

文章编号: 1001-2486(2003)01-0061-04

一种改进的 AVL 系统中车位显示修正算法*

聂洪山¹, 徐捷², 王鸿谷²

(1. 国防科技大学 ATR 实验室, 湖南长沙 410073; 2. 国防科技大学信息工程研究所, 湖南长沙 410073)

摘要: AVL 系统就是 GIS 与 GPS 技术集成的成功应用。AVL 系统中 GPS 信号在 GIS 地图上显示时, 由于各种因素的影响, 存在很大的偏差。对这种偏差的校正是十分必要的。为了解决这个问题, 对以前提出的误差校正算法——“线性抽象算法”进行了改进, 提出了另一种误差校正方法——“缓冲区线性抽象算法”, 实验结果表明这种方法能在很大程度上校正偏差, 不但具有较高的精度, 而且还提高了算法效率。

关键词: AVL 系统; GPS 数据; GIS; MapInfo 地图; 道路拟合

中图分类号: TP399 **文献标识码:** A

A Study on the Corrective Algorithm of Displaying Points of Vehicles in AVL Systems

NIE Hong-shan¹, XU Jie², WANG Hong-gu²

(1. ATR Lab, National Univ. of Defense Technology, Changsha 410073, China;

2. Institute of Information Engineering, National Univ. of Defense Technology, Changsha 410073, China)

Abstract: AVL system is a successful application of the integration of GIS and GPS. When GPS data is displayed on a GIS map, logically the vehicle point should be displayed on the roads. But because of the effect of errors, there are always warps between the true vehicle point and the displayed point on the map. It is necessary to correct these warps. An algorithm—Buffer Linear Abstraction Algorithm (BLAA) is put forward. This algorithm improves a former algorithm—Linear Abstraction Algorithm (LAA). Experimental result shows that BLAA has high precision, and at the same time it is more efficient.

Key words: AVL system; GPS data; GIS; MapInfo map; road fitting

目前, 国际上 3S(GIS, RS, GPS)的研究和应用正向着集成化方向发展。在这种集成应用中, GPS 主要用于实时快速地提供目标的空间位置, RS 用于实时地或准实时地提供目标及其环境的语义或非语义信息, GIS 则是对多种来源的时间、空间数据进行综合处理、集成管理、动态存取^[1]。GPS/AVL(全球定位/自动车辆定位)系统就是 GIS 与 GPS 技术集成的成功应用。

在 GPS/AVL 系统中, 当 GPS 移动目标在地图上显示时, 逻辑上车辆应该在道路上行驶, 但由于地图本身不够精确、地图数字化误差、GPS 系统误差、GPS 接收机误差等的影响, 移动目标在地图上显示时并不能保证其在道路上, 有时甚至相差很远, 而移动目标的精确定位正是系统的关键, 这就要求人为地对其进行校正(校正过程也称为道路拟合)。作者在这个问题的研究过程中又提出了一种“缓冲区线性抽象算法”, 对作者前面提出的“线性抽象算法”^[2]进行了改进。

1 电子地图的预处理

为了达到校正偏差的目的, 在“线性抽象算法”和“缓冲区线性抽象算法”中需要根据算法设计的要求对地图进行加工, 包括创建公路线性图层、创建交叉路口图层以及对公路图层进行处理^[2]。这个过程要能够体现地图本身的特点并且要符合实际情况。

* 收稿日期: 2002-08-31

作者简介: 聂洪山(1975—), 男, 博士生。

2 线性抽象算法

“线性抽象算法”的基本思想是找到离车辆点最近的公路线性图层中的一个线段对象,然后用点到直线的垂直点算法计算得到位于该线段上的点,也就是所求的校正点。其算法流程如下:

A. 以车辆点为圆心选取适当的值为半径画圆(这个圆不必显示出来),判断该圆与交叉路口图层中的对象是否相交,如果在查找过程中发现车辆点到离它最近的交叉路口点的距离满足给定的条件(在圆范围内),则停止查找过程,直接将车辆校正点画在该交叉路口上,取下一个车辆点,返回 A。

B. 以车辆点为圆心,选取适当的值为半径画隐含的圆,判断该圆与公路图层中的对象是否相交。如果相交,则计算公路图层中有几个对象与该圆相交;如果找到多于一个对象,则适当减小圆半径;如没有满足条件的公路图层对象,则适当加大圆半径,直到找到惟一个满足条件的对象,记忆该对象的公路名称字段(记忆功能)。

