

一种基于点和线的视觉惯性SLAM算法

张 杰^{1,2}, 裴 东^{1,2}, 高文辉^{1,2}

¹西北师范大学物理与电子工程学院, 甘肃 兰州

²甘肃省智能信息技术与应用工程研究中心, 甘肃 兰州

收稿日期: 2021年11月6日; 录用日期: 2021年12月6日; 发布日期: 2021年12月13日

摘 要

传统的视觉惯性SLAM (Simultaneous Localization and Mapping)算法, 在弱纹理场景下, 往往存在定位精度差甚至失效的问题, 本文提出一种基于点和线特征的视觉惯性SLAM算法。该算法以开源的VINS-Mono (Monocular Visual-Inertial Systems)系统为基础, 在此基础上增加了线特征, 结合点线特征各自独有的特性, 并且提出一种新的数据选择策略, 减少了因为线特征的加入而增加的计算量, 保证了系统的实时性。实验采用开源数据集Euroc, 通过与其他开源算法做对比, 对本文算法进行评估, 实验结果表明了本文算法的有效性。

关键词

机器视觉, 弱纹理环境, 点线特征, 数据选择策略, 实时性

A Point and Line Based Visual Inertial SLAM Algorithm

Jie Zhang^{1,2}, Dong Pei^{1,2}, Wenhui Gao^{1,2}

¹College of Physics and Electronic Engineering, Northwest Normal University, Lanzhou Gansu

²Engineering Research Center of Gansu Province for Intelligent Information Technology and Application, Lanzhou Gansu

Received: Nov. 6th, 2021; accepted: Dec. 6th, 2021; published: Dec. 13th, 2021

Abstract

Traditional visual inertial SLAM (Simultaneous Localization and Mapping) algorithms often have poor localization accuracy or even failure in weak texture scenes. In this paper, a visual inertial SLAM algorithm based on point and line features is proposed. The algorithm is based on the open source VINS-Mono (Monocular Visual-Inertial Systems) system, which adds line features, com-

bines the unique features of point and line features, and proposes a new data selection strategy, which reduces the computational burden caused by line features. The real-time performance of the system is guaranteed. The experiment uses open source data set Euroc and evaluates the algorithm in this paper by comparing it with other open source algorithms. The experimental results show the effectiveness of the algorithm in this paper.

Keywords

Machine Vision, Low-Texture Environment, Point and Line Feature, Data Selection Strategy, Real-Time

Copyright © 2021 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

视觉惯性系统 VINS (Visual-Inertial System), 是融合相机和 IMU (Inertial Measurement Unit)数据实现 SLAM 的算法, 是机器人领域的一大关键技术。目前已有的视觉惯性 SLAM 系统, 主要采用特征点法进行跟踪, 首先从图像中选取特征点, 常用的特征点包括著名的 PTAM (Parallel Tracking And Mapping) [1]、SIFT (Scale Invariant Feature Transform) [2]、SURF (Speeded Up Robust Features) [3]、ORB (Oriented Fast and Rotated Brief) [4]等等。基于点特征的 SLAM 系统依赖于点特征的数量和质量, 当在走廊、窗户等弱纹理场景下, 往往检测不到足够数量的点特征, 因此会影响系统的精度, 甚至出现跟踪失败的情况。为了改善基于点特征 SLAM 系统的性能, 越来越多的人开始在点特征系统中加入线特征。Pumarola 等[5]在 ORB-SLAM [6]的基础上加入线特征, 提出了 PL-SLAM (Visual SLAM With Points and Lines), 并提出利用线特征进行系统初始化的方法, 是一种点线结合的单目视觉 SLAM 算法。HE Yijia 等[7]提出的基于点线特征结合的视觉惯导联合里程计 PL-VIO (Point-Line VIO)将线特征集成到优化框架中, 以图优化的方法实现了更高的精度。Gomez-Ojeda 等[8]在 SVO (Semi-Direct Monocular Visual Odometry) [9]的基础上提出了基于点线结合的半直接法单目视觉里程计 PLSVO (Point-Line SVO)。谢晓佳[10]则采用普吕克坐标参数化空间直线, 构建了一个点线综合的双目视觉 SLAM 系统。Qiang Fu 等[11]在 VINS-Mono [12]的基础上增加了线特征, 针对线特征的提取算法, 进行了隐参数调整并提出长度抑制策略, 改进最小二乘法, 设计出一种具有点和线特征的实时单目视觉惯性系统。

