



(12) 发明专利

(10) 授权公告号 CN 101464158 B

(45) 授权公告日 2011.04.20

(21) 申请号 200910045401.8

(22) 申请日 2009.01.15

(73) 专利权人 上海交通大学

地址 200240 上海市闵行区东川路 800 号

(72) 发明人 史文欢 闫焱 申抒含 刘允才

(74) 专利代理机构 上海交达专利事务所 31201

代理人 王锡麟 王桂忠

(51) Int. Cl.

G01C 21/32 (2006.01)

G09B 29/00 (2006.01)

审查员 胡贺伟

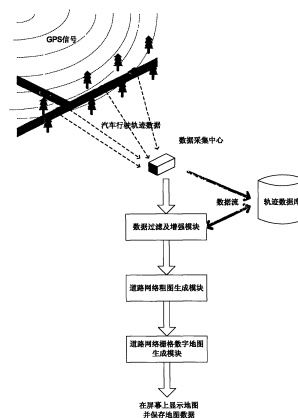
权利要求书 2 页 说明书 6 页 附图 2 页

(54) 发明名称

基于 GPS 定位的道路网络栅格数字地图自动生成方法

(57) 摘要

本发明涉及一种地图测绘领域的基于 GPS 定位的道路网络栅格数字地图自动生成方法。本发明用车载 GPS 设备采集汽车在道路网络上的行驶轨迹数据（即道路信息数据），并用计算机对行驶轨迹数据进行大地坐标到城建坐标的转换，然后存入轨迹数据库；用计算机对轨迹数据库中的行驶轨迹数据进行过滤和增强；然后，采用计算机处理过滤和增强过的行驶轨迹数据，生成道路网络栅格数字地图粗图；最后，采用计算机对上述粗图进行处理，生成道路网络栅格数字地图。本发明所采用数据源获取方便且获取成本较低，其生成的道路网络栅格数字地图完整清晰、精度高，并与实际道路网络同步，因此可更好地满足对道路网络地图及时性和精确性的应用要求。



1. 一种基于 GPS 定位的道路网络栅格数字地图自动生成方法，其特征在于包括如下步骤：

第一步，在车群上加装车载 GPS 设备，实时采集汽车在道路网络上的行驶轨迹数据，然后利用已有地图投影公式将行驶轨迹数据中的大地经纬度数据转换成城建坐标数据，最后存入轨迹数据库；

第二步，采用计算机对轨迹数据库中的行驶轨迹数据进行阈值过滤，将速度大的轨迹点数据删除，同时采用计算机在过滤后同一 ID 相邻时刻的轨迹点数据之间插入二者的中点数据，并删除上述轨迹数据中除城建坐标外的数据项；

第三步，采用计算机对上述离散的轨迹点进行膨胀处理直至离散的轨迹点连成实线，紧接着再进行腐蚀处理来光滑实线边沿，生成边沿光滑的道路网络栅格数字地图粗图；

第四步，采用计算机对上述粗图实施两次膨胀处理后再细化处理，最终得到道路网络栅格数字地图；

所述的地图投影公式为：

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 80085.8307315133 & 3948.762069978173 & -10309402.198608777 \\ -1964.663347251073 & 111740.22784596437 & -4651798.29925682 \end{bmatrix} \cdot \begin{bmatrix} l \\ d \\ 1 \end{bmatrix}, \quad \text{其中 } (x, y) \text{ 为}$$

城建坐标， l ， d 分别为经度和纬度；

所述的阈值过滤是指：将第一步处理过的行驶轨迹数据由数据库分批读入内存，当速度大于 50km/h 时则将该轨迹点数据删除；

所述的插入二者的中点数据，并删除上述轨迹数据中除城建坐标外的数据项是指：计算与当前速度低于 50km/h 的有效轨迹点具有相同 ID 的上一时刻有效轨迹点之间的时间间隔：如果时间间隔在 10 秒以内，则在两个轨迹点数据之间插入一个新轨迹点，新轨迹点 X 轴坐标和 Y 轴坐标分别为前二者 X 轴坐标和 Y 轴坐标的均值，定义一个二维数组 Trace 并将全部元素初始化为 0，逐次将内存中的轨迹点存储到数组中：对于城建坐标为 (x, y) 的轨迹点，将 $\text{Trace}[3800-y][x+800]$ 置为 1，其表示在城建坐标系的 (x, y) 位置上存在有效轨迹点；

所述的膨胀处理是指：定义一个与 Trace 同阶的二维数组 Temp，将 Trace 数组中以 $\text{Trace}[i][j]$ 为中心的 3×3 区域中的元素值复制到 3×3 二维数组 neighbor 中，检查 neighbor 中是否有值为 1 的元素，如有，则将 $\text{Temp}[i][j]$ 置为 1，否则置为 0，用循环命令语句对 Trace 全部元素进行一遍上述处理，最后将 Temp 中的元素值全部复制回 Trace 数组中，重复执行六次；

