



(12) 发明专利申请

(10) 申请公布号 CN 103680177 A

(43) 申请公布日 2014. 03. 26

(21) 申请号 201310643348. 8

(22) 申请日 2013. 12. 03

(71) 申请人 上海交通大学

地址 200240 上海市闵行区东川路 800 号

(72) 发明人 赵亦燃 张阳 张哲慧 甘小莺

田晓华 杨峰 王新兵

(74) 专利代理机构 上海汉声知识产权代理有限

公司 31236

代理人 郭国中

(51) Int. Cl.

G08G 1/0962 (2006. 01)

H04W 4/04 (2009. 01)

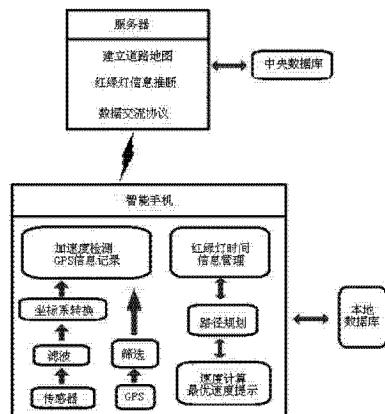
权利要求书2页 说明书9页 附图4页

(54) 发明名称

基于手机的智能车速提示驾驶系统

(57) 摘要

本发明公开一种基于手机的智能车速提示驾驶系统，包括车辆上手机和服务器，所述手机和服务器是通过如下步骤工作的：利用众包来收集车辆上手机 GPS 和加速度信息，并通过无线网在某个时间里将数据传给服务器；通过加速度向量分布图和 GPS 路径确定带红绿灯的路口的位置；使用二次 B 样条曲线来表示道路；利用众包来收集每个路口的加速度事件，并推断该路口红绿灯时间表；利用生成的地图和前方路口的红绿灯时间表，给车辆提示最高速度，使得车辆尽量不在路口遇红灯停止。本发明一方面用了新的方法生成地图，另一方面利用路口结构解决了红绿灯时间表在复杂路口的推算。



1. 一种基于手机的智能车速提示驾驶系统,包括车辆上手机和服务器,其特征在于,所述手机和服务器是通过如下步骤工作的:

步骤一,利用众包来收集车辆上手机 GPS 和加速度信息,并通过无线网在某个时间里将数据传给服务器;

步骤二,通过加速度向量分布图和 GPS 路径确定带红绿灯的路口的位置;

步骤三,绘制和使用二次 B 样条曲线来表示道路,并生成地图数据库;

步骤四,用众包来收集每个路口的加速度事件,并推断该路口红绿灯时间表;

步骤五,利用生成的地图数据库和前方路口的红绿灯时间表,给车辆提示最优速度,使得车辆尽量不在路口遇红灯停止。

2. 根据权利要求 1 所述的基于手机的智能车速提示驾驶系统,其特征在于,所述步骤一中收集车辆上手机 GPS 和加速度信息的方法是:使用手机上集成的加速度计和磁力计,将手机体坐标系中检测到的加速度,通过旋转矩阵,转换到局部 NED 坐标系中,得到车辆加速度在行驶方向上的投影,通过 GPS 数据得到车辆行驶方向,并且记录下车辆行驶的经纬度和时间戳。

3. 根据权利要求 1 所述的基于手机的智能车速提示驾驶系统,其特征在于,所述步骤二中的加速度向量包含车辆 ID、经纬度、经纬度精度、加速方向、时间,所述通过加速度向量分布图和 GPS 路径确定带红绿灯的路口的位置的方法是:当车辆的速度降为零且当这种零速度状态持续了若干秒以上,随后的加速才会被记录下来,这样,加速度向量密集的地方很可能是带红绿灯的路口,采用类似于均值漂移的方法来找出加速度事件发生密度最大的地点,在多次均值漂移的过程中,圆心向着加速度密度最大的点移动,而且漂移的圆的半径不断缩小,当圆的大小近似于一个路口,且当某次移动距离小于一个阈值之后,候选路口的位置就用最后的圆心所表示,然后,利用路口处产生加速度的车辆的 GPS 轨迹判断路口的支路数,验证路口的有效性。

4. 根据权利要求 3 所述的基于手机的智能车速提示驾驶系统,其特征在于,所述利用路口处产生加速度的车辆的 GPS 轨迹判断路口的支路数,验证路口的有效性的方法是:把 GPS 轨迹按候选路口断开,如果有两条轨迹有相同的起始候选路口和结束候选路口,或者有同样的延伸角度,那么他们被归并到同一个路段,对于一个路口,如果它拥有的支路路段个数少于三或者多余五,认为它是无效路口,当一个路口的支路被确定之后,GPS 轨迹就会将一个路口的支路连接起来,构成这个路口的细节结构,有效的路口被给予一个唯一的 ID,并且记录路口支路的连接逻辑。

5. 根据权利要求 4 所述的基于手机的智能车速提示驾驶系统,其特征在于,所述步骤三中,绘制和使用二次 B 样条曲线,生成地图数据库的方法是:在两个直接连接的路口之间,利用采集到的所有经过该路段的车辆的 GPS 数据点,确定一系列锚节点,利用得到的一系列锚节点,求出二次 B 样条曲线的控制点,控制点的求取利用锚节点的位置和方向,将道路曲线用二次 B 样条曲线表示,通过路口坐标、分支信息、道路锚节点信息,生成地图数据库。

6. 根据权利要求 5 所述的基于手机的智能车速提示驾驶系统,其特征在于,所述锚节点的确定是利用均值漂移的方法,只是每个 GPS 点的权重为其精度的倒数,在多条方向一致的 GPS 点中,选取在一定圆形域内的重心点作为该道路上的锚节点,并以平均方向作为

该锚节点的方向，沿着 GPS 轨迹，每隔大约 10 至 20 米计算一次锚节点，计算方法类似于加权的均值漂移，只是圆移动方向被约束在平均前进方向的垂直方向，如果下一个锚节点的方向比上一个改变量大于一定角度，或者距离上一个锚节点大于一定距离，该锚节点才有效，并被记录下来。

7. 根据权利要求 6 所述的基于手机的智能车速提示驾驶系统，其特征在于，所述一定角度为 20 度，所述一定距离为 50 米。

8. 根据权利要求 5 所述的基于手机的智能车速提示驾驶系统，其特征在于，所述步骤四中推断该路口红绿灯时间表的方法是：在路口停止等待红灯的车辆起步加速后，利用手机传感器和 GPS，将路口 ID、入和出的支路编号、零速度状态的起始结束时间，以及检测到加速至发送数据的时间间隔，通过网络发送给服务器，服务器通过某路口采集到的加速度事件集合，动态的推断该路口的红绿灯总周期、状态数、状态发生顺序、状态持续时间。

