

# 基于EM-ORB算法的移动机器人SLAM系统研究

陈昱皓<sup>1,2</sup>, 彭道刚<sup>1,2</sup>, 王志萍<sup>1,2</sup>, 夏飞<sup>1,2</sup>

(1. 上海电力学院 自动化工程学院, 上海 200090;

2. 上海电力学院 上海发电过程智能管控工程技术研究中心, 上海 200090)

**摘要:** 针对 Kinect 相机存在的固有噪声, 在机器人视觉定位与建图中提出一种改进 ORB 特征匹配算法结合改进环境测量模型的 SLAM 系统, 该系统使用改进 ORB 算法提取图像的特征点, 建立相邻帧之间特征点的对应关系, 并对深度图进行滤波; 使用 ICP 算法计算机器人运动, 通过环境测量模型去除误匹配点, 同时进行回环检测, 最后使用  $g^2o$  对位姿进行全局优化, 建立环境点云图。通过实际环境与公开数据集的运行测试, 结果表明, 该 V-SLAM 能够准确地完成相机位姿的更新, 并建立环境点云图。

**关键词:** 同时定位与建图; 移动机器人; Kinect 相机; 环境测量模型

**中图分类号:** TM341 **文献标识码:** A **DOI:** 10.19457/j.1001-2095.dqcd19261

## Research on the SLAM System Based on Improved EM Model with ORB Algorithm and Mobile Robot

CHEN Yuhao<sup>1,2</sup>, PENG Daogang<sup>1,2</sup>, WANG Zhiping<sup>1,2</sup>, XIA Fei<sup>1,2</sup>

(1. College of Automation Engineering, Shanghai University of Electric Power, Shanghai 200090, China;

2. Shanghai Engineering Research Center of Intelligent Management and Control for Power Process,

Shanghai university of Electric Power, Shanghai 200090, China)

**Abstract:** In the 3D V-SLAM, due to the inherent noise of Kinect camera, a V-SLAM system combine the improved ORB features extraction algorithm with the improved environment measurement model was proposed. The improved ORB algorithm was adopted to extract the feature points of the RGB image, compute the descriptors of those points simultaneously, so that the correspondency of the feature points between the adjacent frame could be established. At meanwhile the noised of depth image were cleared. Secondly by adopting the iterative closest point algorithm, the motion estimation of the camera pose were gained. The environment measurement model could be adopted to wipe off the outlier points, refine the camera pose as well. At last, the  $g^2o$  library was used to optimize the global robot pose, construct the sparse point clouds map. According to the real scene and public dataset, the experimental result shows that the V-SLAM system can compute the transformation of the camera pose precisely, and construct the pointclouds successfully.

**Key words:** simultaneous localization and mapping (SLAM); mobile robot; Kinect camera; environment measurement model

随着机器人应用领域的发展, 机器人导航技术得到了广泛的应用, 而其中定位与建图技术是至关重要的。在未知环境中, 机器人首先要明确自己的位置, 记录下自己在地图各个位置的方向和坐标, 为之后的运动规划打下基础。

同时定位与建图(SLAM)技术发展了很长的时间, 传统方法中常用的传感器有激光雷达与单目摄像头。早期的视觉方法都以滤波为主, 由于单目摄像头尺度不确定性等原因, 为了确定图像中物体与相机的距离, 需要大量的计算, 影响了系统的实时性; 激光雷达由于其高昂的价格, 大

大提高了SLAM技术的成本。微软公司推出的 Kinect 摄像头, 由于其低廉的价格, 并且可以同时得到 RGB 图像与其对应的深度图像, 受到研究者广泛的青睐。

使用 RGB-D 相机的 SLAM 方法主要基于特征点提取, 早期由文献[1]通过 SIFT 方法提取特征点并获得对应的深度值, 使用迭代最近点(iterative closest point, ICP)算法计算出相机的位姿, 并且构建出对应的 3D 点云图<sup>[1]</sup>, 后来文献[2]提出了一种基于 FAST 角点提取算法的 ORB (oriented FAST and rotated BRIEF) 特征点提取与对

基金项目: 上海市“科技创新行动计划”社会发展领域项目(16DZ1202500)

