



(12) 发明专利申请

(10) 申请公布号 CN 117782102 A

(43) 申请公布日 2024. 03. 29

(21) 申请号 202311830221.7

G01C 21/16 (2006.01)

(22) 申请日 2023.12.28

G01C 21/00 (2006.01)

(71) 申请人 合肥工业大学

地址 230009 安徽省合肥市包河区屯溪路  
193号

(72) 发明人 张炳力 米向东 郑达 王焱辉  
刘中正 张羊阳

(74) 专利代理机构 安徽省合肥新安专利代理有  
限责任公司 34101

专利代理师 陆丽莉 何梅生

(51) Int. Cl.

G01C 21/20 (2006.01)

G06V 20/58 (2022.01)

G06V 20/56 (2022.01)

G06T 7/73 (2017.01)

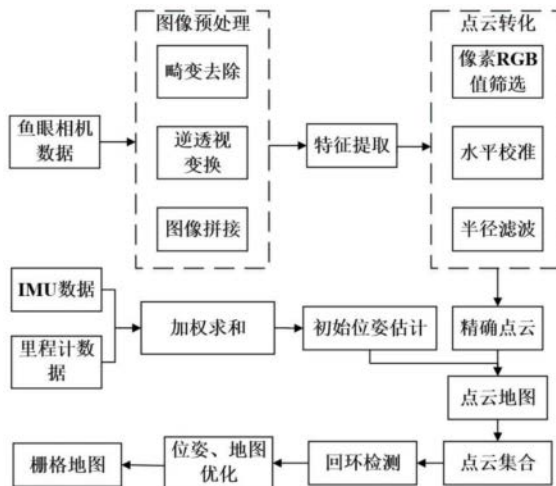
权利要求书4页 说明书8页 附图1页

(54) 发明名称

一种基于环视视觉的全自动泊车定位与建图方法

(57) 摘要

本发明公开了一种基于环视视觉的全自动泊车定位与建图方法,包括:1.通过车辆安装的四个环视鱼眼相机获取车辆周围的环视图像信息;2.利用深度学习方法提取环视图像中的车道线以及障碍物信息;3.对提取的车道线以及障碍物信息进行点云转化处理;4.融合IMU和里程计数据,进行位姿估计并生成全局停车场地图;5.通过回环检测进行地图优化,生成栅格地图用于路径导航规划。本发明结合环视相机、IMU和里程计的各自优势,提升了停车场定位与建图的精度,有助于提高泊车效率与准确度。



1. 一种基于环视视觉的全自动泊车定位与建图方法,其特征在于,包括以下步骤:

步骤1:车辆在移动过程中,通过安装在汽车前、后、左、右的鱼眼相机获取当前t时刻的环视图像后,经过相机畸变去除、逆透视变换与图像拼接的操作,得到一张t时刻的俯视图像 $I_t$ ;

步骤2:采用卷积神经网络的DeepLab V3+模型提取所述俯视图像 $I_t$ 中的车道线以及障碍物信息并进行标注,得到带有车道线以及障碍物信息的图像 $I'_t$ ;

步骤3:对提取的车道线以及障碍物信息进行点云转化;

步骤3.1、将 $I'_t$ 中的车道线以及障碍物信息所对应的像素点转化为点云 $P_t$ ;

步骤3.2、根据t时刻的车辆IMU传感器对点云 $P_t$ 进行水平校准,得到水平校准后的点云 $P'_t$ ;

步骤3.3、对水平校准后的点云 $P'_t$ 进行半径滤波处理,得到精确点云 $P''_t$ ;

步骤4:融合IMU传感器和里程计的数据,对t+1时刻的车辆位姿进行估算并生成全局停车场地图;

步骤4.1、利用式(1)计算t时刻IMU估计的车辆加速度 $\tilde{a}_t^i$ 和车辆角速度 $\tilde{\omega}_t^i$ :

$$\begin{cases} \tilde{a}_t^i = a_t^i + n_{a,t}^i + r_{a,t}^i \\ \tilde{\omega}_t^i = \omega_t^i + n_{\omega,t}^i + r_{\omega,t}^i \end{cases} \quad (1)$$

式(1)中,i表示IMU传感器坐标系, $a_t^i$ 表示t时刻IMU传感器输出的车辆加速度, $n_{a,t}^i$ 表示t时刻车辆加速度的高斯白噪声, $r_{a,t}^i$ 表示t时刻车辆加速度的随机游走噪声, $\omega_t^i$ 表示t时刻IMU传感器输出的车辆角速度; $n_{\omega,t}^i$ 表示t时刻车辆角速度的高斯白噪声, $r_{\omega,t}^i$ 表示t时刻车辆角速度的随机游走噪声;

步骤4.2、利用式(2)计算t+1时刻IMU估计的车辆速度 $v_{t+1}^i$ 和车辆角度 $\theta_{t+1}^i$ :

$$\begin{cases} \Delta_v = \int_0^{\Delta_t} \tilde{a}_t^i dt \\ \Delta_\theta = \int_0^{\Delta_t} \tilde{\omega}_t^i dt \\ v_{t+1}^i = v_t^i + \Delta_v \\ \theta_{t+1}^i = \theta_t^i + \Delta_\theta \end{cases} \quad (2)$$

式(2)中, $\Delta_t$ 表示t时刻和t+1时刻之间的差值, $\Delta_v$ 表示速度的变化量, $\Delta_\theta$ 表示角度的变化量; $v_t^i$ 和 $\theta_t^i$ 分别表示t时刻IMU估计的车辆速度和车辆角度;当t=0时,令 $v_t^i$ 和 $\theta_t^i$ 均为零;

步骤4.3、利用式(3)计算t+1时刻里程计估计的车辆速度 $v_{t+1}^o$ 和车辆角度 $\theta_{t+1}^o$ :

$$\begin{cases} v_{t+1}^o = \frac{v_{r,t+1} + v_{l,t+1}}{2} \\ s_{r,t+1} = v_{r,t+1} \times \Delta_t \\ s_{l,t+1} = v_{l,t+1} \times \Delta_t \\ \theta_{t+1}^o = \theta_t^o + \frac{s_{r,t+1} - s_{l,t+1}}{Len} \end{cases} \quad (3)$$

式(3)中,o表示里程计坐标系, $v_{r,t+1}$ 和 $v_{l,t+1}$ 分别表示t+1时刻里程计输出的右轮速度和

左轮速度,  $s_{r,t+1}$  和  $s_{l,t+1}$  分别表示  $\Delta_t$  内右轮行驶里程和左轮行驶里程,  $Len$  是车身宽度,  $\theta_t^o$  是  $t$  时刻里程计估计的车辆角度; 当  $t=0$  时, 令  $\theta_t^o$  为零;

