

融合光流法和特征匹配的视觉里程计

许广富^{1,2}, 曾继超^{1,2}, 刘锡祥^{1,2*}

¹东南大学仪器科学与工程学院, 江苏 南京 210096;

²微惯性仪表与先进导航技术教育部重点实验室, 江苏 南京 210096

摘要 针对传统视觉里程计中光流法定位精度差、特征点法耗时多的问题,提出一种融合光流法和特征匹配的视觉里程计模型。该模型融合了基于帧间优化的 LK 光流位姿估计和基于关键帧的光流/特征点位姿优化算法。针对传统参考帧/当前帧跟踪方式容易产生累积误差的问题,在光流法的基础上,通过引入局部优化算法对相机进行初步位姿估计;针对特征法中图像插入频率过高、耗时多的问题,在关键帧的基础上,通过构建光流/特征点统一损失函数进行相机位姿优化。在 EuRoC 数据集上进行了算法定位精度测试,结果表明,简单环境下所提算法与特征点法的定位精度相当;在特征点缺失情况下,所提算法的定位精度较特征点法有所提高,具有一定的鲁棒性。运行时间测试结果表明,在保证定位精度的基础上,所提算法的运行时间较特征点法减少了 37.9%,具有一定的实时性。

关键词 机器视觉; 视觉里程计; 光流法; 特征点法; 位姿估计

中图分类号 TP24

文献标志码 A

doi: 10.3788/LOP57.201501

Visual Odometer Based on Optical Flow Method and Feature Matching

Xu Guangfu^{1,2}, Zeng Jichao^{1,2}, Liu Xixiang^{1,2*}

¹School of Instrument Science and Engineering, Southeast University, Nanjing, Jiangsu 210096, China;

²Key Laboratory of Micro-Inertial Instrument and Advanced Navigation Technology, Ministry of Education, Nanjing, Jiangsu 210096, China

Abstract Aiming at the problems that there exist poor accuracy of the optical flow method and time consumption of the feature point method in traditional visual odometers, we propose the model of a visual odometer by integrating optical flow with feature matching. This model mainly fuses the LK optical flow pose estimation based on inter-frame optimization with the optical flow / feature point pose optimization based on key frames. In addition, aiming at the problem that there occur accumulation errors in the traditional reference-frame/current-frame tracking method, we introduce a local optimization algorithm on the basis of the optical flow method to preliminarily estimate the camera's pose. Simultaneously, aiming at the problems that the image insertion frequency is too high and time consumption in the feature method, we construct a unified loss function of optical flow/feature points on the basis of the key frames to optimize the camera's pose. The position accuracy test results of the algorithm on the EuRoC dataset show that the position accuracy of the proposed algorithm in simple environments is equivalent to that of the feature point method, and in the case of missing feature points, the proposed algorithm possesses position accuracy higher than that of the feature point method and has certain robustness. The running time test results show that on the basis of ensuring the positioning accuracy, the running time of the proposed algorithm is 37.9% less than that of the feature point method, and the algorithm has the certain real-time performance.

Key words machine vision; visual odometer; optical flow method; feature point method; pose estimation

OCIS codes 150.4232; 150.4620

收稿日期: 2020-01-13; 修回日期: 2020-02-07; 录用日期: 2020-02-24

基金项目: 国家自然科学基金(61973079, 51979041)

* E-mail: scliuseu@163.com

1 引言

随着定位理论的不完善和计算机视觉科学的不断发展,视觉里程计(VO, Visual Odometry)成为新的研究热点^[1-2]。视觉里程计只利用视觉图像而无需场景和运动先验信息即可对当前相机位姿进行估计,不受移动部件磨损等误差影响,是自主定位导航和视觉导航系统的关键,被广泛应用于AR(Augmented Reality)、机器人导航、无人机、航天航空等各种领域。

目前,常见的视觉里程计的实现方法有特征点法和光流法。特征点法首先对两帧图像进行特征提取和匹配,之后采用对极几何、PNP算法或ICP算法来估计位姿^[3]。文献[4]对此框架进行了改进,在对相邻两帧进行特征提取后,采用优化估计算法对经过PNP估计的位姿进行最优求解,从而得到更为准确的结果。2015年,研究者提出了基于ORB(Oriented FAST and Rotated BRIEF)特征的定位与地图构建算法(ORB-SLAM)^[5];2017年,在ORB-SLAM的基础上,研究者又提出ORB-SLAM2^[6]。ORB-SLAM采用ORB算法^[7]提取特征点和计算描述子,其吸收了PTAM(Parallel Tracking and Mapping)^[8]的优点,是基于特征点法的即时定位与地图构建(SLAM, Simultaneous Localization and Mapping)系统的优秀代表。光流法是利用图像上像素的光流场信息计算两帧之间像素的运动速度,并通过对极几何等进行相机运动估计。由于不需要进行描述子的计算和特征的匹配,因此光流法的运行速度较快。但光流法假设了光照不变性^[9],这个假设在实际应用中往往无法得到满足,从而导致光流法在相机位姿估计精度上不如特征点法高。文献[10]提出了半直接视觉里程计(SVO),其通过直接跟踪稀疏的角点来进行相机位姿的估计,运行速度快,但定位精度方面不及ORB-SLAM等基于特征点的方法。

