射为整型坐标,数据量可以减少一半。

(2) 每一条弧段(包括区域边界和线状地物)只记录其起点坐标(x,y),其后续点坐标用相邻两点间的x,y的偏移量代替,即弧段除起始点以外的其它后续点只需用短整型记录,这样就可以将数据量压缩到原来的1/4左右。

(3) 原始电子地图上,弧段的型值点采样很密集,在其变换到所指定的屏幕后,这些型值点就更加稠密,

以至于弧段上前后相随的几个点在屏幕上落在同一个像素点上,系统实现时只保留弧段上落在同一个像素上的一个点,采用滤点压缩的方法去掉其它的点。

1.4 索引数据结构

由于嵌入式设备显示屏幕较小,不可能像台式机一样显示大范围的地理要素。对于用户来说,要在图幅内漫游一遍来查找自己感兴趣的地理目标是非常低效的。为提高查询速度,系统建立了一种基于行政区划的索引机制。

索引数据文件包括文件头和数据区。在文件头中,首先写入行政区的个数、每个行政区的名称以及它在数据区的入口地址,然后在每个行政区的开始处记录有所有目标图层的入口地址。对于 1:10000 或更大比例尺的电子地图,目标图层包括居民地、道路和服务设施;对于比例尺小于 1:10000 的,目标图层只包括居民地和道路。通过每一目标图层的入口地址,系统能找到行政区内这一图层包含的所有目标信息。建立这样的索引后,在嵌入式 GIS 上进行查询时,系统会自动列出所选行政区所选要素层的全部要素。

2 系统主要功能

该系统主要由嵌入式 GIS 和 GPS 模块组成,通过 CF 卡插槽连接 PDA 和 GPS 接收机,实现 GIS 和 GPS 的集成。模块之间采用无缝集成方式,即集成的几个系统模块共用一个主界面,核心处理模块相互独立,彼此之间可以相互通信。该系统主要实现了以下功能:

(1) 地图显示功能。提供电子地图的显示、多级比例尺的缩放、漫游等基本功能。为提高响应速度,我们采取两种方法来优化程序。一是将视口范围与数据块范围作比较,如果数据块矩形和视口矩形存在交集,则读取该数据块,否则跳过;二是在读取数据块的地理要素时,如果要素的最小矩形与视口矩形存在交集,则显示该要素,否则不显示。实验结果表明优化后的程序基本能够满足用户的要求。

(2) 图层管理功能。控制所有图层的打开/关闭、显示/隐藏。用户可以根据需要打开或关闭图层,也可以仅仅打开或关闭图层的显示。后者避免了频繁调入图层数据,加快了数据的显示速度。在进行漫游或缩放等需要重绘窗口的操作时,可以适当关闭一些图层以提高响应速度,这在 GPS 导航中是非常重要的。因为 GPS 数据每一秒刷新一次,如果不能在一秒钟内将定位信息显示在电子地图上,系统将一直陷于响应状态,无法实现定位和导航的功能。

(3) 查询检索功能。系统提供两种查询方式。一种是在屏幕地图上经由触摸笔点击得到查询目标的相关属性。另一种是基于行政区划的名称查询,查询目标将被重新定位在屏幕地图上。

(4) 分析量算功能。能进行距离、面积量算和最短路径查询等功能。

(5) GPS 定位与导航功能。由 GPS 得到移动目标的定位信息,并根据地图匹配算法将目标位置显示在屏幕电子地图上,能进行轨迹记录、轨迹重放和数据采集等操作。

3 关键算法实现

3.1 地图匹配算法

3.1.1 算法的基本思想

地图匹配的基本思想是通过将车辆的 GPS 航迹与电子地图上矢量化的路段相近匹配,寻找当前行驶的道路,并将车辆当前的 GPS 定位点投影道路上。这样既保证了不会因为定位误差使车辆定位点偏离车辆当前行驶的道路,而且通过投影使车辆定位数据仅残留定位误差在车辆前进方向上的径向分量,从而提高车辆的定位精度。

地图匹配算法从原理上可以分为两个相对独立的过程:(1)寻找车辆当前行驶的道路(如果当前道路为已知,则可以省略这一步骤);(2)将当前定位点投影到车辆行驶的道路上。

3.1.2 算法描述

地图匹配算法分为初始定位和动态定位。初始定位主要用于系统开始启动时或当车辆方位失去控制时

进行车辆的位置判断。动态定位一般用于车辆导航。算法描述如下:

(1) 初始定位

① 获得车辆的当前定位数据 P_g 。

② 取出该车辆 $N(N \geq 10)$ 点最近期定位数据纪录,按时间顺序连接成曲线 l ;取所有与 l 距离为 e (e 为预先设定的定位数据误差的平均值) 的邻近区 $Buffer(l, e)$;搜索所有在 $buffer(l, e)$ 内的路段,载入集合 $S(r)$ 。

③ 在 $S(r)$ 中搜索所有可能路线与曲线 l 相匹配,求得最佳匹配路段作为当前运行路段 R 。

④ 记当前定位数据 P_g 在行车路线上的投影为当前定位点 P 。

(2) 动态定位

① 获得当前车辆定位数据 P_g ,计算 P_g 与上一定位点的距离作为车辆的行程 L 。

② 从前一时刻定位点 P 开始,如果已知道路 R 的前进方向上的终止点在车辆行程距离 L 内,则当前行车路线不变,即 P_g 在 R 上的投影为 P ,转入④;否则,说明前面有道路交叉,转入③。