9. 根据权利要求 6 所述的基于手机的智能车速提示驾驶系统，其特征在于，所述步骤五中，利用生成的地图数据库和前方路口的红绿灯时间表，给车辆提示最优速度的方法是：驾驶员在使用本系统时，首先设定目的地，系统根据到达目的地耗时最小来确定路径，车辆在行驶过程中，将自己的位置、路径中下两个路口的 ID 发送给服务器，服务器将前方路口适合该车辆通过的红绿灯时间表发送到手机，同时手机利用地图数据库中的锚节点算出到下一个路口的距离，这样，系统就向驾驶员提供最佳速度提示。

## 基于手机的智能车速提示驾驶系统

### 技术领域

[0001] 本发明涉及一种智能交通系统,具体是一种基于手机的智能车速提示驾驶系统。

### 背景技术

[0002] 如今,交通灯主导着城市交通、协调不同道路驶来的车辆安全通过路口。但由于驾驶员缺乏交通灯信息,他们很难在行驶中采用合适的速度,因而常常在到达路口时必须停下车等红灯。这种走走停停的驾驶模式导致更多的能源消耗、空气污染、交通拥堵和安全事故。根据美国德克萨斯州交通研究所(Texas Transportation Institute)的数据,2011年,道路拥堵导致美国城市居民花费了额外55亿小时的时间和29亿加仑的汽油,总计额外支出1210亿美元。

[0003] 因此,我们寄希望于动态调整交通信号计时来适应车流,或者告知驾驶员交通信号安排并且提供合适速度的建议。前一种方法中,全球范围内已经存在智能交通信号控制系统,例如 SCATS、SCOOT、RHODES 及后继者 MILOS。但是总体而言,由于基础设施建设与系统维护所需的巨额成本,这类系统很难大范围推广。对于后一种方法,经对现有技术文献的检索发现,0. Servin 等在《IEEE Intelligent Transportation Systems Society, 2006》(电气电子工程师协会智能交通系统委员会,2006) 上发表的“An Energy and Emissions Impact Evaluation of Intelligent Speed Adaptation”(智能速度适应系统对能源和排放影响的评估)。该文中,车载智能速度提示系统对环境的积极影响已经被证实。随着手机科技的进步,如今,智能手机在交通协调系统中极有前景。另外, M. Krause 等在《IEEE Intelligent Vehicles Symposium, 2012》(IEEE 智能车辆研讨会,2012)上发表的“Traffic Light Assistant -Driven in a Simulator”(交通灯助手——在模拟器中接受驾车指导)。该文展示了一个用智能手机作为人机界面的成本效益高的交通灯助手。通过合理设计,未来基于智能手机应用的系统必会更多出现。

[0004] 经对现有技术文献的检索发现,目前存在一些类似系统。E. Koukoumidis 等在《ACM International Conference on Mobile Systems, Applications, and Services, New York, 2011, pp. 127–140》(计算机工程师协会移动系统、应用、服务国际会议,2011年,第 127–140 页) 发表的“SignalGuru:lever-aging mobile phones for collaborative traffic signal schedule advisory”(信号灯大师 :利用移动手机群提供交通信号时间信息)。该文利用了智能手机摄像头和自组网建立起交通灯时刻表,给驾驶员提示合理速度。但是基于对信号灯做图像处理的方法在大城市,特别是路口有多组红绿灯的情况下,该系统很难实现。G. Ning 等在《2<sup>nd</sup> International Conference on Computer Science and Electronics Engineering, 2013》(第二届计算机科学和电子工程国际会议) 中发表的“Adaptive Driving Speed Guiding to Avoid Red Traffic Lights”(自适应速度提示避开红灯系统)。该文中,利用交通部门提供的红绿灯时间和地图信息给驾驶员提供速度建议。但是大部分情况下,交通部门没有准确的交通灯信息。

## 发明内容

[0005] 本发明针对上述现有技术中存在的技术问题，提供一种基于手机的智能车速提示驾驶系统，利用现有智能手机的技术和普遍性，提供一个成本低、无需基础设施、给予驾驶员合理速度建议的系统。该系统的运行仅仅依靠智能手机和互联网服务器。

[0006] 本发明是通过以下技术方案实现的：

[0007] 一种基于手机的智能车速提示驾驶系统，包括车辆上手机和服务器，所述手机和服务器是通过如下步骤工作的：

[0008] 步骤一，利用众包来收集车辆上手机 GPS 和加速度信息，并通过无线网在某个时间里将数据传给服务器；

[0009] 步骤二，通过加速度向量分布图和 GPS 路径确定带红绿灯的路口的位置；

[0010] 步骤三，绘制和使用二次 B 样条曲线来表示道路，并生成地图数据库；

[0011] 步骤四，用众包来收集每个路口的加速度事件，并推断该路口红绿灯时间表；

[0012] 步骤五，利用生成的地图数据库和前方路口的红绿灯时间表，给车辆提示最优速度，使得车辆尽量不在路口遇红灯停止。

[0013] 所述步骤一中收集车辆上手机 GPS 和加速度信息的方法是：使用智能手机上集成的加速度计和磁力计，将手机体坐标系中检测到的加速度，通过旋转矩阵，转换到局部 NED 坐标系中，得到车辆加速度在行驶方向上的投影，通过 GPS 数据得到车辆行驶方向，并且记录下车辆行驶的经纬度和时间戳。

[0014] 所述步骤二中的加速度向量包含车辆 ID、经纬度、经纬度精度、加速方向、时间戳，所述通过加速度向量分布图和 GPS 路径确定带红绿灯的路口的位置的方法是：当车辆的速度降为零且当这种零速度状态持续了若干秒以上，随后的加速才会被记录下来，这样，加速度向量密集的地方很可能是带红绿灯的路口，采用类似于均值漂移的方法来找出加速度事件发生密度最大的地点，在多次均值漂移的过程中，圆心会向着加速度密度最大的点移动，而且漂移的圆的半径不断缩小，当圆的大小近似于一个路口，且当某次移动距离小于一个阈值之后，候选路口的位置就用最后的圆心所表示，然后，利用路口处产生加速度的车辆的 GPS 轨迹判断路口的支路数，验证路口的有效性。