光流法相对于特征点法,不需要进行描述子的计算和特征的匹配,运行速度较快,定位频率高,但无法避免累积误差;而特征点法的定位精度较高,定位频率低,特征缺失时算法容易失效。两者具有较好的互补特性,因此可将两者进行融合。文献[11]提出一种基于松耦合的光流融合特征匹

配的框架,该框架在松耦合模式下采用卡尔曼滤波对基于光流和特征匹配计算得到的位姿进行融合,能够在一定程度上提高计算速度,但松耦合模式丢失了大量信息,定位精度远不如传统特征点法。文献[12]提出了一种将运动约束加入光流跟踪并加权融合特征点法的方法,该方法通过增加约束来优化算法的定位精度,但忽略了前端光流和特征点匹配之间的关系,需要进一步改进。

本文在前人基础上提出了一种基于紧耦合框架的融合光流和特征匹配的视觉里程计。首先,针对传统参考帧/当前帧跟踪方式^[13]中信息利用不充分、容易产生累积误差等问题,在光流法的基础上通过引入帧间优化对相机位姿进行初步估计;其次,图像插入频率过高将导致信息冗余,计算资源消耗过多会导致算法的定位精度下降,所以需要去粗取精,筛选系列图像中的关键帧,在关键帧的基础上构建光流/特征点统一损失函数,从而进行相机位姿优化,以便更快更好地得到位姿估计结果。

2 相关算法

2.1 Lucas-Kanade 光流法

Lucas-Kanade 光流法^[14]是一种经典的图像跟踪算法,简称LK算法。LK算法将模板图像 $T(x)$ 与输入图像 $I(x)$ 进行联合,通过求解最小化目标函数以得到目标状态变量,即

$$\sum_x \{I[\mathbf{W}(x; \mathbf{p} + \Delta \mathbf{p})] - T(x)\}^2, \quad (1)$$

式中: $\Delta \mathbf{p}$ 为状态量的微小变换; $\mathbf{W}(x; \mathbf{p})$ 为两帧图像之间的仿射变换函数; $\mathbf{x} = (x, y)^T$ 为图像像素坐标, T 表示转置; $\mathbf{p} = (p_1, p_2, p_3, p_4, p_5, p_6)^T$ 为目标状态变量,且

$$\mathbf{W}(x; \mathbf{p}) = \begin{pmatrix} 1 + p_1 & p_3 & p_5 \\ p_2 & 1 + p_4 & p_6 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}. \quad (2)$$

对(1)式进行一阶泰勒展开,得到

$$\sum_x \left\{ I[\mathbf{W}(x; \mathbf{p})] + \nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \Delta \mathbf{p} - T(x) \right\}^2, \quad (3)$$

式中: $\nabla I = \left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right)$ 为经过仿射变换后的图像梯度。

再将(3)式展开,并对 $\Delta \mathbf{p}$ 求导,可得

$$2 \sum_x \left(\nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right) \left\{ I[\mathbf{W}(x; \mathbf{p})] + \nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \Delta \mathbf{p} - T(x) \right\}. \quad (4)$$

为了求解(3)式的最小二乘问题,根据高斯牛顿法,令(4)式为0,从而得到

$$\Delta \mathbf{p} = \mathbf{H}^{-1} \sum_x \left(\nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right)^T \{T(\mathbf{x}) - I[\mathbf{W}(\mathbf{x}; \mathbf{p})]\}, \quad (5)$$

$$\mathbf{H} = \sum_x \left(\nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right)^T \left(\nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right). \quad (6)$$

综上可得 LK 算法的流程如下。

- 1) 根据 \mathbf{p} 和 \mathbf{W} 计算图像仿射变换。
- 2) 计算图像误差 $T(\mathbf{x}) - I[\mathbf{W}(\mathbf{x}; \mathbf{p})]$ 。
- 3) 计算经过 \mathbf{W} 变换的图像梯度 ∇I 。
- 4) 计算在 $(\mathbf{x}; \mathbf{p})$ 处的雅克比矩阵 $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ 。
- 5) 计算图像下降梯度 $\nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ 。
- 6) 根据(6)式计算 Hessian 矩阵 \mathbf{H} ;
- 7) 根据(5)式计算得到 $\Delta \mathbf{p}$ 。
- 8) 更新目标状态: $\mathbf{p} = \mathbf{p} + \Delta \mathbf{p}$ 。
- 9) 判断 $\|\Delta \mathbf{p}\| < \epsilon$ (ϵ 为阈值) 是否成立, 成立则算法收敛, 不成立则重新执行步骤 1)~8)。

至此,两帧图像之间的对应关系可以通过由光流法求解得到的目标状态 \mathbf{p} 获得。光流法的运行速度较快,但也存在一些问题,如像素误差直接带来估计误差,累积误差无法消除等,需要对此进行进一步改进。

