

文章编号: 1002-0446(2008)04-0289-09

基于全景视觉的移动机器人同步定位与地图创建研究

许俊勇, 王景川, 陈卫东

(上海交通大学自动化系, 上海 200240)

摘要: 提出了一种基于全景视觉的移动机器人同步定位与地图创建 (Omni-vSLAM) 方法. 该方法提取颜色区域作为视觉路标; 在分析全景视觉成像原理和定位不确定性的基础上建立起系统的观测模型, 定位出路标位置, 进而通过扩展卡尔曼滤波算法 (EKF) 同步更新机器人位置和地图信息. 实验结果证明了该方法在建立环境地图的同时可以有效地修正由里程计造成的累积定位误差.

关键词: 全景视觉; 同步定位与地图创建 (SLAM); 视觉路标; 不确定性; 扩展卡尔曼滤波

中图分类号: TP24

文献标识码: B

Omni-vision-Based Simultaneous Localization and Mapping of Mobile Robots

XU Jun-yong, WANG Jing-chuan, CHEN Wei-dong

(Shanghai Jiaotong University, Shanghai 200240, China)

Abstract: A method of omni-vision-based simultaneous localization and mapping (Omni-vSLAM) for mobile robots is introduced. The presented method extracts the color areas and uses the extracted information as the visual landmarks. The imaging principle and localization uncertainty of the omni-vision are analyzed to build the observation model and locate the landmarks. Then the robot poses and map information are updated synchronously with the help of extended Kalman filter (EKF). The experiment results show that the presented method can build the environment map and correct the accumulated location error caused by the odometer simultaneously.

Keywords: omni-vision; simultaneous localization and mapping (SLAM); visual landmark; uncertainty; extended Kalman filter (EKF)

1 引言 (Introduction)

移动机器人必须具有探索未知区域、创建可靠的环境地图并同时利用地图来进行自主定位的能力, 才能具备完全的自主性. 同步定位与地图创建 (SLAM) 就是实现这一能力的关键技术, 它是机器人定位领域的热门研究课题之一. 传统的 SLAM 主要利用激光、声纳等得以实现^[1,2], 然而这些传感器获得的信息不仅形式单一, 而且信息量少, 在机器人应用环境复杂化和任务需求多样化的趋势中, 已经无法满足 SLAM 发展的需要. 相比而言, 视觉传感器价格低、信息量大, 再加上计算机视觉技术的不断发展, 基于视觉传感器的

SLAM (vSLAM) 具有更广泛的应用领域和研究价值.

目前, vSLAM 的研究工作主要采用基于普通视觉传感器的立体视觉和单目视觉^[3,4]. 然而, 普通视觉传感器在视域上有局限, 在特征跟踪和定位的连续性上存在不足之处. 全景视觉 (Omni-vision) 具有 360° 全方位视角, 在机器人导航、视频监控、多媒体会议等领域都有着广泛的应用^[5]. 它不仅具有普通视觉传感器的优点, 而且通过发挥其观察视野广这一特殊优势, 可以获得既丰富又较完整的环境信息, 增强了跟踪和定位目标的能力, 很好地弥补了目前 vSLAM 研究中的不足.

全景视觉不仅给 vSLAM 问题的研究带来了机

遇,同时也带来了挑战:首先,信息量的增多会加大算法的复杂度,降低实时性;其次,由于全景图像畸变较大,要提取鲁棒的环境特征并匹配是非常困难的,而特征选择的优劣又与vSLAM数据关联的正确性直接相关;另外,目前存在的全景定位方法主要采用对大量环境全景图像组成的数据库进行分析和学习的方法^[6,7],这些方法无法应用于环境地图未知的情况.由此可见,对基于全景视觉的SLAM(Omni-vSLAM)课题的研究不仅能够推动vSLAM自身技术的发展,提供新的思路,而且可以拓宽全景视觉的应用领域(例如月球探索、人员搜救等),具有重要的意义.

本文从研究角度出发,提出了一种Omni-vSLAM的实现方法.该方法提取全景图像中的颜色区域作为视觉路标来建立地图,在对全景视觉成像原理和定位不确定性进行分析的基础上建立起系统的观测模型,定位路标位置,进而利用扩展卡尔曼滤波算法(EKF)同步更新机器人和地图信息,修正由里程计带来的累积定位误差.针对全景图像非线性变化强、成像模型复杂的缺点,该方法分别采用特征筛选和观测等效变换等手段,提高了定位的精度,增强了算法的效率.利用人工视觉路标的实验结果证明了方法的有效性.

2 系统结构(System configuration)

本文采用的Omni-vSLAM系统结构如图1所示:系统的输入是里程计数据和同一时刻采集到的全景图像,系统的输出是扩展卡尔曼滤波算法更新后的机器人位置和地图.

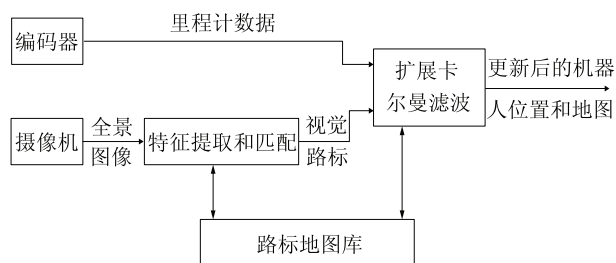


图1 系统结构图

Fig.1 Diagram of system configuration

Omni-vSLAM的地图是由全景图像中提取出的、带有空间位置信息的视觉特征组成,这些特征也称为“视觉路标”.机器人在行进中通过识别和观测这些视觉路标来建立地图,并修正自己的位置.当观测结果与路标地图库中的某个视觉路标匹配时,就把它作为这个视觉路标的观测信息

输出,否则就作为一个新路标信息加入到路标地图库中.

EKF算法模块是整个系统的核心,它是SLAM中最常用的算法之一^[8].实际SLAM系统的运动模型和观测模型可以看作是带高斯噪声的非线性模型,采用EKF能够对系统状态进行良好的跟踪.系统的协方差矩阵包含了机器人位置和地图路标位置的不确定信息^[9],通过观测地图中的路标,协方差矩阵任何子矩阵的行列式单调递减,不确定性逐渐变小,保证了机器人和路标定位误差的收敛性.