所述的腐蚀处理是指：将 Trace 数组中以 $\text{Trace}[i][j]$ 为中心的 3×3 区域中的元素值复制到 3×3 二维数组 neighbor 中，检查 neighbor 中是否有值为 0 的元素，如果有，则将 $\text{Temp}[i][j]$ 置为 0，否则置为 1，用循环命令语句对 Trace 全部元素进行一遍上述处理，最后将 Temp 中的元素值复制回 Trace 数组中；

所述的细化处理是指：1) 定义一个与 Trace 同阶的二维数组 Temp，将 Trace 数组中以 $\text{Trace}[i][j]$ 为中心的 5×5 区域中的元素值复制到 5×5 二维数组 neighbor 中，计算

$nCount = neighbor[1][1] + neighbor[1][2] + neighbor[1][3] + neighbor[2][1] + neighbor[2][2] + neighbor[2][3] + neighbor[3][1] + neighbor[3][2] + neighbor[3][3]$, 定义

布尔变量 $bCondition1$, 如果 $2 \leq nCount \leq 6$, 则将其置为 $true$, 否则置为 $false$;

2) 将 $nCount$ 清零, 逐次判断 $neighbor[1][2] = 0 \ \& \ neighbor[1][1] = 1$ 、 $neighbor[1][1] = 0 \ \& \ neighbor[2][1] = 1$ 、 $neighbor[2][1] = 0 \ \& \ neighbor[3][1] = 1$ 、 $neighbor[3][1] = 0 \ \& \ neighbor[3][2] = 1$ 、 $neighbor[3][2] = 0 \ \& \ neighbor[3][3] = 1$ 、 $neighbor[3][3] = 0 \ \& \ neighbor[2][3] = 1$ 、 $neighbor[2][3] = 0 \ \& \ neighbor[1][3] = 1$ 、 $neighbor[1][3] = 0 \ \& \ neighbor[1][2] = 1$ 是否成立, 如果成立, 则 $nCount$ 每次增 1, 否则不变, 最后定义布尔变量 $bCondition2$, 并判断 $nCount = 1$ 是否成立, 如果成立, 则将其置为 $true$, 否则置为 $false$;

3) 定义布尔变量 $bCondition3$, 判断 $neighbor[1][2] \times neighbor[2][1] \times neighbor[2][3] = 1$ 是否成立, 如果成立, 则将 $bCondition3$ 置为 $true$ 后转到步骤 4), 否则, 先将 $nCount$ 清零, 然后逐次判断 $neighbor[0][2] = 0 \ \& \ neighbor[0][1] = 1$ 、 $neighbor[0][1] = 0 \ \& \ neighbor[1][1] = 1$ 、 $neighbor[1][1] = 0 \ \& \ neighbor[2][1] = 1$ 、 $neighbor[2][1] = 0 \ \& \ neighbor[2][2] = 1$ 、 $neighbor[2][2] = 0 \ \& \ neighbor[2][3] = 1$ 、 $neighbor[2][3] = 0 \ \& \ neighbor[1][3] = 1$ 、 $neighbor[1][3] = 0 \ \& \ neighbor[0][3] = 1$ 、 $neighbor[0][3] = 0 \ \& \ neighbor[0][2] = 1$ 是否成立, 如果成立, 则 $nCount$ 每次增 1, 最后判断 $nCount$ 是否为 1, 如果不为 1, 则将 $bCondition3$ 置为 $true$, 否则置为 $false$;

4) 定义布尔变量 $bCondition4$, 判断

$neighbor[1][2] \times neighbor[2][1] \times neighbor[3][2] = 0$ 是否成立, 如果成立则将 $bCondition4$ 置为 $true$ 并转到步骤 5), 否则, 先将 $nCount$ 清零, 逐次判断

$neighbor[1][1] = 0 \ \& \ neighbor[1][0] = 1$ 、 $neighbor[1][0] = 0 \ \& \ neighbor[2][0] = 1$ 、

$neighbor[2][0] = 0 \ \& \ neighbor[3][0] = 1$ 、 $neighbor[3][0] = 0 \ \& \ neighbor[3][1] = 1$ 、

$neighbor[3][1] = 0 \ \& \ neighbor[3][2] = 1$ 、 $neighbor[3][2] = 0 \ \& \ neighbor[2][2] = 1$ 、

$neighbor[2][2] = 0 \ \& \ neighbor[1][2] = 1$ 、 $neighbor[1][2] = 0 \ \& \ neighbor[1][1] = 1$ 是否成立, 如果成立, 则 $nCount$ 每次增 1, 最后判断 $nCount$ 是否为 1, 如果不为 1, 则将 $bCondition4$ 置为 $true$, 否则置为 $false$;