2.2 特征点法

特征点法包括特征点提取和特征匹配,其中特征提取以 ORB 特征点为例,包括 FAST (Features

from Accelerated Segment Test) 角点提取和 BRIEF (Binary Robust Independent Elementary Features) 描述子计算。当提取 FAST 角点时,需要在图像中选取像素 p , 假设其亮度为 I_p , 以该像素为圆点, 选取半径为 3 的圆上的所有点为考察目标, 当有连续 N 个点的亮度大于 I_p 或者小于 I_p 时即认为像素 p 为 FAST 特征点。为了增强 FAST 角点的鲁棒性, 通过灰度质心法对特征点进行旋转不变性描述。在提取 FAST 关键点之后, 还需要对每个点进行描述子计算, 即 BRIEF 描述子计算。BRIEF 描述子是一种二进制向量, 由 0 和 1 组成, 选择相邻像素 (p, q) 进行对比, 当 p 大于 q 时 BRIEF 描述子取 1, 反之取 0。选择 128 个像素对, 以此获得一个 128 维 BRIEF 描述子。当计算完描述子后, 进行特征匹配, 特征匹配能解决数据关联问题, 常用方法有暴力匹配法、最近邻法和快速近似最近邻法, 这些算法目前已经比较成熟, 并易于实现。通过上述步骤获得相机采集的相邻两幅图像特征点所有对应关系之后, 采用 PNP、光束平差 (BA, Bundle Adjustment) 等算法即可获得相机位姿估计。特征点法估计位姿具有较高的精度, 但计算时间长, 因此需要进行改进。

本文在上述相关算法的基础上, 将光流跟踪与特征匹配进行融合, 具体流程如图 1 所示, 即先进行基于帧间优化的光流位姿估计, 并在此基础上结合关键帧的特征匹配再进行位姿优化。

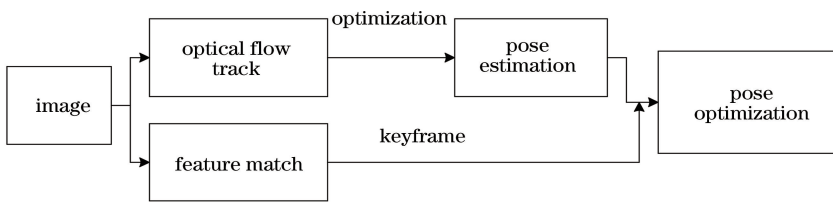


图 1 算法流程图

Fig. 1 Flow chart of algorithm

3 基于帧间优化的 LK 光流位姿估计算法

图 2 所示为传统参考帧/当前帧 VO 的工作示意图, 在这种模式中, 定义了参考帧 (Ref) 和当前帧 (Cur) 两个概念, 其中 Kf1、Kf2 分别为第一个和第二个关键帧, T_{12} 、 T_{23} 、 T_{34} 为各关键帧之间的位姿。以 $(t-1)$ 时刻图像为参考帧, 以 t 时刻的图像作为当前帧, 采用第一节所述的 LK 光流法跟踪这两帧图像, 获得两帧图像的对应关系后, 采用对极几何进

行两帧之间的位姿估计。当运动到 $(t+1)$ 时刻时, 丢弃 $(t-1)$ 时刻的图像, 将 t 时刻的图像定义为新的参考帧, 以 $(t+1)$ 时刻的图像为新的当前帧, 重复上述计算, 从而获得 $(t+1)$ 时刻的估计, 如此往复获得相机连续运动轨迹。

这种模式的 VO 缺少了不同帧之间的约束, 如果参考帧估计不准, 误差将不断累积, 最后会出现严重的漂移。另外, 从信息利用角度来看, 每次计算完位姿之后就丢弃上一帧数据, 显然浪费了很多信息。

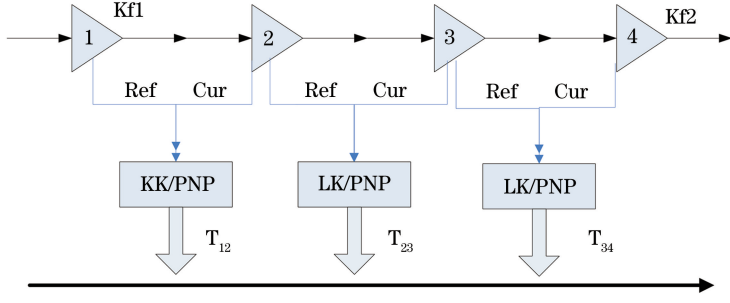


图 2 参考帧/当前帧 VO 的工作示意图

Fig. 2 Schematic of referenced-frame/current-frame VO

因此,本文在传统光流法的基础上通过引入帧间优化方法,在局部范围内对相机位姿进行非线性优化,以此尽可能消除噪声的影响。非线性优化需要建立最小化目标函数,具体推导过程如下。

将(2)式变形为

$$\mathbf{W}(\mathbf{x}; \mathbf{p}) = \begin{pmatrix} 1+p_1 & p_3 & 0 \\ p_2 & 1+p_4 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} + \begin{pmatrix} p_5 \\ p_6 \\ 0 \end{pmatrix}, \quad (7)$$

记变换后的图像像素坐标为 $\mathbf{x}' = (x', y')^T$, 代入(7)式进行齐次变换,得

$$\mathbf{x}' = \mathbf{M}_{3 \times 3} \mathbf{x} + \mathbf{N}_{3 \times 1}, \quad (8)$$