3 全景图像特征提取(Extracting features from omnidirectional image)

本文使用的全景视觉系统是基于单视点折射成像原理实现的^[10].系统由摄像机和安装于其镜头正上方的双曲面镜组成:通过双曲面镜反射到摄像机中的光线都通过同一视点,该视点对应的是双曲面的焦点 F ,摄像机的镜头中心 C 则对应另一个焦点 F' (图2).折反射全景视觉系统具有视域广、价格低、实用性强等特点,但是由于其图像畸变较大,分辨率低,且目标距离和图像位置成非线性变化关系,因此在vSLAM实现上具有挑战性.

传统的基于局部特征的匹配和跟踪算法在畸变较大的全景图像中比较难以应用,因为它们一般都通过特征点邻域图像进行匹配.相反,在光照均匀的环境中,颜色信息在全景图像中较为稳定,识别和跟踪的性能也强,因此可以通过提取和定位显著的颜色区域作为视觉路标来创建地图.

本文采用的提取方法参考RoboCup中使用的颜色区域提取法:基于HSV颜色模型提取颜色,对外界光照条件敏感程度低,识别能力较强^[11].实验中摆放颜色鲜明的柱状物作为人工视觉路标,在进行vSLAM前,先对其颜色进行标定,选取包含路标颜色 H 、 S 、 V 的上下限阈值.运行时,机器人从采集到的全景图像的HSV空间中提取出该阈值范围内的像素作为目标像素,再利用游程长度编码区域分割法^[12]把相邻的目标像素合并成颜色区域,每块颜色区域就代表一个观测到的路标.在分割的同时计算出区域的面积、中点等属性,其中,中点像素 (u,v) 作为该路标的成像位置,也就是EKF算法中输入的观测量.值得注意的是,需要合适地选取颜色阈值,以便在有效提取路标颜色的同时降低干扰颜色的影响.

在图像实际处理过程中，总会不可避免地提取出不属于路标图像的少量干扰颜色点，这些点在区域分割中会被误认为是路标。此外由后面 4.2 节的分析可知，利用距离机器人较远的路标对应的颜色区域定位出的空间位置不确定性较大，精度较低，而且会增大误匹配的概率；而近处的路标不确定性小，定位精度高，才是适合选取的。因此需要对得到的颜色区域特征进行筛选，去掉干扰颜色点和远处路标的成像区域。根据前面的描述，可以得出如下的筛选方法：

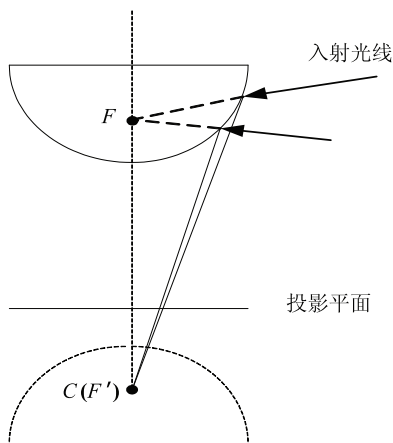


图 2 单视点示意图

Fig.2 Sketch map of single viewpoint

a) 如果颜色区域 P 的面积小于某个阈值，则认为它是干扰颜色点，即：

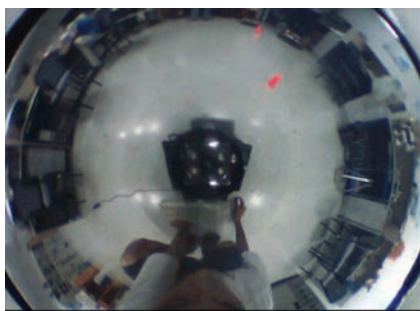
若 $Area_P < Area_{th}$ ，则 $P \notin \text{Landmark}$ 。

b) 如果颜色区域 P 的中心与全景图像中心之间的距离大于某个阈值，则认为它是远处路标图像，即：

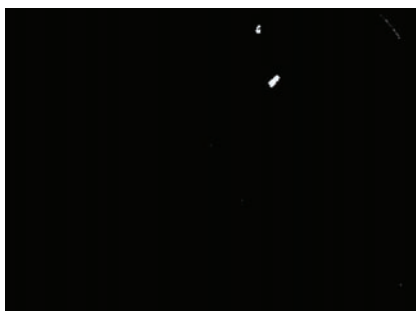
若 $Distance_{C_P-C_1} > Distance_{th}$ ，则 $P \notin \text{Landmark}$ 。

一个实验场景的全景图像处理结果如图 3 所示。原始全景图像（图 3(a)）中有一近一远两个人工视觉路标，在此场景中，路标颜色的 HSV 阈值可选取为： $H_{Low} = 200, H_{High} = 254, S_{Low} = 60, S_{High} = 200, V_{Low} = 100, V_{High} = 200$ 。从图 3(b) 中可以看出，颜色提取不仅把路标图像提取出来，而且还提取出了少量的干扰颜色点。而经过特征筛选后只有近处的路标图像被很好地保存下来，见图 3(c)（实验参数为面积阈值 $Area_{th} = 50$ 像素，距离阈值 $Distance_{th} = 150$ 像素）。

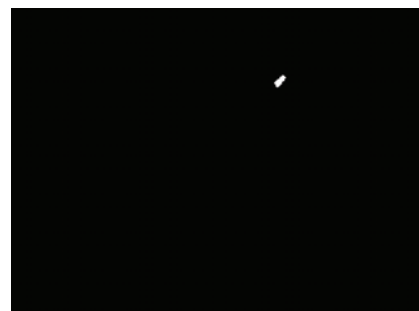
路标颜色区域的中心通过全景成像模型可以定位到二维平面坐标位置；尽管实际选用的是三维柱状人工视觉路标，但是它的几何尺寸不大，由此所产生的定位误差可以融合到观测量 (u, v) 的噪声干扰中。



(a) 原始图像



(b) 颜色提取后的图像



(c) 特征筛选后的图像

图 3 全景图像处理结果

Fig.3 Results of omnidirectional image processing

4 同步定位与地图创建 (Simultaneous localization and mapping)

从系统的角度来看，SLAM 是一个滤波问题 [13]，是根据系统从 0 时刻到 k 时刻的控制输入和观测信息来估计系统当前的状态。SLAM 中系统的状态向量表示为 $\mathbf{X} = (\mathbf{X}_R^T, \mathbf{X}_{F_1}^T, \mathbf{X}_{F_2}^T, \dots, \mathbf{X}_{F_n}^T)^T$ ，由机器人的位置 \mathbf{X}_R 和地图中的路标位置 $\mathbf{X}_{F_i} (i = 1, 2, \dots, n)$ 组成。基于 EKF 的 SLAM 算法包含两个过程：运动