步骤4.3、利用式(4)估算  $t+1$  时刻融合后的车辆速度  $v_{t+1}$  和车辆角度  $\theta_{t+1}$ :

$$\begin{cases} v_{t+1} = v_{t+1}^i \times \mu_1 + v_{t+1}^o \times \mu_2 \\ \theta_{t+1} = \theta_{t+1}^i \times \mu_1 + \theta_{t+1}^o \times \mu_2 \\ \mu_1 + \mu_2 = 1 \end{cases} \quad (4)$$

式(4)中,  $\mu_1$  表示IMU传感器对应的权重,  $\mu_2$  表示里程计对应的权重; 当车辆直行时, 设置  $\mu_1 < \mu_2$ ; 当车辆转弯时, 设置  $\mu_1 > \mu_2$ ;

步骤4.4、以车辆的初始位置为坐标原点, 车辆朝向为Y轴正方向, 车辆朝向沿车辆所在平面顺时针旋转  $90^\circ$  为X轴正方向, 建立二维地图坐标系;

定义车辆在  $t$  时刻的位姿  $W_t = (X_t, Y_t, V_t)$ , 其中,  $X_t, Y_t$  为车辆在  $t$  时刻的二维坐标,  $V_t$  为在  $t$  时刻车辆朝向与X轴正方向的夹角; 当  $t=0$  时, 令  $W_0 = (0, 0, 90^\circ)$ ;

步骤4.5、利用式(5)计算车辆在  $t+1$  时刻的位姿  $W_{t+1} = (X_{t+1}, Y_{t+1}, V_{t+1})$ :

$$\begin{cases} X_{t+1} = X_t + \cos V_t \times \int_0^{\Delta_t} v_{t+1} dt \\ Y_{t+1} = Y_t + \sin V_t \times \int_0^{\Delta_t} v_{t+1} dt \\ V_{t+1} = \theta_{t+1} \end{cases} \quad (5)$$

式(5)中,  $X_{t+1}$  和  $Y_{t+1}$  表示车辆在  $t+1$  时刻的二维坐标,  $V_{t+1}$  表示车辆在  $t+1$  时刻的车辆朝向与X轴正方向的夹角;

步骤4.6、在  $t$  时刻的位姿  $W_t$  上添加精确点云信息  $P''_t$ , 以构成  $t$  时刻的局部点云地图  $M_{p,t}$ ; 根据  $t+1$  时刻的位姿  $W_{t+1}$ , 使用正态分布变换算法计算  $t+1$  时刻的精确点云信息  $P''_{t+1}$  与  $M_{p,t}$  之间的旋转矩阵  $R_{t+1}$ 、平移矩阵  $T_{t+1}$ , 从而利用式(6)对  $t+1$  时刻的点云  $P''_{t+1}$  进行变换操作, 得到  $t+1$  时刻的局部点云地图  $M_{p,t+1}$ , 进而得到  $t+1$  时刻的全局地图  $M_{g,t+1} = M_{p,t} + M_{p,t+1}$ :

$$M_{p,t+1} = R_{t+1} * P''_{t+1} + T_{t+1} \quad (6)$$

步骤5: 通过回环检测对全局地图进行优化, 生成栅格地图用于路径导航规划;

步骤5.1、获取初始时刻到  $t$  时刻的精确点云集合  $P'' = \{P''_m | m=0, 1, 2, 3, \dots, t\}$ , 其中,  $P''_m$  表示第  $m$  时刻的精确点云;

步骤5.2、在  $t+1$  时刻对精确点云集合  $P''$  进行回环检测, 得到  $t+1$  时刻更新后的全局地图  $M'_{g,t+1}$ ;

步骤5.3、将  $t+1$  时刻更新后的全局地图  $M'_{g,t+1}$  转化为  $t+1$  时刻的栅格地图  $G_{t+1}$  并保存后, 用于车辆的全自动泊车。

2. 根据权利要求1所述的基于环视视觉的全自动泊车定位与建图方法, 其特征在于, 步骤3.2中根据  $t$  时刻的车辆IMU传感器对点云  $P_t$  进行水平校准, 得到水平校准后的点云  $P'_t$ 。包括:

步骤3.2.1、定义车辆IMU传感器的平面方程为:  $ax+by+cz+d=0$ , 其中,  $x, y, z$  分别是车辆IMU传感器平面的  $x$  轴、 $y$  轴、 $z$  轴,  $a, b, c$  分别是  $x, y, z$  的法向量,  $d$  为所述车辆IMU传感器平面相对于自身坐标原点的偏移量;

步骤3.2.2、定义车辆坐标系中的竖直向量为L,将所述车辆IMU平面的法向量a、b、c通过所述竖直向量L进行旋转,得到旋转矩阵 $m_r$ ;

步骤3.2.3、根据所述旋转矩阵 $m_r$ ,利用点云旋转函数对 $P_t$ 进行处理,得到旋转后的点云,将旋转后的点云的z轴高度值设置为0,从而得到水平校准后的点云 $P'_t$ 。

3.根据权利要求2所述的基于环视视觉的全自动泊车定位与建图方法,其特征在于,步骤3.3中对水平校准后的点云 $P'_t$ 进行半径滤波处理,得到精确点云 $P''_t$ ,包括:

步骤3.3.1、选取所述水平校准后的点云 $P'_t$ 中的任意一个点记为点A,以点A作为圆心,以一个距离阈值作为半径,形成一个滤波圆,判断所述滤波圆内是否包含至少两个点,若包含,则将点A保留;否则,则将点A删除;

步骤3.3.2、重复执行步骤3.3.1,直至点云 $P'_t$ 中的所有点均完成处理为止,从而将所有保留的点组成精确点云 $P''_t$ 。

4.根据权利要求3所述的基于环视视觉的全自动泊车定位与建图方法,其特征在于,步骤4.6中的正态分布变换算法是按如下步骤进行:

步骤4.6.1、将t时刻的局部点云地图 $M_{p,t}$ 中的点云进行降采样处理后,得到t时刻处理后的局部点云地图 $M'_{p,t}$ ;

步骤4.6.2、将 $M'_{p,t}$ 离散化为三维的网格,并计算每个网格中点云的均值和协方差矩阵,从而得到一个高斯分布 $GD_1$ ;

步骤4.6.3、按照步骤4.6.1和步骤4.6.2对t+1时刻的精确点云信息 $P''_{t+1}$ 进行处理,得到另一个高斯分布 $GD_2$ ;

步骤4.6.4、计算 $GD_1$ 和 $GD_2$ 之间的KL散度,并通过梯度下降法最小化KL散度,从而得到最优的旋转矩阵 $R_{t+1}$ 、平移矩阵 $T_{t+1}$ 。