式中: \mathbf{x}, \mathbf{x}' 为齐次变换前后的图像像素坐标; $\mathbf{M} =$

$$\begin{pmatrix} 1+p_1 & p_3 & 0 \\ p_2 & 1+p_4 & 0 \\ 0 & 0 & 1 \end{pmatrix}; \mathbf{N} = \begin{pmatrix} p_5 \\ p_6 \\ 0 \end{pmatrix}, \text{其中参数为 LK 光}$$

流法所求出的目标状态变量。

相机位姿变化也会使图像发生相应变化,因此根据相机模型,可得

$$\mathbf{x}' = \mathbf{s}^{-1} \mathbf{K} \exp(\hat{\boldsymbol{\xi}}) \mathbf{X}, \quad (9)$$

式中: \mathbf{s} 为图像深度信息; \mathbf{K} 为相机内参; $\hat{\boldsymbol{\xi}}$ 为相机待估计位姿的李群形式; \mathbf{X} 为上一帧通过逆投影恢复的 3D 点的坐标。

由于观测噪声的存在,(8)式和(9)式不会严格相等,因此可联立两式构造出一个误差表达式:

$$\mathbf{e}(\boldsymbol{\xi}) = \mathbf{M} \mathbf{x}_{i'} + \mathbf{N} - \mathbf{s}^{-1} \mathbf{K} \exp(\hat{\boldsymbol{\xi}}) \mathbf{X}_{i'}, \quad (10)$$

式中: $\mathbf{x}_{i'}$ 为特征点的像素坐标; \mathbf{x}_i 为特征点的三维坐标; i' 为特征点的序号; $\mathbf{e}(\boldsymbol{\xi})$ 为两帧之间的误差函数; $\boldsymbol{\xi}$ 为相机待估计位姿的李群形式。之后再对一段连续的图像误差进行求和,即可得到该非线性问题的最小化目标函数:

$$\boldsymbol{\xi}^* = \underset{\boldsymbol{\xi}}{\operatorname{argmin}} \frac{1}{2} \sum_{k=1}^m \sum_{i'=1}^n \cdot \|\mathbf{M}_k \mathbf{x}_{i'}^k + \mathbf{N}_k - (\mathbf{s}_{i'}^k)^{-1} \mathbf{K} \exp(\hat{\boldsymbol{\xi}}_k) \mathbf{X}_{i'}^k\|_2^2, \quad (11)$$

式中: k 为图像序号; $\mathbf{M}_k, \mathbf{N}_k$ 是由 LK 光流法求出的目标状态变量构成的矩阵; $\mathbf{x}_{i'}^k$ 为特征点的像素坐标; $\mathbf{s}_{i'}^k$ 为第 k 幅图像的深度信息; $\hat{\boldsymbol{\xi}}_k$ 为第 k 个关键帧时相机待估计位姿的李群形式; $\mathbf{X}_{i'}^k$ 为特征点的三维坐标; n 是每帧光流跟踪的像素个数; m 为优化周

期的帧数。

在(11)式中, $\boldsymbol{\xi}$ 为待优化的变量,为了求解这个优化问题,需要研究误差随着位姿的变化,所以需要分析其关于位姿的导数形式。本文采用李代数中的扰动模型进行求解,于是,将(10)式改写为

$$\mathbf{e}(\boldsymbol{\xi} + \delta\boldsymbol{\xi}) = \mathbf{M} \mathbf{x}_{i'} + \mathbf{N} - \mathbf{s}^{-1} \mathbf{K} \exp(\hat{\boldsymbol{\xi}}) \exp(\delta\hat{\boldsymbol{\xi}}) \mathbf{X}_{i'} = \mathbf{M} \mathbf{x}_{i'} + \mathbf{N} - \mathbf{s}^{-1} \mathbf{K} \exp(\hat{\boldsymbol{\xi}}) \mathbf{X}_{i'} - \mathbf{s}^{-1} \mathbf{K} \mathbf{q}, \quad (12)$$

式中: $\mathbf{q} = \delta\hat{\boldsymbol{\xi}} \exp(\hat{\boldsymbol{\xi}}) \mathbf{X}_{i'}$ 。对(12)式进行一阶泰勒展开:

$$\mathbf{e}(\boldsymbol{\xi} + \delta\boldsymbol{\xi}) = \mathbf{M} \mathbf{x}_{i'} + \mathbf{N} - \mathbf{s}^{-1} \mathbf{K} \exp(\hat{\boldsymbol{\xi}}) \mathbf{X}_{i'} - \mathbf{s}^{-1} \mathbf{K} \frac{\partial \mathbf{q}}{\partial \delta\boldsymbol{\xi}} \delta\boldsymbol{\xi} = \mathbf{e}(\boldsymbol{\xi}) - \mathbf{s}^{-1} \mathbf{K} \frac{\partial \mathbf{q}}{\partial \delta\boldsymbol{\xi}} \delta\boldsymbol{\xi} = \mathbf{e}(\boldsymbol{\xi}) - \mathbf{J} \delta\boldsymbol{\xi}, \quad (13)$$

