

# 移动机器人视觉 SLAM 过程中 图像匹配及相机位姿求解的研究

林志诚<sup>1</sup>, 郑松<sup>2</sup>

(1. 福州大学电气工程与自动化学院, 福建 福州 350116)

(2. 福建省工业控制信息安全技术企业重点实验室, 福建 福州 350008)

**摘要:** 针对移动机器人视觉 SLAM 过程中的关键环节图像匹配及相机位姿求解问题, 通过 ORB 特征匹配算法匹配移动机器人采集的两帧图像, 得到特征点集。分析了最小二乘法理论特征, 根据两帧图像中特征点不变性, 采用基于最小二乘法的 ICP 算法建立相机运动模型, 使用 SVD 分解法求解运动模型, 估计相机位姿状态, 并通过非线性优化算法验证 SVD 分解法。利用 RGB-D 相机 Kinect 采集图像, 对上述算法进行实验, 得到准确的相机姿态数据, 从而有效解决了三维空间下移动机器人局部移动过程中的定位问题。

**关键词:** 视觉同步定位与地图构建; ORB 特征; 匹配算法; 迭代最近点算法

**中图分类号:** TP242.6

**文献标识码:** A

**文章编号:** 2095-509X(2017)11-0013-06

移动机器人是集环境感知、动态决策与规划、行为控制与执行等功能于一体的综合系统, 是集传感器技术、信息处理、电子工程、计算机工程、自动化控制工程以及人工智能等多种学科的综合研究成果。移动机器人研究方向十分广泛, 其中未知环境中自主导航移动机器人是最具挑战性的研究方向之一。移动机器人自主导航由 3 个部分组成: 定位、建图以及导航<sup>[1]</sup>。SLAM(simultaneous localization and mapping, 同步定位与地图构建)是指利用激光仪、里程计等传感器, 于运动过程中建立周围环境地图模型, 同时估计自己的位姿<sup>[2]</sup>。移动机器人视觉 SLAM 采用相机为传感器, 相机种类主要分为单目相机(monocular)<sup>[3]</sup>、双目相机(stereo)<sup>[4]</sup>和深度相机(RGB-D)<sup>[5]</sup>, 其中 RGB-D 相机不仅能够采集彩色图像, 而且还能获得图像深度信息, 因此采用 RGB-D 相机作为解决 SLAM 问题的外界信息的传感器正逐渐成为一种趋势。本文利用 RGB-D 相机, 得到三维空间点在相机成像平面上的投影值, 即由数字排列而成的复杂矩阵。利用相机在机器人运动过程中采集两帧图像, 检测并匹配图像信息中特征点集, 根据图像中特征点集保持不变的特性, 采用算法求解相机姿态变化, 从而构建局部地图并解决机器人定位问题, 提升移动机器人

自主导航能力, 提高移动机器人的智能化程度以及未知环境下信息处理能力。

## 1 ORB 特征检测匹配算法

为了简化图像处理算法, 选取相机运动时图像中保持不变的点, 只有在确定这些点的基础上, 才能估计相机位姿。在视觉 SLAM 过程中, 这些点称为图像特征点(feature-point)。

特征点由关键点(key-point)和描述子(descriptor)组成。关键点是指特征点在图像中的位置; 描述子是按照特定的方式, 比如向量, 描述该关键点周围像素信息。在当前的视觉 SLAM 领域中, 常用的特征提取算法有 SIFT<sup>[6]</sup>、SURF<sup>[7]</sup>、ORB<sup>[8]</sup>等。SIFT 算法对旋转、尺度缩放、亮度变化保持不变, 对视角变化、仿射变换、噪声也保持一定程度的稳定性。SURF 算法是 SIFT 算法的改进, 其最大特点是采用 Hessian 矩阵提高计算效率, 且计算速度是 SIFT 算法的 3 倍左右。SIFT 和 SURF 算法都具有计算量过大的缺点, 故在处理图像过程中存在时滞效果, 且对硬件有高性能的需求。目前为止, 普通计算机的 CPU 运算速度均无法满足实时计算的要求。ORB 算法采用了 FAST 特征检测算子, 并采用二进制格式的 BRIEF 特征描述算子, 极大地提