以上是目前比较流行的点线结合的 SLAM 系统, 线特征提取大多采用 LSD (Line Segment Detector) [13]算法, 但由于 LSD 算法, 运行较慢, 不适用于 SLAM 这种对实时性要求较高的系统。Cuneyt Akinlar 等[14]在 2011 年提出一种快速线段特征提取算法 EDlines, 其能够在线性时间内给出精确的结果, 并拥有比 LSD 检测算法更快的提取速度, 但精度与 LSD 算法持平。因此本文对 EDlines 算法进行改进, 在保证原有运行速度的基础上, 提高了算法精度; 线段匹配采用 LBD (Lind Band Descriptor) [15]描述子方案, 得到点、线特征数据后, 提出一种数据选择策略, 减少因为线特征的加入而增加的计算量, 保证了系统的实时性, 并以 VINS-Mono 系统为基础搭建了本文的 SLAM 系统。

2. 系统框架

本文 SLAM 系统共分为四大线程: 测量数据预处理、局部视觉 - 惯性 BA 与重定位、回环检测和全

局位姿图优化,如图1所示。首先对相机和IMU数据经行预处理,包括点线特征的提取与匹配、IMU预积分,得到点线特征和IMU预积分数据后进行数据选择,然后采用基于滑动窗口模型的非线性估计器,根据视觉约束、IMU约束以及回环约束构建联合优化函数并求解出滑动窗口内所有帧间位置、速度、旋转及偏置等,通过DboW词袋模型和BRIEF描述子经行回环检测,建立滑动窗口与回环候选帧之间的链接,最后是为了确保基于重定位的结果,对过去的位姿进行全局一致配置。该系统以开源的VINS-Mono系统架构为基础,在此基础上增加了线特征,并结合点线特征各自独有的特性。本研究将着重阐述关于线特征的主要算法。