5) 判断 $bCondition1$ 、 $bCondition2$ 、 $bCondition3$ 、 $bCondition4$ 是否都为 $true$, 如果是, 则将 $Temp[i][j]$ 置为 1, 否则置为 0, 运用循环命令语句, 重复步骤 1) ~ 步骤 5) 直至 $Trace$ 中全部元素都经历一遍运算, 最后将 $Temp$ 中的元素值复制回 $Trace$ 中, $Trace$ 数组中存储的数据即所生成的道路网络栅格数字地图。

2. 根据权利要求 1 所述的基于 GPS 定位的道路网络栅格数字地图自动生成方法, 其特征是, 所述车载 GPS 设备指能够接收 GPS 信号并经内置芯片运算后得到汽车定位数据, 包括汽车 ID、经度、纬度、时间、速度、行驶方向信息, 同时还具有数据传输功能的车载设备。

3. 根据权利要求 1 所述的基于 GPS 定位的道路网络栅格数字地图自动生成方法, 其特征是, 所述离散的轨迹点即汽车行驶轨迹数据存储到位图格式的二维数组中进行处理。

4. 根据权利要求 1 所述的基于 GPS 定位的道路网络栅格数字地图自动生成方法, 其特征是, 所述得到的道路网络栅格数字地图为二值化位图。

基于 GPS 定位的道路网络栅格数字地图自动生成方法

技术领域

[0001] 本发明涉及一种测绘技术领域的地图生成方法，具体地说，涉及的是一种基于 GPS 定位的道路网络栅格数字地图自动生成方法。

背景技术

[0002] 道路网络栅格数字地图（以下简称道路网络地图）是一种描绘道路网络地理特征及结构的位图，有着方便生成、内容细致、容易被用户接受等诸多优点，可广泛应用于汽车导航系统及智能交通系统（ITS）等领域中。随着城市化的进行以及经济的快速发展，道路网络更新速率正日益增大，现有道路网络图生成技术存在着诸如周期长、精度差、成本高等缺点，从而造成了道路网络地图更新总是跟不上实际道路网络变化的困境。开发快速精确的道路网络地图生成技术已愈发显得重要。

[0003] 现有道路网络栅格数字地图生成方法主要分为两类：一类是传统测绘方法，该方法运用测绘仪器及测绘学原理进行实地测绘，并将测绘得到的道路网络地图绘制于图纸上或存储于磁介质中。该方法有着悠久的历史，存在着精度不稳定、周期长、成本高等诸多缺点，因此随着卫星技术的发展其应用范围日益缩小；另一类是基于遥感图像的自动生成方法，该方法将卫星俯拍地面得到的高清晰度图像（即遥感图像）作为数据源，运用计算机和数字图像处理学原理对遥感图像进行处理来自动生成道路网络地图，该方法所需遥感图像购买成本较高，且所购图像通常并非道路网络的当时反映，此外，生成地图的难易度及精确性极易受到遥感图像质量的影响。

[0004] GPS（全球定位系统）是美国政府自上世纪 70 年代开始研制，历时 20 年，耗资 200 亿美元建立的一种基于卫星导航的定位与计时系统，其基本原理为：根据高速运动卫星瞬间位置作为已知的起算数据，采用空间距离后方交会的方法，确定待测点的位置。自上世纪 80 年代美国政府宣布开放部分 GPS 导航信号供民用以来，民用 GPS 定位技术得到了很大的发展。结合定位修正技术（如 DGPS 差分修正技术），民用 GPS 的定位误差通常可控制在 2～3 米以内，部分先进技术甚至实现了误差 1 米以内的高精度定位。此外，由于 GPS 具有全天候、低成本以及实时（无需支付使用费用，便可使用 GPS 接收机接收民用 GPS 信号，经接收机内置芯片运算后即实现实时定位）等诸多优势，因此 GPS 被越来越多地应用于社会生活的各个领域。

[0005] 经对现有技术的文献检索发现，最大似然光谱分类法（高芳琴等. 遥感图像中城市道路信息的自动识别与制图 [J]. 东北测绘, 2001, 24(III) : 27-30）是基于遥感图像自动生成道路网络栅格数字地图的常用方法之一。该方法考虑每个像素 RGB 分量之间的关系，利用特征矩阵和均值向量及像元分量计算各地图要素类（如道路网络、河流、建筑物、绿地等）的判别变量，如果道路网络地图要素类的判别变量值最大，则该将像元归于道路网络地图要素类。该方法可由计算机自动完成，制图速度较快，但存在着以下缺陷：1、购买遥感图像费用较高，且所购图像往往已滞后于道路网络的实际情况；2、当遥感图像上的道路网络图斑和非道路网络图斑的光谱界限不明显时，该方法生成的道路

