doi:10.3772/j.issn.1002-0470.2019.12.008

CFD-SLAM:融合特征法与直接法的快速鲁棒 SLAM 系统①

王化友②[*,**,***] 代 波[*,**,***] 何玉庆[**,***,****]

([*]中国科学院沈阳自动化研究所机器人学国家重点实验室 沈阳 110016)

([**]中国科学院机器人与智能制造创新研究院 沈阳 110016)

([***]中国科学院大学 北京 100049)

([****]广州中国科学院沈阳自动化研究所分所 广州 511458)

摘 要 构建了一个融合特征法与直接法的快速、鲁棒同时定位与地图构建(SLAM)系统 CFD-SLAM,该系统能够同时应用于单目、双目和 RGB-D 相机。SLAM 系统主要由 3 部分组成:追踪、局部建图和闭环检测。追踪部分融合特征法与直接法,并对关键帧和非关键帧分别采用特征法和直接法进行追踪,以提高系统的实时性和在特征缺失环境下的鲁棒性。特征法提取 ORB 特征和计算 BRIEF 描述子,并通过最小化特征点的重投影误差来获得关键帧的位姿估计。直接法通过最小化光度误差来获得非关键帧的位姿估计。对于 RGB-D 相机,逆深度误差被加入到特征法和直接法的优化目标函数中。局部建图部分负责管理局部关键帧和地图点,并通过光束平差法(BA)来优化局部关键帧位姿和局部地图点位置。对于 RGB-D 相机,该 SLAM 系统能够构建八叉树地图,用于机器人导航等高级任务。闭环检测部分通过检测闭环关键帧和执行位姿图优化来提高 SLAM 系统的全局一致性。本文通过在开源数据集上与典型开源 SLAM 系统进行对比实验,证明了本文的 SLAM 系统在保证定位精度的同时,具有较好的实时性和鲁棒性。

关键词 同时定位与地图构建(SLAM);特征法;直接法;光束平差法(BA);闭环检测;重定位;逆深度误差

0 引 言

同时定位与地图构建(simultaneous localization and mapping,SLAM)是一种用来同时估计机器人位姿和构建环境地图的技术。在过去的几十年中,SLAM 技术一直是研究的热点,现在已经广泛应用于无人机、无人车、机器人、AR/VR 等领域。SLAM 技术还被许多学者认为是实现自主机器人的关键[1]。为了更好地解决 SLAM 问题,各种各样的传感器被用来获取环境信息,包括激光雷达、单目、双目和 RGB-D 相机等。由于视觉传感器能够获得较为丰富的环境信息,视觉 SLAM 一直是近几年的研究热点,因而也产生了一批优秀的视觉 SLAM 系统[2-9]。但是各种常用视觉传感器都有其局限性,单目相机无法获取像素点深度信息,因此基于单目相机的 SLAM 系统无法获取机器人运动和环境的实际尺度信息,而且存在尺度偏移问题。双目相机虽然能够通过双目匹配计算获取每个像素点的深度信息,但是获取图像上所有像素点的深度信息需要大量的计算。RGB-D 相机可以直接获取像素点的深度信息,但是深度相机测量范围有限,而且受光照影响较大,所以基于 RGB-D 相机的 SLAM 系统大多只

① 国家自然科学基金(61433016,61503369)和广东省科技计划(2017B010116002)资助项目。

② 男,1994 年生,硕士生;研究方向:视觉 SLAM;联系人,E-mail: wanghuayou@sia.cn

(收稿日期:2019-01-27)

能应用于室内环境。

当前主流的视觉 SLAM 系统框架主要包含 3 部分:前端视觉里程计(visual odometry, VO)、后端优化(back-end optimization)和闭环检测(loop closure)[3-6,9]。前端视觉里程计主要用来估计连续图像帧之间的位姿变换关系。后端优化用来优化前端里程计的误差。闭环检测部分检测是否到达之前到过的位置。前端视觉里程计主要有 2 种方法:特征法和直接法[3-9]。特征法首先提取特征点和计算描述子,通过最小化匹配特征点之间的重投影误差(reprojection error)来估计机器人的位姿。直接法则直接利用图像灰度信息,通过最小化光度误差(photometric error)来估计机器人的位姿。特征法需要提取特征点和计算描述子,不仅耗时,而且不能应用于低纹理等特征缺失的环境中。相比于特征法,直接法在低纹理等特征缺失环境下更加鲁棒,而且不需要耗时来提取特征点和计算描述子。但是直接法要求相邻帧图像具有较大重叠,不能很好处理视角变化较大的场景,而特征法又恰好能够适用于大视角变化场景。直接法的另外一个缺陷就是没有对应的闭环检测方法,基于直接法的 SLAM 系统仍然需要提取特征才能够进行闭环检测。

针对直接法和特征法所存在的问题,一些学者开始尝试将直接法和特征法进行融合,实现 2 种方法的优缺点互补。最典型的研究成果是 SVO (fast semi-direct monocular visual odometry)[4]算法。由于在追踪过程中不需要提取特征点,在普通笔记本上,运行帧率可以达到 300 帧/s。SVO 只在关键帧上提取 FAST (features from accelerated segment test)角点,然后利用直接法,根据 FAST 角点周围图像块的灰度信息进行图像位姿估计。该系统的目标应用平台为无人机的俯视相机,无法适用于机器人运动较为复杂的场景。该系统采用的是单目相机,不能获得机器人运动和环境的真实尺度信息,而且存在漂移。SVO 只是一个 VO,没有后端优化和闭环检测功能,同时也基本没有建图功能。Krombach 等人[10]通过融合基于直接法的 LSD-SLAM (large-scale direct monocular SLAM)[5]和基于特征法的视觉里程计 LIBVISO2 (library for visual odometry 2)[11]构建了一个半稠密的实时立体视觉里程计。对于非关键帧,该系统采用特征点法进行估计,关键帧采用半稠密的直接图像对齐,并将特征点法的估计值作为直接法的初始值。Feng 等人[12]对每幅图像提取 ORB (oriented FAST and rotated BRIEF)特征点,如果能够提取足够数量的特征点,则利用特征点法进行追踪,否则采用直接法进行估计。文中并没有给出细节信息,也没有给出里程计的定位精度对比实验。文献[13]构建了一个融合直接法和特征法的快速双目 SLAM 系统。该系统的里程计部分与 SVO 系统中的位姿估计方法基本一致,但是该作者引入了曝光增益补偿和曝光偏执补偿。

本文构建新的融合特征法与直接法的快速、鲁棒 SLAM 系统,该系统能够同时应用于单目、双目和 RGB-D 相机。本文的主要贡献有:

(1) 提出了融合特征法与直接法的思想,并通过对关键帧和非关键帧采用不同的追踪方法实现了对图像帧快速、鲁棒的追踪过程;

(2) 采用了一个基于缓冲器的初始化策略,使得初始化过程更加宽松和鲁棒;

(3) 针对 RGB-D 相机,本文提出了同时优化重投影误差和逆深度误差的追踪优化方法,并构建了能够用于导航等高级任务的八叉树地图。

本文组织结构如下。第 1 部分给出了 SLAM 系统常用的相机模型和坐标变换表达形式。第 2 部分给出了整个 CFD-SLAM (combining feature-based method and direct method)系统的框架和流程,并对框架中的每一个组成部分进行了详细分析。与典型开源 SLAM 系统的对比实验结果在第 3 部分给出。第 4 部分对本文提出的 CFD-SLAM 系统进行了总结,并给出了未来的研究方向。

1 相机模型与坐标变换

1.1 相机模型

相机模型(camera model)给出相机坐标系下的 3 维点和像素坐标系下的 2 维像素点之间的变换关系。本文采用广泛使用的针孔相机模型,针孔相机模型可以表示为

$$P_{uv} = \pi(P_c) = \left(\frac{X_c f_x}{Z_c} + c_x, \frac{Y_c f_y}{Z_c} + c_y \right)^{\mathrm{T}} \tag{1}$$

其中，$P_{uv} = (u, v)^{\mathrm{T}}$ 是像素坐标系下的一个 2 维点，$P_c = (X_c, Y_c, Z_c, 1)^{\mathrm{T}}$ 是当前齐次相机坐标系下的一个 3 维点。f_x 和 f_y 是相机的焦距，c_x 和 c_y 是针孔相机模型的相机中心坐标系。这个函数通常被称为投影函数。相机坐标系下的 3 维点也能够通过像素坐标系下 2 维点的坐标获得，并且这个函数被称为逆投影模型，它通常被定义为

$$P_c = \pi^{-1}(P_{uv}, Z_c) = \left(\frac{u - c_x}{f_x}Z_c, \frac{v - c_y}{f_y}Z_c, Z_c\right)^{\mathrm{T}} \tag{2}$$