5.根据权利要求4所述的基于环视视觉的全自动泊车定位与建图方法,其特征在于,步骤5.2是按如下步骤进行:

步骤5.2.1、使用Kd-tree算法快速查找 $P''$ 中与t+1时刻的精确点云 $P''_{t+1}$ 相似值最大的精确点云 $P''_k, P''_k \in P'', 0 \leq k \leq t$ ;

步骤5.2.2、计算t+1时刻的精确点云 $P''_{t+1}$ 与 $P''_k$ 的相似性程度 $S_{t+1}$ ;

步骤5.2.3、使用迭代最近点算法计算t+1时刻的精确点云 $P''_{t+1}$ 与 $P''_k$ 的之间的欧氏距离 $d_{t+1}$ ;当 $S_{t+1}$ 大于设定的相似性程度阈值,且 $d_{t+1}$ 小于设定的欧氏距离阈值时,则表示车辆回到之前的历史位置,并执行步骤5.2.4,否则,表示车辆未处于回环状态,则得到t+1时刻更新后的全局地图 $M'_{g,t+1} = M_{g,t+1}$ ;

步骤5.2.4、通过非线性优化算法对t+1时刻的位姿 $W_{t+1}$ 进行优化,以最小化t+1时刻精确点云 $P''_{t+1}$ 与 $P''_k$ 之间的误差,从而得到t+1时刻优化后的位姿 $W'_{t+1}$ ;

根据 $W'_{t+1}$ ,对t+1时刻的局部点云地图 $M_{p,t+1}$ 进行更新,得到t+1时刻更新后的局部点云地图 $M'_{p,t+1}$ ,从而得到t+1时刻更新后的全局地图 $M'_{g,t+1} = M_{g,t} + M'_{p,t+1}$ 。

6.根据权利要求5所述的基于环视视觉的全自动泊车定位与建图方法,其特征在于,步骤5.3是按如下步骤进行:

步骤5.3.1、对栅格地图 $G_{t+1}$ 初始化,包括:设置t+1时刻的栅格地图 $G_{t+1}$ 的分辨率,根据 $M'_{g,t+1}$ 的边界计算栅格地图 $G_{t+1}$ 的边界框,设置 $G_{t+1}$ 的原点也为车辆的初始位置;

步骤5.3.2、对 $M'_{g,t+1}$ 进行半径滤波、降采样处理后,得到t+1时刻轻量化的全局地图

$M''_{g,t+1}$ ;

步骤5.3.3、遍历 $M''_{g,t+1}$ 中的点云并逐一映射到栅格地图 $G_{t+1}$ 中,并将栅格地图中被占据的位置的概率标记为100,未被占据的位置的概率标记为0,从而得到t+1时刻最终的栅格地图 $G_{t+1}$ 。

7.一种电子设备,包括存储器以及处理器,其特征在于,所述存储器用于存储支持处理器执行权利要求1-6中任一所述全自动泊车定位与建图方法的程序,所述处理器被配置为用于执行所述存储器中存储的程序。

8.一种计算机可读存储介质,计算机可读存储介质上存储有计算机程序,其特征在于,所述计算机程序被处理器运行时执行权利要求1-6中任一所述全自动泊车定位与建图方法的步骤。

## 一种基于环视视觉的全自动泊车定位与建图方法

### 技术领域

[0001] 本发明涉及车辆自动驾驶领域,尤其涉及一种全自动泊车定位与建图方法。

### 背景技术

[0002] 随着城市的快速发展和汽车保有量逐年增加,车位短缺以及复杂的泊车环境已经成为了驾驶者的一大困扰,浪费了大量的时间和精力。此外,由于停车操作需要同时兼顾车辆后方以及两侧的情况,且空间紧凑,过程中又无人指导,因此对驾驶员驾驶操作的熟练程度要求很高。即便是操作熟练的驾驶员也会被繁琐的泊车过程和找不到合适的车位等问题所困扰。基于以上需求,智能泊车技术应运而生,这项技术不仅可以解决驾驶员的烦恼,而且对于停车场空间的高效利用、泊车过程的安全性都有较大益处。

[0003] 室内停车场自主泊车定位技术面临的主要问题包括两方面,一是受限于室内环境与GNSS信号常被遮挡无法使用的问题;另一方面则是由于传感器成本较高而导致自主泊车技术普及较为困难的问题。虽然能够通过搭载如激光雷达等传感器增加精确环境感知的能力,但势必会带来成本的增加。在自动驾驶领域不断压低成本,提升产业竞争力的背景下,如何实现在有限成本下优化泊车效果,成为当前亟待解决的问题。

### 发明内容

[0004] 本发明针对有限成本下泊车定位与建图优化问题,提供了一种基于环视视觉的泊车定位与建图方法,以期实现在不改变传感器配置的情况下,通过优化定位与建图来提高建图的精度,从而能保障车辆在地下停车场内安全行驶,并有助于提高泊车效率与准确度。

[0005] 本发明为达到上述发明目的,采用如下技术方案:

[0006] 本发明一种基于环视视觉的全自动泊车定位与建图方法的特点在于,包括以下步骤:

[0007] 步骤1:车辆在移动过程中,通过安装在汽车前、后、左、右的鱼眼相机获取当前 $t$ 时刻的环视图像后,经过相机畸变去除、逆透视变换与图像拼接的操作,得到一张 $t$ 时刻的俯视图像 $I_t$ ;

[0008] 步骤2:采用卷积神经网络的DeepLab V3+模型提取所述俯视图像 $I_t$ 中的车道线以及障碍物信息并进行标注,得到带有车道线以及障碍物信息的图像 $I'_t$ ;

[0009] 步骤3:对提取的车道线以及障碍物信息进行点云转化;

[0010] 步骤3.1、将 $I'_t$ 中的车道线以及障碍物信息所对应的像素点转化为点云 $P_t$ ;

[0011] 步骤3.2、根据 $t$ 时刻的车辆IMU传感器对点云 $P_t$ 进行水平校准,得到水平校准后的点云 $P'_t$ ;

[0012] 步骤3.3、对水平校准后的点云 $P'_t$ 进行半径滤波处理,得到精确点云 $P''_t$ ;

[0013] 步骤4:融合IMU传感器和里程计的数据,对 $t+1$ 时刻的车辆位姿进行估算并生成全局停车场地图;