[0015] 所述利用路口处产生加速度的车辆的 GPS 轨迹判断路口的支路数，验证路口的有效性的方法是：把 GPS 轨迹按候选路口断开，如果有两条轨迹有相同的起始候选路口和结束候选路口，或者有同样的延伸角度，那么他们被归并到同一个路段，对于一个路口，如果它拥有的支路路段个数少于三或者多余五，认为它是无效路口，当一个路口的支路被确定之后，GPS 轨迹就会将一个路口的支路连接起来，构成这个路口的细节结构，有效的路口被给予一个唯一的 ID，并且记录路口支路的连接逻辑。

[0016] 所述步骤三中，绘制和使用二次 B 样条曲线，生成地图数据库的方法是：在两个直接连接的路口之间，利用采集到的所有经过该路段的车辆的 GPS 数据点，确定一系列锚节点，利用得到的一系列锚节点，求出二次 B 样条曲线的控制点，控制点的求取利用了锚节点的位置和方向，将道路曲线用二次 B 样条曲线表示，通过路口坐标、分支信息、道路锚节点信息，生成地图数据库。

[0017] 所述锚节点的确定是利用均值漂移的方法，只是每个 GPS 点的权重为其精度的倒数，在多条方向一致的 GPS 点中，选取在一定圆形域内的重心点作为该道路上的锚节点，并

以平均方向作为该锚节点的方向,沿着 GPS 轨迹,每隔大约 10-20 米计算一次锚节点,计算方法类似于加权的均值漂移,只是圆移动方向被约束在平均前进方向的垂直方向,如果下一个锚节点的方向比上一个改变量大于一定角度,或者距离上一个锚节点大于一定距离米,该锚节点才有效,并被记录下来。

[0018] 所述步骤四中推断该路口红绿灯时间表的方法是:在路口停止等待红灯的车辆起步加速后,利用手机传感器和 GPS,将路口 ID、入和出的支路编号、零速度状态的起始结束时间,以及检测到加速至发送数据的时间间隔,通过网络发送给服务器,服务器通过某路口采集到的加速度事件集合,动态的推断该路口的红绿灯总周期、状态数、状态发生顺序、状态持续时间。

[0019] 所述步骤五中,利用生成的地图数据库和前方路口的红绿灯时间表,给车辆提示最优速度的方法是:驾驶员在使用本系统时,首先设定目的地,系统根据到达目的地耗时最小来确定路径,车辆在行驶过程中,将自己的位置、路径中下两个路口的 ID 发送给服务器,服务器将前方路口适合该车辆通过的红绿灯时间表发送到手机,同时手机利用地图数据库中的锚节点算出到下个路口的距离,这样,系统就向驾驶员提供最佳速度提示。

[0020] 本发明所提供的智能车速提示驾驶系统,一方面用了新的方法生成地图,另一方面利用路口结构解决了红绿灯时间表在复杂路口的推算。系统的运行仅仅依靠智能手机和互联网服务器,成本低、无需基础设施、给予驾驶员最合理的速度建议。

## 附图说明

- [0021] 图 1 是采用均值漂移定位候选路口的方法示意图;
- [0022] 图 2 是图 1 中候选路口中假路口示意图;
- [0023] 图 3 是图 1 中候选路口中真路口示意图;
- [0024] 图 4 (a) 是一段道路的有效锚节点示意图;
- [0025] 图 4 (b) 是图 4 (a) 的锚节点生成的路段示意图;
- [0026] 图 5 (a) 是绘制二次 B 样条曲线表示道路的示意图;
- [0027] 图 5 (b) 是用直线表示道路的示意图;
- [0028] 图 6 是四支路路口示意图;
- [0029] 图 7 是  $P(n)$ 、 $n$ 、 $p$  的关系示意图;
- [0030] 图 8 是四支路路口红绿灯通行的四个状态示意图;
- [0031] 图 9 是本系统的运行流程图。

## 具体实施方式

- [0032] 以下通过具体实施例结合附图对本发明的技术方案做一详细的描述:
- [0033] 本发明所提供的基于手机的智能车速提示驾驶系统,是通过智能手机和互联网服务器运行的,具体运行过程如下:
  - [0034] 步骤一,利用众包来收集车辆上手机 GPS 和加速度信息,并通过无线网络在某个时间里将数据传给服务器;
  - [0035] 手机用到的功能有:
    - [0036] 1) 3 轴加速度计

[0037] 智能手机的加速度计测量施加在手机上的加速度,包括重力,数值是在手机的体坐标系下的。除了加速度计,安卓系统还提供了基于加速度计的重力计,用来估计重力在体坐标系中的方向和大小。

[0038] 2) 3 轴磁力计

[0039] 智能手机也集成了磁力计,能够测出在体坐标系下的磁场大小和方向。把磁场所测出的磁场和加速度计测出的重力当作坐标变换的桥梁,将手机的加速度从体坐标系转换到地球局部 NED 坐标系。

[0040] 3) 全球定位系统(GPS)

[0041] GPS 已经集成到了现在的智能手机上,而且随着 DGPS 的到来, GPS 定位的精度已经大大提高。从测试用的手机反映出, GPS 位置信息的精确度平均在 10-15 米,最好的情况下在 4 米。手机通过 GPS 可以获得的信息有: 经纬度、速度、运动方向、通用协调时(UTC, 即从 1970 年 1 月 1 日开始的毫秒计时)、精度等。

[0042] 加速度事件测量的实现:

[0043] 因为加速度计的测量值受噪声的影响比较大,所以测量数据首先经过一个截止频率在 2Hz 的低通滤波器。然后加速度将从手机体坐标系转换到地球局部 NED 坐标系。

[0044] 坐标转换

[0045] 坐标转换能够将手机的加速度投影到车辆的前进方向上,从而更好地判断车辆加速。坐标的变换遵从 Z-Y-X 的顺序,则坐标变换的旋转矩阵为  $R_{nv|b}$ :

$$[0046] \quad \begin{bmatrix} C(\theta_y)C(\theta_z) & C(\theta_y)S(\theta_z) & -S(\theta_y) \\ S(\theta_x)S(\theta_y)C(\theta_z) & S(\theta_x)S(\theta_y)S(\theta_z) & S(\theta_x)C(\theta_y) \\ -C(\theta_x)S(\theta_z) & +C(\theta_x)C(\theta_z) & \\ C(\theta_x)S(\theta_y)C(\theta_z) & C(\theta_x)S(\theta_y)S(\theta_z) & C(\theta_x)C(\theta_y) \\ +S(\theta_x)S(\theta_z) & -S(\theta_x)C(\theta_z) & \end{bmatrix} \quad \text{式一}$$