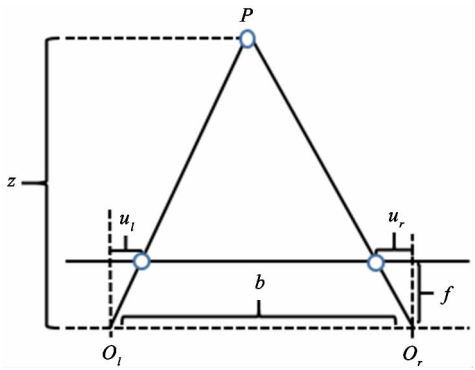

图 1 双目相机模型示意图

双目相机模型如图 1 所示，地图点的深度 z 和左右两侧图像上的像素坐标的对应关系为

$$z = \frac{fb}{d} \tag{3}$$

其中，f 表示相机的焦距，b 表示双目相机的基线长度，$d = u_l - u_r$ 被称为视差，是左右图像的横坐标 u_l 和 u_r 之差。

1.2 3 维刚体坐标变换

刚体坐标变换应用广泛的表达形式是变换矩阵（transform matrix）。变换矩阵主要包括 2 个组成部分：旋转和平移。变换矩阵通常被定义为如下形式：

$$\boldsymbol{T} = \begin{pmatrix} R & t \\ 0^{\mathrm{T}} & 1 \end{pmatrix} \in SE(3) \tag{4}$$

其中，$R \in SO(3)$ 且 $SO(3)$ 代表特殊正交群（special orthogonal group），$t \in R^3$ 且 R^3 表示 3 维空间。变换矩阵 $\boldsymbol{T}$ 是刚体坐标变换 G 的超参数（hyperparameter）表达，这是由于 $\boldsymbol{T}$ 拥有 12 个参数，但是刚体变换只有 6 个自由度（degree of freedom, DOF）。因此，特殊欧式群（special Euclidean group）$SE(3)$ 的李代数（lie algebra）表达 $se(3)$ 常被用来表达刚体变换，那么变换矩阵 $\boldsymbol{T}$ 能够利用一个 6 维向量 $\boldsymbol{\xi}$ 表达。变换矩阵 $\boldsymbol{T}$ 和 6 维向量 $\boldsymbol{\xi}$ 的关系可以表示为

$$\boldsymbol{T} = \exp(\boldsymbol{\xi}^{\wedge}) \tag{5}$$

其中，$\boldsymbol{\xi}^{\wedge}$ 是向量 $\boldsymbol{\xi}$ 的反对称矩阵（antisymmetric matrix），$\exp(\,)$ 表示指数函数。

1.3 重投影函数

基于投影函数和 3 维刚体变换，一个图像上的 2 维像素点位置能够通过另一张图像上对应的 2 维像素点位置来获得，关系式可以表达为

$$P'_{uv} = \tau(P_{uv}, \boldsymbol{T}) = \pi(\boldsymbol{T}\,\pi^{-1}(P_{uv}, D(P_{uv}))) \tag{6}$$

其中，$D(P_{uv})$ 是像素点 P_{uv} 对应的深度，$\boldsymbol{T}$ 是变换矩阵，π 和 π^{-1} 分别为投影函数和逆投影函数。

2 CFD-SLAM 系统框架与流程

CFD-SLAM 是在 ORB-SLAM2 开源系统基础上进行构建和改进的，整个系统框图如图 2 所示。其中无色模块与 ORB-SLAM2 相同，有色模块与 ORB-SLAM2 不同。CFD-SLAM 系统主要由 3 个线程组成，分别为追踪、局部建图和闭环检测线程。下面将对这几部分进行详细分析。

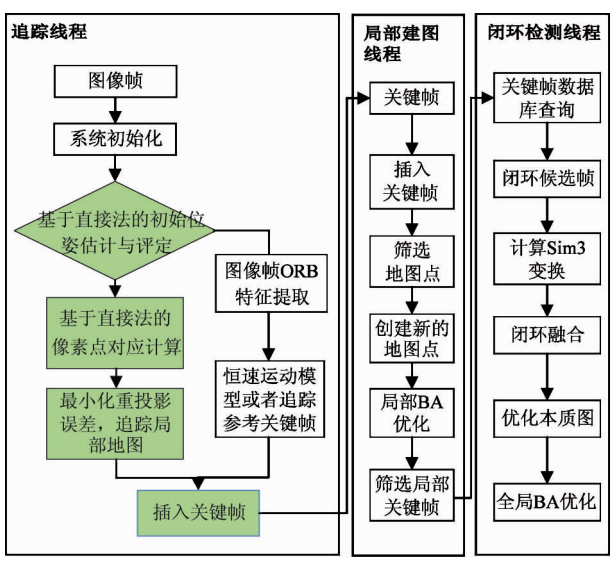

图 2 标准 CFD-SLAM 系统框架

（1）系统初始化。对于单目相机，本文采用的是基于缓冲器的初始化方法，相比于 ORB-SLAM2、

LSD-SLAM 等方法具有更好的初始化速度和有效性。其主要思想是不像 ORB-SLAM2 那样必须使用连续两帧图像进行初始化,而是将前面的图像帧存入到缓冲器中,然后当前帧在缓冲器中搜索参考帧来尝试初始化,直到初始化完成。双目和 RGB-D 相机则可直接进行初始化,不需要单独的初始化步骤。

(2)追踪线程。追踪线程的核心思路是对于每帧图像,先通过直接法进行初始位姿估计,然后对初始位姿估计的结果进行评定。如果利用直接法能够获得一个较好的初始位姿估计,那么就利用关键帧对应地图点与图像像素点的对应关系来优化地图点对应的像素点坐标,最后通过最小化重投影误差来跟踪局部地图。否则,对当前图像帧提取 ORB 特征[14]和计算 BRIEF(binary robust independent elementary features)描述子[15],并执行恒速运动模型或者追踪参考关键帧和追踪局部地图等 ORB-SLAM2 中原有的基于特征点的追踪方法。并通过定义关键帧生成准则来决定是否将当前图像帧生成为关键帧。如果该图像帧被定义为关键帧,则将该关键帧插入到局部地图线程中。

(3)局部建图线程。局部建图线程接收追踪线程创建的关键帧、删除冗余的地图点、创建新的地图点,并执行局部光束平差法(bundle adjustment, BA)来优化局部关键帧的位姿和这些关键帧能够观测到的地图点的位置。为了保证 CFD-SLAM 系统能够用于大范围的环境,局部建图线程还会删除冗余的关键帧。对于 RGB-D 相机,局部建图线程基于关键帧的位姿和深度图像构建了可用于导航、交互等高级任务的八叉树地图。

(4)闭环检测线程。闭环检测线程通过查询关键帧数据库,并通过开源库 DBoWs[16]来获取闭环候选关键帧。一旦得到闭环候选关键帧,这 2 个关键帧通过特征匹配,就能够得到它们之间的位姿变换关系。通过位姿图来优化闭环关键帧的位姿,然后对齐闭环两侧的关键帧,并且融合冗余的地图点,来得到全局一致的相机位姿和环境地图。

2.1 系统初始化

对于单目相机,为了初始化该 SLAM 系统,本文采用了基于缓冲器的初始化策略。在初始化步骤时,当一个新的图像帧到来,在缓冲器中搜索参考帧来实现初始化。如果成功初始化,新的图像帧和参考帧都将被创建为 2 个关键帧。否则,初始化缓冲器将会更新,即将当前图像帧存入缓冲器。为了保证初始化的效率,本文设置了缓冲器的存储阈值,即缓冲器最多存储 20 帧图像。只有当 2 个图像帧之间具有足够的匹配时才能执行初始化过程。初始化过程可以分为 3 步:首先计算单应矩阵(homography matrix)和本质矩阵(essential matrix);然后对 2 种模型进行评估,并选择具有较少外点匹配的模型;通过选择的模型恢复运动和结构,最后通过 BA 来优化初始化结果。

2.2 融合特征法与直接法的追踪线程

追踪线程的目的是通过连续图像帧估计相机的位姿和决定是否将当前图像帧生成为关键帧。为了获得在特征缺失环境下的快速、鲁棒相机位姿估计,追踪线程将直接法和特征点法进行了融合。从图 2 可以看出,该系统通过直接法来获得当前图像帧的初始位姿估计,然后对直接法进行评定。如果直接法的初始位姿估计满足要求,那么将地图点投影到当前帧,通过最小化图像块的光度误差,对地图点对应的当前帧像素点位置进行优化,最后再通过优化地图点与匹配像素点之间的重投影误差来获得精确的相机位姿。如果直接法的初始位姿估计不满足要求,那么追踪线程将首先对当前图像帧提取 ORB 特征[14]和计算 BRIEF 描述子[15],然后利用恒速运动模型和追踪参考关键帧来获得当前帧的初始位姿估计。最后追踪局部地图来获得更加精确的位姿估计。如果追踪线程跟丢,那么基于词袋法(bag of words, BOWs)的重定位模块则对相机进行重定位。追踪线程最后决定是否将当前图像帧生成为关键帧。由于该系统是基于 ORB-SLAM2,所以本文将主要对改进部分进行分析。