收稿日期: 2017-08-18

作者简介: 林志诚(1994—), 男, 福建莆田人, 福州大学硕士研究生, 主要研究方向为移动机器人 SLAM、视觉处理、多移动机器人协同控制。

升了计算速度。本文将在经典的 BRIEF 特征描述算子的基础上进一步改进,并通过实验改进的算法,有效提高了运算效率。

### 1.1 FAST 特征检测算子

FAST 特征检测算子是在以某点为中心的领域,检测并比较该点与领域边界上像素点的灰度值大小,如果灰度值明显不同,则该点为特征点。如果该领域边界上的像素点个数为 9,则称为 FAST-9。具体检测过程如下:

- 1) 在图像中选取像素点  $q$ , 假设灰度值为  $B_q$ 。
- 2) 设置阈值  $T$ 。
- 3) 以  $q$  点为中心, 选取半径为 2 的圆上的 9 个像素点。
- 4) 如果以点  $q$  为中心的圆边界上连续 9 个点的灰度值都大于  $(B_q + T)$  或者小于  $(B_q - T)$ , 那么可以认为其是特征点。
- 5) 循环以上 4 个步骤, 对每个像素点依次操作。

在特征点位置处, 计算特征点的 Harris 响应值, 然后选取其中  $N$  个响应值大的特征点作为最终的特征点集合。ORB 通过灰度质心法对 FAST 特征点添加方向的描述, 具体操作步骤如下:

- 1) 选取以特征点  $O$  为几何中心的图像块, 定义该图像块的矩为:

$$m_{pq} = \sum a^p b^q I(a, b) \quad p, q = \{0, 1\} \quad (1)$$

式中  $I(a, b)$  为特征点  $(a, b)$  处的灰度值。

- 2) 通过矩可以找到图像块的质心

$$C \left( \frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right)$$

- 3) 连接图像块的几何中心  $O$  和质心  $C$ , 得到一个向量, 这个向量与水平的偏转角度可以用于描述特征点的方向, 于是该方向为:

$$\theta = \arctan(m_{01}/m_{10}) \quad (2)$$

至此, FAST 特征点便有了旋转方向的描述。

### 1.2 BRIEF 特征描述算子

得到 FAST 特征点后, 需要以某种方式描述这些特征点的属性。这些属性称为该特征点的描述子。ORB 算法采用二进制格式的 BRIEF 特征描述算子, BRIEF 特征描述算子是由许多个 0 和 1 组成的向量, 0 和 1 的编码由像素点  $L$  与特征点  $O$  的灰度值大小关系决定:

$$\tau(L, O) = \begin{cases} 1 & k(L) < k(O) \\ 0 & k(L) \geq k(O) \end{cases} \quad (3)$$

式中:  $k(L)$ 、 $k(O)$  是像素点  $L$  与特征点  $O$  处的灰度值。因此特征点属性可以由  $n$  维 0, 1 组成的向量表示。BRIEF 特征描述算子的具体求解步骤如下:

• 14 •

- 1) 为了抗噪声扰动, 先对图像进行高斯滤波 (方差为 2, 高斯窗口包含  $9 \times 9$  个像素点)。

- 2) 以 FAST 特征点为中心, 取  $31 \times 31$  个像素的邻域大窗口, 然后随机选取两个小窗口, 对小窗口内的像素灰度值的和进行比较, 并通过式 (3) 进行二进制赋值。

- 3) 在大窗口中随机选取 128 对子窗口, 重复步骤 2) 的二进制赋值, 形成一个二进制编码, 这个编码就是对特征点的描述, 即特征描述算子。

### 1.3 特征提取与匹配实验

特征匹配是视觉 SLAM 过程中极为关键的一步, 它涉及到视觉 SLAM 过程中数据关联的问题, 通过提取图像的 FAST 特征检测算子及 BRIEF 特征描述算子, 对描述算子进行精确匹配, 为后续的相机的姿态估计、优化等操作提供重要的基础。采用最简单的暴力匹配 (brute-force matcher), 更能凸显出 ORB 特征检测算子的性能。图 1 为移动机器人通过 RGB-D 相机采集到的原始图像, 图 2 为移动机器人平移变换 5cm 后 RGB-D 相机采集到的变换图像, 图 3 为采用 FAST 特征检测算子和 BRIEF 特征描述算子对图 2 进行特征提取的图像, 图 4 为匹配测试结果。