更新和观测更新，分别是由系统的运动模型和观测模型描述的。下面通过建立 Omni-vSLAM 的运动模型和观测模型，分析其 EKF 算法的实现。

运动更新

机器人在水平面上运动时的坐标系表示如图 4 所示。

机器人的世界坐标为 $\mathbf{X}_R = (x_R, y_R, \theta_R)^T$ ，初始时刻机器人的局部坐标系被定义为世界坐标系，起始坐标和协方差矩阵设定为：

$$\mathbf{X}_R(0) = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\mathbf{P}_R(0) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0.01 \end{pmatrix}$$

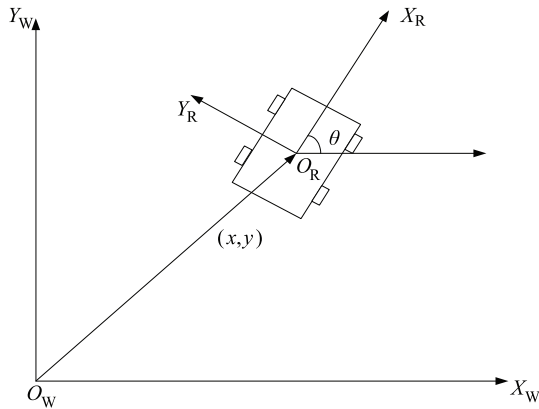


图 4 机器人局部坐标系和世界坐标系

Fig.4 Robot local coordinate frame and global coordinate frame

假设机器人在 k 时刻的位置为 $\mathbf{X}_R(k) = (x_R(k), y_R(k), \theta_R(k))^T$ ，由里程计得到的控制输入为 $\mathbf{u}(k) = (\Delta x_R(k), \Delta y_R(k), \Delta \theta_R(k))^T$ ，则在 $k+1$ 时刻可得：

$$\mathbf{X}_R(k+1) = \begin{pmatrix} x_R(k+1) \\ y_R(k+1) \\ \theta_R(k+1) \end{pmatrix}$$

$$= \begin{pmatrix} x_R(k) + \Delta x_R(k) \cos \theta_R(k) - \Delta y_R(k) \sin \theta_R(k) \\ y_R(k) + \Delta x_R(k) \sin \theta_R(k) + \Delta y_R(k) \cos \theta_R(k) \\ \theta_R(k) + \Delta \theta_R(k) \end{pmatrix}$$

$$= f_R(\mathbf{X}_R(k), \mathbf{u}(k))$$

上述表达式是机器人的运动方程，系统输入 $\mathbf{u}(k)$ 所包含的高斯噪声为 $\mathbf{v}(k)$ ，协方差矩阵为 $\mathbf{Q}(k)$ ，是对角阵。由此得到 $k+1$ 时刻机器人位置的协方差矩阵：

$$\mathbf{P}_R(k+1) = \frac{\partial f_R}{\partial \mathbf{X}_R} \mathbf{P}_R(k) \left(\frac{\partial f_R}{\partial \mathbf{X}_R} \right)^T + \frac{\partial f_R}{\partial \mathbf{u}} \mathbf{Q}(k) \left(\frac{\partial f_R}{\partial \mathbf{u}} \right)^T$$

得到机器人的运动方程后，系统的运动模型也就容易求解了。令系统在 k 时刻的状态和控制

输入为：

$$\mathbf{X}(k) = \begin{pmatrix} \mathbf{X}_R(k) \\ \mathbf{X}_{F_1}(k) \\ \mathbf{X}_{F_2}(k) \\ \vdots \\ \mathbf{X}_{F_n}(k) \end{pmatrix}$$

$$\mathbf{P}(k) = \begin{pmatrix} \mathbf{P}_{X_R X_R}(k) & \mathbf{P}_{X_R X_{F_1}}(k) & \mathbf{P}_{X_R X_{F_2}}(k) & \cdots & \mathbf{P}_{X_R X_{F_n}}(k) \\ \mathbf{P}_{X_{F_1} X_R}(k) & \mathbf{P}_{X_{F_1} X_{F_1}}(k) & \mathbf{P}_{X_{F_1} X_{F_2}}(k) & \cdots & \mathbf{P}_{X_{F_1} X_{F_n}}(k) \\ \mathbf{P}_{X_{F_2} X_R}(k) & \mathbf{P}_{X_{F_2} X_{F_1}}(k) & \mathbf{P}_{X_{F_2} X_{F_2}}(k) & \cdots & \mathbf{P}_{X_{F_2} X_{F_n}}(k) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{X_{F_n} X_R}(k) & \mathbf{P}_{X_{F_n} X_{F_1}}(k) & \mathbf{P}_{X_{F_n} X_{F_2}}(k) & \cdots & \mathbf{P}_{X_{F_n} X_{F_n}}(k) \end{pmatrix}$$

$$\mathbf{u}(k) = \begin{pmatrix} \Delta x_R(k) \\ \Delta y_R(k) \\ \Delta \theta_R(k) \end{pmatrix}$$

则在 $k+1$ 时刻，对系统新状态的预测为：

$$\mathbf{X}(k+1|k) = f(\mathbf{X}(k), \mathbf{u}(k))$$

$$= \begin{pmatrix} f_R(\mathbf{X}_R(k), \mathbf{u}(k)) \\ \mathbf{X}_{F_1}(k) \\ \mathbf{X}_{F_2}(k) \\ \vdots \\ \mathbf{X}_{F_n}(k) \end{pmatrix}$$

$$\mathbf{P}(k+1|k) = \frac{\partial f}{\partial \mathbf{X}} \mathbf{P}(k) \left(\frac{\partial f}{\partial \mathbf{X}} \right)^T + \frac{\partial f}{\partial \mathbf{u}} \mathbf{Q}(k) \left(\frac{\partial f}{\partial \mathbf{u}} \right)^T$$