2.2.1 基于直接法的位姿估计

本文直接法追踪借鉴了 SVO[4]的基本思想,所以直接法追踪的过程主要分为 3 步:稀疏图像对齐(sparse image alignment)、特征对齐(feature alignment)、位姿与结构优化(pose and structure optimization)。稀疏图像对齐是通过最小化相同 3 维点投

影到当前帧和参考帧像素点之间的光度误差，来估计这两帧图像之间的相对位姿变换。特征对齐是通过对齐特征块来修正 3 维地图点对应的 2 维像素点坐标。位姿与结构优化通过最小化上一步对齐特征之间的重投影误差来优化位姿和结构。下面将对这 3 步进行详细分析，并给出针对 RGB-D 相机的特殊处理。

（1）稀疏图像对齐

稀疏图像对齐步骤通过最小化相同 3 维点投影到当前帧和参考帧像素点之间的光度误差，来估计这两帧图像之间的初始位姿变换，如图 3 所示。由于本文中提取的是特征点，只能通过三角化得到特征点像素位置对应的深度信息，所以在计算光度误

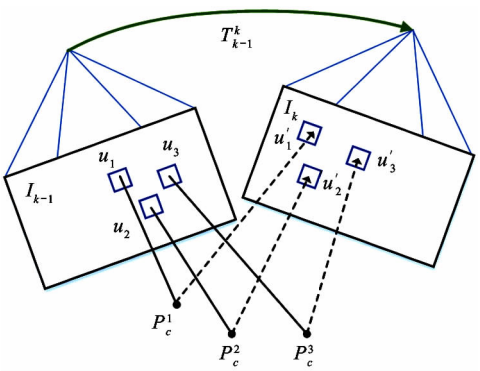

图 3　特征图像对齐示意图

差时采用特征点像素位置周围 4 ×4 的图像块。本文通过最小化两帧图像所有对应图像块之间的光度误差来计算相机位姿变换关系，公式如下：

$$\boldsymbol{T}^k_{k-1} = \underset{\boldsymbol{T}^k_{k-1}}{\operatorname{argmin}} \frac{1}{2} \sum_{i \in \overline{\Re}} \| \delta I(\boldsymbol{T}^k_{k-1}, P^i_{uv}) \|^2 \tag{7}$$

其中 $\overline{\Re}$ 和 $\delta I(\boldsymbol{\xi}, P^i_{uv})$ 分别表示图像块的集合和对应图像块之间的光度误差，分别可表示为

$$\overline{\Re} = \{P_{uv} \mid P_{uv} \in R_{k-1} \wedge \pi(\boldsymbol{T} \cdot \pi^{-1}(P_{uv}, Z_c)) \in \Omega_k\} \tag{8}$$

$$\delta I(\boldsymbol{\xi}, P^i_{uv}) = I_k(\pi(\widehat{\boldsymbol{T}^k_{k-1}}, P^i_c)) - I_{k-1}(\pi(\boldsymbol{T}(\boldsymbol{\xi}) \cdot P^i_c)) \tag{9}$$

其中 $P^i_c = \pi^{-1}(P^i_{uv}, Z^i_c)$，$R_{k-1}$ 表示深度已知的像素点集合，Ω_k 表示的是图像区域。

为了简化计算，本文中假设光度误差是具有单位方差的正态分布（normal distribution）。鲁棒核函数（robust kernel function）也被用来解决遮挡和误匹配等问题。

由于式（7）是非线性方程，所以本文采用迭代高斯-牛顿法（Gauss-Newton，GN）进行求解。给定一个初始的相对位姿变换估计，然后不断地通过增量更新公式来更新位姿估计矩阵。由于位姿变换矩阵是有约束的，所以上面的优化问题是有约束的优化问题，为了将该问题转化为无约束的优化问题，本文将 $SO(3)$ 流形上的表达转化为李代数的表达形式。参考文献[9]，本文采用逆计算公式，即更新参考帧的位姿，更新步骤的逆被应用到当前帧，得到：

$$\widehat{\boldsymbol{T}^k_{k-1}} \leftarrow \widehat{\boldsymbol{T}^k_{k-1}} \cdot \boldsymbol{T}(\boldsymbol{\xi})^{-1} \tag{10}$$

为了加速计算，本文未调整图像块，因为相邻两帧之间的运动较小而且图像块也比较小。使用逆计算方法可以保证雅可比矩阵在迭代优化过程中保持不变，可以提前计算出来，加速整个计算过程。

（2）直接法初始位姿估计评定

直接法初始位姿估计步骤采用稀疏图像对齐，即通过最小化相同 3 维点投影到当前帧和参考帧像素点之间的光度误差，来估计这两帧图像之间的初始位姿变换。为了得到良好的初始位姿估计，需要保证有足够的 3 维点同时投影到当前帧和参考帧图像中，因此直接法只能应用于小视角运动。本文首先通过是否有足够的 3 维地图点投影到两帧图像来判定是否能够通过稀疏图像对齐得到良好的初始位姿估计，通过稀疏图像对齐获得初始位姿变换后，将参考帧图像像素点投影到当前帧，并计算两者图像块之间的光度误差，通过将该像素光度误差与设定阈值进行比较来判定初始位姿估计是否满足要求。根据估计位姿计算光度误差公式与式（9）一致。

（3）特征对齐

在稀疏图像对齐步骤中，本文通过将当前图像与上一帧图像进行对齐，即最小化图像块之间的光度误差来获得两帧图像之间的相对位姿变换关系。通过获得的相对位姿变换关系能够得到 3 维地图点在当前帧图像中对应的像素坐标位置。由于 3 维地图点坐标位置和相机位姿估计的不精确性，地图点对应的像素点位置估计仍然能够被提高。为了减小误差，相机位姿应该和整个地图对齐，而不仅仅是和参考图像帧对齐。

在初始位姿估计下,将当前帧能够观测到的所有 3 维地图点投影到当前帧中,得到地图点在当前帧中对应的 2 维特征位置。对于每个投影地图点,找到对应具有最近观测角度的能够观测到该地图点的关键帧。然后特征对齐步骤优化当前帧图像中 2 维特征位置。这是通过最小化当前帧图像块和参考关键帧图像块之间光度误差来实现的。

$$P'_{uv} = \underset{P'_{uv}}{\operatorname{argmin}} \frac{1}{2} \| I_k(P'_{uv}) - Ai \cdot I_r(P^i_{uv}) \|^2 \tag{11}$$

这个特征对齐方程是通过逆计算 Lucas-Kanade 算法求解的[17]。相比于上一步,特征对齐步骤对参考图像块采用了仿射变换(affine transformation),因为更大的 8×8 的图像块被使用,并且参考关键帧和当前帧之间具有较大的运动。

(4) 位姿与结构优化

在上一步中,已经获得了亚像素级精度的特征匹配。因此,通过最小化特征点的重投影误差来优化相机位姿:

$$\boldsymbol{T}^k_w = \underset{\boldsymbol{T}^k_w}{\operatorname{argmin}} \frac{1}{2} \sum_i \| P^i_{uv} - \pi(\boldsymbol{T}^k_w \cdot P^i_w) \|^2 \tag{12}$$

这是被广泛使用的运动 BA 问题,它能够通过迭代非线性最小二乘算法求解,例如高斯-牛顿法。3 维地图点坐标也通过最小化重投影误差进行优化。

(5) 双目相机和 RGB-D 相机的特殊处理

在直接法追踪过程中,双目相机和单目相机在算法层面没有本质区别,只是可以在两帧图像上同时进行直接法的追踪。

RGB-D 相机由于能够直接提供深度信息,在追踪的过程中,本文将同时利用多视角立体信息和深度信息。即在进行直接法初始位姿估计时不仅最小化光度误差项,还引入了几何误差项。因此,本文针对 RGB-D 相机,通过最小化光度和几何误差项来估计相机的位姿。目标函数如下:

$$\hat{\boldsymbol{\xi}} = \underset{\boldsymbol{\xi}}{\operatorname{argmin}}(r_p + \lambda r_g) \tag{13}$$

其中 r_p 表示光度误差项,与式(7)的表达形式一致。r_g 表示几何误差。λ 是一个超参数,根据实际实验结果进行调整。由于稀疏图像对齐步骤已经给出了光度误差的表达形式,所以下面将对几何误差项进行分析。基于文献[18]的研究成果,对于几何误差项,本文选择通过最小化逆深度误差,而不是深度误差。对于单个地图点的逆深度误差项被定义为

$$e^i_g = \frac{1}{d^{c,i}_c} - \frac{1}{\boldsymbol{d}^{\mathrm{T}}_z \exp(\boldsymbol{\xi}^\wedge) P^{r,i}_c} \tag{14}$$