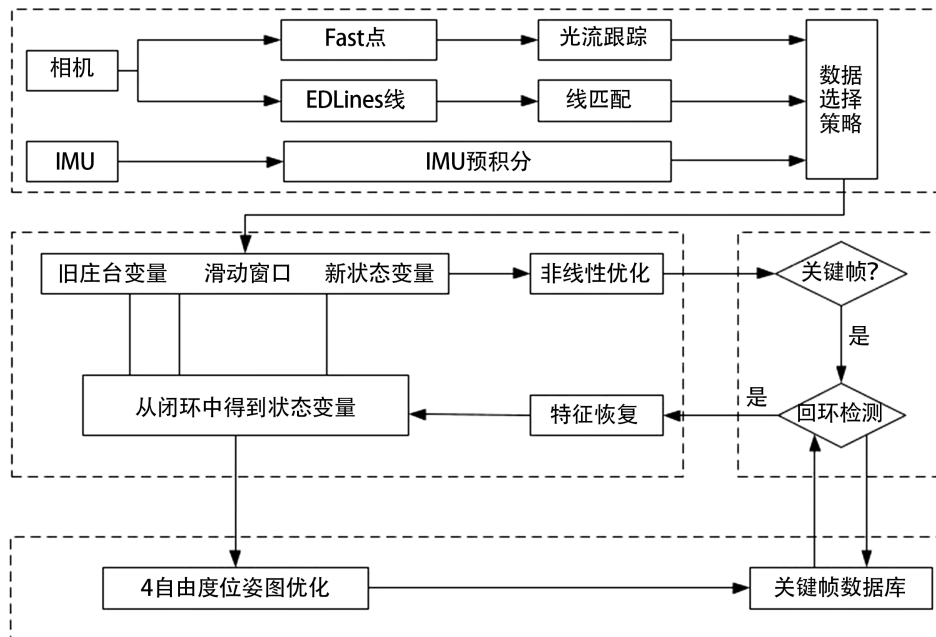


Figure 1. System frame
图1. 系统框架图

3. 线特征提取

3.1. EDlines 直线提取算法

EDLines 是一种高效且快速的线段检测算法,无需参数调整只需为所有类型的图像运行一组默认参数即可,是一种快速、无参数的线段检测器,详细原理如下:首先提取边缘,具体分为以下4步:1)将灰度图通过滤波器进行处理后输出平滑图像;2)计算图像中各像素的梯度方向、梯度值;3)将梯度图中梯度值的峰值点设为锚点;4)连接锚点并绘制成边。

将上一步由锚点连接而成的边缘,利用最小二乘线拟合的方法进行线段的拟合提取,其原理是偏差绝对值之和最小:

$$\min \sum_{i=1}^m |\delta_i| = \sum_{i=1}^m |\varphi(x_i) - y_i| \quad (1)$$

其中 x_i, y_i 是像素点坐标, $y = \varphi(x)$ 是近似曲线, $\delta_i = \varphi(x_i) - y_i$ 是近似曲线在点 (x_i, y_i) 出的偏差。

得到的线段需要经过评估才能被认定为线段,评估的方法采用 Helmholtz 原理,通过计算 NFA (Number of False Alarms)来确定提取的线段是否有效,具体原理为:对于一段由 n 个锚点组成的线段,其

中 k 个点的梯度方向与线段的方向垂直, 若其 NFA 计算的结果小于 1, 则认为线段有效, 否则判定为无效线段, 将之剔除。

$$\text{NFA}(n, k) = N^4 \sum_{i=k}^n p^i (1-p)^{n-i} \quad (2)$$

3.2. EDlines 算法改进

由于提取出的线特征存在多种干扰点, 最小二乘法拟合直线时, 因其需要尽可能包含全部局外点, 会造成直线与局内点存在偏差, 所以本文使用随机抽样一致性(RANSAC) [16]算法解决线特征含有高噪声点的问题, RANSAC 算法的原理是通过局内点的数据得出模型, 拟合出的直线包含局内点的概率高。

RANSAC 算法拟合像素链的步骤如下: 首先假定单个需要拟合的像素链中有 N 个像素点, 随机选择两个像素点, 通过选择的两个点计算出两点所表示直线的模型方程 $y = ax + b$ 。然后设定阈值 T , 计算其余 $N - 2$ 个像素点与模型方程的距离, 统计出距离小于阈值的点的数量 M , 将这些点称为内点。最后按以上步骤迭代 K 次, M 值最大时所得到的直线即为目标直线。