作者简介: 陈昱皓(1993-), 男, 硕士研究生, Email: kotokoruru@gmail.com

通讯作者: 彭道刚(1977-), 男, 教授, 博士(后), Email: pengdaogang@126.com

应描述子计算的方法,该算法通过计算特征点周围像素点的灰度值来确定图像的质心,从而计算出特征点的方向,使得FAST算法具有旋转不变性,所谓特征点的描述子,则是通过在特征点周围随机选择一组像素点,通过比较灰度值大小关系而得到一组描述该特征点的二进制向量。由于ORB算法具有较好的鲁棒性,相比传统的SIFT与SURF算法,有着极快的提取速度,因此基于特征点提取的方法使得视觉SLAM在工程上的应用成为可能。但目前SLAM系统的主要问题在于减少传感器固有噪声带来的精度影响。

综上所述,本文创新点主要在以下方面:

1)结合文献[3]提出的环境测量(environment measurement, EM)模型,改进深度值标准差的计算方法,通过随机一致性采样(random sample consensus, RANSAC)算法来过滤帧与帧之间误匹配的特征点,提高计算出的相机运动精度,通过观测似然性的计算,评价相机运动的好坏、筛选位姿。在后端优化中采用李代数的表示方式,简化目标函数的约束,优化点云图的精确度。2)通过改进的ORB算法,对图像进行栅格化,通过二叉树结构,计算并细分每一层图像金字塔栅格块的特征点,保证提取到更多的特征点数量,同时使其更均匀地分布在图像的各个部分,避免回环检测时图像特征不够丰富,影响检测准确性。

## 1 系统算法原理

### 1.1 SLAM系统框架

本研究提出的SLAM系统如图1所示,主要包括以下几个模块:改进的基于二叉树结构的ORB特征提取与匹配模块、点云与深度信息存储模块及EM模型、后端优化与回环检测模块等3个主要模块。

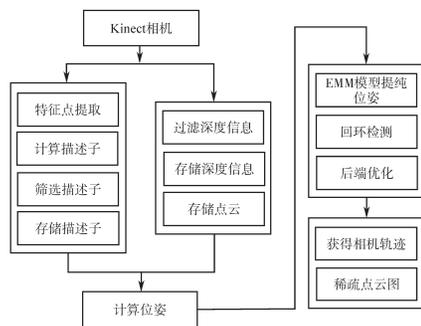


图1 SLAM系统架构

Fig.1 Structure of SLAM system

### 1.2 系统算法流程

1)特征匹配与提取。首先通过Kinect获得相机每一RGB彩色图像与深度图像,提取每一帧的特征点并计算其对应的描述子,在相邻帧中进行特征点匹配。再通过相机内参,得到像素坐标到3D空间的射影关系。

2)点云与深度图。通过相机矫正与标定,得到相机的畸变系数以及内参矩阵,对齐深度图和彩色图之后,存储下每个特征点的深度值,通过相机内参即可计算出像素点在空间中的3D坐标,生成彩色点云。

3)EM模型。由于传感器本身存在误差,为了减小精度,通过改进的EM模型来计算特征点在位姿坐标系转换时,对应特征点3D深度值的概率分布。以此来评价求得相机相对运动的好坏。

4)回环检测与后端优化。在室内环境中,通过g<sup>2</sup>o优化库,进行位后端姿矫正。

最后,使用位姿图的方式将相机关键帧的位姿作为图节点,而相对运动作为图的边,并使用李代数表达待优化的变量,以此来构建全局的非线性优化,最终矫正相机的位姿。

### 1.3 基于二叉树的ORB改进算法

ORB特征提取法由于其具有旋转与尺度不变性,在光照和模糊场景下有着很好的鲁棒性,因此在视觉SLAM系统中有着广泛的应用,二叉树改进的ORB算法的思路在于:在图像的每个部分上都提取到特征点,避免特征点过于集中,使得路标信息不够丰富,影响回环检测的精度。

首先对图像构建金字塔,并计算每一层图像的边缘。将每层金字塔的图像分成N个30×30的栅格窗口节点。在每层的每个窗口区域中使用FAST算法提取角点作为特征点之后,构造二叉树进行节点剔除,每次构造完节点后,判断节点数量是否大于所要求的特征点数量,若不满足,再次将每个节点划分成4个子节点区域,如图2所示。直到提取到足够的特征点为止,最后取每个节点的Harris响应值最大的特征点为最终特征点集合。