其中,$d^{c,i}_c$ 是当前帧第 i 个特征点的深度,并且这个特征点对应于参考关键帧中特征对一个 3 维点 $P^{r,i}_c$。$\boldsymbol{d}_z$ 是一个 3 维向量,被定义为 $\boldsymbol{d}_z = [0, 0, 1]$,$\pi()$ 仍然是投影函数。几何误差项可以表示为

$$\begin{aligned} r_g &= \sum_{i=1}^{n} \omega_p \left(\frac{(e^i_g)^2}{\sigma^2_g} \right) \\ &= \sum_{i=1}^{n} \omega_p \left(\frac{\left(\frac{1}{d^{c,i}_c} - \frac{1}{\boldsymbol{d}^{\mathrm{T}}_z \exp(\boldsymbol{\xi}^\wedge) P^{r,i}_c} \right)^2}{\sigma^2_g} \right) \end{aligned} \tag{15}$$

为了估计误差项 r_g 的标准方差,将 RGB-D 相机当作双目相机进行处理。逆深度可以被定义为

$$\rho = \frac{d}{fb} \tag{16}$$

其中,d 为视差,f 是相机的焦距,b 是基线长度。逆深度参数[18]线性依赖于视差 d。逆深度误差的标准方差可以被定义为

$$\sigma_\rho = \frac{\partial \rho}{\partial d} \sigma_d = \frac{\sigma_d}{fb} \tag{17}$$

2.2.2 基于特征法的位姿估计

基于特征法的追踪过程首先对当前图像帧提取 ORB 特征[14]和计算 BRIEF 描述子[15],然后通过恒速运动模型或者追踪参考关键帧来获得图像帧的初始位姿估计。最后通过追踪局部地图来优化该图像帧的位姿。

(1) ORB 特征提取

本文对关键帧提取 ORB 特征,主要是因为 ORB 特征是在 FAST 角点的基础上,通过灰度质心法(intensity centroid method)和图像金字塔(image pyramid)实现了旋转不变性(rotation invariance)和尺度不变性(scale invariance)。同时 ORB 特征相比于 SIFT (scale-invariant feature transform) 和 SURF (speed up robust features) 等特征具有较快的提取速度。在同一幅图像中提取 1 000 个特征点时,

ORB 约花费 15 ms,SURF 需要 300 ms,SIFT 约需要 1 000 ms。

(2) 恒速运动与追踪参考关键帧

恒速运动模型是假设当前帧和上一帧图像具有相同的运动,给出当前帧的初始位姿估计。然后将上一帧图像中与当前帧特征匹配的特征点对应的地图点投影到当前帧,最小化匹配特征点之间的重投影误差来估计当前帧的位姿。追踪参考关键帧是将参考关键帧中与当前帧匹配的特征点对应的地图点投影到当前帧。优化的目标函数可以表达为

$$\hat{\boldsymbol{\xi}} = \underset{\boldsymbol{\xi}}{\operatorname{argmin}} r_r = \sum_{i=1}^{n} \omega_p\left(\frac{(e_r^i)^2}{\sigma_r^2}\right) = \sum_{i=1}^{n} \omega_p\left(\frac{(P_{uv}^{c,i} - \pi(\exp(\boldsymbol{\xi}^\wedge)P_c^{r,i}))^2}{\sigma_r^2}\right) \tag{18}$$

其中 $\omega_p()$ 表示 Huber 鲁棒核函数,可以被定义为

$$\omega_p(e) = \begin{cases} \dfrac{e^2}{2\delta} & |e| \leqslant \delta \\ |e| - \dfrac{\delta}{2} & \text{其他} \end{cases} \tag{19}$$

待优化的目标函数的求解是通过高斯-牛顿法进行求解的。为了使用高斯-牛顿法,误差函数相对于待优化变量的雅可比矩阵需要被首先计算出来,然后通过增量方程求解待优化变量的增量。下面将给出待优化重投影误差的目标函数相对于待优化位姿的雅可比矩阵,表达形式如下:

$$\frac{\partial e_r^i}{\partial \delta\boldsymbol{\xi}} = -\frac{\partial \pi(\exp(\boldsymbol{\xi}^\wedge)P_c^{r,i})}{\partial \delta\boldsymbol{\xi}} = -\frac{\partial \pi(P_c^{c,i})}{\partial P_c^{c,i}} \frac{\partial \exp(\boldsymbol{\xi}^\wedge)P_c^{r,i}}{\partial \delta\boldsymbol{\xi}} \tag{20}$$

其中,

$$\frac{\partial \pi(P_c^{c,i})}{\partial P_c^{c,i}} = -\begin{pmatrix} \dfrac{f_x}{Z_c^{c,i}} & 0 & -\dfrac{f_x X_c^{c,i}}{(Z_c^{c,i})^2} \\ 0 & \dfrac{f_y}{Z_c^{c,i}} & -\dfrac{f_y Y_c^{c,i}}{(Z_c^{c,i})^2} \end{pmatrix} \tag{21}$$

$$\frac{\partial \exp(\boldsymbol{\xi})P_c^{r,i}}{\partial \delta\boldsymbol{\xi}} = \begin{pmatrix} I & -(\exp(\boldsymbol{\xi}^\wedge)P_c^{r,i})^\wedge \\ 0^{\mathrm{T}} & 0^{\mathrm{T}} \end{pmatrix} = \begin{pmatrix} I & -(P_c^{c,i})^\wedge \\ 0^{\mathrm{T}} & 0^{\mathrm{T}} \end{pmatrix} \tag{22}$$

雅可比矩阵可以表达为如下形式:

$$\frac{\partial e_r^i}{\partial \delta\boldsymbol{\xi}} = \begin{pmatrix} \dfrac{f_x}{Z_c^{c,i}} & 0 & -\dfrac{f_x X_c^{c,i}}{(Z_c^{c,i})^2} & J_{03} & J_{04} & -\dfrac{f_x Y_c^{c,i}}{Z_c^{c,i}} \\ 0 & \dfrac{f_y}{Z_c^{c,i}} & -\dfrac{f_y Y_c^{c,i}}{(Z_c^{c,i})^2} & J_{13} & J_{14} & \dfrac{f_y X_c^{c,i}}{Z_c^{c,i}} \end{pmatrix} \tag{23}$$

其中,

$$J_{03} = -\frac{f_x X_c^{c,i} Y_c^{c,i}}{(Z_c^{c,i})^2} \tag{24}$$

$$J_{04} = f_x + \frac{f_x (X_c^{c,i})^2}{(Z_c^{c,i})^2} \tag{25}$$

$$J_{13} = -f_y - \frac{f_y (Y_c^{c,i})^2}{(Z_c^{c,i})^2} \tag{26}$$

$$J_{14} = \frac{f_y X_c^{c,i} Y_c^{c,i}}{(Z_c^{c,i})^2} \tag{27}$$

当前帧位姿的估计求解是通过 g^2o[19] 求解器来实现的,并采用了高斯-牛顿法。

(3) 追踪局部地图

追踪局部地图是将局部地图点投影到当前帧,并将地图点与特征点进行匹配,通过最小化重投影误差的方式来估计当前图像帧的位姿。重投影误差的表达式为

$$e_r^i = P_{uv}^{c,i} - \pi(\exp(\boldsymbol{\xi}^\wedge)P_c^{r,i}) \tag{28}$$

$$r_r = \sum_{i=1}^{n} \omega_p\left(\frac{(e_r^i)^2}{\sigma_r^2}\right) = \sum_{i=1}^{n} \omega_p\left(\frac{(P_{uv}^{c,i} - \pi(\exp(\boldsymbol{\xi}^\wedge)P_c^{r,i}))^2}{\sigma_r^2}\right) \tag{29}$$

该式的求解与式(18)是完全一致的。

(4) RGB-D 相机的特殊处理

双目相机可以直接通过在左右两张图像上提取特征并进行三角化直接获得特征点的深度信息,不需要多视角的三角化。在度量重投影误差时,从单目 2 维像素坐标扩展到了 3 维,增加了右侧图像横坐标像素位置误差项。

对于 RGB-D 相机,由于能够直接获得深度信息,所以仍然在前面最小化重投影误差的基础上引入逆深度误差项,则目标函数可以表达为

$$\hat{\boldsymbol{\xi}} = \underset{\boldsymbol{\xi}}{\operatorname{argmin}}(r_r + \lambda r_g) \tag{30}$$