[0014] 步骤4.1、利用式(1)计算 $t$ 时刻IMU估计的车辆加速度 $\vec{a}_t^i$ 和车辆角速度 $\vec{\omega}_t^i$ :

$$[0015] \quad \begin{cases} \tilde{a}_t^i = a_t^i + n_{a,t}^i + r_{a,t}^i \\ \tilde{\omega}_t^i = \omega_t^i + n_{\omega,t}^i + r_{\omega,t}^i \end{cases} \quad (1)$$

[0016] 式(1)中,  $i$ 表示IMU传感器坐标系,  $a_t^i$ 表示 $t$ 时刻IMU传感器输出的车辆加速度,  $n_{a,t}^i$ 表示 $t$ 时刻车辆加速度的高斯白噪声,  $r_{a,t}^i$ 表示 $t$ 时刻车辆加速度的随机游走噪声,  $\omega_t^i$ 表示 $t$ 时刻IMU传感器输出的车辆角速度;  $n_{\omega,t}^i$ 表示 $t$ 时刻车辆角速度的高斯白噪声,  $r_{\omega,t}^i$ 表示 $t$ 时刻车辆角速度的随机游走噪声;

[0017] 步骤4.2、利用式(2)计算 $t+1$ 时刻IMU估计的车辆速度 $v_{t+1}^i$ 和车辆角度 $\theta_{t+1}^i$ :

$$[0018] \quad \begin{cases} \Delta_v = \int_0^{\Delta_t} \tilde{a}_t^i dt \\ \Delta_\theta = \int_0^{\Delta_t} \tilde{\omega}_t^i dt \\ v_{t+1}^i = v_t^i + \Delta_v \\ \theta_{t+1}^i = \theta_t^i + \Delta_\theta \end{cases} \quad (2)$$

[0019] 式(2)中,  $\Delta_t$ 表示 $t$ 时刻和 $t+1$ 时刻之间的差值,  $\Delta_v$ 表示速度的变化量,  $\Delta_\theta$ 表示角度的变化量;  $v_t^i$ 和 $\theta_t^i$ 分别表示 $t$ 时刻IMU估计的车辆速度和车辆角度;当 $t=0$ 时,令 $v_t^i$ 和 $\theta_t^i$ 均为零;

[0020] 步骤4.3、利用式(3)计算 $t+1$ 时刻里程计估计的车辆速度 $v_{t+1}^o$ 和车辆角度 $\theta_{t+1}^o$ :

$$[0021] \quad \begin{cases} v_{t+1}^o = \frac{v_{r,t+1} + v_{l,t+1}}{2} \\ s_{r,t+1} = v_{r,t+1} \times \Delta_t \\ s_{l,t+1} = v_{l,t+1} \times \Delta_t \\ \theta_{t+1}^o = \theta_t^o + \frac{s_{r,t+1} - s_{l,t+1}}{Len} \end{cases} \quad (3)$$

[0022] 式(3)中,  $o$ 表示里程计坐标系,  $v_{r,t+1}$ 和 $v_{l,t+1}$ 分别表示 $t+1$ 时刻里程计输出的右轮速度和左轮速度,  $s_{r,t+1}$ 和 $s_{l,t+1}$ 分别表示 $\Delta_t$ 内右轮行驶里程和左轮行驶里程,  $Len$ 是车身宽度,  $\theta_t^o$ 是 $t$ 时刻里程计估计的车辆角度;当 $t=0$ 时,令 $\theta_t^o$ 为零;

[0023] 步骤4.3、利用式(4)估算 $t+1$ 时刻融合后的车辆速度 $v_{t+1}$ 和车辆角度 $\theta_{t+1}$ :

$$[0024] \quad \begin{cases} v_{t+1} = v_{t+1}^i \times \mu_1 + v_{t+1}^o \times \mu_2 \\ \theta_{t+1} = \theta_{t+1}^i \times \mu_1 + \theta_{t+1}^o \times \mu_2 \\ \mu_1 + \mu_2 = 1 \end{cases} \quad (4)$$

[0025] 式(4)中,  $\mu_1$ 表示IMU传感器对应的权重,  $\mu_2$ 表示里程计对应的权重;当车辆直行时,设置 $\mu_1 < \mu_2$ ;当车辆转弯时,设置 $\mu_1 > \mu_2$ ;

[0026] 步骤4.4、以车辆的初始位置为坐标原点,车辆朝向为Y轴正方向,车辆朝向沿车辆所在平面顺时针旋转 $90^\circ$ 为X轴正方向,建立二维地图坐标系;

[0027] 定义车辆在 $t$ 时刻的位姿 $W_t = (X_t, Y_t, V_t)$ ,其中,  $X_t, Y_t$ 为车辆在 $t$ 时刻的二维坐标,  $V_t$ 为在 $t$ 时刻车辆朝向与X轴正方向的夹角;当 $t=0$ 时,令 $W_0 = (0, 0, 90^\circ)$ ;

[0028] 步骤4.5、利用式(5)计算车辆在 $t+1$ 时刻的位姿 $W_{t+1} = (X_{t+1}, Y_{t+1}, V_{t+1})$ :

$$[0029] \quad \begin{cases} X_{t+1} = X_t + \cos V_t \times \int_0^{\Delta t} v_{t+1} dt \\ Y_{t+1} = Y_t + \sin V_t \times \int_0^{\Delta t} v_{t+1} dt \\ V_{t+1} = \theta_{t+1} \end{cases} \quad (5)$$

[0030] 式(5)中, $X_{t+1}$ 和 $Y_{t+1}$ 表示车辆在t+1时刻的二维坐标, $V_{t+1}$ 表示车辆在t+1时刻的车辆朝向与X轴正方向的夹角;

[0031] 步骤4.6、在t时刻的位姿 $W_t$ 上添加精确点云信息 $P''_t$ ,以构成t时刻的局部点云地图 $M_{p,t}$ ;根据t+1时刻的位姿 $W_{t+1}$ ,使用正态分布变换算法计算t+1时刻的精确点云信息 $P''_{t+1}$ 与 $M_{p,t}$ 之间的旋转矩阵 $R_{t+1}$ 、平移矩阵 $T_{t+1}$ ,从而利用式(6)对t+1时刻的点云 $P''_{t+1}$ 进行变换操作,得到t+1时刻的局部点云地图 $M_{p,t+1}$ ,进而得到t+1时刻的全局地图 $M_{g,t+1} = M_{p,t} + M_{p,t+1}$ ;

$$[0032] \quad M_{p,t+1} = R_{t+1} * P''_{t+1} + T_{t+1} \quad (6)$$

[0033] 步骤5:通过回环检测对全局地图进行优化,生成栅格地图用于路径导航规划;

[0034] 步骤5.1、获取初始时刻到t时刻的精确点云集合 $P'' = \{P''_m | m=0, 1, 2, 3, \dots, t\}$ ,其中, $P''_m$ 表示第m时刻的精确点云;