利用FAST方法提取完特征点之后,为了使特征点具有旋转不变性,通过计算图像点质心的方法,可以求出特征点的角度。

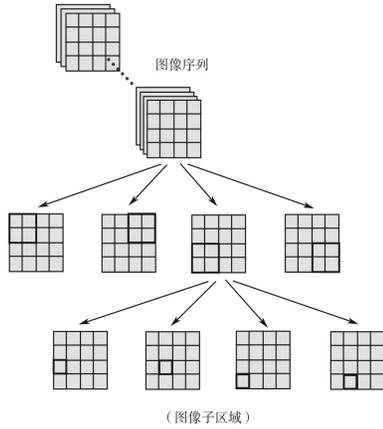


图2 使用四叉树结构分割图像

Fig.2 The segment images using quadtree

#### 1.4 相机深度值滤波

由于Kinect相机红外摄像头存在传感器固有噪声,使用文献[3]提出的误差模型对Kinect相机的深度数据进行滤波,减少像素深度值的误差,改善相机运动估计值的误差,从而提高相机位姿的估计精度。

#### 1.5 EM模型评估相机位姿

误差评价中改进了文献[4]中提出的环境测量模型来优化计算出的相机位姿,EM模型主要对传感器的测量误差概率分布进行建模,并将机器人前后运动时产生的观测值纳入到计算中以消除传感器误差,减小相机传感器噪声造成的误差,只在以激光雷达为主的2D-SLAM中有着广泛的应用。EM模型改进部分主要包含:1)通过计算特征匹配点的马氏距离初步优化相机运动的估计值;2)结合观测似然性模型评价相机运动估计值的好坏。

##### 1.5.1 相机运动的初步优化

定义相机的位姿与相机运动表达为

$$T_{ij} = \begin{bmatrix} \mathbf{R}_{ij} & \mathbf{t}_{ij} \\ \mathbf{0} & 1 \end{bmatrix} \quad (1)$$

式中: $\mathbf{R}$ ,  $\mathbf{t}$ 分别为相机的旋转矩阵与平移向量。

通过RnSAC算法随机选取取出两帧之间的匹配点,通过ICP算法计算出初始运动 $T_{n-1, n}$ ,从而得到特征点估计值的误差,由于Kinect第1代的有效深度值为1.2~3.5 m,在水平方向的视角为 $57.5^\circ$ ,垂直方向的视角为 $43.5^\circ$ ,分辨率为 $640 \times 480$ 。那么图像中的像素在空间坐标中 $x, y$ 轴上的标准差为

$$\sigma_x = \sigma_d \cdot \tan(\theta_x/m_x) \quad (2)$$

$$\sigma_y = \sigma_d \cdot \tan(\theta_y/m_y) \quad (3)$$

式中: $\theta_x, \theta_y$ 分别为相机在水平和垂直方向的视角大小; $m_x, m_y$ 分别为相机在水平和垂直方向的图像分辨率; $\sigma_d$ 为滤波之后的深度值的标准差,经过深度滤波之后,得到平滑之后的深度值。

使用文献[5]中提出的深度值的标准差计算公式计算 $\sigma_d$ :

$$\sigma_d = \left(\frac{m}{fb}\right)z^2 \quad (4)$$

式中: $m$ 为红外相机和RGB相机的视差; $b$ 为基线; $f$ 为相机的焦距; $z$ 为滤波之后得到的深度值。

文中将 $\left|\frac{m}{fb}\right|$ 作为误差系数取定值为 $2.85e-5$ ,由于经过深度滤波方法的处理,实际中误差系数取经验值 $1.5e-5$ 。最后,特征点在某一相机坐标系下的协方差矩阵可以写为

$$\Sigma_{xy} = \begin{bmatrix} \sigma_x^2 & & \\ & \sigma_y^2 & \\ & & \sigma_z^2 \end{bmatrix} \quad (5)$$

而同一特征点的协方差矩阵在不同相机位姿坐标系下的估计值计算为

$$\tilde{\Sigma}_{xy}^{(n)} = \mathbf{R}_{n-1, n}^T \Sigma_{xy}^{(n-1)} \mathbf{R}_{n-1, n} \quad (6)$$

式中: $\mathbf{R}_{n-1, n}$ 为相机位姿从第 $n-1$ 帧变换到第 $n$ 帧时的旋转矩阵; $\Sigma_{xy}^{(n-1)}$ 为特征点在第 $n-1$ 帧中的协方差矩阵; $\tilde{\Sigma}_{xy}^{(n)}$ 为同一特征点在第 $n$ 帧相机坐标系下的协方差矩阵估计值。