其中,  $C(\theta)$ 、 $S(\theta)$  分别表示  $\cos(\theta)$ 、 $\sin(\theta)$ 。

[0047] 为了计算旋转矩阵,需要用到两个参考向量,即重力和磁场。这里将地磁的北极视为地球的北极,它们的区别不会对系统造成太大影响。求出旋转矩阵之后,在地球局部 NED 坐标系下的加速度  $\vec{a}_{nv}$  表达式为

$$[0048] \quad \vec{a}_{nv} = \begin{pmatrix} a_{x_nv} \\ a_{y_nv} \\ a_{z_nv} \end{pmatrix} = R_{nv|b} \begin{pmatrix} a_{x_b} \\ a_{y_b} \\ a_{z_b} \end{pmatrix} = R_{nv|b} \cdot \vec{a}_b \quad \text{式二}$$

其中,  $\vec{a}_b$  是体坐标系

里的加速度向量。

[0049] 车辆的行驶方向是指其前进方向与该地点指向北的向量的顺时针夹角,并且可以直接从 GPS 数据中获得。我们将转换到地球局部 NED 坐标系中的加速度投影到车辆停止前的行驶方向,以便检测车辆从静止到加速的事件。加速度向量的投影可以自然地滤掉与前进方向垂直的噪声或者是用户的摆弄。为了进一步减少加速误检测,我们设置了一个约 1 秒的时间区间,来测量这个区间内的累积加速。因为用户摆弄产生的加速会被一个大致同等大小的减速抵消,所以该时间区间内总的累积加速度很小,不会触发误测。

[0050] 本系统将加速度事件发生的时间和地点记录,同时记录 GPS 数据,包括经纬度、速度、运动方向、时间戳、精度等。

[0051] 步骤二,通过加速度向量分布图和 GPS 路径确定带红绿灯的路口的位置;

[0052] 定位候选路口

[0053] 为了找出可能的路口信息,采用类似于均值漂移(Mean Shift)的方法来找出加速度事件发生密度最大的地点。在多次均值漂移的过程中,圆心会向着加速度密度最大的点移动,而且漂移的圆的半径不断缩小。当圆的大小近似于一个路口,且当某次移动距离小于一个阈值之后,候选路口的位置就大致用最后的圆心所表示。图 1 展示了这个均值漂移定位候选路口的过程,图中的箭头为加速度向量。

[0054] 通过分析候选路口中的加速度向量的形式可以排除掉一些假路口。如果同方向的加速向量的平均向量指向朝外,那么它不是真正的路口,如图 2 所以只有平均向量指向朝里的才会被留下来。如图 3 所示。

[0055] 路口结构分析

[0056] GPS 轨迹可以用来推测路口支路的数量和方向。把 GPS 轨迹按候选路口断开,如果有两条轨迹有相同的起始候选路口和结束候选路口,或者有同样的延伸角度,那么他们被归并到同一个路段。对于一个路口,如果它拥有的支路路段个数少于三或者多余五,认为它是无效路口。当一个路口的支路被确定之后,有向 GPS 轨迹就会将一个路口的支路连接起来,构成这个路口的细节结构。这样一来,本发明系统就可以支持单行道等特殊路段。

[0057] 剩下来的路口被认为有效,并且会被给予一个唯一的 ID,而且路口支路的连接逻辑也会被记录下来。

[0058] 步骤三,使用二次 B 样条曲线来表示道路,并生成地图数据库;

[0059] 为了尽可能地记录下道路的形状以便计算距离,需要用一些带有方向的点来存储路段信息。在多条方向一致的 GPS 点中,选取在一定圆形域内的重心点作为该道路上的锚节点,并以平均方向作为该锚节点的方向。沿着 GPS 轨迹,每 10 米计算一次锚节点,计算方法类似于加权的均值漂移,只是圆移动方向被约束在平均前进方向的垂直方向。如果下一个锚节点的方向比上一个改变量大于 20 度,或者距离上一个锚节点大于 50 米,该锚节点才有效,并被记录下来。图 4 (a) 表示了一段路的有效锚节点,为图中的圆心表示,并且箭头表示其方向。

[0060] 得到一系列锚节点后,用二次 B 样条曲线来拟合道路的形状,并计算距离。二次 B 样条曲线在控制点处为一阶导数连续,符合道路的性质,而且在计算控制点上也比较方便。为了在两个相临的锚节点之间用二次 B 样条曲线表示道路,需要求出控制点。如图 5 (a),A 和 B 是锚节点,虚线 11 和 12 分别是 A 和 B 的方向上的延长线,交于 C 点。然后控制点则为 11 上的 D 和 12 上的 E,使得  $AD=AC$ , $CB=BE$ 。这样,AB 间的道路就可以大致用由 DCE 为控制点形成的 B 样条曲线表示,其长度也可以用积分计算。但是如果 A 和 B 本身大致在一条直线上,那么可能无法用 B 样条正确的表示,如图 5 (b),此时就直接用直线表示道路即可。图 4 (a) 中的锚节点生成的路段如图 4 (b) 所示。这样,所有的道路都可以用路口以及锚节点较为精确的表示。

[0061] 构建的地图数据

[0062] 地图数据包括路口 ID,路口位置,路口的出入的支路号,以及每个支路所连接的路

口,以及该路段的锚节点,都储存在数据库中,其结构如表 1。

[0063] 表 1. 地图数据库示意

[0064]

路口 ID	纬度	经度	支路 R <sub>e</sub> 的角度	支路 R <sub>e</sub> 连接的路口	支路 R <sub>e</sub> 的长度	R <sub>e</sub> 上的锚节点数目	锚节点 i 的纬度	锚节点 i 的经度	锚节点 i 的角度
001	121.346	30.135	15	00n	325	10	121.347	30.136	56

[0065] 步骤四,用众包来收集每个路口的加速度事件,并推断该路口红绿灯时间表;

[0066] 智能手机下载到地图数据之后,其后的加速事件都会跟一个路口对应。当车辆停在一个路口时,其等候所在的支路可以由路口连接关系推出。当 GPS 速度信息(近似)为零时,手机记录下零速度状态的起始时间,当检测到加速度时,系统再记下零速度状态的结束时间。当车辆驶出路口,其驶出支路的编号可以被 GPS 信息确定,然后,手机将路口 ID、入和出的支路编号、零速度状态的起始结束时间,以及检测到加速至发送数据的时间间隔发送给服务器。记录零速度状态时间的原因是为了排除在同一个队列里等候的车辆先后加速带来的多重加速事件,通过零速度状态时间可以找到最开始加速的时刻,作为该车流方向红灯转绿的时刻。以一个典型的 4 支路的路口为例,如图 6 所示,首先给支路编号,构建路口的出入支路表 2。