[0035] 步骤5.2、在t+1时刻对精确点云集合 $P''$ 进行回环检测,得到t+1时刻更新后的全局地图 $M'_{g,t+1}$ ;

[0036] 步骤5.3、将t+1时刻更新后的全局地图 $M'_{h,t+1}$ 转化为t+1时刻的栅格地图 $G_{t+1}$ 并保存后,用于车辆的全自动泊车。

[0037] 本发明所述的基于环视视觉的全自动泊车定位与建图方法的特点也在于,步骤3.2中根据t时刻的车辆IMU传感器对点云 $P_t$ 进行水平校准,得到水平校准后的点云 $P'_t$ 。包括:

[0038] 步骤3.2.1、定义车辆IMU传感器的平面方程为: $ax+by+cz+d=0$ ,其中,x、y、z分别是车辆IMU传感器平面的x轴、y轴、z轴,a、b、c分别是x、y、z的法向量,d为所述车辆IMU传感器平面相对于自身坐标原点的偏移量;

[0039] 步骤3.2.2、定义车辆坐标系中的竖直向量为L,将所述车辆IMU平面的法向量a、b、c通过所述竖直向量L进行旋转,得到旋转矩阵 $m_r$ ;

[0040] 步骤3.2.3、根据所述旋转矩阵 $m_r$ ,利用点云旋转函数对 $P_t$ 进行处理,得到旋转后的点云,将旋转后的点云的z轴高度值设置为0,从而得到水平校准后的点云 $P'_t$ 。

[0041] 步骤3.3中对水平校准后的点云 $P'_t$ 进行半径滤波处理,得到精确点云 $P''_t$ ,包括:

[0042] 步骤3.3.1、选取所述水平校准后的点云 $P'_t$ 中的任意一个点记为点A,以点A作为圆心,以一个距离阈值作为半径,形成一个滤波圆,判断所述滤波圆内是否包含至少两个点,若包含,则将点A保留;否则,则将点A删除;

[0043] 步骤3.3.2、重复执行步骤3.3.1,直至点云 $P'_t$ 中的所有点均完成处理为止,从而将所有保留的点组成精确点云 $P''_t$ 。

[0044] 步骤4.6中的正态分布变换算法是按如下步骤进行:

[0045] 步骤4.6.1、将t时刻的局部点云地图 $M_{p,t}$ 中的点云进行降采样处理后,得到t时刻处理后的局部点云地图 $M'_{p,t}$ ;

[0046] 步骤4.6.2、将 $M'_{p,t}$ 离散化为三维的网格,并计算每个网格中点云的均值和协方差

矩阵,从而得到一个高斯分布 $GD_1$ ;

[0047] 步骤4.6.3、按照步骤4.6.1和步骤4.6.2对 $t+1$ 时刻的精确点云信息 $P''_{t+1}$ 进行处理,得到另一个高斯分布 $GD_2$ ;

[0048] 步骤4.6.4、计算 $GD_1$ 和 $GD_2$ 之间的KL散度,并通过梯度下降法最小化KL散度,从而得到最优的旋转矩阵 $R_{t+1}$ 、平移矩阵 $T_{t+1}$ 。

[0049] 步骤5.2是按如下步骤进行:

[0050] 步骤5.2.1、使用Kd-tree算法快速查找 $P''$ 中与 $t+1$ 时刻的精确点云 $P''_{t+1}$ 相似值最大的精确点云 $P''_k, P''_k \in P'', 0 \leq k \leq t$ ;

[0051] 步骤5.2.2、计算 $t+1$ 时刻的精确点云 $P''_{t+1}$ 与 $P''_k$ 的相似性程度 $S_{t+1}$ ;

[0052] 步骤5.2.3、使用迭代最近点算法计算 $t+1$ 时刻的精确点云 $P''_{t+1}$ 与 $P''_k$ 的之间的欧氏距离 $d_{t+1}$ ;当 $S_{t+1}$ 大于设定的相似性程度阈值,且 $d_{t+1}$ 小于设定的欧氏距离阈值时,则表示车辆回到之前的历史位置,并执行步骤5.2.4,否则,表示车辆未处于回环状态,则得到 $t+1$ 时刻更新后的全局地图 $M'_{g,t+1} = M_{g,t+1}$ ;

[0053] 步骤5.2.4、通过非线性优化算法对 $t+1$ 时刻的位姿 $W_{t+1}$ 进行优化,以最小化 $t+1$ 时刻精确点云 $P''_{t+1}$ 与 $P''_k$ 之间的误差,从而得到 $t+1$ 时刻优化后的位姿 $W'_{t+1}$ ;

[0054] 根据 $W'_{t+1}$ ,对 $t+1$ 时刻的局部点云地图 $M_{p,t+1}$ 进行更新,得到 $t+1$ 时刻更新后的局部点云地图 $M'_{p,t+1}$ ,从而得到 $t+1$ 时刻更新后的全局地图 $M'_{g,t+1} = M_{g,t} + M'_{p,t+1}$ 。

[0055] 步骤5.3是按如下步骤进行:

[0056] 步骤5.3.1、对栅格地图 $G_{t+1}$ 初始化,包括:设置 $t+1$ 时刻的栅格地图 $G_{t+1}$ 的分辨率,根据 $M'_{g,t+1}$ 的边界计算栅格地图 $G_{t+1}$ 的边界框,设置 $G_{t+1}$ 的原点也为车辆的初始位置;

[0057] 步骤5.3.2、对 $M'_{g,t+1}$ 进行半径滤波、降采样处理后,得到 $t+1$ 时刻轻量化的全局地图 $M''_{g,t+1}$ ;

[0058] 步骤5.3.3、遍历 $M''_{g,t+1}$ 中的点云并逐一映射到栅格地图 $G_{t+1}$ 中,并将栅格地图中被占据的位置的概率标记为100,未被占据的位置的概率标记为0,从而得到 $t+1$ 时刻最终的栅格地图 $G_{t+1}$ 。

[0059] 本发明一种电子设备,包括存储器以及处理器的特点在于,所述存储器用于存储支持处理器执行所述全自动泊车定位与建图方法的程序,所述处理器被配置为用于执行所述存储器中存储的程序。

[0060] 本发明一种计算机可读存储介质,计算机可读存储介质上存储有计算机程序的特点在于,所述计算机程序被处理器运行时执行所述全自动泊车定位与建图方法的步骤。

[0061] 与现有技术相比,本发明的有益效果在于:

[0062] 1.本发明通过融合环视鱼眼相机、IMU和里程计信息,获取丰富的车位信息与障碍物信息,从而在不增加传感器成本的情况下提升泊车环境感知能力,克服了单独使用摄像头带来的定位不准确问题。