为了将两个相邻帧的噪声情况都考虑进来,协方差矩阵 $\Sigma_{xy}^{(n-1)}$ 与协方差矩阵估计值 $\tilde{\Sigma}_{xy}^{(n)}$ 之和为

$$\Sigma_c = \Sigma_{xy}^{(n-1)} + \tilde{\Sigma}_{xy}^{(n)} \quad (7)$$

得到特征点3D坐标估计值计算公式为

$$\mu_{xy}^{(j)} = T_{n-1, n} \mu_{xy}^{(i)} \quad (8)$$

式中: $\mu_{xy}^{(i)}$ 表示特征点在第 $n-1$ 帧中的3D坐标的测量值; $\mu_{xy}^{(j)}$ 为同一特征点经过坐标系转换在第 $n$ 帧中的3D坐标估计值。

通过式(8)我们即可将相邻帧匹配好的特征点转换到同一个坐标系下,并计算重投影误差:

$$\Delta \mu_{xy} = \tilde{\mu}_{xy}^{(j)} - \mu_{xy}^{(j)} \quad (9)$$

改进EM模型使用马氏距离来评价特征点的测量值和估计值误差:

$$D_m^2 = \Delta \mu_{xy}^T \Sigma_c^{-1} \Delta \mu_{xy} \quad (10)$$

若特征点的估计值和测量值的马氏距离平方小于某一阈值(取为9),那么这一特征点则被选为即正确投影点(inlier),并将筛选出的inlier

集合作为下一运动优化时所要使用的特征点。通过反复迭代计算出优化好的位姿运动  $T_{n-1, n}$ 。

1.5.2 图像的观测似然性计算

观测似然性主要是基于概率分布提出的一种计算方法,主要是观测相邻帧对应特征点的深度值信息,将其作为条件概率,增加了模型的信息层次,从而可以评价经过 1.5.1 节计算出来的结果。

通过相机的运动  $T_{ij}$ ,把像素点投影到不同相机坐标系下的结果分为 2 类:正确投影点(inliers)、错误投影点(outliers)。分别统计 2 种投影点的数量,从而确定相机运动过后的位姿  $T_{ij}$  是否为关键帧。先定义在第  $n$  帧观测到特征点  $p_j$  时,在第  $n-1$  帧可以观测到另一特征点  $p_i$  的条件概率的计算公式为

$$P(p_i|p_j) = \eta(p_i, p_j, \sigma_{ij}) \quad (11)$$

其中 
$$\sigma_{ij} = \sqrt{\sigma_i^2 + \sigma_j^2}$$

式中:  $\sigma_i^2, \sigma_j^2$  分别为点  $p_i$  和  $p_j$  深度值的方差。

使用高斯分布来计算似然性。

观测似然性的具体计算方法如图 3 所示,设  $n$  帧的某一像素点为  $p_j$ ,同时记其深度值为  $z_j$ ,根据 1.5.1 节所计算得出的相机运动  $T_{ij}$  与标定好的内参矩阵,将像素点  $p_j$  投影到第  $n-1$  帧中的相机坐标系下,得到像素点  $p'_i$ 。在第  $n-1$  帧中建立以  $p'_i$  为中心,3 个像素为半径的像素连通区域,遍历该区域中的像素,根据式(11),分别计算以  $z_j$  为均值、 $\sigma_{ij}$  为方差的高斯分布,得到该像素点的累积分布。设定阈值范围(取经验值 0.002 ~ 0.9)。

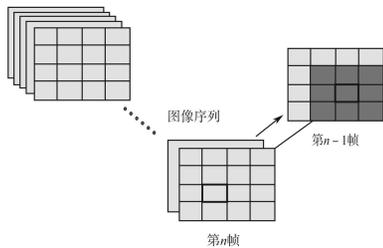


图3 观测似然性方法计算

Fig.3 The method that compute the observation likelihood

若在阈值范围内,则设定为正确投影点,若概率值小于此阈值,那么评判为遮蔽点,高于此则为错误投影点,并且分别统计 2 类点的数量。

最后建立基于 2 个评价指标的模型:运动指标与确定性指标。运动指标为 inliers 点数量与 inliers 和 outliers 点总数的比值,阈值取经验值

0.6;确定性指标为 inliers 点数量和所有特征点数量的比值,阈值取经验值 0.25。若相邻帧的像素点坐标在相机运动  $T_{ij}$  的坐标系变换下满足该评价模型,那么  $T_{ij}$  则为好的模型,设置该变换下产生的位姿节点为关键帧。