式中: $\frac{\partial \mathbf{q}}{\partial \delta\boldsymbol{\xi}} = [\mathbf{I}, -\hat{\mathbf{q}}]$, 其中 $\hat{\mathbf{q}}$ 为 \mathbf{q} 的李群形式; $\mathbf{J} = \mathbf{s}^{-1} \mathbf{K} \frac{\partial \mathbf{q}}{\partial \delta\boldsymbol{\xi}}$ 。

至此我们推导出了误差关于位姿李代数的雅克比矩阵,之后使用高斯牛顿法计算增量,以对极几何求解出的位姿作为初始值,进行迭代优化求解以估计出更为精确的位姿。

4 基于关键帧的光流/特征点位姿优化算法

虽然光流法的定位频率较高,但定位精度相对较低,且会产生累计误差;而特征点法的定位精度较高,但定位频率较低,因此可以将两者进行融合。本

文采用紧耦合的方式将两者进行融合,融合示意图如图 3 所示。

图 3 左边为纯 LK 方法的视觉里程计,其中长方形为普通图像帧,每一帧图像都可通过光流法进行位姿估计。右边是融合了特征匹配的视觉里程计结构,该结构的核心是联合了关键帧的特征匹配,其中大圆为位姿图中的待优化的相机位姿节点,一端连接光流法位姿估计,一端连接关键帧的特征匹配(菱形),即该节点具有两个约束边,因此可建立统一损失函数进行联合优化。

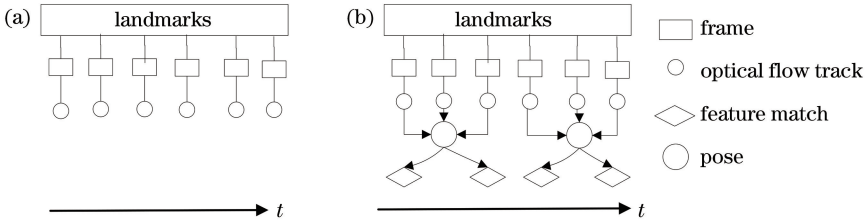


图 3 光流/特征匹配融合示意图。(a)光流法;(b)光流法加入特征匹配

Fig. 3 Schematic of optical-flow/feature-matching fusion. (a) Optical flow method; (b) feature matching introduced in optical flow method

4.1 关键帧的选取与光流位姿累积

在融合过程中,因为特征匹配方法和光流法的频率不同(如图 4 所示),所以本文提出的优化位姿算法借鉴了视觉惯性融合中的预积分思想,计算两关键帧之间所有的 LK 位姿并进行累积(如图 4 中

Acu-LK 所示),将累积结果作为两关键帧之间的光流位姿约束,再利用该约束与两关键帧之间经特征点匹配后的位姿进行联合优化。图 4 中 i 和 j 分别表示第 i 个和第 j 个关键帧, Δt 表示两幅图像之间的时间间隔。

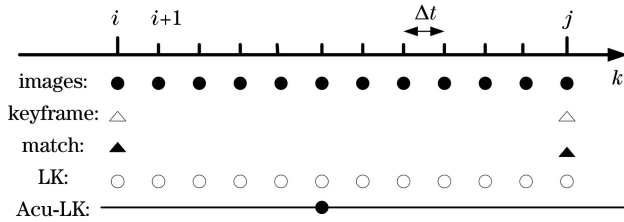


图 4 不同帧率间的关系

Fig. 4 Relationship among different frame rates

本文采用当前关键帧与上一关键帧的距离作为衡量指标来判断关键帧。

$$\| \prod_i \Delta T_i \| < \| \gamma \|, \quad (14)$$

式中: γ 为关键帧判断阈值; ΔT_i 是相邻两帧之间的转移矩阵。位姿不断累积并与阈值相比较,当累积位姿大于阈值时,定义此时图像帧 i 为关键帧。

联合优化模型以关键帧为单位,因此需要对两关键帧之间的所有位姿进行累积。根据(8)式中的 $M_{3 \times 3}$ 、 $N_{3 \times 1}$,推导两关键帧 i 和 j 间的相对位姿变换情况。

由于

$$\begin{cases} x_2 = M_1 x_1 + N_1 \\ x_3 = M_2 x_2 + N_2, \\ x_4 = M_3 x_3 + N_3 \end{cases} \quad (15)$$

递推可得

$$x_4 = M_3 [M_2 (M_1 x_1 + N_1) + N_2] + N_3, \quad (16)$$

因此关键帧 i 和 j 之间的变换可表示为

$$x_j = \Pi(x_i), \quad (17)$$

式中: $\Pi(\cdot)$ 函数为包含 $M_i, M_{i+1}, \dots, M_{j-1}, N_i, N_{i+1}, \dots, N_{j-1}$ 参数的转移函数。

因此两关键帧之间光流法的误差方程为

$$e_1^k = \Pi(x_{i'}) - s_{i'}^{-1} K \exp(\hat{\xi}_k) X_{i'}, \quad (18)$$

式中: e_1^k 为光流法的误差方程; $\hat{\xi}_k$ 为关键帧 k 和

$k+1$ 之间转移矩阵的李群。

4.2 重投影误差与统一损失函数

为了抑制光流法的累积误差,通过引入特征匹配法增加了两个位姿节点的约束,从而对累计误差进行修正。本文以重投影误差(即观测到的投影位置与3D点按照当前估计的位姿进行投影得到的位置的误差)作为特征点法的误差方程,图5所示为重投影误差示意图。