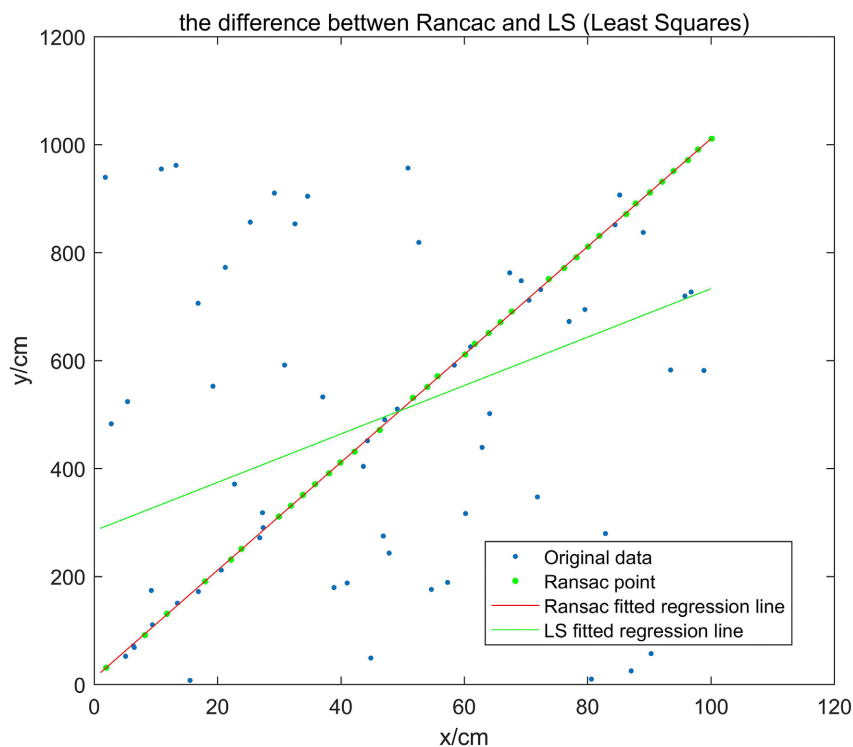


Figure 2. Comparison of linear fitting methods

图 2. 直线拟合算法对比

使用 RANSAC 算法拟合直线, 需要找出合适的算法参数, 提高检测的准确度与合理概率。图 2 所示为一组数据分别使用不同的算法作对比, 其中 RANSAC 的参数设定: 模型阈值 $T = 2$; 迭代次数 $K = 25$ 。在图 2 中就可以明显看出直线位置, 但最小二乘法生成的结果却是错误的, 这是由于图中存在较多局外点, 当只有部分的数据符合模型时, 最小二乘法拟合作用较差, 无法使用, RANSAC 算法可以有效解决此弊端。

由于噪声的存在, 最小二乘法得到的拟合结果与预期结果相差甚远, 使用 RANSAC 算法在生成的边

缘图中拟合直线，减少了局外噪声点对结果产生干扰，在实际 SLAM 系统中经过多次测试发现，当 $T=3, K=28$ 时直线拟合效果最好，运行效果图如图 3 所示，图 3(a) 是改进前的运行效果图，图 3(b) 是改进后的运行效果图，改进后的 EDLines 算法，能够剔除低图像梯度的锚点，减少许多无效的短线段，提取出更多有效的长线段。

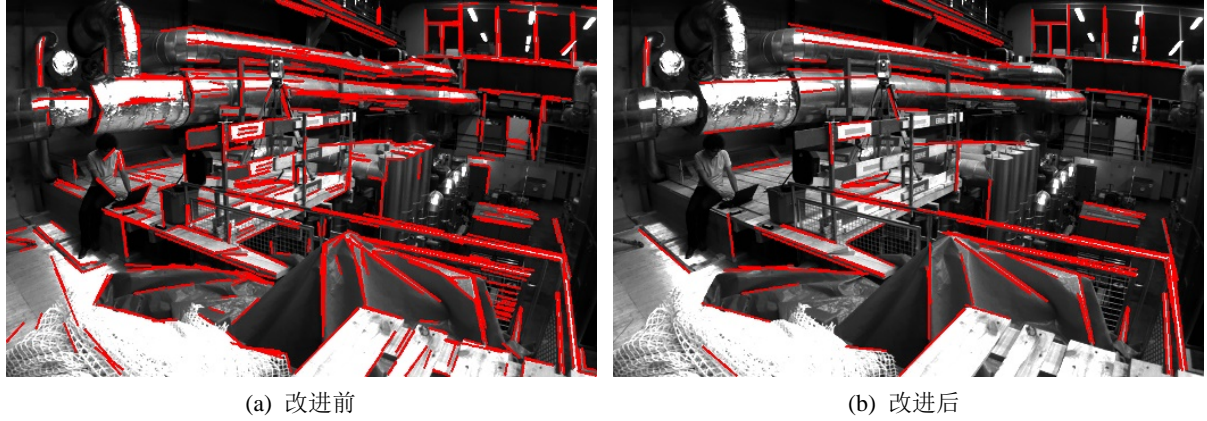


Figure 3. Comparison of actual effects
图 3. 实际效果对比

4. 线特征匹配

提取到线特征之后要对线特征进行描述并加以匹配。线特征的描述和匹配方法有 Zhiheng Wang 等人提出的 MSLD 描述子[17]，该描述子具有旋转、光照不变性等优点。Lilian Zhang 等人在此基础上提出了 LBD 描述子[15]，LBD 描述子引入了全局、局部的高斯权重系数，相对于 MSLD 具有更好的匹配效果，同时拥有更快的计算速度。