网络栅格数字地图会出现道路网络图像模糊和不完整等情况，从而给地图的后续处理带来较大的困难。

发明内容

[0006] 本发明的目的在于针对现有技术的不足，提出一种基于 GPS 定位的道路网络栅格数字地图自动生成方法，采用 GPS 定位技术采集汽车在道路网络上的行驶轨迹数据，并采用计算机对轨迹数据进行自动处理，最后得到精确的道路网络栅格数字地图，解决了现有方法工作期长、精度差、成本高、具有时间滞后性等缺点，及时且精确的道路网络图可以更好地满足汽车导航系统和智能交通系统的应用需要。

[0007] 本发明是通过以下技术方案实现的，首先利用车载 GPS 设备采集汽车在道路网络上的行驶轨迹数据，并采用计算机对行驶轨迹数据进行大地坐标到城建坐标的转换，然后存入轨迹数据库；紧接着，采用计算机对轨迹数据库中的行驶轨迹数据进行过滤和增强；然后，采用计算机处理过滤和增强过的行驶轨迹数据，生成边沿光滑的道路网络栅格数字地图粗图（简称道路网络粗图）；最后，采用计算机对上述粗图进行处理，生成较精确的道路网络栅格数字地图。

[0008] 本发明方法包括以下步骤：

[0009] 第一步，在使用率较高的车群（如出租车）上加装车载 GPS 设备，实时采集汽车在道路网络上的行驶轨迹数据，然后利用已有地图投影公式将行驶轨迹数据中的大地经纬度数据转换成城建坐标数据，最后存入轨迹数据库。

[0010] 由于汽车的行驶轨迹一般位于道路上，因此其数据通常可视作实际道路网络的实际反映。车载 GPS 设备是指能够接收 GPS 信号并经内置芯片运算后得到汽车定位数据（包括汽车 ID、经度、纬度、时间、速度、行驶方向等数据项）的设备。此外，本发明中的车载 GPS 设备还需具备数据传输功能。值得说明的是，在国内外不少城市（如上海）现有的出租车调度系统中，大量的出租车已经加装了具有车载 GPS 设备上述功能的调度设备，因此加装车载 GPS 设备所产生的费用甚至可以避免。地图投影公式指利用技术手段建立的可将经纬网上的点投影到城建坐标平面上的坐标转换公式，利用该公式可实现从经纬度数据到城建坐标数据的精确转换。

[0011] 第二步，采用计算机对轨迹数据库中的行驶轨迹数据进行阈值过滤，将速大的轨迹点数据删除，同时采用计算机在过滤后同一 ID 相邻时刻的轨迹点数据之间插入二者的中点数据，并删除上述轨迹数据中除城建坐标外的数据项；

[0012] GPS 定位误差通常都和汽车的行驶速度有着较密切的关系，研究发现汽车行驶速度越大则定位误差也越大，因此采用计算机对轨迹数据库中的行驶轨迹数据进行阈值过滤，将速度大（比如大于 50km/h）的轨迹点数据删除从而增强轨迹数据的精确性。同时，由于汽车的行驶轨迹数据由离散分布在道路网络上的单个轨迹点数据组成，轨迹点稀疏的轨迹数据会给后续处理带来较大的困难，因此采用计算机在 ID 相同且记录时刻相邻的过滤后轨迹点数据之间插入二者的中点数据（存储于两个过滤后的轨迹点数据之间，坐标取二者的均值），同时删除轨迹数据中除城建坐标外的数据项。

[0013] 第三步，由前两步得到的行驶轨迹数据仍是分布在道路网络上能够一定程度反应道路网络地理特征和结构的离散轨迹点，并不是道路网络栅格数字地图。为此，采用

计算机对上述离散轨迹点进行膨胀处理直至连成实线（代表道路），紧接着进行腐蚀处理来光滑实线边沿，最后生成的实线图形即道路网络粗图。

[0014] 所述的离散轨迹点（汽车行驶轨迹数据）存储到位图格式的二维数组中进行处理。

[0015] 第四步，采用计算机对上述粗图实施两次膨胀后再细化处理，最终得到精度更高的道路网络栅格数字地图。

[0016] 道路网络粗图可基本反应道路网络的实貌，但精度不高，为此采用计算机对道路网络粗图实施两次膨胀后再细化处理，最终生成精度更高的道路网络栅格数字地图（膨胀后再细化处理相当于对轨迹点进行均值降噪，可达到进一步减小定位误差的效果）。所生成的道路网络栅格数字地图为二值化位图（道路网络元素与非道路网络元素分别用不同的数字表示）。

[0017] 所述车载 GPS 设备指能够接收 GPS 信号并经内置芯片运算后得到汽车定位数据（包括汽车 ID、经度、纬度、时间、速度、行驶方向等信息），同时还具有数据传输功能的车载设备。