其中：

$$\frac{\partial f}{\partial \mathbf{X}} = \begin{pmatrix} \frac{\partial f_R}{\partial \mathbf{X}_R} & 0 & 0 & \cdots & 0 \\ 0 & \mathbf{I} & 0 & \cdots & 0 \\ 0 & 0 & \mathbf{I} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I} \end{pmatrix}$$

$$\frac{\partial f}{\partial \mathbf{u}} = \begin{pmatrix} \frac{\partial f_R}{\partial \mathbf{u}} & 0 & 0 & \cdots & 0 \end{pmatrix}$$

状态 $\mathbf{X}(k)$ 到 $\mathbf{X}(k+1|k)$ 的转移函数 f 就是系统的运动模型。从 f 中可以发现地图路标的位置 $\mathbf{X}_{F_i}(i=1,2,\dots,n)$ 并没有更新，也就是说 $\mathbf{X}_{F_i}(k) = \mathbf{X}_{F_i}(k+1)(i=1,2,\dots,n)$ ，这是因为从里程计得到的控制输入并不包含地图路标的位置信息。运动更新过程的结果是得到对系统状态的初步估计 $\mathbf{X}(k+1|k)$ 。

观测更新

对于全景视觉来讲，人工视觉路标 $F_i(i = 1, 2, \dots, n)$ 的观测值 z_{F_i} 就是其二维平面坐标 $(x_{F_i}, y_{F_i})^T$ 在全景图像上的成像像素 $(u_{F_i}, v_{F_i})^T$ ，其高斯噪声为 $w(k)$ ，协方差矩阵为 $R(k)$ 。为了简化观测模型，减小计算量，本文采用类似于激光传感器的观测量 $(r_{F_i}, \theta_{F_i})^T$ 来等效替代像素坐标 $(u_{F_i}, v_{F_i})^T$ ，其中 r_{F_i} 表示机器人与路标位置之间的距离， θ_{F_i} 表示路标在机器人局部坐标系下的方位角。该变换的推导过程如下：图5是全景视觉成像的示意图，全景视觉装置安装在机器人的正上方，摄像机镜头安装高度为 h ，中心为 C （也就是另一个焦点 F' ），焦距为 f 。双曲面镜的镜面参数是 $\frac{z^2}{a^2} - \frac{x^2 + y^2}{b^2} = 1$ ，离心率 $e = \sqrt{a^2 + b^2}$ 。全景图像的中心像素为 $(u_{ctr}, v_{ctr})^T$ ，路标 F_i 的成像像素为 $(u_{F_i}, v_{F_i})^T$ 。

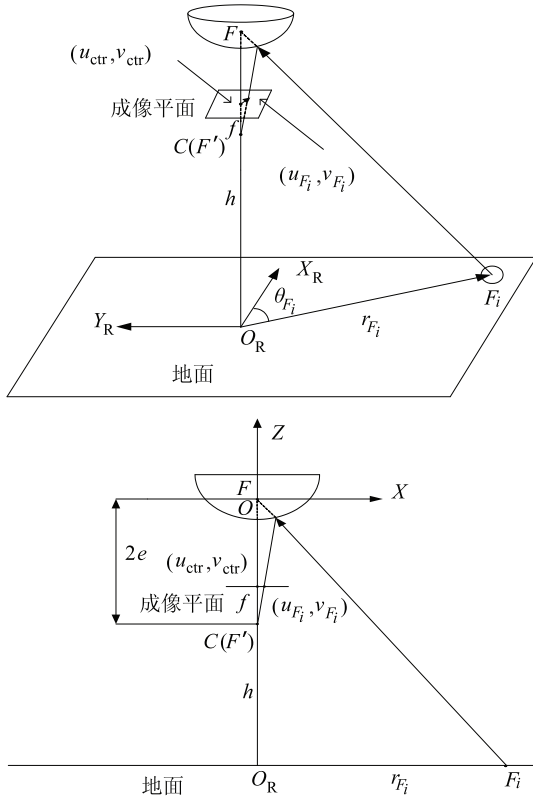


图5 全景视觉成像示意图

Fig.5 Sketch map of omni-vision imaging

显然，由示意图可知：

$$\theta_{F_i} = \text{atan2}(u_{F_i} - u_{ctr}, v_{F_i} - v_{ctr}) \quad (1)$$

$\text{atan2}(y, x)$ 表示坐标点 (x, y) 的象限角，取值范围为 $-\pi \sim \pi$ 。

以镜面焦点 F 为原点建立二维坐标系， X 轴沿机器人到路标之间的位置矢量方向， Z 轴朝向正上

方，这时曲面方程就变为双曲线方程：

$$\frac{(z+e)^2}{a^2} - \frac{x^2}{b^2} = 1$$

入射光的反射光线通过镜头中心 $C = (0, -2e)^T$ ，斜率为：

$$k = \frac{f}{\sqrt{(u_{ctr} - u_{F_i})^2 + (v_{ctr} - v_{F_i})^2}}$$

因此反射光线的直线方程就为： $z = kx - 2e$ 。假设在该二维坐标系中反射光线与镜面双曲线的交点坐标为 $(x_m, z_m)^T$ ，则联立求解上述两式可得到交点 $(x_m, z_m)^T$ ：

$$\begin{cases} \frac{(z_m + e)^2}{a^2} - \frac{x_m^2}{b^2} = 1 \\ z_m = kx_m - 2e \end{cases} \Rightarrow \begin{cases} x_m = \frac{b^2}{ke - \sqrt{a^2(1+k^2)}} \\ z_m = \frac{kb^2}{ke - \sqrt{a^2(1+k^2)}} - 2e \end{cases}$$

从几何关系上又可知：

$$\frac{r_{F_i}}{2e + h} = -\frac{x_m}{z_m} \Rightarrow r_{F_i} = \frac{b^2(2e + h)}{k(a^2 + e^2) - 2ea\sqrt{1+k^2}} \quad (2)$$

所以式 (1)、(2) 就是旧观测量 $(u_{F_i}, v_{F_i})^T$ 到新观测量 $(r_{F_i}, \theta_{F_i})^T$ 的变换公式，在此基础上可求得新观测量的协方差矩阵为：

$$R_{F_i}(k) = \begin{pmatrix} \frac{\partial r_{F_i}}{\partial u_{F_i}} & \frac{\partial r_{F_i}}{\partial v_{F_i}} \\ \frac{\partial \theta_{F_i}}{\partial u_{F_i}} & \frac{\partial \theta_{F_i}}{\partial v_{F_i}} \end{pmatrix} R(k) \begin{pmatrix} \frac{\partial r_{F_i}}{\partial u_{F_i}} & \frac{\partial r_{F_i}}{\partial v_{F_i}} \\ \frac{\partial \theta_{F_i}}{\partial u_{F_i}} & \frac{\partial \theta_{F_i}}{\partial v_{F_i}} \end{pmatrix}^T$$