LBD 描述子首先会在线段处建立一个矩形的线段支持域 LSR (Line Support Region)。利用条带 $\{B_1; B_2; \dots; B_m\}$ 形式表示，用 m 表示 LSR 条带数目， w 表示条带像素宽度。在这个矩形支持域内，定义直线所朝的方向 d_L ，与直线方向顺时针垂直的方向 d_{\perp} ，这两种方向构成局部 2D 坐标系，局部坐标系的原点是该线段的中点。将 LSR 中的每个像素的梯度投影到上述局部坐标系中。

$$g' = (g^T \cdot d_{\perp}, g^T \cdot d_L)^T \triangleq (g'_{d_{\perp}}, g'_{d_L})^T \quad (3)$$

g 是图片坐标系中的像素梯度， g' 是局部坐标系中的像素梯度。引入全局高斯函数 f_g 来缓和在线段垂直方向上微小变化的敏感度，以及引入局部高斯函数 f_l 降低边缘效应，避免描述符的突变。计算每个条带 B_j 对应的特征向量 BD_j ，将所有特征向量合并，形成 LBD 描述子：

$$\text{LBD} = (BD_1^T, BD_2^T, \dots, BD_m^T)^T \quad (4)$$

将 B_j 与相邻条带每一行的局部梯度进行分别求和。对于 k 行，有：

$$\begin{aligned} v1_j^k &= \lambda \sum_{g'_{d_{\perp}} > 0} g'_{d_{\perp}}, v2_j^k = \lambda \sum_{g'_{d_{\perp}} < 0} g'_{d_{\perp}} \\ v3_j^k &= \lambda \sum_{g'_{d_L} > 0} g'_{d_L}, v4_j^k = \lambda \sum_{g'_{d_L} < 0} g'_{d_L} \end{aligned} \quad (5)$$

其中， $\lambda = f_g(k)f_l(k)$ 为高斯系数。每行的和放在一起，可以构成 BD_j 对应的特征描述矩阵：

$$BDM_j = \begin{pmatrix} v1_j^1 & v1_j^2 & \cdots & v1_j^n \\ v2_j^1 & v2_j^2 & \cdots & v2_j^n \\ v3_j^1 & v3_j^2 & \cdots & v3_j^n \\ v4_j^1 & v4_j^2 & \cdots & v4_j^n \end{pmatrix} \in R^{4 \times n} \quad (6)$$

$$n = \begin{cases} 2w, j = 1, m \\ 3w, \text{else} \end{cases}$$

最后对 BDM_j 的均值向量 M_j^T 和标准方差 S_j^T 分别进行归一化, 得到特征向量:

$$BD_j = (M_j^T, S_j^T)^T \in R^s \quad (7)$$

最后得到 LBD 特征:

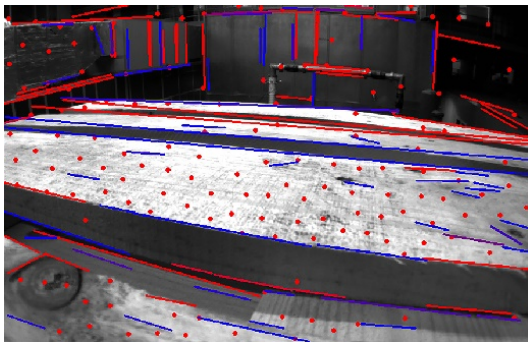
$$LBD = (M_1^T, S_1^T, M_2^T, S_2^T, \dots, M_m^T, S_m^T)^T \in R^{Sm} \quad (8)$$