[0067] 表 2. 出入支路表

[0068]

	I <sub>1</sub>	I <sub>2</sub>	I <sub>3</sub>	I <sub>4</sub>
O <sub>1</sub>	/	/	1	4
O <sub>2</sub>	2	/	/	3
O <sub>3</sub>	1	4	/	/
O <sub>4</sub>	/	3	2	/

[0069] 表 2 中, (O<sub>i</sub>, I<sub>j</sub>) 表示车流从路口的支路 I<sub>j</sub> 驶向支路 O<sub>i</sub>, 并简记为 (i, j)。表中斜杠表示该方向的车流我们系统里不考虑,比如右转和掉头。表中的数字表明该单元对应车流的所属的红绿灯阶段编号,这个会在之后解释。假设每一个阶段在一个红绿灯循环周期内只发生一次,而且周期时间长度、阶段的变换次序、每个阶段的时间长度基本稳定。

[0070] 红绿灯阶段和发生次序推断

[0071] 首先,系统要找到红绿灯周期的时间长度 T。记每个 (i, j) 为一个状态 S<sub>ij</sub>, 令 S 为路口状态的集合,并用 N<sub>s</sub> 表示路口的状态数量。为了方便,我们把 S 里的状态编号,即 S<sub>1</sub>-S<sub>N<sub>s</sub></sub>, 这里不考虑状态的顺序。在状态 S<sub>i</sub> 红灯转绿灯时,第一个车辆的加速事件发生在 t<sup>k</sup><sub>i</sub>, k 是 S<sub>i</sub> 所发生的事件的数目。我们把 N<sub>s</sub> 个状态的加速事件做成表 3,对于状态 S<sub>i</sub>,令 N<sub>i</sub> 表示 S<sub>i</sub>

在目前时刻发生过的事件数目。

[0072] 表 3.  $N_s$  个状态的加速事件表

[0073]

$S_1$	$t_1^1$	$t_1^2$	...	$t_1^{N_1}$
$S_2$	$t_2^1$	$t_2^2$	...	$t_2^{N_2}$
...	...	...	...	...
$S_{N_s}$	$t_{N_s}^1$	$t_{N_s}^2$	...	$t_{N_s}^{N_{N_s}}$

[0074] 发生时间间隔在 5 秒以内的两个状态可以归并为一个状态, 归并之后, 这两个状态的事件也进行归并, 表 3 就减少一行。如果未能归并, 那么事件的时间将被加到该事件行的末端。

[0075] 从表 3 中可以推出  $T$  的一个近似估计值。首先找出每一行中后一事件减去前一事件的时间间隔, 在  $S_i$  行的所有差值中找出最小的  $\Delta t_i = \min\{t_i^k - t_{i-1}^k, k=2, 3 \dots N_i\}$ , 在  $\{\Delta t_i, i=1, 2 \dots N_s\}$  中, 我们想要在很大概率上确定  $\min\{\Delta t_i, i=1, 2 \dots N_s\}$  就是  $T$ 。我们首先计算在  $S_i$  所在行中, 经过  $n$  个红绿灯周期后  $\Delta t_i$  仍不是  $T$  的近似的概率  $X_i(n)$ 。也就是说, 在  $n$  个红绿灯周期内, 没有两个相邻的周期都有加速事件的概率为  $X_i(n)$ 。设  $S_i$  平均每个红绿灯周期里有车辆从停止到加速的事件的概率为  $p_i$ , 那么有:

$$[0076] X_i(n+2) = (1 - p_i)X_i(n+1) + p_i(1 - p_i)X_i(n) \quad \text{式三}$$

[0077] 解得:

$$[0078] X_i(n) = C_1 \left(\frac{1 - p_i + \sqrt{\Delta}}{2}\right)^n + C_2 \left(\frac{1 - p_i - \sqrt{\Delta}}{2}\right)^n$$

[0079] 其中,

$$[0080] \Delta = -3p_i^2 + 2p_i + 1, C_1 = \frac{1+p_i}{2\sqrt{\Delta}} + \frac{1}{2}, C_2 = 1 - C_1.$$

[0081] 于是, 在  $n$  个红绿灯周期中, 能够找到相邻的周期都有加速事件, 且时间间隔约等于  $T$  的概率为:  $P(n) = 1 - X_i(n)$ 。

[0082] 假设某一状态在一个红绿灯周期内有加速事件发生的概率为  $p$ , 我们想通过  $P(n)$ 、 $n$ 、 $p$  的关系, 如图 7 所示, 确保  $P(n)$  对于所有  $N_s$  个状态都在 0.8 以上, 从而在  $N_s$  个状态的  $\{\Delta t_i, i=1, 2 \dots N_s\}$  中找出  $T$  的概率为  $1 - (1 - 0.8)^{N_s}$ 。当  $N_s = 4$  时, 这个概率为 0.9984, 十分接近 1, 于是, 只要系统等够足够多的时间, 让  $P(n)$  都在 0.8, 就能在很大程度上找到  $T$  的近似值。为了估计  $p$ , 假设红绿灯周期在 2 分钟, 使用每 2 分钟内平均出现加速事件的次数来表示  $p$ , 然后通过反求出  $n$ , 系统就能够知道要等够 2\*n 分钟后再计算  $T$ 。比如, 在某路口的一个状态里, 平均每 2 分钟有加速事件的概率为 0.6, 那么找出  $P(n)=0.8$  的  $n$  对应值, 即  $n=4$ , 那么, 该路口该状态经过  $4*2=8$  分钟就可以计算  $\Delta t$  了, 等到  $N_s$  个状态的  $\Delta t$  得到, 就可以求出  $T$  的近似值。

[0083] 当路口的红绿灯周期  $T$  近似确定之后, 要为  $S$  里的所有状态找出发生的次序。为此, 找到每个状态后面, 离该状态发生最近的状态(也需要间隔 10 秒以上), 即计算  $\min\{t_j^i - t_k^i\}, 1 \leq i \leq N_s, 1 \leq j \leq N_s, t_j^i > t_k^i$ 。如果  $S_j$  紧跟着  $S_k$ , 间隔为  $\Delta t_{kj}$ , 那么  $s_k \rightarrow s_j$  组成一个链。把所有链首尾相接, 如果能够组成一个环链, 且  $|\sum(\Delta t_{ij}) - T| < 10s$