其中重投影误差和逆深度误差和前面的表达式是一致的。求解也是通过高斯-牛顿法进行,并通过

g^2o[19] 实现。

2.2.3 关键帧提取策略

由于本文的追踪过程采用的是直接法和特征法的融合,而后端是利用共视图(covisibility graph)进行优化,所以传入到后端的关键帧是提取了 ORB 特征点的图像帧。所以,本文给出了直接法和特征法的切换条件:

(1) 当前帧利用直接法获得初始位姿估计后,地图点投影到当前帧的点数少于 30 个点;

(2) 连续执行 20 帧的直接法追踪图像帧;

(3) 距离上次关键帧的运动超过设定阈值。

在 ORB-SLAM2 原有关键帧提取策略的基础之上,本文针对直接法和特征法的融合引入了关键帧提取策略:

(1) 从上一次关键帧生成或者重定位开始,至少已经处理了 20 帧图像;

(2) 当前帧至少追踪 50 个特征点;

(3) 当前帧追踪少于 90% 的参考关键帧追踪的特征点;

(4) 如果连续使用直接法追踪的帧数大于 20 帧,或者运动超过阈值,那么对该图像帧提取,并进行特征法追踪,并将该图像帧定义为关键帧。

2.3 局部建图线程

局部建图线程(图 4)处理关键帧、执行局部 BA 优化和建立局部地图。当一个关键帧被插入,建图线程首先更新共视图。接着,冗余和错误三角化的稀疏地图点被丢弃。然后,局部 BA 优化被执行来优化当前帧、所有共视关键帧和被这些关键帧观测到的地图点。其他能够观测到这些地图点的关键帧也被插入到优化过程中,但是它们的位姿保持不变。局部 BA 优化方程被定义为

$$E = \sum_{i=1}^{m}\sum_{j=1}^{n}\left\{\omega_p\left(\frac{(e_r^{i,j})^2}{\sigma_r}\right)+\lambda\omega_p\left(\frac{(e_g^{i,j})^2}{\sigma_g}\right)\right\} \tag{31}$$

其中 m 是关键帧的数量,n 是对应于这些关键帧的地图点的数量。ω_p 仍然是鲁棒核函数,σ_r 和 σ_g 分别代表 $e_r^{i,j}$ 和 $e_g^{i,j}$ 的标准方差。$e_r^{i,j}$ 表示关键帧 i 中的地图点 j 的重投影误差,它能够被定义为

$$e_r^{i,j} = P_{uv}^{i,j} - \pi(\exp(\hat{\boldsymbol{\xi}}_i)P_w^j) \tag{32}$$

其中,$\pi()$仍然是投影函数,P_w^j 代表世界坐标系中的地图点 j。$P_{uv}^{i,j}$ 是在关键帧 i 中对应于地图点 j 的 2 维像素坐标点。$\hat{\boldsymbol{\xi}}_i$ 是 $\boldsymbol{\xi}_i$ 的反对称矩阵。

逆深度误差项 $e_g^{i,j}$ 可以被定义为

$$e_g^{i,j} = \frac{1}{d_c^{i,j}} - \frac{1}{\boldsymbol{d}_z^{\mathrm{T}}\exp(\hat{\boldsymbol{\xi}}_i)P_w^j} \tag{33}$$

其中,$\boldsymbol{d}_z$ 是一个 3 维向量,被定义为 $\boldsymbol{d}_z = [0, 0, 1]$。$d_c^{i,j}$ 表示关键帧 i 中特征点 j 对应的深度值。最终,BA 优化过程也是通过 g^2o 优化器[19] 进行求解,能够得到优化的关键帧位姿和地图点位置。

在局部 BA 优化之后,冗余的关键帧被丢弃,用来减少局部 BA 优化的复杂性和实现 life-long 操作。在相同的环境中,本文的 CFD-SLAM 系统不会一直增加关键帧的数量。如果一个关键帧观测到的地图点有 90% 以上被至少其他 3 个关键帧观测到,那么这个关键帧就被定义为冗余关键帧。

对于 RGB-D 相机,通过使用所有关键帧位姿和关键帧对应的深度图,表达环境信息的点云地图能够通过将每个关键帧对应的地图点投影到世界坐标系获得。然而,点云表达是高度冗余的,并且需要大量的计算和存储资源。为了解决这个问题,表达环境信息的占据栅格地图(occupancy grid maps)被建立,这个地图的建立是基于 OctoMap 开源地图库[20]。那么这个地图就是由体素构成,每个体素的占据信息已经给出,并且每个体素的占据信息可以通过式(34)进行更新。

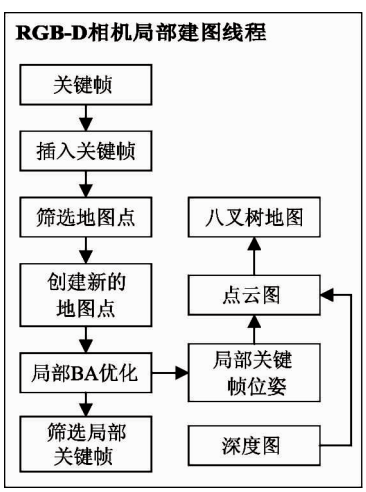

图 4 RGB-D 相机局部建图线程流程图

$$P(n \mid z_{1:T}) = \left[1 + \frac{1 - P(n \mid z_T)}{P(n \mid z_T)} \frac{1 - P(n \mid z_{1:T-1})}{P(n \mid z_{1:T-1})} \frac{P(n)}{1 - P(n)}\right]^{-1} \tag{34}$$

相比于点云地图表达,这种占据栅格地图能够更加方便地应用于高级任务,例如机器人导航和探索任务。

2.4 闭环检测线程

闭环检测线程查询关键帧数据库,通过开源库 DBoWs[16] 来搜索闭环候选关键帧,并在此基础上对每个闭环候选关键帧进行深度一致性检测。深度一致性检测是通过计算所有匹配的特征点的平均深度误差来完成的。并且定义了一个阈值,如果平均深度误差小于阈值,那么这个关键帧就可以被认为是一个闭环候选关键帧。

一旦闭环候选关键帧被检测到,那么这 2 个关键帧之间的 ORB 特征匹配就能够得到,进而,2 个关键帧之间的相对位姿变换即可求出。然后对闭环关键帧进行对齐并删除重复的地图点。最后,执行作用在所有关键帧上的位姿图优化来优化所有关键帧的位姿,但是不优化地图点的位置来实现全局一致性。变换矩阵形式的位姿图优化可以表示为

$$\{\widehat{\boldsymbol{T}_w^1}, \cdots, \widehat{\boldsymbol{T}_w^j}, \cdots, \widehat{\boldsymbol{T}_w^k}, \cdots, \widehat{\boldsymbol{T}_w^m}\} = \underset{\{\boldsymbol{T}_w^1, \cdots, \boldsymbol{T}_w^m\}}{\operatorname{argmin}} \sum_{j,k} \boldsymbol{\omega}_p(e_{j,k}^{\mathrm{T}} \boldsymbol{\Omega}_{j,k}^{-1} e_{j,k}) \tag{35}$$

其中,m 表示需要被优化的相对位姿的数量,$\boldsymbol{\omega}_p$ 仍然是鲁棒核函数,$\boldsymbol{\Omega}_{j,k}$ 是与图像尺度相关的协方差矩阵,$\boldsymbol{T}_w^j$ 表示关键帧 j 在世界坐标系下的位姿。误差项 $e_{j,k}$ 定义为如下形式:

$$e_{j,k} = \log(\boldsymbol{T}_j^k \boldsymbol{T}_w^k \boldsymbol{T}_w^{j\ -1}) \tag{36}$$

其中,$\boldsymbol{T}_j^k$ 表示关键帧 k 和 j 之间的相对位姿变换矩阵,$\boldsymbol{T}_w^k$ 表示关键帧在世界坐标系下的位姿矩阵,$\boldsymbol{T}_w^{j\ -1}$ 表示变换矩阵 $\boldsymbol{T}_w^j$ 的逆,log( )表示对数函数。

3 实验与结果分析

为了评估本文提出的 CFD-SLAM 系统的性能,本文将在各种环境下开源数据集上进行实验,并与开源 SLAM 系统进行对比,通过统计和数值分析给出系统的精度、实时性和鲁棒性的对比结果。

3.1 开源数据集精度对比实验

由于本文的 CFD-SLAM 系统能够适用于单目、双目和 RGB-D 相机,所以为了评估该系统的性能,本文基于 KITTI[21]、EuRoc[22] 和 TUM RGB-D[23] 3 个典型开源数据集,与 SLAM 系统进行了对比实验,来验证该系统的精度、实时性和在无结构、低纹理、特征缺失等环境下的鲁棒性。所有的实验都是在一个具有 Intel Core i7-7700 (4 Cores @ 2.40 GHz) 和 16 GB RAM 的 ThinkPad T470P 笔记本上完成的。本文中的每个实验结果都是在单个数据集上面执行 5 次取平均值的结果,以避免实验结果的偶然性。