图 1 原始图像



图 2 平移变换后图像



图 3 特征提取图像

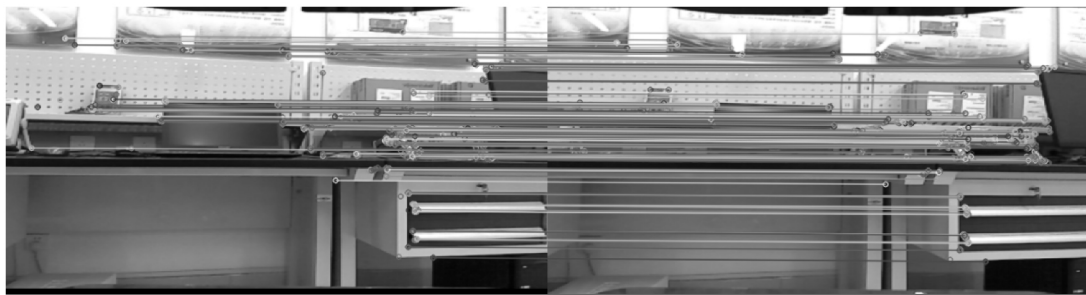


图 4 特征匹配图像

## 2 非线性优化

### 2.1 状态估计问题

在机器人视觉 SLAM 过程中,对机器人进行状态估计,即已知输入数据  $u$  (惯性传感器数据) 和观测数据  $z$  (机器人相机图像特征) 的条件下,计算状态  $x$  的条件概率分布:  $P(x|z, u)$ 。

为了简化计算,通常忽略惯性传感器数据,而将观测数据看作图像中一系列数据点集,从而将问题转换为如何从许多图像中重建三维空间结构。根据贝叶斯法则,机器人状态变量的后验概率为:

$$P(x|z) = \frac{P(z|x)P(x)}{P(z)} \quad (4)$$

在先验概率  $P(x)$  已知的条件下,求解最大后验概率  $P(x|z)$  相当于求解  $x$  最大似然  $P(z|x)$ 。机器人观测方程有:

$$z_{kj} = h(y_j, x_k) + v_{kj} \quad (5)$$

式中:  $h(y_j, x_k)$  为相机模型;  $y_j$  为图像中特征点;  $x_k$  为相机位姿状态变量;  $v_{kj}$  为噪声项,且满足高斯分布  $v_{kj} \sim N(0, Q_{kj})$ 。所以观测数据的条件概率为:

$$P(z_{kj}|x_k, y_j) = N(h(y_j, x_k), Q_{kj}) \quad (6)$$

式(6)是一个高斯分布,为了求该式最大似然估计,就必须求使它最大化的  $x_k, y_j$ 。对于多维高斯分布  $x \sim N(\mu, \delta)$ ,通常采用以下形式来求解高斯分布的最大似然估计:

$$-\ln P(x) = \frac{1}{2} \ln [(2\pi)^N \det(\delta)] + \frac{1}{2} (x - \mu)^T \delta^{-1} (x - \mu) \quad (7)$$

对原分布求最大化相当于对负对数求最小化。在式(7)中等号右边第一项与  $x$  无关,可以略去。只要求等号右边第二项二次型的最小化,就得到了对状态的最大似然估计。代入视觉 SLAM 的观测模型,相当于求解:

$$x = \min [ (z_{kj} - h(x_k, \mu_j))^T Q_{kj}^{-1} (z_{kj} - h(x_k, \mu_j)) ] \quad (8)$$