③ 作 P_g 到所有搜索到的路段的投影,记投影距离最短的路段为 R' ;记 R' 为行车路线 R ;考虑到车辆大角度转弯时必须减速的实际情况,当在前进方向上 R 与 R' 的夹角小于 30° 时,记 P_g 到 R' 上的投影为 P ,否则,记搜索到的距离 P_g 最近的路段交叉口为 P 。

④ 判断 P 是否在 P_g 的邻近区 $Buffer(P_g, e)$ 内,是则转入(1),否则转入⑤。

⑤ 转入初始定位算法的②。

该算法的定位精度与实际的道路状况有很大关系,当道路较复杂时,误差较大。此外,当道路网数据量很大时,该算法的运算量较大。为提高定位精度,可在进行道路匹配时引入行驶方向和拓扑约束条件。为减少运算量,可对道路网进行分级管理,因为市区内的交通干道是车辆行驶的主要路线,在实现车辆路径定位时可以对主干道优先搜索,计算匹配度量,只有当匹配度量值超过一定的经验门限时,才去搜索较低等级的道路。这一方面减少了判断失误,另一方面也降低

了运算量。

3.2 最短路径算法

3.2.1 算法的基本思想

最短路径问题是车载导航系统中的一个重要问题,经典的图论与不断发展完善的计算机数据结构及算法的有效结合使得新的最短路径算法不断涌现,它们在空间复杂度、时间复杂度、易实现性及应用范围等方面各具特色,其中 A* 算法由于在路径规划中引入启发信息而得到了广泛的应用。针对车辆自导航系统的特点,我们使用改进的 A* 算法进行最短路径选取^[2],其核心思想是:将当前临时标记结点到源点的最短路径、当前临时标记结点到目标结点的最小费用和当前临时标记结点前一点到目标结点的最小费用三者之和作为此临时标记结点的属性值,这个属性值将作为从临时标记结点集中选取永久标记结点的依据,这种优化方法称为改进的 A* 算法,其当前临时标记结点 i 的估计函数定义为:

$$f^*(i) = g(i) + h^*(i) + h^*(i-1)$$

其中 $g(i)$ 是从起点到当前临时标记结点 i 的实际费用的量度, $h^*(i)$ 是从当前临时标记结点 i 到终点的最小费用的估计, $h^*(i-1)$ 是从当前临时标记结点前一点到目标结点的最小费用。

3.2.2 算法描述

我们采用优先级队列来实现这种最短路径算法,为了避免搜索时对未标号弧的重复搜索,具体实现时,我们通过用各弧在算法中不断被松弛的距起点的最小费用值来构造优先级队列,它删除的是累计权值最小的元素。设图存储后获得的链表为 $pLinkLinkList$, 优先级队列为 $pPathInfoList$, 临时路径队列为 $pTempWayList$, 回溯追踪的路径队列为 $pWayList$ 。算法描述如下:

(1) 判断 $pLinkLinkList$ 中弧的首点是否等于起点,是,则求出 $pPathInfoList$ 中的最小值,将最小值删除并添加到 $pTempWayList$ 中;不是,则终止;

(2) 判断 $pPathInfoList$ 中弧的末点标识是否等于终点,等于,则执行(4);不等于,则判断当前弧是否被

遍历过,是,则执行(3);不是,则对当前弧进行标记,获得当前弧的关联弧结构,判断关联弧是否被遍历过,是,则执行(3);不是,则将该关联弧添加到 $pPathInfoList$ 中;

(3) 求出 $pPathInfoList$ 中的最小值,将最小值删除并添加到 $pTempWayList$ 中,转入(2)执行;

(4) 从 $pTempWayList$ 尾部向前搜索,查找末点标识等于终点的弧,将其添加到 $pWayList$ 中;

(5) 判断 $pWayList$ 中弧的首点标识是否等于起点,是,则终止。不是,则执行(6);

(6) 从 $pTempWayList$ 尾部向前搜索,查找末点标识等于 $pWayList$ 中弧的首点标识的弧,将其添加到 $pWayList$ 中;

(7) 判断 $pWayList$ 中弧的首点标识是否等于起点,是,则终止。不是,则转入(6)执行。

经过分析,该算法的时间复杂度为 $O(n \log n)$,而传统的基于邻接矩阵或邻接表的 Dijkstra 算法的时间复杂度为 $O(n^2)$ 。经测试,该算法具有较好的运行效果。

4 结束语

本文主要介绍了一个基于 PDA 的在 Windows CE 上实现的嵌入式 GIS,它可以实时的为移动对象提供基于地理空间位置的综合信息服务。由于时间仓促和实验手段不足,目前我们所做的系统并不完善,在以后的工作中还应该完善数据模型、改进地图匹配算法,最终在 linux 操作系统上实现该系统。

参考文献

- 1 陈涛,车辆导航系统中大区域路径规划算法的设计与实现[D],测绘学院硕士论文,2005。
- 2 段莉琼,基于城市交通网络的路径分析与应用[D],测绘学院硕士论文,2004。
- 3 高博,车辆导航系统中数据处理、地图匹配和路径规划的研究,测绘学院硕士论文,2001。