1.6 回环检测

回环检测的目的主要是用来判断机器人是否回到之前到达过的地方,消除因传感器自身引起的累计误差,提高位姿计算精度。

回环矫正过程中使用  $g^2o$ <sup>[6]</sup> 优化库提供的图优化框架,把相机的每一帧位姿作为“节点”,而位姿之间的运动误差函数(见 1.7 节)则作为连接“节点”的“边”,以此构造整个相机位姿图。当相机接收到新一帧时,将其构造为新的节点,并且以前一帧为中心开始搜索回环帧。

搜索方法为:

- 1) 临近节点搜索。在当前图中以第 1 个关键帧为起点,第  $n-1$  帧为终点,通过 Dijkstra 算法搜索其最短路径,保存经过的节点位姿为回环后选帧。
- 2) 前序节点搜索。将  $n-1$  帧之前的相邻帧直接保存,作为回环后选帧。
- 3) 随机节点搜索。在整个位姿图中,随机选取几帧作为回环候选帧。

最后将以上 3 种搜索方法获得的位姿节点作为回环候选帧,比较其和当前帧的相似度,在新关键帧与回环帧之间分别构造误差函数,作为节点之间的“边”,消除重投影误差,消除传感器自身导致的累积误差。

1.7 后端优化

后端优化属于全局范围内,对相机位姿的估计值进行优化,  $g^2o$  库是一个以 C++ 语言写成的数值优化库,主要用于 2D 或 3D 的位姿估计,其主要的方法在于将机器人的位姿与误差构建成一个图结构,在文中,相机作为图的“节点”,而连接节点之间的“边”则为相机运动的误差函数,并将整个“位姿图”构建为一个非线性最小二乘问题<sup>[7]</sup>:

$$F(x) = \sum_{k \in C} e_k(x_k, z_k)^T \Omega_k e_k(x_k, z_k) \quad (12)$$

式中:  $F(x)$  为目标误差函数;  $x_k$  为相机在  $k$  点的位姿;  $z_k$  为机器人测量到的特征点的值;  $\Omega_k$  为与  $x_k$  相关约束的信息矩阵。

通过计算误差函数  $F(x)$  的极小值来得到相机最

优位姿:

$$x^* = \arg \min F(x) \quad (13)$$

简记  $e_k(x_k, z_k) = e_k(x)$ 。由于目标函数非线性,为了求解极值,通常在  $x_k$  附近做泰勒展开,得:

$$e_k(x_k + \Delta x_k) = e_k + J_k \Delta x_k \quad (14)$$

式中:  $J_k$  为误差函数的雅各比矩阵。

同理对  $F(x)$  进行泰勒展开后得到:

$$\begin{aligned} F(x + \Delta x) &= e_k(x + \Delta x) \Omega_k e_k(x + \Delta x) \\ &= (e_k + J_k \Delta x)^\top \Omega_k (e_k + J_k \Delta x) \end{aligned} \quad (15)$$

简写后得到:

$$F(x + \Delta x) = c_k + 2b_k \Delta x + \Delta x^\top H_k \Delta x \quad (16)$$

其中  $H_k = J_k^\top \Omega_k J_k$   $b_k = 2e_k^\top \Omega_k J_k$   $c_k = e_k^\top \Omega_k e_k$

系统只优化相机的位姿,而忽略特征点的3D坐标优化,由于回环的存在,每个位姿顶点有多条边连接构成多个约束,不同节点的多个边都由各个位姿节点的运动变换矩阵  $\Delta T_{ij}$  构成,而每条边优化得到的  $\Delta T_{ij}$  存在误差,因此构建误差函数为

$$e_k = \Delta T_{ij} - T_i^{-1} \cdot T_j \quad (17)$$

式中:  $T_i, T_j$  为相机的位姿。

由于变换矩阵  $T$  中的旋转矩阵  $R$  必须为正交矩阵,由其作为优化变量会引入额外约束,不便于求导,考虑到位姿变换矩阵属于特殊欧氏群 (SE(3)),并且在实数空间是连续的,因此也属于李群 (Lie group),因此考虑使用李代数来表达位姿  $T$ ,消除旋转矩阵  $R$  的约束。