定义数据与估计值之间存在以下误差:

$$e_{yjk} = z_{kj} - h(x_k, y_j) \quad (9)$$

根据式(9)求该误差的平方和:

$$J(x) = \sum_k \sum_j e_{yjk}^T Q_{kj}^{-1} e_{yjk} \quad (10)$$

综上所述,得到最小二乘问题(least square

problem) 的模型。它的最优解等价于状态  $x$  最大似然估计。根据运动方程和观测方程估计整体误差值,求解这个误差的最小值是一个典型最小二乘法优化的过程。

### 2.2 非线性最小二乘法

在实际机器人系统中,环境变量不满足高斯分布,误差方程为非线性方程,非线性最小二乘法在机器人实际控制模型解析中起到至关重要的作用。非线性最小二乘法是以误差的平方和最小为准则来估计非线性模型参数的一种参数估计方法。对于三维空间中的一点  $s$ ,为了得到目标点  $h$ ,对  $s$  点进行旋转位移变换得到  $R_s$ 。根据式(9)构建  $R_s$  与  $h$  的误差公式,从而建立非线性最小二乘法模型,其中  $R$  是三维矩阵,用来描述刚体旋转。利用三维向量即李代数来表示刚体旋转变化,李代数与旋转矩阵通过矩阵指数函数建立映射关系。

对于利用最小二乘法模型求误差最小值,传统方法为令目标函数导数为零,然后求解最小值。李代数导数形式如下:

$$\frac{\partial(h - R_s)}{\partial R} = \frac{\partial(h - \exp(\hat{\varphi})s)}{\partial \varphi} = \lim_{\varphi \rightarrow 0} \frac{-(h - \exp(\hat{\varphi})s)^T J_l \partial \varphi}{\partial \varphi} \quad (11)$$

式中:  $J_l$  为雅可比矩阵;  $\hat{\varphi}$  为三维旋转变换向量组成的对称矩阵。该计算过程过于繁琐,且只适用于线性方程。针对非线性最小二乘法采用迭代的方式求解,具体步骤如下:

- 1) 给定某个初始值  $x_0$ 。
- 2) 开始进行迭代,对于迭代过程,需找一个下降矢量  $\Delta x$ ,使得  $\|f(x_k + \Delta x)\|_2^2$  达到极小值。
- 3) 若  $\Delta x$  足够小,则停止迭代。否则返回重复步骤 2)。

为了寻找步骤 2) 的极小值,必须寻找一个梯度,将  $f(x)$  进行一阶泰勒展开得到:

$$f(x + \Delta x) \approx f(x) + f'(x_k) \Delta x \quad (12)$$

式中:  $f'(x_k)$  实际上是一个矩阵的导数,也就是一个雅可比矩阵。为了寻找下降矢量  $\Delta x$ ,使得  $\|f(x + \Delta x)\|_2^2$  达到最小值,则相当于构建一个线性方程:

$$\mathbf{x}^* = \arg \min_{\Delta \mathbf{x}} \frac{1}{2} \|f(\mathbf{x}) + f'(\mathbf{x}_k) \Delta \mathbf{x}\|^2 \quad (13)$$

分解式(13)得到:

$$\frac{1}{2} \|f(\mathbf{x}) + f'(\mathbf{x}_k) \Delta \mathbf{x}\|^2 = \frac{1}{2} (\|f(\mathbf{x})\|_2^2 + 2f(\mathbf{x})^T f'(\mathbf{x}) \Delta \mathbf{x} + \Delta \mathbf{x}^T f'(\mathbf{x})^T f'(\mathbf{x}) \Delta \mathbf{x}) \quad (14)$$

求关于  $\Delta \mathbf{x}$  的导数,并令其为 0,经过简化可以得到:

$$f'(\mathbf{x})^T f'(\mathbf{x}) \Delta \mathbf{x} = -f'(\mathbf{x})^T f(\mathbf{x}) \quad (15)$$

由于求解的变量为  $\Delta \mathbf{x}$ ,因此式(15)为线性方程。上述方法称为高斯牛顿法,具体步骤如下:

1) 给定初始值  $\mathbf{x}_0$ 。

2) 对于  $k$  次迭代,求出当前雅可比矩阵  $f'(\mathbf{x})$  和矩阵函数  $f(\mathbf{x})$ 。

3) 求解公式(15)。

4) 若  $\Delta \mathbf{x}$  足够小,则停止迭代。否则,令  $\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta \mathbf{x}$ ,返回步骤 2)。

该算法能令  $f'(\mathbf{x})^T f'(\mathbf{x})$  近似为二阶 Hessian 矩阵,从而忽略计算  $f'(\mathbf{x})^T f'(\mathbf{x})$ 。高斯牛顿法通过求解非线性方程的误差平方和达到最小从而求解机器人状态方程,虽然收敛速度较慢,但是在视觉 SLAM 过程中仍有大量的应用。

### 3 相机位姿求解

基于上述数学问题,在视觉 SLAM 过程中采用 RGB-D 相机采集图像,同时获得图像深度距离,通过特征点进行相机姿态的估计。

假设通过 ORB 算法采集到一组匹配好的特征点集  $p$  和  $q$ :  $p = \{p_1, \dots, p_n\}$ ,  $q = \{q_1, \dots, q_n\}$ 。

点集  $p$  通过旋转、平移变化,得到点集  $q$ :

$$\forall i, q_i = R p_i + t \quad (16)$$

式中:  $R$  为旋转矩阵;  $t$  为平移向量。

关于相机位姿的求解,可通过图像特征点匹配计算出图像移动过程,从而映射出相机移动过程。该方法需要建立两个点集之间的对应关系,利用迭代最近点(iterative closest point, ICP)寻找对应关系,求解移动机器人 3D 空间的位姿。对于中等大小的点集来说,该方法具备计算速度快、鲁棒性强、误差小等特性,因此被广泛应用于机器人运动过程中相机的姿态计算。

#### 3.1 ICP 算法

ICP 算法即 3D 点集匹配算法,是基于最小二乘法的最优匹配算法,即通过重复计算旋转矩阵以及平移向量使得初始点集与最终点集误差满足最小度量。设初始点集为特征点集  $p$ ,最终点集为特征点集  $q$ ,通过寻找初始点集到最终点集的最小距离,使得初始集每个点到最终点集都会有映射,其具体步骤如下:

1) 寻找初始点集  $p$  与最终点集  $q$ 。

2) 计算初始点集对应点  $p_k$  和最终点集对应点  $q_k$ 。

3) 计算旋转变换矩阵  $R_k$ , 平移变换向量  $t_k$ , 使得公式  $\sum_{k=1}^n \|q_k - (R_k p_k + t_k)\|_2^2$  达到最小值。

4) 计算公式  $p_{k+1}' = R_k p_k' + t$ , 其中  $\forall p_k' \in p$  给定最小误差值,直到最小均方差  $d(\frac{1}{n} \sum_{k=1}^n \|q_k' - p_k'\|_2^2)$  小于或者到达预期迭代次数  $\tau$  为止,否则重复步骤 2)。

#### 3.2 SVD 方法

ICP 算法根据给定的初始点集与最终点集误差建立最小二乘法模型,从而进行相机旋转平移位姿求解,初始点集与最终点集中第  $i$  个点的误差项:

$$e_i = p_i - (R q_i + t) \quad (17)$$

式中:  $p_i$  为点集  $p$  的去质心坐标点;  $q_i$  为点集  $q$  的去质心坐标点。

求点集中所有点误差和达到最小的旋转变换矩阵  $R$  和平移变换向量  $t$ 。利用 ICP 算法构建的最小二乘法模型为

$$\min_{R, t} J = \frac{1}{2} \sum_{i=1}^n \|p_i - (R q_i + t)\|_2^2 \quad (18)$$

忽略平移变换向量  $t$ , 将式(18)展开得:

$$\min \frac{1}{2} \sum_{i=1}^n \|p_i - R q_i\|_2^2 = \frac{1}{2} \sum_{i=1}^n (p_i p_i + q_i^T R^T R q_i - 2 p_i^T R q_i) \quad (19)$$