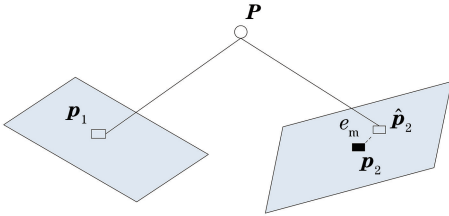


图5 重投影误差示意图

Fig. 5 Schematic of reprojection error

P 为3D点, p_1 为该点在前一帧上的投影, \hat{p}_2

为3D点在当前帧相机坐标系中投影的实际值,而 p_2 为投影点理论值,两者之间具有一定的距离, e_m 即为当前帧上的重投影误差,因此调整相机位姿,使两者距离最小,从而实现位姿优化。

根据相机模型可得重投影误差为

$$e_m^{i',k} = x_{i'} - s_i^{-1} \mathbf{K} \exp(\hat{\xi}_k) \mathbf{X}_{i'}, \quad (19)$$

式中: $e_m^{i',k}$ 为特征匹配法的误差; $\hat{\xi}_k$ 为关键帧 k 和 $k+1$ 之间转移矩阵的李群。

至此已推导出光流位姿误差方程和特征匹配误差方程,将(18)式和(19)式进行联立,构建统一的损失函数为

$$J(\xi) = \sum_{i'=1}^Q \sum_{k=1}^K (e_m^{i',k})^T e_m^{i',k} + \sum_{r=1}^R \sum_{k=1}^K (e_l^{r,k})^T e_l^{r,k}, \quad (20)$$

式中: $e_l^{r,k}$ 为光流法的误差; r 为光流点的序号; K 为关键帧的数量; Q 为特征点的数量; R 为光流点数量。

求解(20)式即可得相机位姿为

$$\xi^* = \underset{\xi}{\operatorname{argmin}} \left[\sum_{i'=1}^Q \sum_{k=1}^K (e_m^{i',k})^T e_m^{i',k} + \sum_{r=1}^R \sum_{k=1}^K (e_l^{r,k})^T e_l^{r,k} \right]. \quad (21)$$

5 实验结果与分析

为了验证本文所提出的融合光流法和特征点法的视觉里程计的有效性,在 EuRoC 数据集(数据集网址为 <https://projects.asl.ethz.ch/datasets/doku.php?id=knavvisualinertialdatasets>)中进行实验。数据集分为简单(easy)、中等(medium)和复杂(difficult)三种情况,包含了弱纹理和光线变化等环境因素。考虑算法的复杂度和运行效率,本文采用数据集中的相机左目图像,每隔20帧用特征点法修正一次,并从算法定位精度和运行时间方面进行不同算法的对比。算法运行平台为 Intel i5-4200 处理器,2.5 GHz 主频,8G 内存,不使用图形处理器加速。

5.1 算法的定位精度实验

首先,在 EuRoC 数据集的 V2_01_easy 图像中进行实验,将本文所提算法、ORB_SLAM2 特征点法和 LK 光流法进行对比,三种算法所估计出来的轨迹与真实轨迹关系如图6所示。

从图6可以看出,LK 光流法估计的轨迹和真实轨迹之间有较大的误差,说明 LK 光流法估计位姿的精度较差;而 ORB_SLAM2 算法估计的轨迹和本文所提算法估计的轨迹均与真实轨迹重合较好,

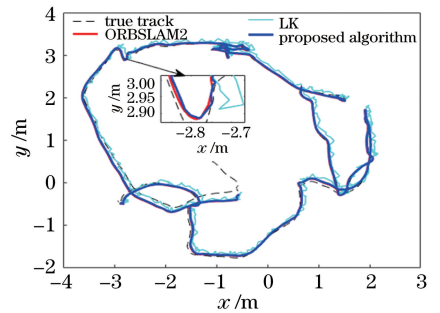


图6 所提算法的轨迹图

Fig. 6 Trace map of proposed algorithm

表明本文所提算法不仅能够有效地估计出相机位姿,还具有与特征点法相当的定位精度。但在一些运动剧烈变化的条件下,本文所提算法估计的轨迹会出现偏差较大的情况,这是由于此时光流法和特征匹配失效,统一损失函数值变大。

为了进一步证明本文算法的有效性,在 EuRoC 其他数据集中也进行了相同实验,部分结果如图7所示。

由图7可见,特征点法估计的轨迹与真实轨迹重合较好,而 LK 光流法估计的轨迹贴合度较差。由于本文设置的关键帧的阈值较小,因此本文所提算法的定位精度与特征点法相当,但当出现大范围的光线波动时,所提算法的定位精度低于特征点法。