进一步可推得由观测产生的路标位置 $(x_{F_i}, y_{F_i})^T$ 的不确定性，也就是协方差矩阵为：

$$P_{F_i-OB}(k) = \begin{pmatrix} \frac{\partial x_{F_i}}{\partial r_{F_i}} & \frac{\partial x_{F_i}}{\partial \theta_{F_i}} \\ \frac{\partial y_{F_i}}{\partial r_{F_i}} & \frac{\partial y_{F_i}}{\partial \theta_{F_i}} \end{pmatrix} R_{F_i}(k) \begin{pmatrix} \frac{\partial x_{F_i}}{\partial r_{F_i}} & \frac{\partial x_{F_i}}{\partial \theta_{F_i}} \\ \frac{\partial y_{F_i}}{\partial r_{F_i}} & \frac{\partial y_{F_i}}{\partial \theta_{F_i}} \end{pmatrix}^T$$

由于这里不考虑机器人本身位置的不确定性，因此由观测造成的路标全局坐标和局部坐标的不确定性是相等的，即：

$$\begin{aligned} \frac{\partial x_{F_i}}{\partial r_{F_i}} &= \frac{\partial r_{F_i} \cos \theta_{F_i}}{\partial r_{F_i}} = \cos \theta_{F_i} \\ \frac{\partial y_{F_i}}{\partial r_{F_i}} &= \frac{\partial r_{F_i} \sin \theta_{F_i}}{\partial r_{F_i}} = \sin \theta_{F_i} \\ \frac{\partial x_{F_i}}{\partial \theta_{F_i}} &= \frac{\partial r_{F_i} \cos \theta_{F_i}}{\partial \theta_{F_i}} = -r_{F_i} \sin \theta_{F_i} \\ \frac{\partial y_{F_i}}{\partial \theta_{F_i}} &= \frac{\partial r_{F_i} \sin \theta_{F_i}}{\partial \theta_{F_i}} = r_{F_i} \cos \theta_{F_i} \end{aligned}$$

不同远近的路标在全景图像中的成像像素虽然含有相同的高斯噪声分布 $R(k)$ ，但是噪声对由观测模型测量出的路标位置的不确定性大小

$\mathbf{P}_{F_i\text{-OB}}(k)$ 却是不同的. 如图 6 所示, 从全景图像中心像素出发, 从水平方向开始, 每隔 45° 取一条射线, 在每条射线上等间距 40 像素采样 4 个点, 共采样 8 组 32 个点, 根据实际实验参数求出这些点对应的路标位置和协方差矩阵, 其中不确定性用 95% 置信度的协方差椭圆表示. 从定位结果可以看出, 距离机器人越远的路标定位的不确定性越大, 定位效果越差, 而且图像上的线性变化对应的是空间位置的非线性变化, 因此只有选取合适位置的路标建立地图才能保证定位的精度.

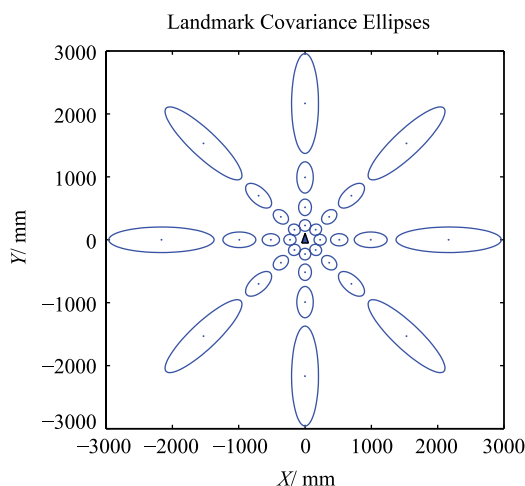


图 6 视觉路标定位不确定性分析

Fig.6 Analysis of visual landmarks' localization uncertainty

采用新观测量 $(r_{F_i}, \theta_{F_i})^T$ 之后, 依据图 5 的几何关系, 可以得到它与机器人位置 $\mathbf{X}_R(k)$ 之间的关系式:

$$\begin{aligned} \mathbf{z}_{F_i}(k) &= \begin{pmatrix} r_{F_i}(k) \\ \theta_{F_i}(k) \end{pmatrix} \\ &= \begin{pmatrix} \sqrt{(x_{F_i}(k) - x_R(k))^2 + (y_{F_i}(k) - y_R(k))^2} \\ \text{atan2}(y_{F_i}(k) - y_R(k), x_{F_i}(k) - x_R(k)) - \theta_R(k) \end{pmatrix} \\ &= h_{F_i}(\mathbf{X}(k)) \end{aligned}$$

该式也是新观测量与系统状态之间的关系函数 h_{F_i} , 也就是系统的观测模型. 与旧观测量相比, 其观测模型大大简化, 并且实际实现时可以把全景图像上所有可能的观测像素点 $(u, v)^T$ 到新观测量

$(r, \theta)^T$ 之间的转换结果保存成一个查询表, 这样每次转换可以通过查表进行, 缩短了计算时间, 增强了实时性.

观测更新过程的前提是要获得观测量 \mathbf{z} 与地图中路标 F_i 之间的匹配关系, 也就是 SLAM 中的数据关联问题. 常用的数据关联算法有两种: 单一相容最近邻域法 (ICNN, Individual Compatibility Nearest Neighbor) 和联合相容分枝界限法 (Joint Compatibility Branch and Bound, JCBB) [14]. 本文提出的 Omni-vSLAM 方法采用 JCBB 算法实现数据关联. 与 ICNN 相比, JCBB 逐个求解匹配时不仅考虑观测值与路标之间的单一相容性 (IC), 还考虑了该匹配与其他匹配之间的联合相容性 (JC), 从中找出满足该性质的匹配数最多的结果作为正确匹配结果. 由于考虑到匹配之间的相互关系, 错误匹配满足联合相容性的概率很小, 因此提高了正确性. 匹配的正确性很重要, 错误的匹配会引起 EKF 算法发散, 导致定位失败.