(近似相等), 则认为该环链有效, 可以组成一个周期内依次发生的红绿灯通行阶段。一个路口可能有多条有效的环链, 而且, 在车辆稀少的情况下, 同一状态也可能自己构成环链 ( $s_k \rightarrow s_k \rightarrow s_k$ )。但是, 当两个环链里的阶段在某个红绿灯周期内同时发生, 那么这两个阶段会被归并为一个阶段, 而且两个环链会相互内插、合并, 变为一条环链。如果车流量足够, 那么典型的情况是, 一个 4 条支路的路口有 4 个阶段,  $N_s=4$ ,  $s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow s_4 \rightarrow s_1$ 。图 8 展示了一个典型路口的 4 个状态, 解释了表 2 里的数字。

[0084] 红绿灯时刻表计算和修正

[0085] 每一个有效的环链都可以单独计算该环链中每个阶段的时间表, 环链里的状态这里就是一个红绿灯通行阶段, 也用  $S_i$  表示一个阶段。举例来说, 一个有  $N$  个阶段的环链, 其中  $S_i$  的持续时间为  $dts_i$ 。不失一般性, 假设阶段  $S_1$  的加速事件在  $t_s$  发生, 之后的环链中的各阶段有事件发生时, 系统按照推测的  $T$  和已知的阶段的次序, 算出各个阶段的发生次数, 并且存入表 4。这里比如, 假设  $t_1^1$  在时刻  $S_1$  发生,  $S_1-S_N$  发生的次数记为  $ns_1-ns_N$ , 于是有第一行。

[0086] 表 4. 发生的事件表

[0087]

$t_1^1$	$ns_1$	$ns_2$	...	$ns_N$
$t_1^2$	$ns'_1$	$ns'_2$	...	$ns'_N$
$t_2^1$	$ns''_1$	$ns''_2$	...	$ns''_N$
$t_3^1$	$ns'''_1$	$ns'''_2$	...	$ns'''_N$
...	...	...	...	...

[0088] 当属于该环链的加速事件不断积累, 系统在表的行数大于某个门限  $M$  的时候做一次计算, 更新  $dts_i$ , ( $1 \leq i \leq N$ ), 和  $T$ 。方法如下, 首先将表的每一列变成一个向量, 即  $t$ ,  $ns_1 \cdots ns_N$ , 共  $N+1$  个向量。为了找出最接近真实的  $dts_i$  和开始时间  $t_0$ , 构建一个代表与真实事件发生时间的平均平方误差的方程  $F$ :

$$F = (t_0 - t_s)^2 + \sum_{i=1}^M (t_0 + ns_1(i) \cdot dts_1 + ns_2(i) \cdot dts_2 + \dots + ns_N(i) \cdot dts_N - t(i))^2 \quad \text{式四}$$

[0090] 对式四求导, 可以求出最佳的  $t_0$ 、 $dts_i$ :

[0091] 每次修正过后, 表中的前 5 行会被去掉, 然后当新进来 5 个数据之后, 系统再次按照新的  $t$ ,  $ns_1 \cdots ns_N$  做一次修正:

$$\frac{\partial F}{\partial t_0} = 0, \frac{\partial F}{\partial dts_i} = 0, i = 1, 2, \dots, N$$

[0093] 在大多数情况下, 系统都能利用充足的历史数据使得各阶段的时间表误差尽可能小。

[0094] 步骤五, 利用生成的地图和前方路口的红绿灯时间表, 给车辆提示最优速度, 使得车辆尽量不在路口遇红灯停止;

[0095] 首先, 司机在使用本系统时, 先要设定目的地, 然后系统会向服务器请求出发地和

目的地之间圆形区域的道路的平均车速信息。系统根据道路长度和平均车速,求得各个路段耗时,根据 Dijkstra 算法选择耗时最小的路径。

[0096] 一旦路径确定下来,系统根据下个路口和再后面一个路口,向服务器请求下个路口合适的通行时间段。服务器收到请求后返回该路口的目标阶段的变绿剩余时间  $t_g$ ,以及变红的剩余时间  $t_r$ ,以及红绿灯周期  $T$ 。这样,车辆应该在  $t(n)$  时刻到达路口,  $t(n)$  满足  $t_g+t_{sg}+nT < t(n) < t_r-t_{sr}+nT$ 。 $t_{sg}$  和  $t_{sr}$  是为了减小误差带来的影响而设定的前后容错时间间隔,一般设为 10 秒。

[0097] 为了计算当前车辆位置到下一个路口的距离,智能手机结合下载的地图信息,将自己当前的地点与要经过的锚节点用 B 样条曲线表示,然后对距离做积分,求出剩余距离。

[0098] 剩余的距离  $d$  和剩余的时间  $t(n)$  求出后,最佳的速度应该满足

$$\frac{d}{t_r-t_{sr}+nT} < v_{opt} < \frac{d}{t_g+t_{sg}+nT} \text{ 式五}$$

[0099]  $n$  应该是一个使得  $v_{opt}$  不超过速限的最小整数。智能手机在每次经过一个路口,或在路段上每经过 1 分钟都会向服务器请求红绿灯信息。在每次 GPS 定位更新后,系统都会重新计算距离。

[0100] 最后,系统会通过手机界面显示一个合理的车速,并用提示前方路口的红绿灯状态、剩余时间、车速是否过快或者过慢。

[0101] 综上所述,本系统中的手机和服务器的运行流程如图 9 所示。

[0102] 本系统利用少数车辆由于误差导致遇到红灯停止然后加速的事件,给尽可能多的其他车辆提供车速建议。

[0103] 尽管本发明的内容已经通过上述优选实施例作了详细介绍,但应当认识到上述的描述不应被认为是对本发明的限制。在本领域技术人员阅读了上述内容后,对于本发明的多种修改和替代都将是显而易见的。因此,本发明的保护范围应由所附的权利要求来限定。