5. 数据选择策略

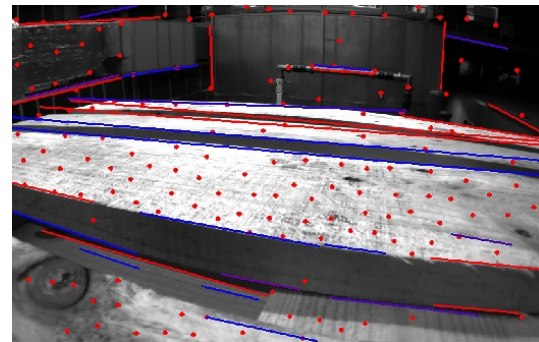
线特征的加入, 无疑会增大计算量, 降低系统实时性。为提高系统的实时性, 文献[18]针对点线特征 SLAM 算法中, 存在图像局部密集区域提取大量相似线特征、同一直线上的线段过度分割等弊端, 利用梯度密度滤波器剔除图像中特征密集区域, 降低了线特征的误匹配率加速了特征提取过程; 文献[19]提出一种最优候选帧的关键帧选取策略, 首先将疑似关键帧作为候选帧, 并从多个方面评估选出最优帧作为关键帧, 提高了整个系统的鲁棒性。文献[20]通过引入线段分组策略, 合并潜在的同源线段、生成高质量长线段, 提高了系统精度和实时性。为了降低引入线特征后的计算量, 本节提出一种针对点特征与线特征的数据选择策略, 对线特征的数量设置阈值, 以点特征为主, 线特征为辅, 当点特征较多时, 将线特征阈值减小, 取少量线特征; 当遇到弱纹理情况, 对应点特征较少时, 相应增大线特征阈值, 进而提取出更多线特征, 如此既保证了精度, 亦保证了系统实时性。在实际系统中, 阈值设置如表 1 所示, 点特征阈值设置为固定数值 150, 线特征数目阈值上下限分别为 100、20, 其数目随点特征的数目变化而变化。