3.1.1 EuRoc 数据集

为了评估本文 CFD-SLAM 系统的性能,首先通过在 EuRoc 数据集上运行该系统来进行验证。EuRoc 数据集[22] 是通过无人机在 2 个不同房间和 1 个大型工业环境下飞行采集的,采集到的数据包含双目图像(20 Hz)、同步的 IMU (inertial measurement unit) 测量信息(200 Hz)和通过 VICON 动补系统获得的真实状态。该数据集通过飞机飞行的速度、遮挡、纹理等信息,将数据集分为简单、中等和困难 3 个级别。

表 1 给出了在所有 EuRoc 图像序列中,本文 CFD-SLAM 系统与 ORB-SLAM2 系统的绝对平移均方根误差(root-mean-square error, RSME)[21] 的对比结果,其中 CFD-MONO、CFD-STEREO、ORB-MONO 和 ORB-STEREO 分别代表 CFD-SLAM 和 ORB-SLAM 在单目和双目情况下的运行结果。表 1 的实验结果表明,CFD-SLAM 系统具有与 ORB-SLAM2 近似的定位精度,即误差处在相同的量级上。本文的系统虽然引入了直接法,但是由于设定的特征提取阈值、关键帧插入方式和后端优化保证了系统整体的定位精度,该特征法与直接法的融合算法并不会影响系统的定位精度。而且 ORB-SLAM2 在 V2_03_difficul 图像序列中会出现跟丢的情况,而 CFD-SLAM 由于引入了直接法的特性,可以在保证定位精度的前提下,提高系统的鲁棒性,尤其是在特征缺失和快速运动造成图像模糊的场景。

表 1　CFD-SLAM 与 ORB-SLAM2 在 EuRoc 数据集上的精度对比结果

	关键帧轨迹 RSME(m)			
	ORB-STEREO	CFD-MONO	ORB-MONO	CFD-STEREO
MH _01 _easy	0.098	**0.070**	0.039	**0.035**
MH _02 _easy	0.084	**0.066**	0.028	**0.018**
MH _03 _medium	0.096	**0.071**	0.029	**0.028**
MH _04 _difficult	0.083	**0.081**	0.164	**0.119**
MH _05 _difficult	0.079	**0.060**	0.073	**0.060**
V1 _01 _easy	0.024	**0.015**	0.046	**0.035**
V1 _02 _medium	0.029	**0.020**	0.030	**0.020**
V1 _03 _difficult	**0.087**	X	0.062	**0.048**
V2 _01 _easy	0.018	**0.015**	0.040	**0.037**
V2 _02 _medium	**0.037**	0.017	0.052	**0.035**
V2 _03 _difficult	X	X	**0.097**	X

注: X 表示"该 SLAM 系统不能在这个图像序列上运行"

3.1.2　KITTI 数据集

为了评估本文 CFD-SLAM 系统在大尺度场景和不完全运动环境中的运行性能,本文将通过 KITTI 数据集来进行验证。KITTI 数据集[21]是通过一辆汽车在城市和高速环境下采集的,数据集中主要包含双目图像序列。立体传感器具有 54 cm 的基线长度,10 Hz 的频率,其中图像具有 1 240 × 376 的分辨率。本文将 CFD-SLAM 与 ORB-SLAM2[9] 和 STEREO LSD-SLAM[24]进行了对比,采用了 2 种不同的度量方式,一种是文献[21]提出的绝对变换均方根误差 t_{abs} ,另一种是文献[23]提出的相对平移误差 t_{rel} 和相对旋转误差 r_{rel}。表 2 给出了 3 个 SLAM 系统在 11 个图像序列上的对比结果。CFD-SLAM 系统具有与 ORB-SLAM2 和 STEREO LSD-SLAM 近似的定位精度,这与前面在单目情况下的分析结果是一致的。证明了该直接法和特征法融合算法的有效性。同时,图 5 给出了几个 CFD-SLAM 在 KITTI 图像序列上运行得到的轨迹,并与真实轨迹进行对比,从图中可以看出,系统具有较小的偏移误差,可以很好地恢复真实轨迹。同时图 6 也给出了和轨迹相匹配的特征点地图。

表 2　CFD-SLAM 与 ORB-SLAM2 在 KITTI 数据集上的对比实验结果

序列	CFD-SLAM(stereo)			ORB-SLAM2(stereo)			STEREO SLD-SLAM		
	t_{rel} (%)	r_{rel} (deg/100m)	t_{abs} (m)	t_{rel} (%)	r_{rel} (deg/100m)	t_{abs} (m)	t_{rel} (%)	r_{rel} (deg/100m)	t_{abs} (m)
00	0.76	0.27	1.61	0.70	**0.25**	1.30	**0.63**	0.26	**1.0**
01	1.83	0.32	12.61	**1.39**	**0.21**	9.98	2.36	0.36	**9.0**
02	0.78	**0.23**	8.43	**0.76**	**0.23**	6.17	0.79	**0.23**	**2.6**
03	0.80	0.22	0.93	**0.71**	**0.18**	**0.68**	1.01	0.28	1.2
04	0.48	0.25	0.28	0.48	**0.13**	0.26	**0.38**	0.31	**0.2**
05	0.58	**0.15**	**0.80**	**0.40**	0.16	**0.80**	0.64	0.18	1.5
06	0.64	0.17	0.92	**0.51**	**0.15**	0.77	0.71	0.18	1.3
07	0.62	0.29	0.59	**0.50**	**0.28**	**0.57**	0.56	0.29	0.5
08	1.10	0.33	3.57	**1.05**	0.32	**3.44**	1.11	**0.31**	3.9
09	0.95	0.26	9.44	**0.87**	0.27	**3.12**	1.14	**0.25**	5.6
10	0.70	0.29	2.15	**0.60**	**0.27**	**1.06**	0.72	0.33	1.5

3.1.3　TUM RGB-D 数据集

为了评估本文提出的 CFD-SLAM 系统在 RGB-D 相机中的定位精度,将 CFD-SLAM 系统与能够应用于 RGB-D 相机的 SLAM 系统进行了对比,包括 ORB-SLAM2[9]和 RGBD-SLAM[25]。表 3 给出了 3 种系统移动误差的比较结果。均方根误差被用来作为移动误差的度量指标。这 3 个系统都会在某些图像序列中跟丢,只对鲁棒追踪部分的图像帧进行了定位精度的比较。从这 2 个表中可以得出,相比于 ORB-SLAM2 和 RGBD-SLAM,CFD-SLAM 系统能够获得了更高的定位精度,该定位精度和 ORB-SLAM2 基本处在同一量级。针对RGB-D相机,本文更好地

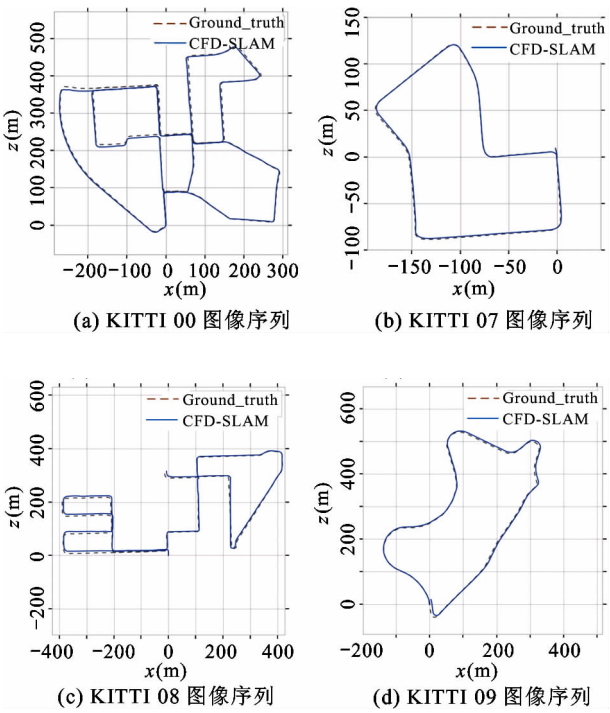

图 5 CFD-SLAM 在 KITTI 00,07,08 和 09 图像序列运行得到的估计轨迹

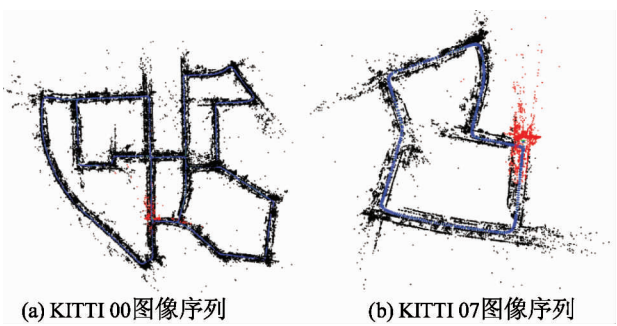

图 6 CFD-SLAM 在 KITTI 00 和 07 序列上得到的特征点地图

表 3 CFD-SLAM 与 ORB-SLAM2 和 RGBD-SLAM 在 TUM RGBD 数据集上的精度对比实验结果(m)