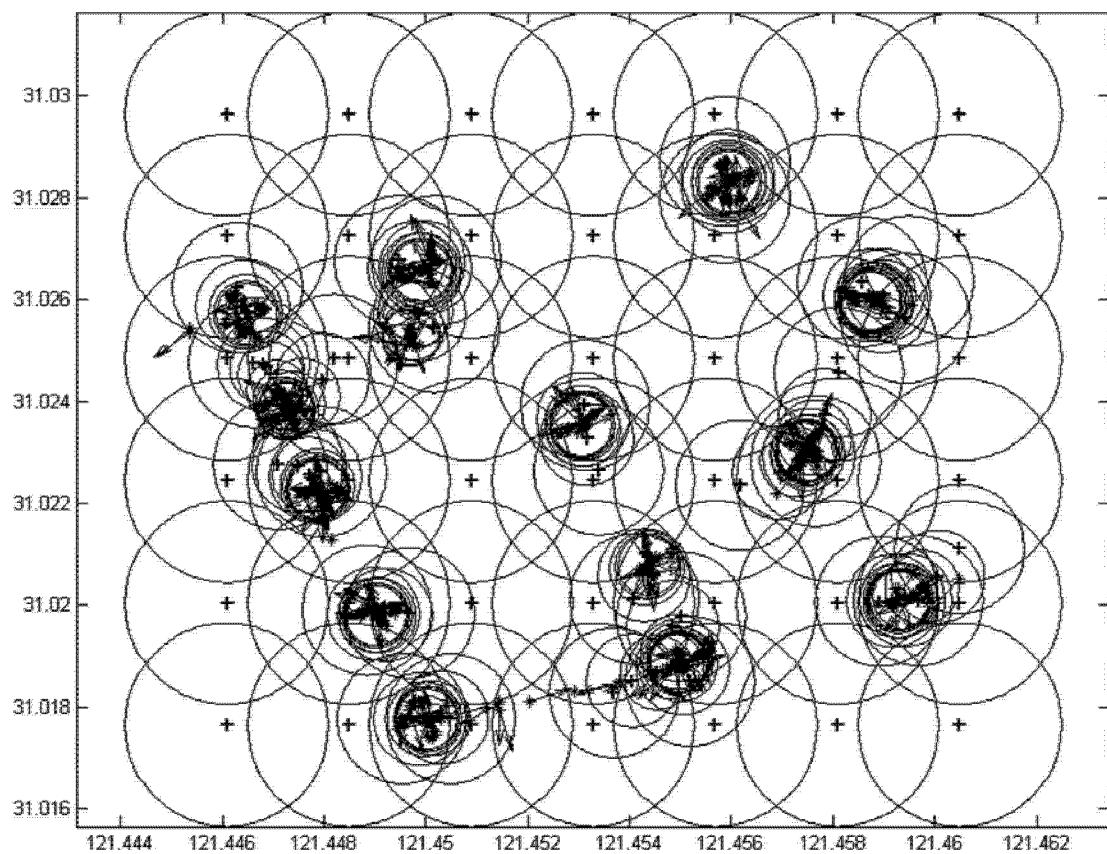


图 1

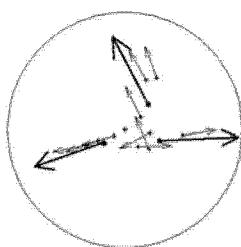


图 2

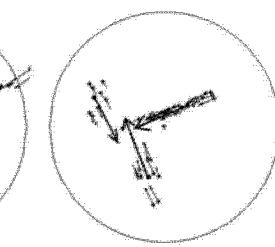
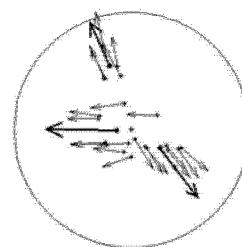