位姿矩阵  $T$  以李代数表达:

$$\xi^\wedge = \begin{bmatrix} \phi^\wedge & \rho \\ \mathbf{0}^\top & \mathbf{0} \end{bmatrix} \quad (18)$$

其中

$$\phi^\wedge = \Phi a^\wedge$$

式中,  $\xi$  为一个6维向量;  $\Phi$  为旋转角;  $\phi^\wedge$  的指数映射  $e^\phi$  为旋转矩阵  $R$  的李代数表达;  $\rho$  的指数映射  $e^\rho$  为平移向量的李代数表达。

根据文献[8],经推导给出旋转矩阵  $R$  的李代数表达:

$$R = e^{\phi^\wedge} = \cos \Phi + (1 - \cos \Phi) a a^\top + \sin \Phi a^\wedge \quad (19)$$

通过对式(18)取对数则可求得  $\phi^\wedge$ ,  $\phi^\wedge$  为旋转角  $\Phi$  和一个单位向量  $a^\wedge$  之积,符号 $\wedge$ 为从向量到矩阵的映射, $\vee$ 为从矩阵到向量的映射。为求得  $\rho$ ,变换矩阵的指数映射  $e^\xi$  的展开为:

$$\begin{aligned} T = e^\xi &= \sum_{n=0}^{\infty} \frac{1}{n!} (\xi^\wedge)^n \\ &= \begin{bmatrix} \sum_{n=0}^{\infty} \frac{1}{n!} (\phi^\wedge)^n & [\sum_{n=0}^{\infty} \frac{1}{(n+1)!} (\phi^\wedge)^n] \rho \\ \mathbf{0}^\top & \mathbf{0} \end{bmatrix} \end{aligned} \quad (20)$$

由式(20)可得平移向量  $t$  的李代数表达:

$$t = J\rho = [\sum_{n=0}^{\infty} \frac{1}{(n+1)!} (\phi^\wedge)^n] \rho \quad (21)$$

$$J = \frac{\sin \Phi}{\Phi} + (1 - \frac{\sin \Phi}{\Phi}) a a^\top + \frac{1 - \cos \Phi}{\Phi} a^\wedge$$

通过  $J$  和  $t$  可以求得  $\rho$ 。有了变换矩阵  $T$  对应的李代数,则很容易对目标函数  $e_k$  进行求导,并进行泰勒展开,把误差函数  $e_k$  中的位姿矩阵  $T$  改写为李代数得:

$$\begin{aligned} e_k &= \ln(\Delta T_{ij}^{-1} T_i^{-1} T_j)^\vee \\ &= \ln(e^{-\xi_j^\wedge} \cdot e^{-\xi_i^\wedge} \cdot e^{\xi_j^\wedge}) \end{aligned} \quad (22)$$

按李代数求导方式对  $e^{-\xi_i^\wedge}$  和  $e^{-\xi_j^\wedge}$  各左乘一个扰动,由于 BCH 公式 (baker-campbell-hausdorff formula),且李代数乘法的伴随性质,可以把  $\xi_i$  侧的扰动乘在右边得:

$$\dot{e}_k = \ln(e^{-\xi_j^\wedge} \cdot e^{-\xi_i^\wedge} \cdot e^{\xi_j^\wedge}) + \ln(e^{(-Ad(e^{-\xi_j^\wedge}) \cdot e^{-\xi_i^\wedge})^\vee} \cdot e^{(-Ad(e^{-\xi_j^\wedge}) \cdot e^{-\xi_i^\wedge})^\vee}) \quad (23)$$

其中

$$Ad(e^{-\xi_j^\wedge}) = \begin{bmatrix} e^{-\phi_j^\wedge} & (t_j)^\wedge e^{-\phi_j^\wedge} \\ \mathbf{0} & e^{-\phi_j^\wedge} \end{bmatrix}$$

对式(23)进行求导得:

$$\dot{e}_k \approx e_k + \frac{\partial e_k}{\partial \delta \xi_i} \delta \xi_i + \frac{\partial e_k}{\partial \delta \xi_j} \delta \xi_j \quad (24)$$

式中关于2个位姿  $T_i$  和  $T_j$  扰动的雅各比矩阵为

$$\begin{aligned} \frac{\partial e_k}{\partial \delta \xi_i} &= -\mathfrak{J}_r^{-1}(e_k) Ad(e^{-\xi_j^\wedge}) \\ \frac{\partial e_k}{\partial \delta \xi_j} &= -\mathfrak{J}_r^{-1}(e_k) Ad(e^{-\xi_i^\wedge}) \end{aligned} \quad (25)$$