[0018] 与现有技术相比，本发明实时采集与道路网络同步的道路信息数据（汽车行驶轨迹数据），避免了现有技术道路信息数据源滞后的缺陷；此外，本发明道路信息数据源（GPS 信号）获取方便，且获取成本较低；由于所生成的道路网络栅格数字地图可完整清晰地涵盖所有能汽车通行的道路，并具有较高精度（误差小于 1 ~ 3m，优于一般的现有技术），因此可更好地满足汽车导航系统和智能交通系统对道路网络数字地图的应用需要。

附图说明

[0019] 图 1 为本发明基于 GPS 定位的道路网络栅格数字地图自动生成方法流程图。

[0020] 图 2 为本发明实施例所生成的道路网络栅格数字地图的全图显示。

[0021] 图 3 为本发明实施例所生成的道路网络栅格数字地图的局部放大显示。

具体实施方式

[0022] 下面结合附图对本发明的实施例作详细说明：本实施例在以本发明技术方案为前提下进行实施，给出了详细的实施方式和具体的操作过程，但本发明的保护范围不限于下述的实施例。

[0023] 以吉林省吉林市昌邑区部分道路网络上采集的汽车行驶轨迹数据实施本发明自动生成道路网络栅格数字地图为例（计算机处理部分采用 Microsoft VisualStudio 2005 编译环境实现），其具体过程分为如下四个模块（如图 1 所示）：

[0024] 1. 在数据采集中心接收车载 GPS 设备以 5 秒为周期发送的汽车行驶轨迹数据，行驶轨迹数据由大量离散轨迹点数据组成，每个轨迹点数据包括汽车 ID、经度、纬度、时间、速度、行驶方向共 6 个数据项。根据已有的大地坐标系到吉林市地面道路图城建坐标系的地图投影公式：

$$[0025] \quad \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 80085.8307315133 & 3948.762069978173 & -10309402.198608777 \\ -1964.663347251073 & 111740.22784596437 & -4651798.29925682 \end{bmatrix} \cdot \begin{bmatrix} l \\ d \\ 1 \end{bmatrix}, \quad (1)$$

[0026] 其中 (x, y) 为城建坐标, l, d 分别为经度和纬度, 不断将离散轨迹点的城建坐标按式 (1) 进行运算, 从而将行驶轨迹数据的经度和纬度数据替换成城建坐标的 X 轴坐标数据和 Y 轴坐标数据, 然后按如下表所示的格式存储到数据库中。

[0027]

汽车 ID	X 坐标 (m)	Y 坐标 (m)	时间	速度 (km/h)	行驶方向
0001	-327	1316	08-11-0706:25:39	33	7

[0028] 注: 上述吉林市地面道路图城建坐标系以该市火车站站前广场雕塑所在位置为城建坐标系原点, 其大地经度和纬度分别为 126.5670305556° E 和 43.8558166667° N, 坐标轴单位长度为一米。

[0029] 2. 将第 1 步处理过的行驶轨迹数据由数据库分批读入内存, 当速度大于 50km/h 时则将该轨迹点数据删除, 同时计算与当前有效轨迹点 (速度低于 50km/h) 具有相同 ID 的上一时刻有效轨迹点之间的时间间隔: 如果时间间隔在 10 秒以内, 则在两个轨迹点数据之间插入一个新轨迹点, 新轨迹点 X 轴坐标和 Y 轴坐标分别为前二者 X 轴坐标和 Y 轴坐标的均值。定义一个二维数组 Trace 并将全部元素初始化为 0, 逐次将内存中的轨迹点存储到数组中: 对于城建坐标为 (x, y) 的轨迹点, 将 Trace[3800-y][x+800] 置为 1, 其表示在城建坐标系的 (x, y) 位置上存在有效轨迹点。下表为所得 Trace 数组 (元素为 Trace[i][j]) 的一部分:

[0030]

j \ i	623	624	625	626	627	628
201	0	0	0	0	0	1
202	0	0	0	0	1	0
203	0	0	0	1	0	0
204	0	1	1	1	0	1
205	0	1	0	1	0	1
206	0	0	1	0	0	0

[0031] 3. 对第 2 步处理得到的 Trace 数组进行如下处理:

[0032] 1) 定义一个与 Trace 同阶的二维数组 Temp, 将 Trace 数组中以 Trace[i][j] 为中心的 3x3 区域中的元素值复制到 3x3 二维数组 neighbor 中, 检查 neighbor 中是否有值为 1 的元素, 如有, 则将 Temp[i][j] 置为 1, 否则置为 0, 用循环命令语句对 Trace 全部元素进行一遍上述处理, 最后将 Temp 中的元素值全部复制回 Trace 数组中。重复执行步骤 (1) 六次;