TUM RGB-D 序列	CFD-SLAM	ORB-SLAM2	RGBD-SLAM
fr1 _ desk	0.016	0.016	0.026
fr1 _ desk2	0.021	0.022	X
fr1 _ room	0.045	0.047	0.087
fr1 _ xyz	0.011	0.015	X
fr2 _ desk	0.011	0.009	0.057
fr2 _ xyz	0.004	0.004	X
fr3 _ office	0.012	0.010	X
fr3 _ nostructure _ texture _ near _ withloop	0.014	0.013	X
fr3 _ structure _ texture _ far	0.011	0.012	X

注:X 表示“该 SLAM 系统不能在这个图像序列上运行”

利用了深度信息,将深度信息和 RGB 信息融合在了同一个优化框架下进行优化。图 7 给出了系统在 TUM RGBD 数据集上运行得到的八叉树地图。

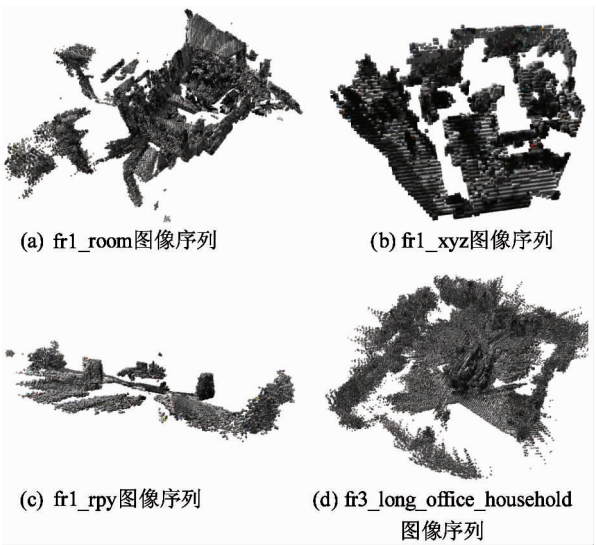

图 7 CFD-SLAM 系统在 TUM RGBD 数据集 fr1 _ room, fr1 _ xyz, fr2 _ rpy 和 fr3 _ long _ office _ household 4 个图像序列上构建的八叉树地图

3.2 实时性

本文提出融合特征法与直接法,目的是提高系统的实时性。为了对本文 CFD-SLAM 系统进行更加完善的评估,对 CFD-SLAM 系统的单帧图像追踪时间与 ORB-SLAM2 进行了对比。表 4 给出了在相同参数设置的情况下,CFD-SLAM 和 ORB-SLAM2 单帧图像的追踪时间。实验结果表明,CFD-SLAM 系统在绝大多数数据集下都消耗了更少的追踪时间,其中最优的情况,单帧图像追踪消耗的时间只有 ORB-SLAM2 的 1/3。这主要是因为本文中对大多数图像帧都不需要提取特征和计算描述子,不需要进行特征匹配,只需要计算光度误差即可。当然,表中还有一些图像序列,CFD-SLAM 和 ORB-SLAM2 具有近似的单帧图像追踪时间,这主要是由图像本身决定的,可能是直接法在这些图像中没有获得较好的位姿估计,大多数图像帧需要提取 ORB 特征并进行匹配。在 TUM RGBD 数据集上,虽然本文将深度信息引入了追踪过程的优化框架中,但仍然能够保证在大多数情况下,CFD-SLAM 具有比 ORB-SLAM2 在 RGB-D 数据集上更快的追踪效果。

表 4　CFD-SLAM 与 ORB-SLAM2 的单帧图像追踪时间对比结果(s)

		ORB-MONO	CFD-MONO	ORB-STEREO	CFD-STEREO	ORB-RGB-D	CFD-RGB-D
EuRoc	MH _ 01 _ easy	0.032	**0.014**	0.056	**0.021**	X	X
	MH _ 02 _ easy	0.027	**0.015**	0.056	**0.029**	X	X
	MH _ 03 _ medium	0.025	**0.013**	0.057	**0.039**	X	X
	MH _ 04 _ difficult	0.028	**0.012**	0.054	**0.018**	X	X
	MH _ 05 _ difficult	0.031	**0.018**	0.067	**0.019**	X	X
	V1 _ 01 _ easy	0.025	**0.014**	0.053	**0.031**	X	X
	V1 _ 02 _ medium	0.033	**0.010**	0.075	**0.014**	X	X
	V1 _ 03 _ difficult	0.031	**0.009**	0.071	**0.013**	X	X
	V2 _ 01 _ easy	0.029	**0.014**	0.075	**0.016**	X	X
	V2 _ 02 _ medium	0.026	**0.012**	0.059	**0.011**	X	X
	V2 _ 03 _ difficult	0.027	**0.011**	0.062	**0.015**	X	X
KITTI	00	0.037	**0.031**	0.085	**0.039**	X	X
	01	0.034	**0.032**	0.112	**0.057**	X	X
	02	0.038	**0.031**	0.095	**0.082**	X	X
	03	0.032	**0.028**	0.098	**0.045**	X	X
	04	0.034	**0.032**	0.104	**0.031**	X	X
	05	0.036	**0.031**	0.088	**0.026**	X	X
	06	0.034	**0.030**	0.094	**0.038**	X	X
	07	0.035	**0.030**	0.078	**0.028**	X	X
	08	0.037	**0.029**	0.084	**0.079**	X	X
	09	0.037	**0.029**	0.080	**0.025**	X	X
	10	0.038	**0.028**	0.077	**0.023**	X	X
TUM RGB-D	fr1 _ 360	0.020	**0.019**	X	X	0.028	**0.027**
	fr1 _ desk	0.026	**0.021**	X	X	0.034	**0.027**
	fr1 _ desk2	0.021	**0.020**	X	X	0.037	**0.026**
	fr1 _ floor	0.022	**0.019**	X	X	0.024	**0.023**
	fr1 _ room	0.022	**0.022**	X	X	0.033	**0.024**
	fr1 _ xyz	0.031	**0.018**	X	X	0.031	**0.021**
	fr2 _ 360 _ hemi	0.020	**0.015**	X	X	0.027	**0.019**
	fr2 _ 360 _ kidnap	0.020	**0.016**	X	X	0.021	**0.020**
	fr2 _ desk	0.028	**0.011**	X	X	0.035	**0.017**
	fr2 _ large _ withloop	**0.020**	0.021	X	X	**0.019**	0.022
	fr2 _ large _ no _ loop	**0.021**	**0.021**	X	X	0.029	**0.018**
	fr2 _ pioneer _ 360	**0.025**	0.026	X	X	0.024	**0.023**
	fr2 _ pioneer _ slam	0.024	**0.022**	X	X	**0.026**	0.027
	fr2 _ pioneer _ slam2	**0.023**	0.025	X	X	0.025	**0.024**
	fr2 _ pionner _ slam3	**0.022**	0.032	X	X	**0.025**	**0.025**
	fr2 _ rpy	0.024	**0.014**	X	X	0.026	**0.018**
	fr2 _ xyz	0.025	**0.016**	X	X	0.029	**0.019**

注: X 表示“该 SLAM 系统不能在这个图像序列上运行”

3.3 鲁棒性

本文将直接法和特征法进行融合的目的是提高系统在特征缺失、低纹理等环境下的鲁棒性。为了验证本文 CFD-SLAM 系统在特征缺失环境下的鲁棒性,本文将该系统作用在 TUM RGBD 数据集中有无纹理和有无结构的图像序列中,并与 ORB-SLAM2 进行比较。度量的是系统运行 20 次,系统跟丢的次数,实验结果见表 5。ORB-SLAM2 在 TUM RGB-D 数据集中的 fr3 _ nostructure _ notexture _ far, fr3 _ structure _ notexture _ far 和 fr3 _ structure _ notexture _ near 图像序列中都跟丢了,而 CFD-SLAM 都可以很好地追踪这些图像序列。从表中可以看出,相比于 ORB-SLAM2,CFD-SLAM 系统能够跟踪大多数无结构、无纹理场景,具有较少的跟丢次数。因此,可以得出融合直接法和特征法对于特征缺失环境比只用特征法具有更好的鲁棒性。之所以能够实现这样的效果,主要是因为 CFD-SLAM 系统将直接法和特征法进行了结合,实现了两者优点的互补,直接法能够适用于特征缺失和结构缺失的环境中,特征法能够适用于快速运动、强旋转等环境中。