C. 判断本次循环中取得的公路对象的名称和上次循环中取得的公路对象的名称是否相同,相同则表示两个点在同一条公路上(公路的名称一致),转到 D; 如果不同,那么车辆很有可能转弯了,但是不好判断车辆是向左转还是向右转(见图 1 和图 2),所以采取对这个点不进行处理,跳过这个点,但仍记录和车辆点最近的公路对象的名称以备下一次循环时比较之用。取下一个点,返回 A。

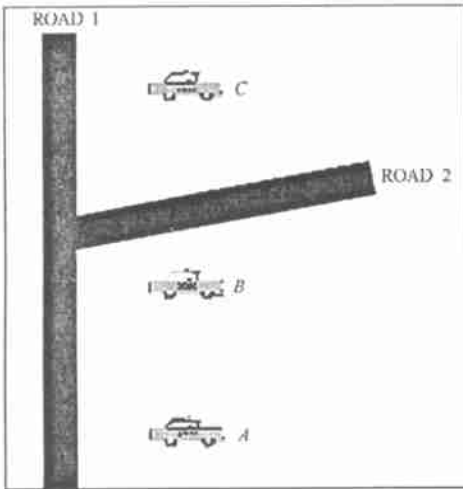


图 1 C 的校正点应放在公路 1 上

Fig. 1 The corrected point of point C should be put on road 1

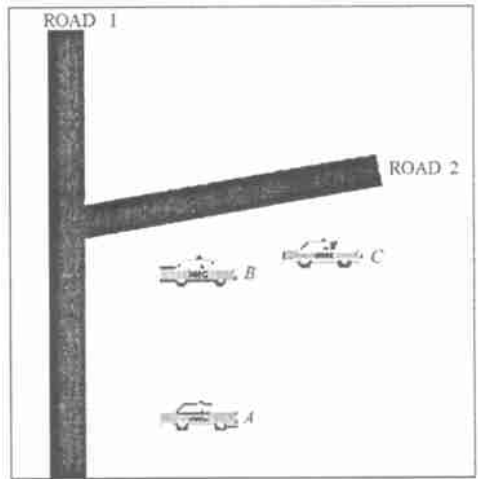


图 2 C 的校正点应放在公路 2 上

Fig. 2 The corrected point of point C should be put on road 2

D. 以车辆点为圆心、选取适当的值为半径画隐含的圆,按照与 B 相似的办法找到离车辆点最近的公路线性对象(即公路线性图层中的对象)。

E. 在 MapInfo 中可以取得公路线性图层中直线段对象的两个端点,已知车辆点和直线段两个端点坐标可以求出车辆点到该直线段的垂直点,这个垂直点就是想要的校正点。如图 3 所示。

F. 在求得的垂直点上画一个标志点作为车辆点的校正点。取下一个车辆点,返回 A。

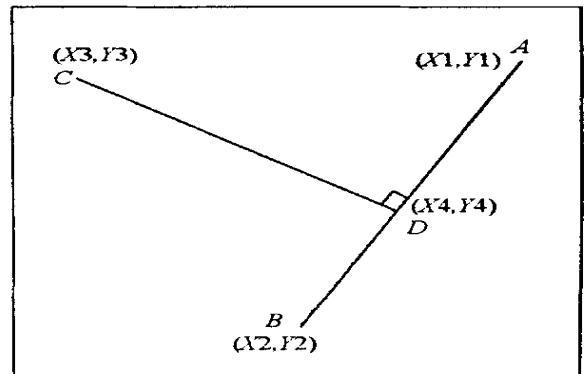


图 3 求点到直线的垂直点

Fig. 3 To calculate the vertical point of a point to a line

3 缓冲区线性抽象算法

在“线性抽象算法”中,在判断公路图层、公路线性图层和交叉路口图层与圆的关系时,查询语句是对整个图层进行操作的,可见其效率是相对较低的,尤其当地图很大、公路线性图层含有很多个对象时查询将耗费很多时间。所以对“线性抽象算法”进行

了改进, 加入了公路对象的缓冲区, 以节省查询时间, 把这种算法称为“缓冲区线性抽象算法”, 其实现思路如下:

(1) 如果车辆定位点附近(100m 内) 有交叉路口, 则把校正点放在该交叉路口上。

(2) 如果车辆点附近没有交叉路口, 则找到离该点最近的一个公路对象, 该公路对象应该位于同一条公路上两个交叉路口之间(在对地图的预处理中就要做到, 即使是同一条公路也要根据交叉路口分割成若干个公路段), 然后做该公路段的缓冲区, 缓冲区的半径可根据 GPS 接收机的精度来确定, 对于单独的 GN-77 接收机可选定 100m。

(3) 判断车辆点与缓冲区的关系, 如果车辆点与该缓冲区相交, 则把与该缓冲区相交的公路线性图层中的若干个对象放入一个临时表, 在这个临时表中做车辆的校正算法(这个临时表显然要比整个公路线性图层小很多, 这正是提高查询效率的关键)。

(4) 如果车辆点与缓冲区不相交, 可以判断车辆正在偏离原来的公路, 则跳回到 1。

(5) 把校正点画到地图上去。取下一个点, 返回(1)。

(6) 对特殊的情况采取特殊处理。

“缓冲区线性抽象算法”是采用做公路对象的缓冲区的方法对“线性抽象算法”进行改进, 对于提高系统的查询效率和处理能力有很大帮助, 同时这种算法还保证了即使车辆点实际离另一条路很近, 只要车辆点落到当前公路对象的缓冲区内, 就用当前公路线性对象进行校正(没有到达交叉路口之前车辆不可能跑到另一条路上)。这种算法的优点在于把车辆点与整个公路图层的的关系转变为车辆点与其中某一个对象的关系。即在很多时候只需判断车辆点与一个公路对象的缓冲区对象的空间位置关系, 而不用在整个公路图层中查找一个满足条件的公路对象。但是在 MapInfo 中创建对象的缓冲区也是很费时间的。如果地图内容很少, 包含的对象很少, “缓冲区线性抽象算法”的优势将得不到体现, 其效率甚至还不如“线性抽象算法”高, 但是如果地图的表很大时, 这种算法的优势就很明显了。

4 模拟结果^[3~6]

MapInfo 是当今世界上流行的桌面地理信息系统。MapBasic 是理想的在 MapInfo 平台上开发用户定制的应用程序的编程语言。作者通过将 MapBasic 集成在 Visual Basic 中, 实现了“线性抽象算法”和“缓冲区线性抽象算法”。算法采用长沙市北区地图, 采用作者模拟的 GN-77 接收板的数据作为 GPS 数据, 结果如图 4。图 4 中公路上的点是校正的结果点。其中, A 处校正点放在交叉路口上, 作者在程序中设置如果在距车辆点 100m 范围内有交叉路口(如果交叉路口的个数超过一个, 则缩小搜索范围, 找其中离车辆点最近的交叉路口点), 则把车辆点的校正点画在该交叉路口上, 然后转到下一个点进行校正运算; B、C 两点的情况和 A 点处的情况相同, 所以校正点都画在了交叉路口上; 但 D 点的情况有