假设系统在 $k+1$ 时刻得到了地图库中 m 个路标的观测值 $\mathbf{z}_{F_i,j}(j=1,2,\dots,m; i \in \{1,2,\dots,n\})$, 根据 EKF 算法有:

$$\begin{aligned} \mathbf{X}(k+1|k+1) &= \mathbf{X}(k+1|k) + \mathbf{W}_{F_i,j}(k+1)\mathbf{v}_{F_i,j}(k+1) \\ \mathbf{P}(k+1|k+1) &= \mathbf{P}(k+1|k) - \mathbf{W}_{F_i,j}(k+1)\mathbf{S}_{F_i,j}(k+1) \cdot \\ &\quad \mathbf{W}_{F_i,j}(k+1)^T \end{aligned}$$

其中:

$$\begin{aligned} \mathbf{v}_{F_i,j}(k+1) &= \mathbf{z}_{F_i,j}(k+1) - h_{F_i}(\mathbf{X}(k+1|k)) \\ \mathbf{W}_{F_i,j}(k+1) &= \mathbf{P}(k+1|k) \left(\frac{\partial h_{F_i}}{\partial \mathbf{X}} \right)^T \mathbf{S}_{F_i,j}(k+1)^{-1} \\ \mathbf{S}_{F_i,j}(k+1) &= \frac{\partial h_{F_i}}{\partial \mathbf{X}} \mathbf{P}(k+1|k) \left(\frac{\partial h_{F_i}}{\partial \mathbf{X}} \right)^T + \mathbf{R}_{F_i}(k+1|k) \\ \frac{\partial h_{F_i}}{\partial \mathbf{X}} &= \begin{pmatrix} \frac{\partial h_{F_i}}{\partial X_R} & 0 & \dots & \frac{\partial h_{F_i}}{\partial X_{F_i}} & 0 & \dots \end{pmatrix} \end{aligned}$$

若某个观测量 \mathbf{z} 没有与地图库中的任何路标匹配上, 那么 \mathbf{z} 对应的就是一个新的视觉路标 F_{n+1} , 位置为:

$$\mathbf{X}_{F_{n+1}}(k+1) = \begin{pmatrix} x_{F_{n+1}}(k+1) \\ y_{F_{n+1}}(k+1) \end{pmatrix} = \begin{pmatrix} x_R(k+1|k) + r_{F_{n+1}}(k+1) \cos(\theta_R(k+1|k) + \theta_{F_{n+1}}(k+1)) \\ y_R(k+1|k) + r_{F_{n+1}}(k+1) \sin(\theta_R(k+1|k) + \theta_{F_{n+1}}(k+1)) \end{pmatrix} = g(\mathbf{z}_{F_{n+1}}(k+1))$$

把它加入到原系统状态中得到：

$$\mathbf{X}(k+1|k+1) = \begin{pmatrix} \mathbf{X}(k+1|k) \\ \mathbf{X}_{F_{n+1}}(k+1) \end{pmatrix}$$

$$\mathbf{P}(k+1|k+1) = \begin{pmatrix} \mathbf{P}(k+1|k) & \mathbf{0} \\ \mathbf{0} & \left(\frac{\partial \mathbf{g}}{\partial \mathbf{z}_{F_{n+1}}} \mathbf{R}_{F_{n+1}}(k+1) \frac{\partial \mathbf{g}}{\partial \mathbf{z}_{F_{n+1}}} \right)^T \end{pmatrix}$$

从上面的结果可以看出，观测更新完成了对系统状态的修正和 Omni-vSLAM 地图的建立。有 m 个观测值，就有 m 次独立的观测更新，这些更新过程可以有效地减小系统的不确定性，提高定位的精度。由于系统状态中同时包含了机器人位置和地图信息，因此就实现了同步定位与地图创建。

5 实验及结果分析 (Experiments and results analysis)

实验共分两个部分：仿真和真实环境实验。仿真结果说明了全景视觉大视域的特点在解决 SLAM 问题上的优势，真实环境实验结果则证明了本文所提出的 Omni-vSLAM 方法的有效性。

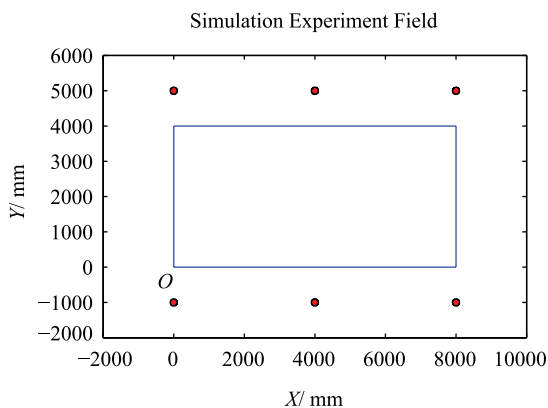


图 7 仿真实验场地示意图
Fig.7 Diagram of simulation experiment field

仿真

普通摄像机的视角约为 60° ，现假设有两个机器人，机器人 1 的传感器感知范围为前向 60° ，机器人 2 为全向 360° ，感知半径都是 3 m。如图 7 所示，仿真场地中共有 6 个路标，两个机器人同时从原点出发以 0.1 m/s 的速度沿 8 m×4 m 的矩形边缘逆时针运动 4 周。该仿真共进行两组，第一组施加较小的高斯噪声，第二组施加较大的高斯噪声，以此来模拟不同条件的环境。为了显示一般性，每组仿真进行 10 次，得到 10 个独立的结果。图 8 比较了两个机器人在不同环境下最终 SLAM 定位误差

的均方差及最大、最小偏差。从图中可以看出机器人 2 的定位误差较小，这是因为感知范围越大，观测路标的时间越长，修正定位误差的能力也越强；反之，感知范围越小，不但得到观测值的次数越少，而且建地图的能力也越弱。上述仿真结果说明了全景视觉大视域的优点加强了特征跟踪的连续性，从而提高了 vSLAM 定位的精度。