式中: 等号右边括号里第一项与  $R$  无关, 等号右边括号里第二项由于  $R^T R = I$ , 亦与  $R$  无关, 因此目标函数变为:

$$\sum_{i=1}^n p_i^T R q_i = -\text{tr}(R \sum_{i=1}^n q_i p_i^T) \quad (20)$$

采用奇异值分解方法(SVD)将旋转矩阵分解为 3 个特殊矩阵的乘积,先定义矩阵:

$$W = \sum_{i=1}^n p_i q_i^T \quad (21)$$

对  $W$  进行 SVD 分解得:

$$W = U \Sigma V^T \quad (22)$$

式中:  $\Sigma$  为奇异值组成的对角矩阵;  $U$  和  $V$  为正交矩阵。当  $W$  满秩时,有:

$$R = UV^T \quad (23)$$

解出  $R$ , 可求得平移变换向量  $t$ :

$$t = p' - R q' \quad (24)$$

通过以上的旋转矩阵以及平移向量,对观察模型进行持续性的变换,使得求解出来的变换后的点集趋近于真实点集。在使用 SVD 方法求解过程中,并没有引入相机的模型,因此显著地减少了计算量,为求解机器人移动过程旋转矩阵以及平移向量带来了极大的方便。

#### 3.3 非线性优化方法求解

在视觉 SLAM 过程中通过非线性优化方法求

解误差最小二乘法问题,从而估计相机状态。该方法精确度高、可靠性强,但是存在计算复杂的缺点,面对复杂非线性函数难以通用化,故通常利用非线性优化方法检验并优化 SVD 方法求解结果。假设 3D 特征点集  $q$  中点  $q_i$ ,其投影至图像的像素点坐标为  $u_i$ ,  $\exp(\xi^A) p_i$  表示特征点集  $p$  中点  $p_i$  旋转平移变换后的像素点位置,与空间点位置的关系如下:

$$u_i = \exp(\xi^A) p_i \quad (25)$$

利用最小二乘法建立模型:

$$e(\xi) = \min_{\xi} \frac{1}{2} \sum_{i=1}^n \|u_i - \exp(\xi^A) q_i\|_2^2 \quad (26)$$

李代数  $\xi^A$  的表达式如下:

$$\xi^A = \begin{bmatrix} \mu & \varphi \\ 0 & 0 \end{bmatrix} \quad (27)$$

式中:  $\mu$  为旋转矩阵;  $\varphi$  为平移矩阵。将  $e(x)$  函数线性化为:

$$e(\xi + \Delta\xi) \approx e(\xi) + J\Delta\xi \quad (28)$$

便可以使用高斯牛顿法进行求解。式中:  $\Delta\xi$  为李代数增量;  $J$  为雅可比矩阵,

$$J = \begin{bmatrix} \frac{f_x}{z'} & 0 & -\frac{f_x x'}{z'^2} & -\frac{f_x y'}{z'^2} & f_x + \frac{f_x x'^2}{z'^2} & \frac{f_x y'}{z'} \\ 0 & \frac{f_y}{z'} & -\frac{f_y x'}{z'^2} & -\frac{f_y y'}{z'^2} & \frac{f_y x'}{z'^2} & \frac{f_y y'}{z'} \end{bmatrix}^T$$

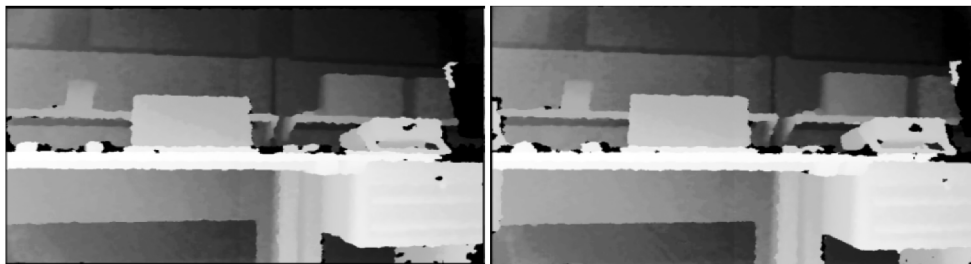


图6 深度图

最终得出矩阵数组为:

$$W = \begin{bmatrix} 12\ 454.90 & 997.29 & 7\ 982.10 \\ 967.10 & 1\ 498.90 & -4\ 141.20 \\ 8\ 490.50 & -4\ 094.90 & 38\ 316.60 \end{bmatrix}$$

$$V = \begin{bmatrix} 0.277\ 312 & 0.943\ 324 & -0.182\ 312 \\ -0.092\ 535 & 0.215\ 095 & 0.972\ 199 \\ 0.956\ 313 & -0.252\ 733 & 0.146\ 939 \end{bmatrix}$$

$$U = \begin{bmatrix} 0.267\ 770 & 0.945\ 497 & -0.185\ 298 \\ -0.093\ 294 & 0.216\ 862 & 0.971\ 734 \\ 0.958\ 955 & -0.242\ 914 & 0.146\ 278 \end{bmatrix}$$

$$R = \begin{bmatrix} 0.999\ 947\ 651 & -0.001\ 553\ 141 & -0.010\ 113\ 548\ 3 \\ 0.001\ 540\ 665 & 0.999\ 998\ 043 & -0.001\ 241\ 249\ 38 \\ 0.010\ 115\ 456 & 0.001\ 225\ 602\ 7 & 0.999\ 948\ 086\ 37 \end{bmatrix}$$

$$t = [-0.561\ 249\ 643\ 6 \quad -0.011\ 017\ 809\ 8 \quad -1.088\ 716\ 244\ 5]$$

其中  $(x' y' z')$  为初始点集  $p$  中点  $p_i$  坐标  $(x, y, z)$  变换到相机坐标系下的坐标;  $f_x, f_y, f_z$  为相机内部参数。通过高斯牛顿法进行迭代,解出李代数  $\xi^A$ ,从而解析出旋转矩阵与平移矩阵。

### 3.4 实验验证

ICP 算法主要利用三维点集数据估计机器人移动姿态,因此采用常用的 RGB-D 相机 Kinect 1 代进行图像采集,Kinect 1 代最大的特点是可以发送一束红外光,遇到物体时红外光反射,Kinect 1 代接收反射光,计算从发射到返回之间光线的飞行时间,确定物体与相机之间的距离,因此通过 Kinect 1 代可以获得整个图像的像素深度。Kinect 1 代除了普通的摄像头以外,还会有 1 个发射器和 1 个接收器,结构如图 5 所示,通过 1 中匹配好的图像来验证 SVD 方法求解最小二乘法模型的过程,其中 Kinect 1 代采集到的图 1 及图 2 的深度图像如图 6 所示。



图5 Kinect 1 代结构图

通过非线性优化方法求解式(25),得到总体误差实验数据见表 1。

表1 总体误差实验数据

迭代次数	总体误差	耗时 /s
0	2 307 924 136.843 756	0.000 100 815
1	2 307 911 777.515 920	0.000 168 614
2	2 307 905 803.967 105	0.000 234 950
3	2 307 902 851.811 679	0.000 300 715
4	2 307 901 383.941 958	0.000 367 889
5	2 307 900 652.705 269	0.000 430 786
6	2 307 900 288.157 070	0.000 493 706
7	2 307 900 106.333 046	0.000 556 517
8	2 307 900 115.609 100	0.000 611 769

由表可以发现,在第 6 次迭代后总体误差基本稳定不变,说明算法已经收敛,且非线性优化方法求解式(25)模型的变换矩阵为:

$$\xi^A = \begin{bmatrix} 0.999\ 950 & -0.001\ 670 & -0.009\ 830 & -0.568\ 204 \\ 0.001\ 665 & 0.999\ 998 & -0.001\ 277 & -0.010\ 851 \\ 0.009\ 833 & 0.001\ 261 & 0.999\ 951 & -1.087\ 120 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

根据式(26), $\xi^A$  矩阵左上角  $3 \times 3$  子矩阵为旋转矩阵  $R'$ ,右上角  $3 \times 1$  子矩阵为平移向量  $t'$ ,SVD 方法求解的旋转矩阵  $R$  与平移向量  $t$  相比几乎无异样,说明 SVD 方法已经给出解析解,且给出的结果是相机位姿的最优值。