[0063] 2.本发明提出了一种根据车辆姿态进行水平校准的方法,当车辆经过减速带和坑洼地带时,能够即使纠正车辆受到道路因素干扰时产生的偏差,利用车辆IMU传感器信息对当前获取的点云进行水平校准,提高了获取停车位和障碍物位置信息的准确性。

[0064] 3.本发明提出了一种融合IMU传感器和里程计测量值的方法,根据车辆行驶状态和传感器的特性进行权重分配。有利于结合IMU传感器在车辆转弯、里程计在车辆直行时具

有相对较高精度的优势,进一步提升了定位与建图的精度。

## 附图说明

[0065] 图1为本发明提供的定位与建图方法流程图;

[0066] 图2为本发明提供的回环检测优化后的全局点云地图示意图;

## 具体实施方式

[0067] 下面结合说明书附图和具体实施例对本发明的技术方案做进一步详细说明。

[0068] 本实施例中,如图1所示,一种基于环视视觉的全自动泊车定位与建图方法,包括以下步骤:

[0069] 步骤1:车辆在移动过程中,通过安装在汽车前、后、左、右的鱼眼相机获取当前 $t$ 时刻的环视图像后,经过相机畸变去除、逆透视变换与图像拼接的操作,得到一张 $t$ 时刻的俯视图 $I_t$ ;实际应用中,鱼眼相机水平视场角为 $196^\circ$ ,垂直视场角为 $147^\circ$ ,检测距离 $>10m$ ,环视图像经过视频解码器输送至车辆搭载的全自动泊车控制器的视觉处理单元进行俯视图图像的生成操作。

[0070] 步骤2:采用卷积神经网络的DeepLab V3+模型提取俯视图 $I_t$ 中的车道线以及障碍物信息并进行标注,得到带有车道线以及障碍物信息的图像 $I'_t$ ;

[0071] 步骤2.1、在Ubuntu系统中安装Labelme软件,使用图像标注工具Labelme对采集的2000张停车场俯视图图片进行标注,标注出车道线以及障碍物信息,得到标注好的停车场图片数据集D;

[0072] 步骤2.2、将标注好的停车场图片数据集D输入DeepLab V3+模型进行训练,进行前向传播,计算损失函数,然后通过反向传播调整模型参数,最终得到优化后的模型M;

[0073] 步骤2.3、使用优化后的模型M对俯视图 $I_t$ 进行处理,提取出车道线以及障碍物信息,不同信息类型会被标注为不同的颜色,例如车道线标注为红色,RGB值分别为255,0,0。障碍物标注为蓝色,RGB值分别为0,0,255;进而得到带有车道线以及障碍物信息的图像 $I'_t$ 。

[0074] 步骤3:对提取的车道线以及障碍物信息进行点云转化;

[0075] 步骤3.1、将 $I'_t$ 中的车道线以及障碍物信息所对应的像素点转化为点云 $P_t$ ;默认设置点云格式为pcl::PointXYZRGB格式。

[0076] 步骤3.2、根据 $t$ 时刻的车辆IMU传感器对点云 $P_t$ 进行水平校准,得到水平校准后的点云 $P'_t$ ;

[0077] 步骤3.2.1、定义车辆IMU传感器的平面方程为: $ax+by+cz+d=0$ ,其中, $x$ 、 $y$ 、 $z$ 分别是车辆IMU传感器平面的 $x$ 轴、 $y$ 轴、 $z$ 轴, $a$ 、 $b$ 、 $c$ 分别是 $x$ 、 $y$ 、 $z$ 的法向量, $d$ 为车辆IMU传感器平面相对于自身坐标原点的偏移量;

[0078] 步骤3.2.2、定义车辆坐标系中的竖直向量为 $L$ ,默认车辆坐标系中的竖直向量 $L$ 为 $(0,0,1)$ ,将车辆IMU平面的法向量 $a$ 、 $b$ 、 $c$ 通过竖直向量 $L$ 进行旋转,得到旋转矩阵 $m_t$ ;

[0079] 步骤3.2.3、根据旋转矩阵 $m_t$ ,利用点云旋转函数对 $P_t$ 进行处理,得到旋转后的点云,将旋转后的点云的 $z$ 轴高度值设置为0,从而得到水平校准后的点云 $P'_t$ 。

[0080] 步骤3.3、对水平校准后的点云 $P'_t$ 进行半径滤波处理,得到精确点云 $P''_t$ ;

[0081] 步骤3.3.1、选取水平校准后的点云 $P'_t$ 中的任意一个点记为点A,以点A作为圆心,以一个距离阈值作为半径,默认设置距离阈值为0.1m,形成一个滤波圆,判断滤波圆内是否包含至少两个点,若包含,则将点A保留;否则,则将点A删除;

[0082] 步骤3.3.2、重复执行步骤3.3.1,直至点云 $P'_t$ 中的所有点均完成处理为止,从而将所有保留的点组成精确点云 $P''_t$ 。

[0083] 步骤4:融合IMU传感器和里程计的数据,对t+1时刻的车辆位姿进行估算并生成全局停车场地图;

[0084] 步骤4.1、利用式(1)计算t时刻IMU估计的车辆加速度 $\tilde{a}_t^i$ 和车辆角速度 $\tilde{\omega}_t^i$ :

$$[0085] \quad \begin{cases} \tilde{a}_t^i = a_t^i + n_{a,t}^i + r_{a,t}^i \\ \tilde{\omega}_t^i = \omega_t^i + n_{\omega,t}^i + r_{\omega,t}^i \end{cases} \quad (1)$$

[0086] 式(1)中,i表示IMU传感器坐标系, $a_t^i$ 表示t时刻IMU传感器输出的车辆加速度, $n_{a,t}^i$ 表示t时刻车辆加速度的高斯白噪声, $r_{a,t}^i$ 表示t时刻车辆加速度的随机游走噪声, $\omega_t^i$ 表示t时刻IMU传感器输出的车辆角速度; $n_{\omega,t}^i$ 表示t时刻车辆角速度的高斯白噪声, $r_{\omega,t}^i$ 表示t时刻车辆角速度的随机游走噪声。

[0087] 步骤4.2、利用式(2)计算t+1时刻IMU估计的车辆速度 $v_{t+1}^i$ 和车辆角度 $\theta_{t+1}^i$ :

$$[0088] \quad \begin{cases} \Delta_v = \int_0^{\Delta_t} \tilde{a}_t^i dt \\ \Delta_\theta = \int_0^{\Delta_t} \tilde{\omega}_t^i dt \\ v_{t+1}^i = v_t^i + \Delta_v \\ \theta_{t+1}^i = \theta_t^i + \Delta_\theta \end{cases} \quad (2)$$