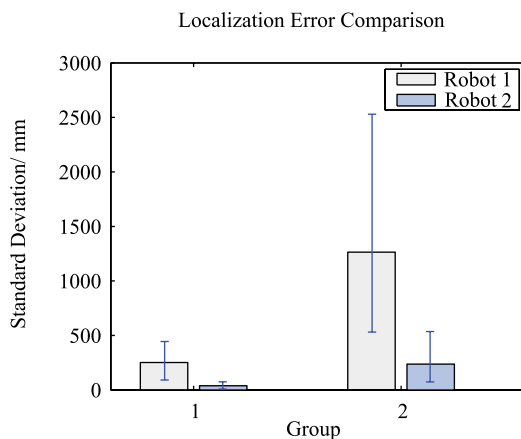


图 8 不同感知域机器人的 SLAM 定位误差比较
Fig.8 Comparison between SLAM localization errors of robots with different perceptual fields

真实环境实验

本文采用的实验平台是上海交通大学自动化系机器人与智能信息处理研究所自主开发的 Frontier-II 移动机器人 (图 9)。该机器人使用 PentiumM 1.2 G 处理器，内存为 512 M，配备了无线网络系统和全景视觉系统，其摄像机分辨率为 640×480 像素。



图 9 Frontier-II 机器人
Fig.9 Frontier-II robot

实验场地是约 6 m×8 m 的实验室环境，光照均匀，在场地的固定位置摆放 8 个柱状人工视觉

路标,通过远程控制让机器人以约 0.15 m/s 的速度从原点出发,沿水平长度为 2.7 m、竖直长度为 5.4 m 的“L”形路径逆时针来回运动 3 周,最后回到原点(图 10).第 1 周运动时,机器人通过全景视觉观测计算出人工视觉路标的位置并建立地图,这时,里程计定位的累积误差还比较小,里程计的轨迹与 Omni-vSLAM 的轨迹差别不大;而且由于初期观测误差对 Omni-vSLAM 的定位精度影响较大,因此会出现 Omni-vSLAM 定位误差大于里程计定位误差的现象.图 11(a) 和 11(b) 显示了第 2、第 3 周后的定位结果,虚线代表里程计运动轨迹,实线代表 Omni-vSLAM 运动轨迹,圆点代表路标,路标周围是 95%置信度的协方差椭圆.机器人运动到第 2、3 周时,里程计定位的累积误差逐渐显现,不断递增;而由于有地图帮助修正定位误差,Omni-vSLAM 的运动轨迹偏离很小,定位误差能保持在一定范围内.

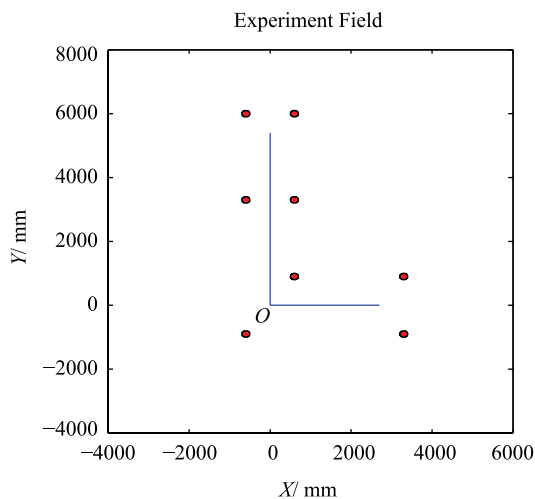
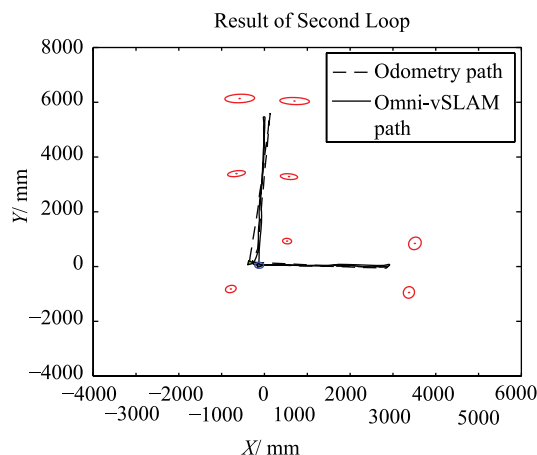


图 10 实验场地示图

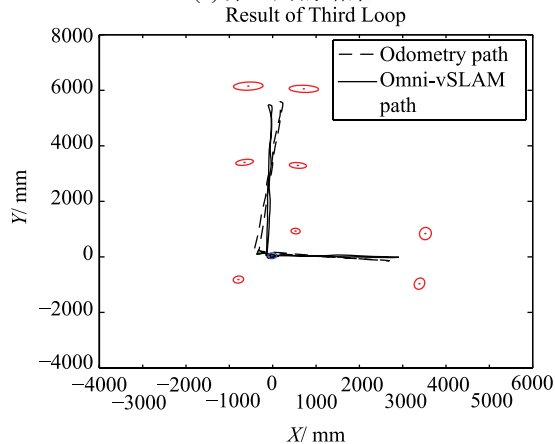
Fig.10 Diagram of experiment field

仿真中已经提到:全景视觉对 vSLAM 定位精度的提高来源于对目标跟踪能力的加强,图 12 反映了实验中路标的观察情况:左图是每个运行周期观察到的路标个数,右图是通过假设观察范围被限制在前向 60°、对原有观测值进行筛选而得到一个受限视域的观察结果.不难看出,全景视觉在观察和跟踪能力上具有很强的优势,使得 Omni-vSLAM 始终能够保持较好的更新系统状态、修正定位误差的能力.从实验过程中还可以发现,全景视觉在初期就能够通过观察身后的路标来建立地图,进行定位;在转弯处,机器人也可以始终锁定目标,而初期快速有效地建立环境地图和在大运动中不丢失目标的能力对 vSLAM 的性能具有非常

重要的影响,这些恰恰是传统视觉传感器存在的不足.图 13 是最终里程计与 Omni-vSLAM 的定位误差的比较,可以看出里程计的整体定位误差是发散的,而 Omni-vSLAM 则是收敛的,平均误差在 50 mm 左右,从而验证了本文提出的 Omni-vSLAM 实现方法是有效的.