其中

$$-\mathfrak{J}_r^{-1}(e_k) \approx I + \frac{1}{2} \begin{bmatrix} \phi_{e_k}^\wedge & \rho_{e_k}^\wedge \\ \mathbf{0} & \phi_{e_k}^\wedge \end{bmatrix}$$

式(16)中前2项的雅各比矩阵  $J_k$  可近似约等于为位姿  $T_i$  的雅各比矩阵。式(16)第3项的  $H$  矩阵为一个高维海塞矩阵,为减少计算量,在优化中忽略此矩阵。最后使用 L-M (levenberg-marquart) 或 G-N (gauss-newton) 法迭代来得到目标误差函数的极小值,从而求得相机的局部最优位姿。

## 2 实验分析

### 2.1 改进 ORB 算法的特征点提取

为验证改进 ORB 算法提取特征点的有效性,进行如下测试。

对2幅相同图像同时用原始算法与改进提取特征点,并统计2种方法计算出的特征点在图像中像素坐标分布,如图4所示,“x”为改进算法的特征点,“●”为改进前的特征点,由于使用四叉树结构对图像金字塔各层进行角点提取,改进后算法的特征点分布得更广,位置集中在横坐标为[0,640]像素区域,而改进前的图像则集中在横坐标为[0,400]的像素区域,缓解一些由于相似特征点集中于某一区域而产生的误匹配,提高位姿估计的精度。

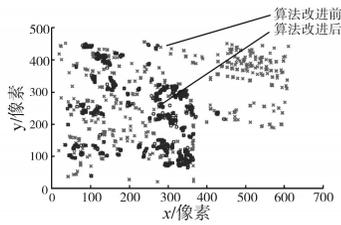


图4 特征点分布

Fig.4 Distribution of the feature points

在回环检测中,同样需要利用图片特征点描述子来对图片相似性进行比对,确定是否为之前的场景,若特征点分布的位置越广,那么可以说图像包含的特征信息就越丰富。这样改进的ORB算法为之后的回环检测的鲁棒性提供保证。

在给定800张图像的情况, Linux平台下由于改进算法对图像金字塔细分的每个区域都进行计算,计算花费的时间在0.02 s/帧左右,比起原始算法0.015 s/帧的速度,运算速度略有下降。

## 2.2 公开数据集测试

### 2.2.1 相机轨迹估计

前文从理论上描述了EM模型、回环检测、后端优化等技术,为了测试算法实际的效果,实验使用慕尼黑工业大学的公开数据集进行测试<sup>[9]</sup>,数据集包含着不同场景下的相机运动,通过8个高速摄像机捕捉Kinect相机的真实运动轨迹,适合本实验进行算法验证。

表1为原始算法与改进算法的误差与运行时间的对比,原始算法使用改进前的ORB算法,并且不使用基于深度值概率分布的EM模型来评价和矫正位姿,由于加入了像素点的似然观测性计算,可以减少计算误差,其计算要多次迭代,所以时间复杂度要高于原始算法,而算法的大部分时间都花费在特征点的提取与描述子计算上,计算时间基本上能保持在31~35帧/s左右,满足实时性要求。

表1 算法误差(RMSE)耗时评估

Tab.1 The evaluation of RMSE and time consuming

数据包	未采用EM模型算法		本文算法	
	时间/s	误差/m	时间/s	误差/m
FR1/xyz	0.022 7	0.080 6	0.016 7	0.010 1
FR2/xyz	0.022 1	0.045 2	0.025 3	0.004 1
FR1/desk	0.028 8	0.068 3	0.029 2	0.015 1
FR1/rpy	0.026 1	0.083 3	0.020 4	0.022 1
FR1/room	0.028 1	0.141 3	0.029 6	0.021 3
FR1/plant	0.029 1	0.135 1	0.027 1	0.018 1
FR2/desk	0.029 3	0.071 2	0.021 0	0.008 7