Table 1. New data selection strategy
表 1. 新数据选择策略

线特征阈值	点特征
20	>120
60	50~120
100	<50



(a) 点特征较多改进前



(b) 点特征较多改进后

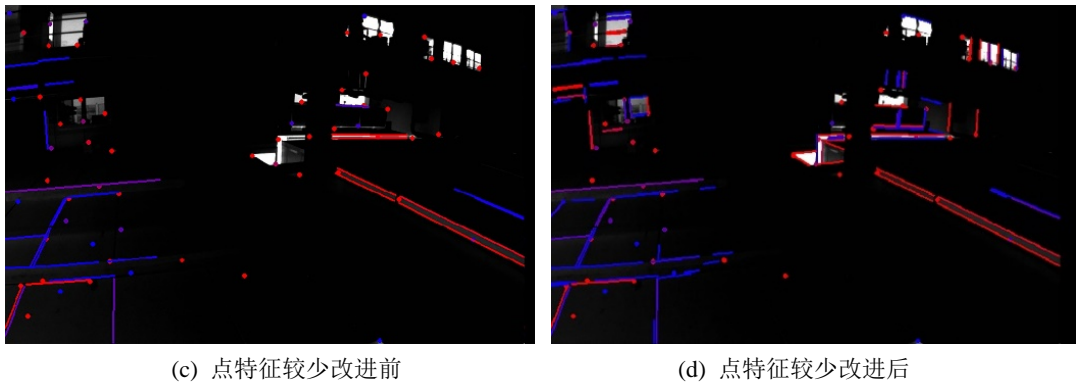


Figure 4. Improved renderings
图 4. 改进效果

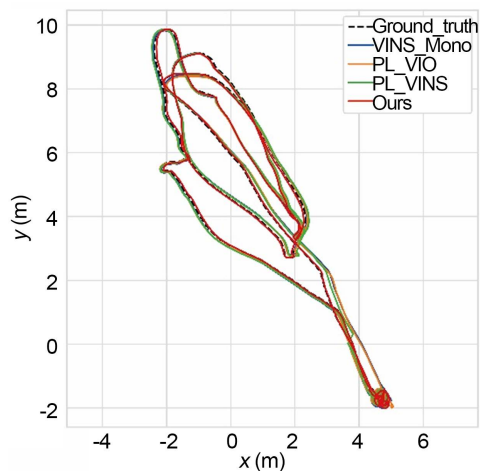
图 4 展示了点特征较多和较少, 这两种情况下, 改进前后的实际运行效果图。其中图 4(a)、图 4(b) 是点特征较多时改进前后效果图, 图 4(a)是改进前, 此时追踪到的点线特征数目较多, 但为了系统的实时性, 利用数据选择策略, 保持点特征阈值不变, 减小线特征阈值, 因此将减少提取到的线特征, 改进后如图 4(b)所示; 图 4(c)、图 4(d)是点特征较少时改进前后效果图, 图 4(c)是改进前, 此时图中光线较暗, 属于典型弱纹理场景, 点线特征较少, 此时增大线特征阈值, 获得更多的线特征, 提高系统精度, 改进后如图 4(d)所示。利用此策略能够在保证系统精度的前提下, 同样保证系统实时性。

6. 实验结果分析

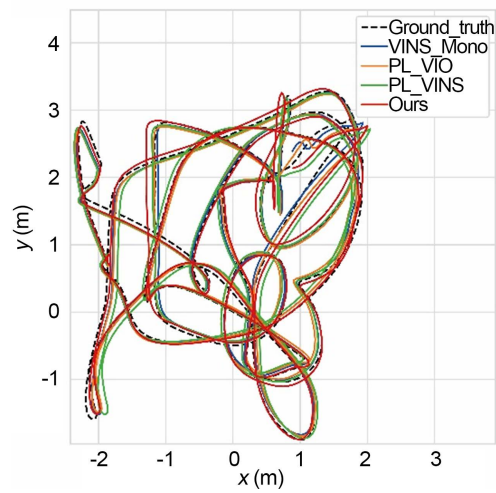
为了评估本文算法的性能, 使用标准视觉惯性数据集 Euroc [21], 该数据集集中有许多弱纹理场景。因为本文算法以 VINS-Mono 系统为基础开发, 所以将本文算法与 VINS-Mono 及以 VINS-Mono 为基础开发的点线特征系统 PL-VIO、PL-VINS 在同一台机器上进行对比。

测试平台为 CPU 为 Intel i7-8700@3.2 GHz, 8 GB 内存, 系统为 ubuntu18.04。

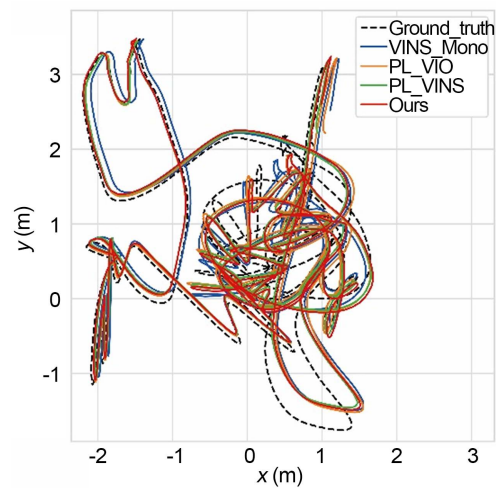
利用 SLAM 精度评定工具 EVO (Evaluation of Odometry), 对比内容是多种 SLAM 算法的均方根误差 (Root Mean Square Error, RMSE)与轨迹。轨迹对比如图 5 所示, 以 MH-02-easy、V1-02-medium、V1-03-difficult、V2-03-difficult 数据集为例, SLAM 算法轨迹与真实轨迹越接近, 精度越高, RMSE 则选择 Euroc 全部的数据集, RMSE 值越小说明系统的整体定位精度越高。