为了定量分析定位精度,本文采用均方根误差

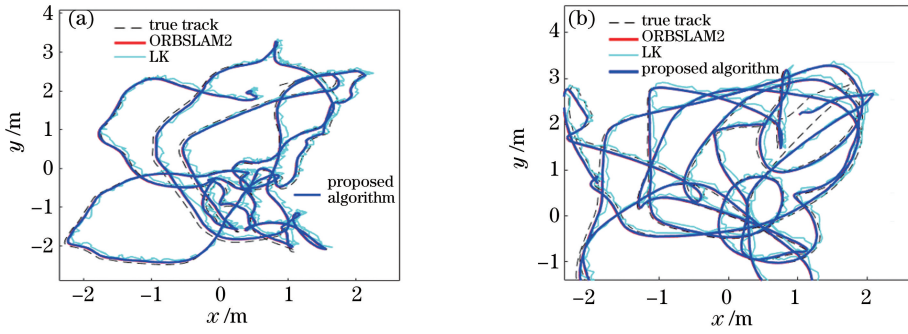


图7 EuRoC数据集上不同算法的估计轨迹。(a) V1_01_easy 图像;(b) V1_02_medium 图像

Fig. 7 Estimated traces of different algorithms on EuRoC dataset. (a) V1_01_easy image; (b) V1_02_medium image

RMSE(Root Mean Square Error)进行计算,其计算公式为

$$r_{\text{RMSE}} = \sqrt{\frac{1}{N} \sum_{h=1}^N |\mathbf{x}_p^h - \mathbf{x}_t^h|^2}, \quad (22)$$

式中: N 采样点的数量; $\mathbf{x}_p^i, \mathbf{x}_t^i$ 分别为算法估计出的坐标点和真实轨迹坐标点; h 为坐标点序号。

三种算法的均方根误差如表1所示。由表1可以发现,光流法的定位精度整体上不如特征点法和本文算法。本文算法在简单场景下(V1_01_

easy, V2_01_easy)与 ORBSLAM2 算法的定位精度相当;在复杂场景中(V2_03_difficult)尤其是在特征点容易丢失的场景下(V1_02_medium),本文算法的定位精度比 ORBSLAM2 更好。这是因为在特征丢失场景和复杂环境下,特征匹配性能下降,光度信息在统一损失函数中的占比更大,有利于获取较好的优化结果。因此,本文算法能实现较好的定位结果,在复杂环境应用中也具有鲁棒性。

表1 三种算法的均方根误差

Table 1 RMSEs of three algorithms

unit: m

Image	Proposed algorithm	ORBSLAM2	LK
V1_01_easy	0.21841943	0.18432932	0.32365693
V1_02_medium	0.40135902	0.49316423	0.50254364
V1_03_difficult	0.466393203	0.44301304	0.62435742
V2_01_easy	0.174869403	0.17046982	0.36542354
V2_02_medium	0.509312352	0.43591642	0.71354231
V2_03_difficult	0.392569843	0.45330634	0.48636982

5.2 算法的运行时间测试

在 V1_02_medium 数据集上对特征点法、光流法和本文算法进行了运行时间测试。图8(a)、(b)为特征点法和光流法的运行时间测试结果,可以看出,特征点法单帧耗时较多,时间变化分布在50 ms附近,且耗时会因特征数量变化而出现较大波动,而光流法单帧耗时较少,时间变化分布在20 ms附近。图8(c)为本文算法的运行时间测试结果,整体上单帧耗时分布在25 ms附近,充分发挥了光流法耗时少的优点,部分图像单帧耗时较多,超过40 ms,这是联合优化时计算量变大的结果。图8(d)为特征点法、本文算法和光流法的耗时均值,分别为0.0477, 0.0296, 0.0209 s。由此可见,本文算法在定位精度超过光流法的基础上,保留了光流法耗时少的特点;在达到特征点法定位精度的基础上,耗时较特征点法减少了37.9%。

6 结 论

针对视觉里程计中特征点法耗时较多和光流法定位精度差的问题,结合光流和特征匹配构建了一个融合光流法和特征点法的紧耦合模型。首先采用光流法对每两帧间的位姿进行估计,大大提高了运行速度,通过增加帧间约束来进行帧间优化,从而降低了像素误差的影响;然后选取图像中的关键帧,在两关键帧之间对光流法的位姿进行累积,并联合特征匹配误差方程构建统一损失函数,这既抑制了累积误差,又减少了计算量。实验结果表明,所提算法在普通场景下能够保持与特征点法相当的定位精度,在特征缺失环境中的表现优于特征点法;同时,在保持定位精度的基础上提高了视觉里程计的运行速度。所提算法扬长避短,保留了光流法耗时少的优点,并充分发挥了特征点法定位精度高的特点,具