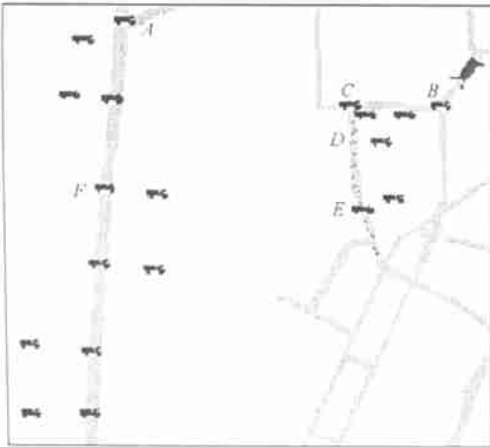


图 4 模拟结果示意
Fig. 4 Simulating results

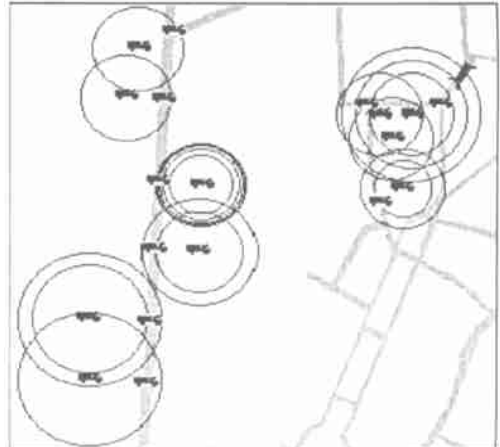


图 5 LAA 算法执行过程
Fig. 5 The calculating process of LAA

些不同, D 点并没有校正点, 这是因为 BC 公路段和 DE 公路段的名称不同(名称属性的改变发生在 C 附近的交叉路口处), 而现在并不知道 D 点的校正点究竟是放在 BC 公路段上还是放在 DE 公路段上, 所以跳过 D 点; 而在 E 点处, 因为 DE 两点在同一条公路上, 所以只用正常的校正算法即可, 先在公路图层中取得 DE 公路段的名称以判断再下一个点是否和 E 点在同一条公路上, 然后在公路线性图层中找到距离车辆点最近的公路线性对象, 求得校正点的位置, 将校正点画出即可; 其余的点像 F 点等都和 E 点的情况类似。

图 5 显示了“线性抽象算法”的执行过程, 图中所画的圆是查询公路对象、公路线性对象和交叉路口图层时用到的。在地图预处理过程中, 公路图层中公路对象是按照实际地理特征划分的, 即按照公路的名称信息以及宽度、车道、是否单行道(当两条路相距很近时作为辅助信息)等划分, 有时还可能根据地图的应用目的把地图上的某些小路删除掉。

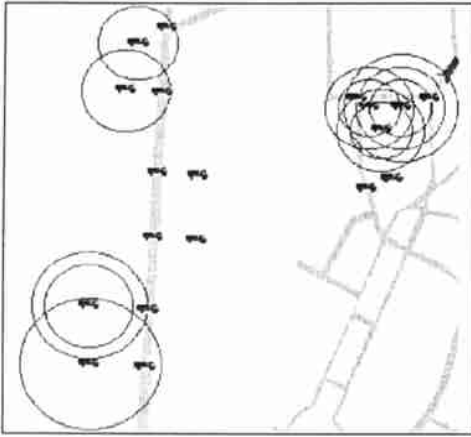


图 6 BLAA 算法执行过程

Fig. 6 The calculating process of BLAA

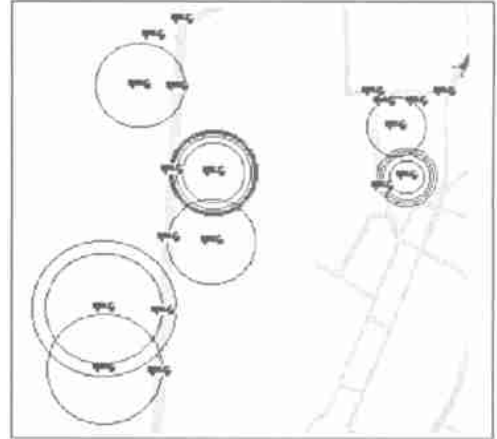


图 7 BLAA 算法执行过程

Fig. 7 The calculating process of BLAA

图 6 和图 7 显示了“缓冲区线性抽象算法”的执行过程, 图 6 中所画的圆是算法在查找离车辆点最近的公路图层中的对象时用到的, 图 7 中所画的圆是在找到了离车辆点最近的公路, 求得了与该公路对象的缓冲区相交的公路线性图层中的临时表后, 在该临时表中做校正时所用到的圆。

5 结束语

“线性抽象算法”能够很精确地对车辆点进行定位, 最大限度模拟车辆点的实际位置, 算法中加入了一次记忆功能, 在对交叉路口处进行处理时较好地解决了问题。

但是, “线性抽象算法”一个最大的问题在于当公路图层很大(公路线性图层也很大)时的效率问题, 因为算法对每个点进行校正时都对整个公路图层、公路线性图和交叉路口图层等进行几次搜索。

“缓冲区线性抽象算法”的基本思想和“线性抽象算法”是一样的, 只是处理过程有所不同, 所以两者的精度是一致的, 但前者的实现方法更为先进, 效率也更高。

参考文献:

- [1] 李德仁. 论 RS, GPS 与 GIS 集成的定义、理论与关键技术[J]. 遥感学报, 1997, (2): 65- 68.
- [2] 聂洪山, 徐捷. AVL 系统中车位显示修正算法的研究[J]. 国防科技大学学报, 2000, 22(3): 109- 113.
- [3] MapBasic: User's Guide, 3rd Edition October[R]. 1996.
- [4] MapBasic: Professional Reference, 3rd Edition October[R], 1996.
- [5] MapInfo Professional(4.5) 用户指南[R], 1996.
- [6] MapInfo Professional(4.5) 参考手册[R], 1996.