通过 MATLAB 软件的 Robotic Toolbox 逆解旋转矩阵  $R$ ,得到旋转角度为  $0.01^\circ$ ,基本满足实际情况,平移矩阵  $t$  第一维为  $-0.56$ ,鉴于 Kinect 1 代为深度相机,即图中特征点 3D 位置已知,因此具备视觉尺度确定性。Kinect 1 代相机内部参数矩阵已标定,故所获取的图像以  $8\text{cm}$  为单位测量实际环境,因此  $-0.56$  表示相机向左平移  $4.48\text{cm}$ 。由图 1 得到图 2,与实际情况相机左移  $5\text{cm}$  大致相同,从而验证了上述算法的正确性。

#### 4 结束语

在移动机器人视觉 SLAM 过程中,本文采用 ORB 算法对图像特征点进行提取和匹配,利用已匹配的图像,通过 ICP 算法建立相机运动位姿模型,求解过程采用 SVD 方法,并通过实验得出机

器人运动过程中旋转矩阵和平移矩阵。该方法不仅可以应用于 RGB-D 相机,同样适用于激光传感器的位姿估计,对于移动机器人 SLAM 以及导航控制的研究具有一定的参考价值。

#### 参考文献:

- [1] STACHNISS C. Robotic Mapping and Exploration[M]. Berlin: Springer Berlin Heidelberg, 2009.
- [2] LIU H, ZHANG G, BAO H. A survey of monocular simultaneous localization and mapping[J]. Journal of Computer - Aided Design & Computer Graphics, 2016, 28(6): 530 - 535.
- [3] DAVISON A J. Real - time simultaneous localisation and mapping with a single camera[C]// IEEE International Conference on Computer Vision, 2003. Proceedings. New York: IEEE, 2003: 1403 - 1410.
- [4] KOLTER J Z, KIM Y, NG A Y. Stereo vision and terrain modeling for quadruped robots[C]// IEEE International Conference on Robotics and Automation. New York: IEEE, 2009: 1557 - 1564.
- [5] HENRY Peter, KRAININ M, HERBST E, et al. RGB - D Mapping: Using Depth Cameras for Dense 3D Modeling of Indoor Environments[M]. Heidelberg: Springer Berlin Heidelberg, 2014: 647 - 663.
- [6] LOWE D G. Distinctive image features from scale - invariant keypoints[J]. International Journal of Computer Vision, 2004, 60(2): 91 - 110.
- [7] BAY H, TUYTELAARS T, GOOL L V. SURF: speeded up robust features[J]. Heidelberg: Springer Berlin Heidelberg, 2006: 404 - 417.
- [8] RUBLEE E, RABAUDE V, KONOLIGE K, et al. ORB: an efficient alternative to SIFT or SURF[J]. Proceedings, 2011, 58(11): 2564 - 2571.

### Research on image matching and camera pose resolution in mobile robot vision SLAM process

LIN Zhicheng<sup>1</sup>, ZHENG Song<sup>2</sup>

(1. College of Electrical Engineering and Automation, Fuzhou University, Fujian Fuzhou, 350116, China)

(2. Fujian Key Laboratory of Industrial Control and

Information Security Technology, Fujian Fuzhou, 350008, China)

**Abstract:** In order to solve the key problem of image matching and camera pose in mobile robot visual SLAM process, it proposes ORB feature matching algorithm and matches the two frames captured from the mobile robot, obtains a set of feature points. It analyses the least squares theory characteristics, uses ICP method to establish the camera motion model according to the feature points of two frame image invariance, applies SVD camera pose estimation to build the motion model. It verifies the SVD decomposition method based on the nonlinear optimization algorithm, uses the RGB-D Kinect camera to capture images, tests the algorithm. The obtained camera pose data solve effectively the problem of three - dimensional positioning of mobile robot under local mobile process.

**Key words:** visual SLAM; ORB feature; matching algorithm; ICP algorithm