表 5 CFD-SLAM 与 ORB-SLAM2 的鲁棒性比较结果

TUM RGB-D 序列	ORB-SLAM2	CFD-SLAM
fr3 _ nostructure _ notexture _ far	20	**0**
fr3 _ nostructure _ notexture _ near _ withloop	20	**3**
fr3 _ nostructure _ texture _ far	12	**0**
fr3 _ nostructure _ texture _ near _ withloop	**0**	**0**
fr3 _ structure _ notexture _ far	20	**0**
fr3 _ structure _ notexture _ near	20	**0**
fr3 _ structure _ texture _ far	**0**	**0**
fr3 _ structure _ texture _ near	**0**	**0**

这 2 种方法的融合使得 CFD-SLAM 系统能够很好地处理快速运动、无纹理和无结构的环境。而且对于 RGB-D 相机,本文更好地利用了深度信息,在追踪过程将逆深度误差也引入到优化框架中,这也是使得系统更加鲁棒的一个重要因素。

4 结 论

本文构建了融合特征法与直接法的快速、鲁棒 SLAM 系统,该系统能够同时应用于单目、双目和 RGB-D 相机。本文将直接法和特征法相结合,弥补了单一方法的不足,实现了快速、鲁棒的 CFD-SLAM 系统。本文针对 RGB-D 相机,提出了同时优化重投影误差和逆深度误差的方法,提高了系统的鲁棒性。同时,本文在大量的数据集和实际机器人上对系统进行了测试,并与 SLAM 系统进行了比较,验证了本文系统具有较好的定位精度,大场景下的运行效果、实时性和在快速运动、无结构、特征缺失等特殊环境下的鲁棒性。

本文对直接法和特征法的结合实际上是半直接法和特征法的结合,后续会进一步研究纯特征法与直接法的融合方式,以提高系统的定位精度、鲁棒性和实时性能。后续还会探讨视觉与惯导融合的 SLAM 系统,并尝试通过提取语义信息的方式来提高系统的鲁棒性,构建能够用于更高级任务的语义地图。

参考文献

[1] Durrant-Whyte H, Bailey T. Simultaneous localization and mapping: part I[J]. *IEEE Robotics & Automation Magazine*, 2006, 13(2): 99-110

[2] Davison A J, Reid I D, Molton N D, et al. Monoslam: real-time single camera slam[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007, 29(6): 1052-1067

[3] Klein G, Murray D. Parallel tracking and mapping for small AR workspaces[C]//IEEE and ACM International Symposium on Mixed and Augmented Reality, Piscataway, USA, 2008: 250-259

[4] Forster C, Pizzoli M, Scaramuzza D. SVO: fast semi-direct monocular visual odometry[C]//IEEE International Conference on Robotics and Automation, Piscataway, USA, 2014: 15-22

[5] Engel J, Schops T, Cremers D. LSD-SLAM: large-scale direct monocular SLAM[C]// European Conference on Computer Vision, Berlin, Germany, 2014: 834-849

[6] Mur-Artal R, Montiel J M M, Tardós J D. ORB-SLAM: a versatile and accurate monocular SLAM system[J]. *IEEE Transactions on Robotics*, 2015, 31(5): 1147-1163

[7] Kaess M, Johannsson H, Roberts R, et al. iSAM2: incremental smoothing and mapping using the bayes tree [C]// IEEE International Conference on Robotics and Automation, Piscataway, USA, 2012: 216-235

[8] Engel J, Koltun V, Cremers D. Direct sparse odometry [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018, 40(3): 611-625

[9] Mur-Artal R, Tardós J D. ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras[J]. *IEEE Transactions on Robotics*, 2017, 33(5): 1255-1262

[10] Krombach N, Droeschel D, Behnke S. Combining feature-based and direct methods for semi-dense real-time stereo visual odometry[C]// International Confersence on Intelligent Autonomous Systems, Berlin, Germany, 2016: 855-868

[11] Geiger A, Ziegler J, Stiller C. StereoScan: dense 3D reconstruction in real-time[C]// Intelligent Vehicles Symposium, Piscataway, USA, 2011: 963-968

[12] Feng J L, Zhang C J, Sun B, et al. A fusion algorithm of visual odometry based on feature-based method and direct method[C]// The Chinese Automation Congress, Jinan, China, 2017: 1854-1859

[13] 张国良, 姚二亮, 林志林, 等. 融合直接法与特征法的快速双目 SLAM 算法[J]. 机器人, 2017, 39(6): 879-888

[14] Rublee E, Rabaud V, Konolige K, et al. ORB: an efficient alternative to SIFT or SURF[C]// IEEE International Conference on Computer Vision, Piscataway, USA, 2011: 2564-2571

[15] Calonder M, Lepetit V, Strecha C, et al. BRIEF: binary robust independent elementary features[C]// European Conference on Computer Vision, Berlin, Germany, 2010: 778-792

[16] Gálvez-López D, Tardos J D. Bags of binary words for fst place recognition in image sequences[J]. *IEEE Transactions on Robotics*, 2012, 28(5): 1188-1197

[17] Baker S, Matthews I. Lucas-Kanade 20 years on: a unifying framework: part I[J]. *International Journal of Computer Vision*, 2002, 56(3): 221-255

[18] Civera J, Davison A J, Montiel M J. Inverse depth parametrization for monocular SLAM[J]. *IEEE Transactions on Robotics*, 2008, 24(5): 932-945

[19] Kuemmerle R, Grisetti, G, Strasdat H, et al. g^2o: a general framework for graph optimization[C]// IEEE International Conference on Robotics and Automation, Piscataway, USA, 2011: 3607-3613

[20] Hornung A, Wurm K M, Bennewitz M, et al. OctoMap: an efficient probabilistic 3D mapping framework based on octrees[J]. *Autonomous Robots*, 2013, 34(3): 189-206

[21] Geiger A, Lenz P, Stiller C, et al. Vision meets robotics: the KITTI dataset[J]. *International Journal of Robotics Research*, 2013, 32(11): 1231-1237

[22] Burri M, Nikolic J, Gohl P, et al. The EuRoC micro aerial vehicle datasets[J]. *International Journal of Robotics Research*, 2016, 35(10): 1157-1163

[23] Sturm J, Engelhard N, Endres F, et al. A benchmark for the evaluation of RGB-D SLAM systems[C]// IEEE/RSJ International Conference on Intelligent Robots and Systems, Piscataway, USA, 2012: 573-580

[24] Engel J, Stückler J, Cremers D. Large-scale direct SLAM with stereo cameras[C]// IEEE/RSJ International Conference on Intelligent Robots and Systems, Piscataway, USA, 2015: 1935-1942

[25] Endres F, Hess J, Sturm J, et al. 3-D mapping with an RGB-D camera [J]. *IEEE Transaction on Robotics*, 2014, 30(1): 177-187

CFD-SLAM: a fast and robust SLAM system combining feature-based method and direct method

Wang Huayou[*,**,***], Dai Bo[*,**,***], He Yuqing[**,***,****]

([*] State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016)

([**] Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110016)

([***] University of Chinese Academy of Sciences, Beijing 100049)

([****] Shenyang Institute of Automation (Guangzhou), Chinese Academy of Sciences, Guangzhou 511458)

Abstract

This work proposes a fast and robust simultaneous localization and mapping (SLAM) system combining feature-based method and direct method (CFD-SLAM), which can be used for monocular, stereo and RGB-D cameras. The SLAM system consists of 3 parts: tracking, local mapping and loop closure. The tracking part combines feature-based method and direct method, feature-based method and direct method are utilized for keyframes and non-keyframes tracking respectively to improve real-time performance and robustness in low-texture environments. Feature-based method extracts ORB (oriented FAST and rotated BRIEF) features and computes BRIEF (binary robust independent elementary features) descriptors, and obtains the pose estimation of keyframes by minimizing reprojection error of feature points. Direct method obtains the pose estimation of non-keyframes by minimizing photometric error. For RGB-D cameras, inverse depth error is added into optimization cost function of both feature-based method and direct method. The local mapping part is in charge of managing local keyframes and map points, and optimizing both poses of local keyframes and position of local map points by using bundle adjustment (BA). To perform high level tasks such as navigation, the Octomap of the environment can be constructed for RGB-D cameras. The loop closure part improves the global consistency of the SLAM system by detecting loop keyframes and executing pose graph optimization. Finally, experiments on public datasets against other state-of-the-art SLAM systems demonstrate that the system is faster and more robust than the state-of-the-art SLAM system without precision reduction.

Key words: simultaneous localization and mapping (SLAM), feature-based method, direct method, bundle adjustment (BA), loop closure, relocalization, inverse depth error