图 3

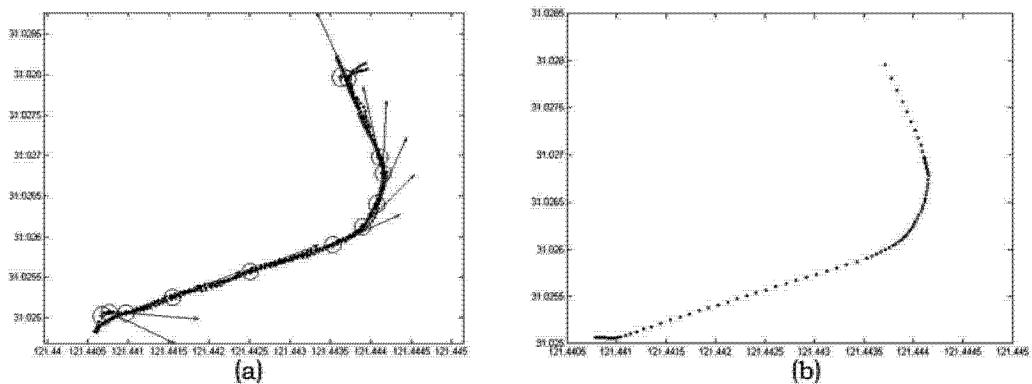


图 4

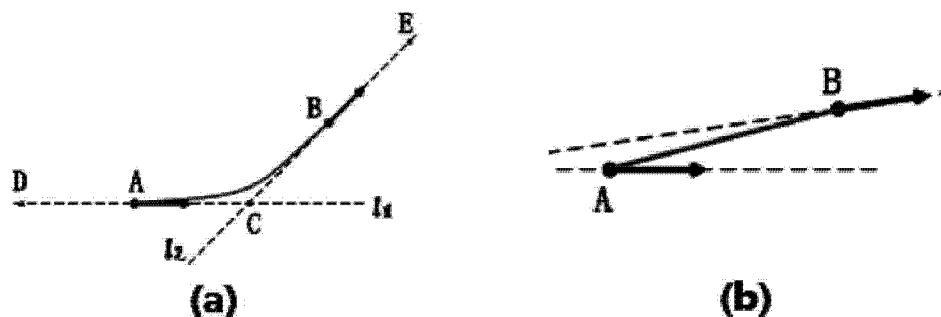


图 5

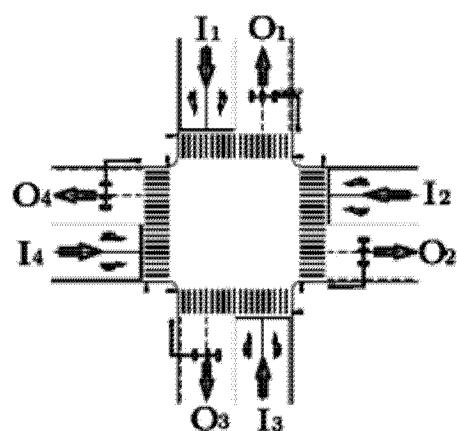


图 6

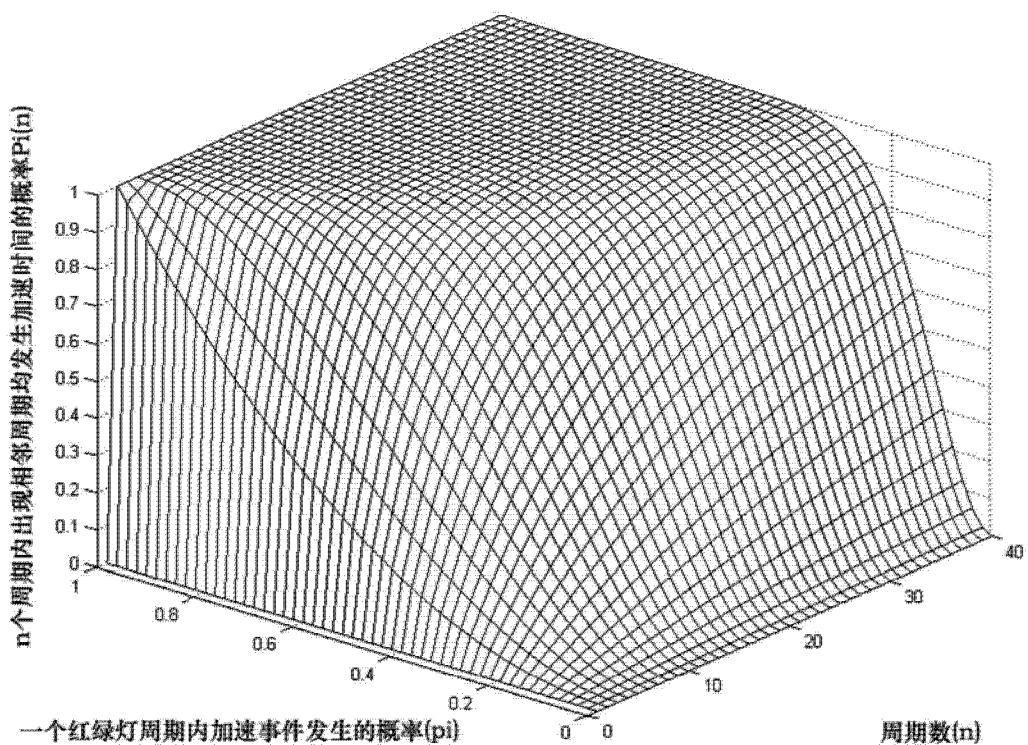


图 7

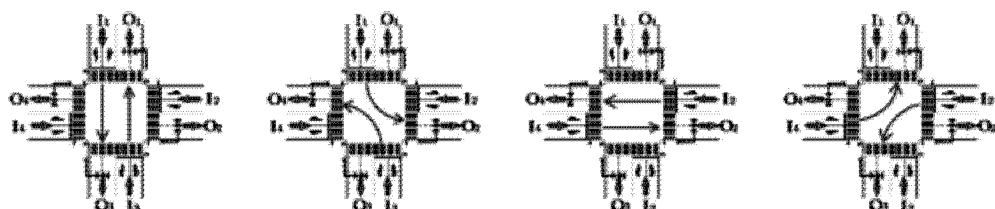


图 8

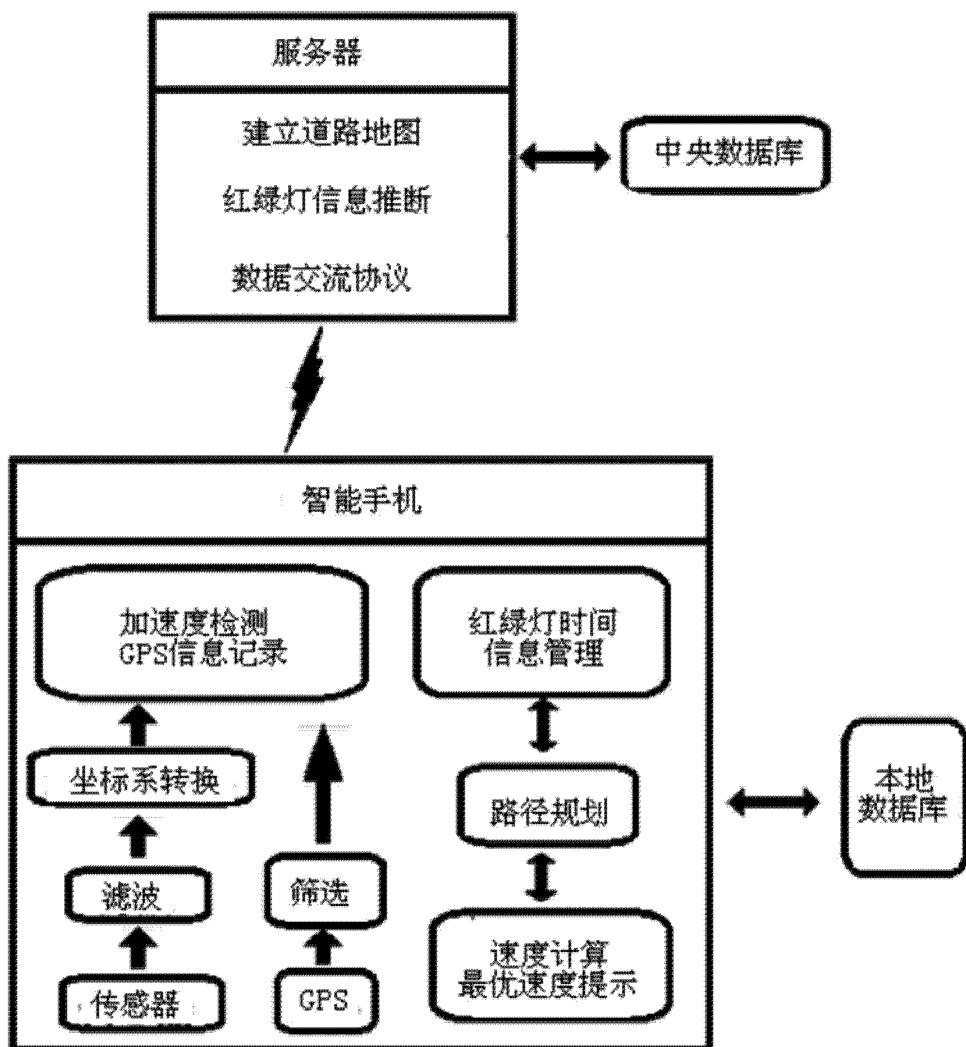


图 9