(a) 第 2 周的结果



(b) 第 3 周的结果

图 11 Omni-vSLAM 定位结果

Fig.11 Localization results of SLAM using Omni-vSLAM

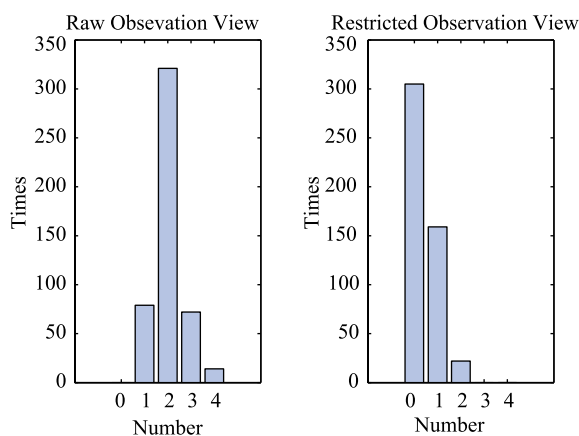


图 12 路标观察数量统计

Fig.12 Number statistic of observed landmarks

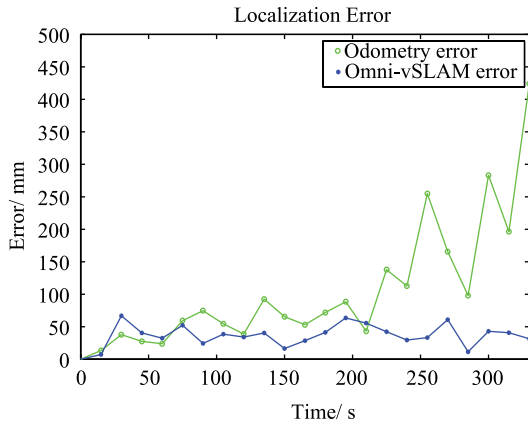


图13 定位误差比较

Fig.13 Localization error comparison

总结上述实验结果,可以得到该 Omni-vSLAM 方法具有以下优势:

(1) 全景视觉可以通过对路标的连续性观测,提高 SLAM 定位的精度。

(2) 全景视觉由于能够获得更完整的环境信息,从而可以在初期快速有效地建立地图,对较单一的环境具有很好的适应性。

(3) 由于采用单摄像机,避免了多摄像机视觉系统负荷过大、实时性不高的问题。

6 结论 (Conclusion)

全景视觉视野广大,观测和跟踪目标的能力强,在机器人导航和定位方面有独到的优点。本文提出了一种基于全景视觉实现机器人同步定位与地图创建的方法 (Omni-vSLAM),并且通过实验结果证明了该方法在建立环境地图的同时可以有效地修正里程计造成的累积定位误差。从应用角度考虑,本方法目前存在局限性,如对远处路标的定位性能差,只适用于观测一定距离内的路标。下一步的工作将在提高该方法性能的同时尝试如何从全景图像中提取鲁棒性更好的自然环境特征,并且从建立二维地图扩展到建立三维地图。

参考文献 (References)

- [1] Tardos J D, Neira J, Newman P, *et al.* Robust mapping and localization in indoor environments using sonar data[J]. *The International Journal of Robotics Research*, 2002, 21(4): 311~330.
- [2] Guivant J, Nebot E, Baiker S. Localization and map building using laser range sensors in outdoor applications[J]. *Journal of Robotic Systems*, 2000, 17(10): 565~583.
- [3] Se S, Lowe D G, Little J J. Vision-based global localization and mapping for mobile robots[J]. *IEEE Transactions on Robotics*,

2005, 21(3): 364~375.

- [4] Davison A J. Real-time simultaneous localisation and mapping with a single camera[A]. *Proceedings of the IEEE International Conference on Computer Vision*[C]. Piscataway, NJ, USA: IEEE, 2003. 1403~1410.
- [5] Ishiguro H. Development of low-cost compact omnidirectional vision sensors and their applications[DB/OL]. <http://www.ai.soc.i.kyoto-u.ac.jp/publications/98/98conf04.pdf>, 1998.
- [6] Bunschoten R. Mapping and Localization from a Panoramic Vision Sensor[DB/OL]. <http://www.science.uva.nl/research/ias/alumni/ph.d.theses/theses/RolandBunschoten.pdf>, 2003.
- [7] Andreasson H, Treptow A, Duckett T. Localization for mobile robots using panoramic vision, local features and particle filter[A]. *Proceedings of the IEEE International Conference on Robotics and Automation*[C]. Piscataway, NJ, USA: IEEE, 2005. 3348~3353.
- [8] Durrant-Whyte H, Bailey T. Simultaneous localization and mapping: Part I[J]. *IEEE Robotics and Automation Magazine*, 2006, 13(2): 99~110.
- [9] Dissanayake M W M G, Newman P, Clark S, *et al.* A solution to the simultaneous localization and map building (SLAM) problem[J]. *IEEE Transactions on Robotics and Automation*, 2001, 17(3): 229~241.
- [10] Zivkovic Z, Booi O. How did we built our hyperbolic mirror omnidirectional camera-practical issues and basic geometry[DB/OL]. <http://staff.science.uva.nl/~zivkovic/Publications/trIAS-UVA-05-04.pdf>, 2005.
- [11] Sural S, Qian G, Pramanik S. Segmentation and histogram generation using the HSV color space for image retrieval[A]. *Proceedings of the 2002 International Conference on Image Processing*[C]. Piscataway, NJ, USA: IEEE, 2002. 589~592.
- [12] 王磊, 陈卫东. 面向自主机器人的实时彩色视觉系统[J]. *上海交通大学学报*, 2003, 37(11): 1645~1648. Wang Lei, Chen Wei-dong. An autonomous robot oriented real-time color vision system[J]. *Journal of Shanghai Jiaotong University*, 2003, 37(11): 1645~1648.
- [13] 罗荣华, 洪炳镛. 机器人同时定位与地图创建研究进展[J]. *机器人*, 2004, 26(2): 182~186. Luo Rong-hua, Hong Bing-rong. The progress of simultaneous localization and mapping for mobile robot[J]. *Robot*, 2004, 26(2): 182~186.
- [14] Neira J, Tardos J D. Data association in stochastic mapping using the joint compatibility test[J]. *IEEE Transactions on Robotics and Automation*, 2001, 17(6): 890~897.

作者简介:

许俊勇 (1983-), 男, 硕士生. 研究领域: 机器视觉, vSLAM.

王景川 (1979-), 男, 研究实习员. 研究领域: 机器人控制.

陈卫东 (1968-), 男, 教授. 研究领域: 智能机器人控制, 多机器人协作.