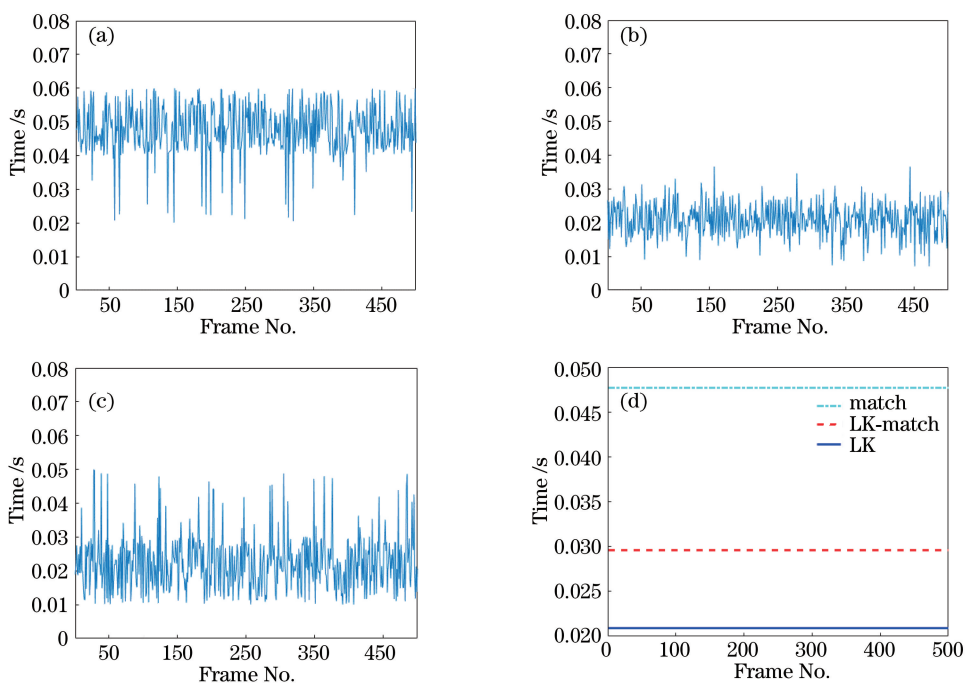


图 8 三种方法的耗时测试结果。(a)特征法;(b)光流法;(c)本文算法;(d)三种方法的时间均值

Fig. 8 Time consumption test results of three methods. (a) Feature method; (b) optical flow method; (c) proposed algorithm; (d) time average of three methods

有一定的实时性和鲁棒性。后续工作将通过融合惯性测量单元,将惯性测量单元测得的数据作为帧间约束并引入到联合优化方程中,以提高算法在高动态环境下的鲁棒性。

参 考 文 献

[1] Lin F C, Liu Y H, Zhou J F, et al. Optimization of visual odometry algorithm based on ORB feature[J]. Laser & Optoelectronics Progress, 2019, 56(21): 211507.

林付春, 刘宇红, 周进凡, 等. 基于 ORB 特征的视觉里程计算法优化[J]. 激光与光电子学进展, 2019, 56(21): 211507.

[2] Xing J W, Tian H F, Wang F. Study on object position and pose estimation method of monocular camera [J]. Navigation Positioning and Timing, 2019, 6(4): 71-77.

邢加伟, 田海峰, 王芳. 单目相机物体位姿估计方法研究[J]. 导航定位与授时, 2019, 6(4): 71-77.

[3] Hou Y H, Liu Y, Lü H L, et al. An autonomous navigation systems of UAVs based on binocular vision[J]. Journal of Tianjin University (Science and Technology), 2019, 52(12): 1262-1269.

侯永宏, 刘艳, 吕华龙, 等. 一种基于双目视觉的无人机自主导航系统[J]. 天津大学学报(自然科学与工程技术版), 2019, 52(12): 1262-1269.

[4] Opower H. Multiple view geometry in computer vision[J]. Optics and Lasers in Engineering, 2002, 37(1): 85-86.

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

[6] Mur-Artal R, Tardos 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.

[7] Rublee E, Rabaud V, Konolige K, et al. ORB: an efficient alternative to SIFT or SURF [C] // 2011 International Conference on Computer Vision, November 6-13, 2011, Barcelona, Spain. New York: IEEE, 2011: 2564-2571.

[8] Klein G, Murray D. Parallel tracking and mapping for small AR workspaces [C] // 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, November 13-16, 2007, Nara, Japan. New York: IEEE, 2007: 225-234.

[9] Guo R F, Jia R. Research on multi-information fusion target tracking algorithm based on LK optical flow method [J]. Modern Electronics Technique, 2019, 42(18): 55-59.

郭瑞峰, 贾榕. LK 光流法的多信息融合目标跟踪算法研究[J]. 现代电子技术, 2019, 42(18): 55-59.

- [10] Forster C, Pizzoli M, Scaramuzza D. SVO: Fast semi-direct monocular visual odometry [C] // 2014 IEEE International Conference on Robotics and Automation (ICRA), May 31- June 7, 2014, Hong Kong, China. New York: IEEE, 2014: 15-22.
- [11] Zhang G L, Yao E L, Lin Z L, et al. Fast binocular SLAM algorithm combining the direct method and the feature-based method[J]. Robot, 2017, 39(6): 879-888.
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.
- [12] Qi N X, Yang X G, Li X F, et al. Visual odometry algorithm based on ORB features and LK optical flow [J]. Chinese Journal of Scientific Instrument, 2018, 39(12): 216-227.
齐乃新, 杨小冈, 李小峰, 等. 基于 ORB 特征和 LK 光流的视觉里程计算法[J]. 仪器仪表学报, 2018, 39(12): 216-227.
- [13] Cadena C, Carlone L, Carrillo H, et al. Past, present, and future of simultaneous localization and mapping: toward the robust-perception age[J]. IEEE Transactions on Robotics, 2016, 32(6): 1309-1332.
- [14] Baker S, Matthews I. Lucas-Kanade 20 years on: a unifying framework [J]. International Journal of Computer Vision, 2004, 56(3): 221-255.