[0089] 式(2)中, $\Delta_t$ 表示t时刻和t+1时刻之间的差值, $\Delta_v$ 表示速度的变化量, $\Delta_\theta$ 表示角度的变化量; $v_t^i$ 和 $\theta_t^i$ 分别表示t时刻IMU估计的车辆速度和车辆角度;当t=0时,令 $v_t^i$ 和 $\theta_t^i$ 均为零。

[0090] 步骤4.3、利用式(3)计算t+1时刻里程计估计的车辆速度 $v_{t+1}^o$ 和车辆角度 $\theta_{t+1}^o$ :

$$[0091] \quad \begin{cases} v_{t+1}^o = \frac{v_{r,t+1} + v_{l,t+1}}{2} \\ s_{r,t+1} = v_{r,t+1} \times \Delta_t \\ s_{l,t+1} = v_{l,t+1} \times \Delta_t \\ \theta_{t+1}^o = \theta_t^o + \frac{s_{r,t+1} - s_{l,t+1}}{Len} \end{cases} \quad (3)$$

[0092] 式(3)中,o表示里程计坐标系, $v_{r,t+1}$ 和 $v_{l,t+1}$ 分别表示t+1时刻里程计输出的右轮速度和左轮速度, $s_{r,t+1}$ 和 $s_{l,t+1}$ 分别表示 $\Delta_t$ 内右轮行驶里程和左轮行驶里程,Len是车身宽度, $\theta_t^o$ 是t时刻里程计估计的车辆角度;当t=0时,令 $\theta_t^o$ 为零。

[0093] 步骤4.3、利用式(4)估算t+1时刻融合后的车辆速度 $\mu_{t+1}$ 和车辆角度 $\theta_{t+1}$ :

$$[0094] \quad \begin{cases} v_{t+1} = v_{t+1}^i \times \mu_1 + v_{t+1}^o \times \mu_2 \\ \theta_{t+1} = \theta_{t+1}^i \times \mu_1 + \theta_{t+1}^o \times \mu_2 \\ \mu_1 + \mu_2 = 1 \end{cases} \quad (4)$$

[0095] 式(4)中, $\mu_1$ 表示IMU传感器对应的权重, $\mu_2$ 表示里程计对应的权重;当车辆直行时,设置 $\mu_1 < \mu_2$ ;当车辆转弯时,设置 $\mu_1 > \mu_2$ ;默认设置车辆直行时 $\mu_1$ 为30%, $\mu_2$ 为70%。车辆转弯时 $\mu_1$ 为70%, $\mu_2$ 为30%。

[0096] 步骤4.4、以车辆的初始位置为坐标原点,车辆朝向为Y轴正方向,车辆朝向沿车辆所在平面顺时针旋转90°为X轴正方向,建立二维地图坐标系;

[0097] 定义车辆在t时刻的位姿 $W_t = (X_t, Y_t, V_t)$ ,其中, $X_t, Y_t$ 为车辆在t时刻的二维坐标, $V_t$ 为在t时刻车辆朝向与X轴正方向的夹角;当 $t=0$ 时,令 $W_0 = (0, 0, 90^\circ)$ 。

[0098] 步骤4.5、利用式(5)计算车辆在t+1时刻的位姿 $W_{t+1} = (X_{t+1}, Y_{t+1}, V_{t+1})$ :

$$[0099] \quad \begin{cases} X_{t+1} = X_t + \cos V_t \times \int_0^{\Delta t} v_{t+1} dt \\ Y_{t+1} = Y_t + \sin V_t \times \int_0^{\Delta t} v_{t+1} dt \\ V_{t+1} = \theta_{t+1} \end{cases} \quad (5)$$

[0100] 式(5)中, $X_{t+1}$ 和 $Y_{t+1}$ 表示车辆在t+1时刻的二维坐标, $V_{t+1}$ 表示车辆在t+1时刻的车辆朝向与X轴正方向的夹角。

[0101] 步骤4.6、在t时刻的位姿 $W_t$ 上添加精确点云信息 $P''_t$ ,以构成t时刻的局部点云地图 $M_{p,t}$ ;根据t+1时刻的位姿 $W_{t+1}$ ,使用正态分布变换算法计算t+1时刻的精确点云信息 $P''_{t+1}$ 与 $M_{p,t}$ 之间的旋转矩阵 $R_{t+1}$ 、平移矩阵 $T_{t+1}$ ;

[0102] 步骤4.6.1、将t时刻的局部点云地图 $M_{p,t}$ 中的点云使用pcl::VoxelGrid函数进行降采样处理后,得到t时刻处理后的局部点云地图 $M'_{p,t}$ ;

[0103] 步骤4.6.2、将 $M'_{p,t}$ 离散化为三维的网格,默认设置网格的分辨率为1m。并计算每个网格中点云的均值和协方差矩阵,从而得到一个高斯分布 $GD_1$ ;

[0104] 步骤4.6.3、按照步骤4.6.1和步骤4.6.2对t+1时刻的精确点云信息 $P''_{t+1}$ 进行处理,得到另一个高斯分布 $GD_2$ ;

[0105] 步骤4.6.4、计算 $GD_1$ 和 $GD_2$ 之间的KL散度,并通过梯度下降法最小化KL散度,默认设置梯度下降法的迭代步长为0.01m,最大迭代次数为50次,从而得到最优的旋转矩阵 $R_{t+1}$ 、平移矩阵 $T_{t+1}$ 。

[0106] 利用式(6)对t+1时刻的点云 $P''_{t+1}$ 进行变换操作,得到t+1时刻的局部点云地图 $M_{p,t+1}$ ,进而得到t+1时刻的全局地图 $M_{g,t+1} = M_{p,t} + M_{p,t+1}$ :

$$[0107] \quad M_{p,t+1} = R_{t+1} * P''_{t+1} + T_{t+1} \quad (6)$$

[0108] 步骤5:通过回环检测对全局地图进行优化,生成栅格地图用于路径导航规划;

[0109] 步骤5.1、获取初始时刻到t时刻的精确点云集合 $P'' = \{P''_m | m=0, 1, 2, 3, \dots, t\}$ ,其中, $P''_m$ 表示第m时刻的精确点云。

[0110] 步骤5.2、在t+1时刻对精确点云集合 $P''$ 进行回环检测:

[0111] 步骤5.2.1、使用Kd-tree算法快速查找 $P''$ 中与t+1时刻的精确点云 $P''_{t+1}$ 相似值最大的精确点云 $P''_k, P''_k \in P'', 0 \leq k \leq t$ ;