[0033] 2) 将 Trace 数组中以 Trace[i][j] 为中心的 3x3 区域中的元素值复制到 3x3 二维数组 neighbor 中, 检查 neighbor 中是否有值为 0 的元素, 如果有, 则将 Temp[i][j] 置为 0,

否则置为 1，用循环命令语句对 **Trace** 全部元素进行一遍上述处理，最后将 **Temp** 中的元素值复制回 **Trace** 数组中。执行本步骤后得到的 **Trace** 数组所存储的信息即道路边沿光滑过的道路网络粗图。

[0034] 4. 对第 3 步处理得到的道路网络粗图进行如下处理：

[0035] 1) 定义一个与 **Trace** 同阶的二维数组 **Temp**，将 **Trace** 数组中以 **Trace[i][j]** 为中心的 5x5 区域中的元素值复制到 5x5 二维数组 **neighbor** 中，计算 $nCount = neighbor[1][1] + neighbor[1][2] + neighbor[1][3] + neighbor[2][1] + neighbor[2][2] + neighbor[2][3] + neighbor[3][1] + neighbor[3][2] + neighbor[3][3]$ 布尔变量 **bCondition1**，如果 $2 \leq nCount \leq 6$ ，则将其置为 true，否则置为 false；

[0036] 2) 将 **nCount** 清零，逐次判断 $neighbor[1][2] = 0 \& neighbor[1][1] = 1$ 、 $neighbor[1][1] = 0 \& neighbor[2][1] = 1$ 、 $neighbor[2][1] = 0 \& neighbor[3][1] = 1$ 、 $neighbor[3][1] = 0 \& neighbor[3][2] = 1$ 、 $neighbor[3][2] = 0 \& neighbor[3][3] = 1$ 、 $neighbor[3][3] = 0 \& neighbor[2][3] = 1$ 、 $neighbor[2][3] = 0 \& neighbor[1][3] = 1$ 、 $neighbor[1][3] = 0 \& neighbor[1][2] = 1$ 是否成立，如果成立，则 **nCount** 每次增 1，否则不变。最后定义布尔变量 **bCondition2**，并判断 **nCount** = 1 是否成立，如果成立，则将其置为 true，否则置为 false；

[0037] 3) 定义布尔变量 **bCondition3**，判断 $neighbor[1][2] \times neighbor[2][1] \times neighbor[2][3] = 1$ 是否成立，如果成立，则将 **bCondition3** 置为 true 后转到步骤 (4)。否则，先将 **nCount** 清零，然后逐次判断 $neighbor[0][2] = 0 \& neighbor[0][1] = 1$ 、 $neighbor[0][1] = 0 \& neighbor[1][1] = 1$ 、 $neighbor[1][1] = 0 \& neighbor[2][1] = 1$ 、 $neighbor[2][1] = 0 \& neighbor[2][2] = 1$ 、 $neighbor[2][2] = 0 \& neighbor[2][3] = 1$ 、 $neighbor[2][3] = 0 \& neighbor[1][3] = 1$ 、 $neighbor[1][3] = 0 \& neighbor[0][3] = 1$ 、 $neighbor[0][3] = 0 \& neighbor[0][2] = 1$ 是否成立，如果成立，则 **nCount** 每次增 1，最后判断 **nCount** 是否为 1，如果不为 1，则将 **bCondition3** 置为 true，否则置为 false；

[0038] 4) 定义布尔变量 **bCondition4**，判断 $neighbor[1][2] \times neighbor[2][1] \times neighbor[3][2] = 0$ 是否成立，如果成立则将 **bCondition4** 置为 true 并转到步骤 (5)。否则，先将 **nCount** 清零，逐次判断 $neighbor[1][1] = 0 \& neighbor[1][0] = 1$ 、 $neighbor[1][0] = 0 \& neighbor[2][0] = 1$ 、 $neighbor[2][0] = 0 \& neighbor[3][0] = 1$ 、 $neighbor[3][0] = 0 \& neighbor[3][1] = 1$ 、 $neighbor[3][1] = 0 \& neighbor[3][2] = 1$ 、 $neighbor[3][2] = 0 \& neighbor[2][2] = 1$ 、 $neighbor[2][2] = 0 \& neighbor[1][2] = 1$ 、 $neighbor[1][2] = 0 \& neighbor[1][1] = 1$ 是否成立，如果成立，则 **nCount** 每次增 1，最后判断 **nCount** 是否为 1，如果不为 1，则将 **bCondition4** 置为 true，否则置为 false；

[0039] 5) 判断 **bCondition1**、**bCondition2**、**bCondition3**、**bCondition4** 是否都为 true，如果是，则将 **Temp[i][j]** 置为 1，否则置为 0。运用循环命令语句，重复步骤 (1) ~ (5) 直至 **Trace** 中全部元素都经历一遍运算，最后将 **Temp** 中的元素值复制回 **Trace** 中。**Trace** 数组中存储的数据即所生成的道路网络栅格数字地图，将其存入一个文本文件中以便后续处理和再显示。