表2选取了部分数据包与采用直接法的LSD-SLAM<sup>[10]</sup>进行误差比较,结果均远优于LSD-SLAM。

表2 与LSD-SLAM误差对比

Tab.2 Comparison between our algorithm and LSD-SLAM

数据包	LSD-SLAM	本文算法
	误差/m	误差/m
FR1/xyz	0.090 0	0.010 1
FR2/desk	0.045 7	0.008 7

图5为未采用EM模型的算法估计出的相机轨迹和真实轨迹的对比,图6为采用了改进ORB算法和EM模型的对比图,所用的数据集为FR1/plant,由于采用了EM模型与回环检测,算法的误差从原始的相邻帧数算法的0.135 1 m下降到0.018 1 m,下降了大约86.6%,精度得到极大的提高。

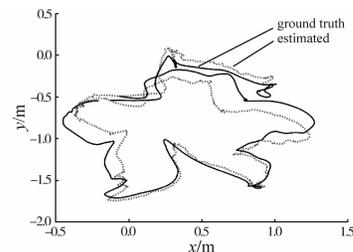


图5 未采用EMM时与真实轨迹值对比

Fig.5 Comparison with the truth trajectory when the algorithm without EMM adopted

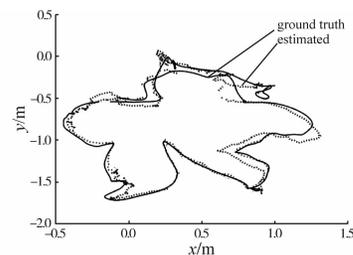


图6 采用EMM时与真实轨迹值对比

Fig.6 Comparison with the truth trajectory when the algorithm with EMM adopted

### 2.2.2 数据集建图效果

得到相机位姿的估计值之后,为便于存储,建立稀疏点云图,图7为数据集FR1/desk以桌子为中心点,实验室3D环境重建的结果。



图7 数据集3维点云

Fig.7 3D-pointcloud of data set

### 2.3 实际环境测试

在实际测试中,使用实验室自主研发的移动机器人,如图8所示,机器人搭载了室内Kinect相机模块,电池、WIFI等模块,内部搭载的工控机运行ROS分布式机器人控制系统。机器人可以由键盘远程控制运动。



图8 实验室移动机器人

Fig.8 Mobile robot in lab

实验中通过WIFI模块来进行工控机到远程PC端的数据交换,计算完机器人的运动轨迹之后,再进行离线建图。

实验室环境如图9所示,控制移动机器人低速环绕图中电脑桌沿长方形轨迹移动,图10为系统估计得出的机器人运动的最终轨迹。



图9 实验室环境

Fig.9 Environment of lab

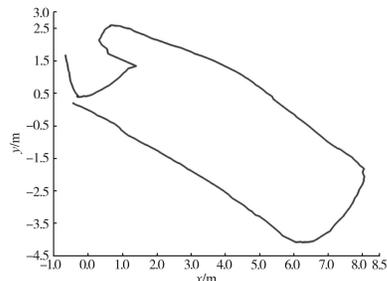
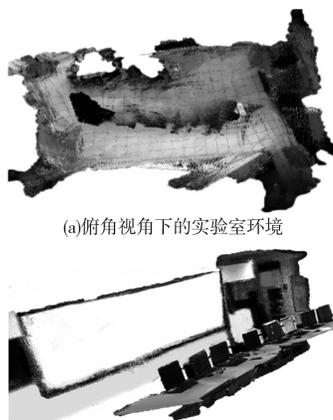


图10 机器人运动轨迹

Fig.10 Trajectory of robot

之后建立三维点云图。效果如图11a所示,图中显示了俯角视角下的实验室环境,基本符合实验室构造,由于存在玻璃、白墙等纹理较少的环境,使得点云图的精度受到了一定的影响,由于拍摄到天花板,对俯角下的图11b中的一排计算机产生了一定遮挡。



(a)俯角视角下的实验室环境

(b)俯角视角下的一排计算机

图11 实验室3维点云

Fig.11 3D-pointCloud of lab

## 3 结论

提出的SLAM系统在实验中对对比了ORB算法和改进ORB算法,并采用公开数据集验证算法的有效性,同时将系统结合实验室机器人,应用于实际环境,验证了系统算法的可行性。本算法的最高精度达到0.009 5 m,平均误差也在0.03 m内,去除3D点云地图创建的时间,每帧在30 ms左右,可以满足室内实时定位的要求。

提出的改进EM模型的局限性在于传感器噪声模型单一,算法的可移植性较差,需要对噪声重新建模,特征点提取的算法对纹理较少的物体鲁棒性比较差。