[0112] 步骤5.2.2、计算t+1时刻的精确点云 $P''_{t+1}$ 与 $P''_k$ 的相似性程度 $S_{t+1}$ ；

[0113] 步骤5.2.3、使用迭代最近点算法计算t+1时刻的精确点云 $P''_{t+1}$ 与 $P''_k$ 的之间的欧氏距离 $d_{t+1}$ ；默认设置相似性程度阈值为90%，欧氏距离阈值为0.03m。当 $S_{t+1}$ 大于设定的相似性程度阈值，且 $d_{t+1}$ 小于设定的欧氏距离阈值时，则表示车辆回到之前的历史位置，并执行步骤5.2.4，否则，表示车辆未处于回环状态，则得到t+1时刻更新后的全局地图 $M'_{g,t+1} = M_{g,t+1}$ 。

[0114] 步骤5.2.4、通过非线性优化算法对t+1时刻的位姿 $W_{t+1}$ 进行优化，以最小化t+1时刻精确点云 $P''_{t+1}$ 与 $P''_k$ 之间的误差，从而得到t+1时刻优化后的位姿 $W'_{t+1}$ ；

[0115] 根据 $W'_{t+1}$ ，对t+1时刻的局部点云地图 $M_{p,t+1}$ 进行更新，得到t+1时刻更新后的局部点云地图 $M'_{p,t+1}$ ，从而得到t+1时刻更新后的全局地图 $M'_{g,t+1} = M_{p,t} + M'_{p,t+1}$ ，如图2所示。

[0116] 步骤5.3、将t+1时刻更新后的全局地图 $M'_{g,t+1}$ 转化为t+1时刻的栅格地图 $G_{t+1}$ ：

[0117] 步骤5.3.1、对栅格地图 $G_{t+1}$ 初始化，包括：设置t+1时刻的栅格地图 $G_{t+1}$ 的分辨率，根据 $M'_{g,t+1}$ 的边界计算栅格地图 $G_{t+1}$ 的边界框，设置 $G_{t+1}$ 的原点也为车辆的初始位置；

[0118] 步骤5.3.2、对 $M'_{g,t+1}$ 进行半径滤波、降采样处理后，得到t+1时刻轻量化的全局地图 $M''_{g,t+1}$ ；

[0119] 步骤5.3.3、遍历 $M''_{g,t+1}$ 中的点云并逐一映射到栅格地图 $G_{t+1}$ 中，并将栅格地图中被占据的位置的概率标记为100，未被占据的位置的概率标记为0，从而得到t+1时刻最终的栅格地图 $G_{t+1}$ 并保存后，用于车辆的全自动泊车。

[0120] 本实施例中，一种电子设备，包括存储器以及处理器，该存储器用于存储支持处理器执行上述方法的程序，该处理器被配置为用于执行该存储器中存储的程序。

[0121] 本实施例中，一种计算机可读存储介质，是在计算机可读存储介质上存储有计算机程序，该计算机程序被处理器运行时执行上述方法的步骤。

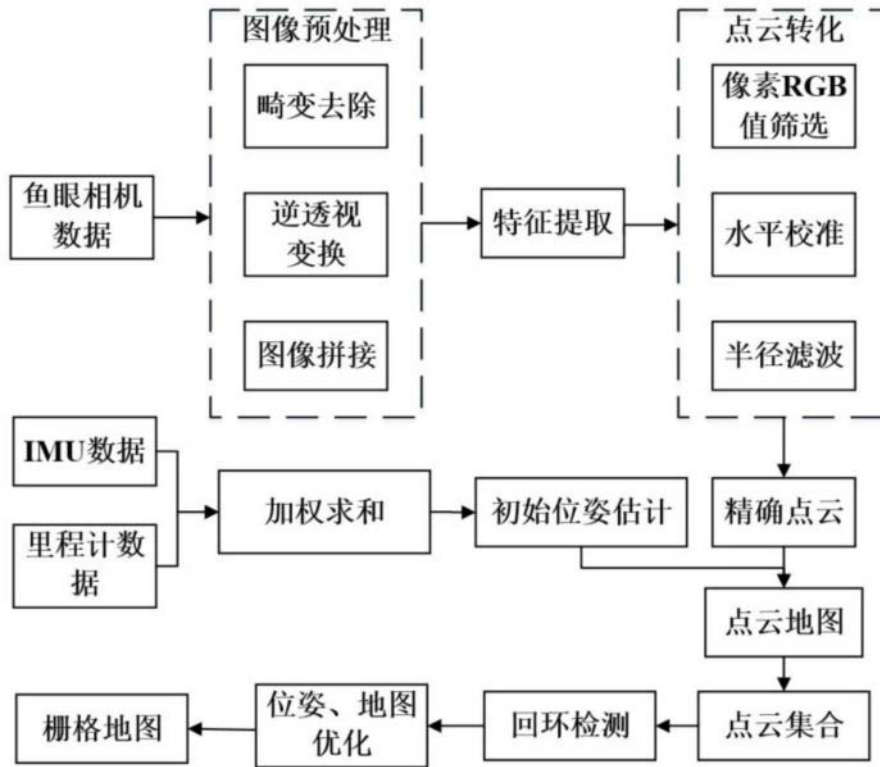


图1

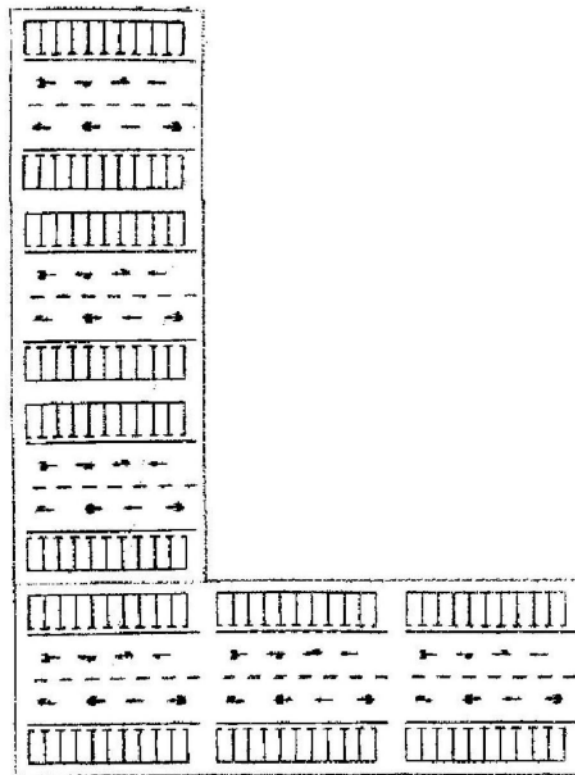


图2