[0040] 如图 2、3 所示，按照如下规则进行道路网络栅格数字地图显示：如果 **Trace[i][j]** = 1 成立，则在屏幕坐标系 (j, i) 处显示一个黑色地图像素点，否则什么也不做。将生

成的道路网络地图（如图 2、3 所示）和吉林市旅游交通图（哈尔滨地图出版社出版，2007 年 10 月第一版）进行对比后，发现该图完整清晰地涵盖了吉林市昌邑区道路网络（包括旅游交通图中未涵盖的新建道路），同时还具有较高的精度。由于本实施例所生成的道路网络栅格数字地图具有及时（可与道路网络的实际情况同步）、清晰（二值化数字地图）以及较高精度（高精度的道路信息数据结合精度提高措施可使所生成的道路网络栅格数字地图具有较高精度）等优点，所以其在汽车导航系统和智能交通系统等领域有着较好的应用前景，能更好地满足上述领域对道路网络数字地图的应用需要。

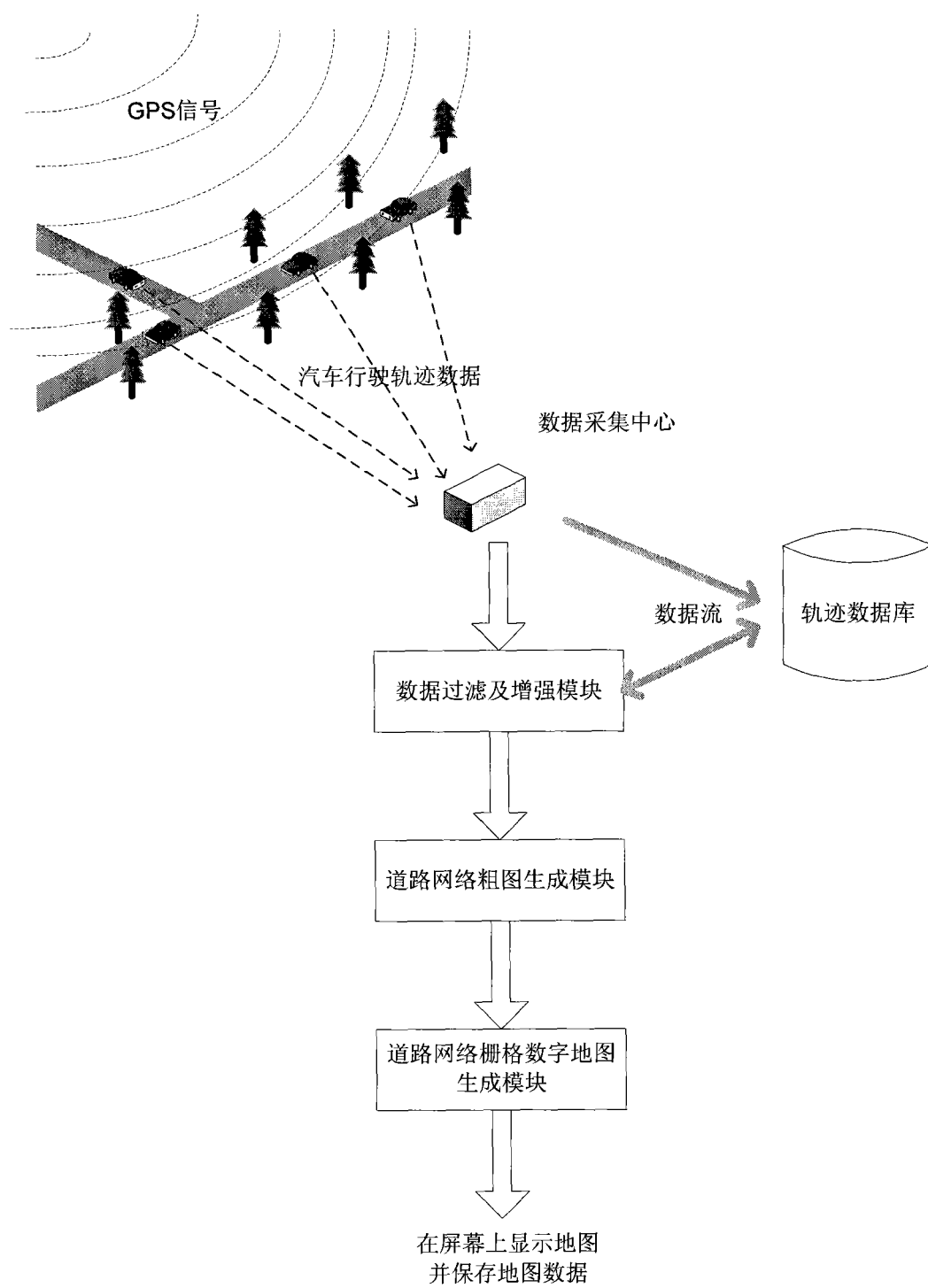


图 1

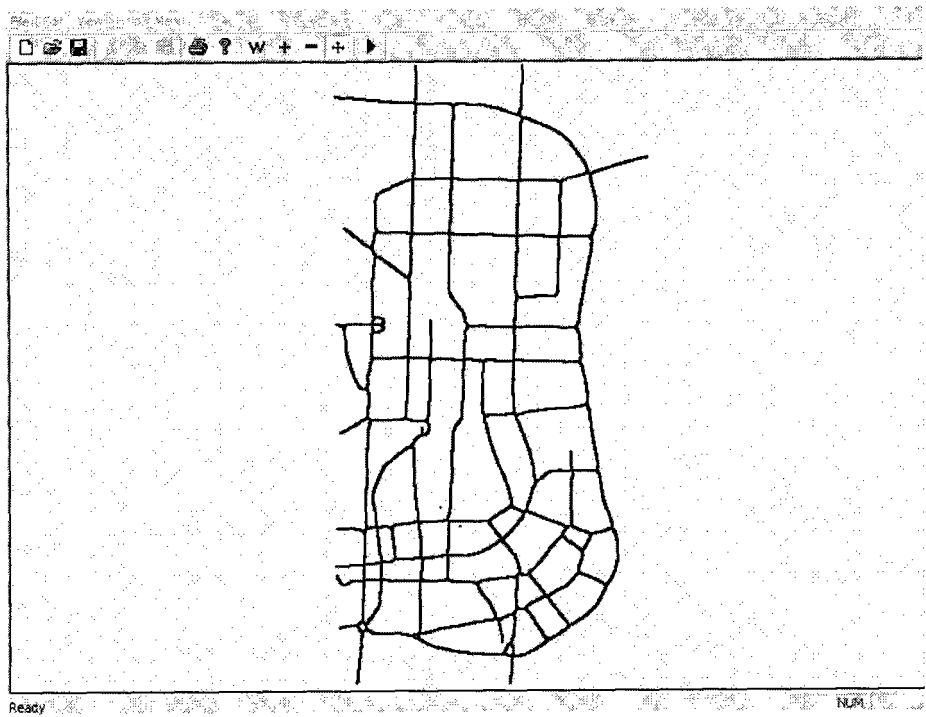


图 2

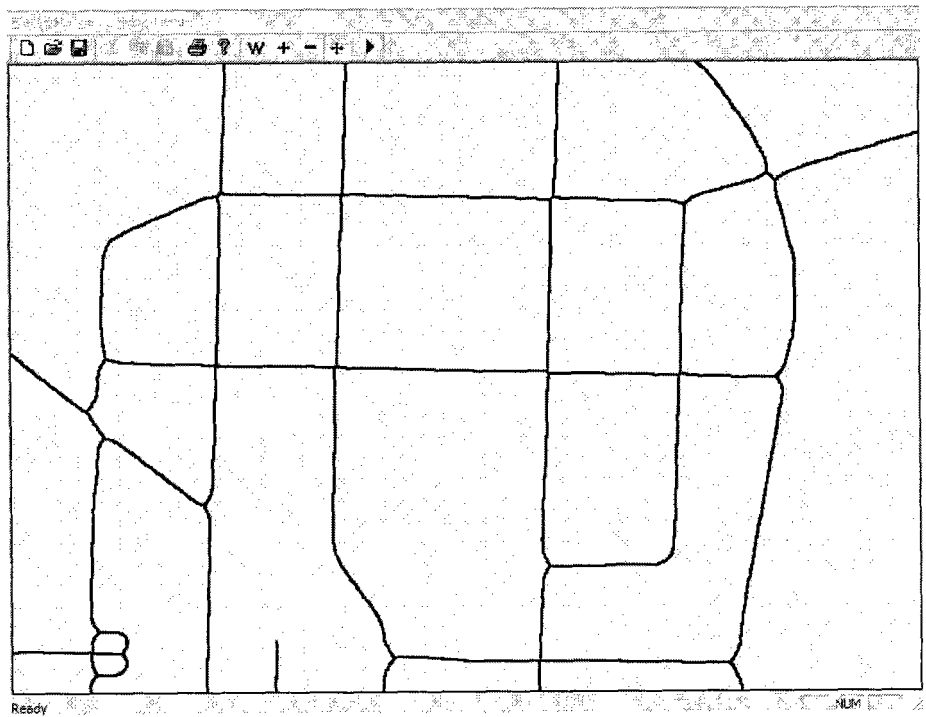


图 3