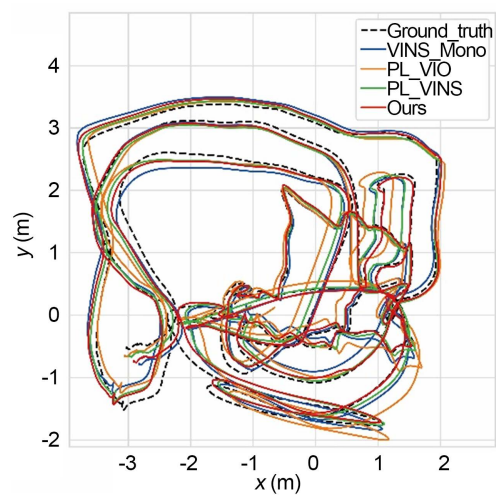
(a) MH-02-easy



(b) V1-02-medium



(c) V1-03-difficult



(d) V2-03-difficult

Figure 5. Running track

图 5. 运行轨迹图

通过与其他几种算法的轨迹对比,结果如图5所示,本文算法在不同难度数据集上均有良好的表现,由于MH-02-easy这个数据集比较简单,本文算法轨迹与其它3种SLAM算法轨迹基本一致,但是在V1-02-medium、V1-03-difficult、V1-03-difficult这三个数据集上,本文算法轨迹与真实轨迹更接近,精度要高于其它3种算法。

Table 2. RMSE comparison of various SLAM algorithms
表 2. 多种 SLAM 算法的 RMSE 对比

数据集	VINS-Mono	PL-VIO	PL-VINS	Ours
MH_01_easy	0.1707	0.1516	0.1415	0.1361
MH_02_easy	0.1918	0.1732	0.1684	0.1592
MH_03_medium	0.2257	0.1943	0.1901	0.1823
MH_04_difficult	0.3536	0.3154	0.2933	0.3014
MH_05_difficult	0.3096	0.2765	0.2684	0.2443
V1_01_easy	0.1165	0.0912	0.0816	0.0596
V1_02_medium	0.1361	0.1132	0.1056	0.0893
V1_03_difficult	0.2125	0.1913	0.1839	0.1765
V2_01_easy	0.0983	0.0958	0.0844	0.0731
V2_02_medium	0.1863	0.1619	0.1511	0.1455
V2_02_difficult	0.3134	0.2890	0.2749	0.2681

表2表明,将使用改进的EDLines算法提取到的线特征整合到视觉惯性里程计中可以有效提高SLAM系统运动估计精度。本文提出视觉惯性SLAM系统,基于改进的EDLines算法,并通过点特征的数量对线特征设置不同阈值,在除MH_04_difficult以外的数据集中都得到了更高的精度。

7. 结论

本文以VINS-Mono为基础,提出了一种点线结合的视觉惯性SLAM系统,在原本点特征的基础上,增加线特征,并对线特征提取算法进行改进,得到了更高的精度,并提出了一种新的数据选择策略,既保证了精度,也保证了系统运行的实时性,解决了传统视觉惯性里程计在弱纹理场景下,由于特征信息缺失,导致定位精度差甚至失效的问题。并在开源视觉惯性数据集EuRoc上的实验结果表明本文算法能够获取更多有效的待匹配线段,提高了线特征匹配的准确度,同时具有很高的实时性,相比其它算法更具优越性。本文只是将线特征加入到后端优化中,基于此研究方向,可以继续改进算法,构建线特征词袋库,将线特征加入回环检测中,进一步提高精度。

致 谢

首先感谢香港科技大学科研团队开源了VINS-Mono系统,让我学习到了一种非常优秀的视觉惯性SLAM框架,本文就是以VINS-Mono为基础,设计的点线结合的视觉惯性SLAM系统;感谢我的导师,本文是在导师耐心的指导下完成的,每一步都倾注了导师的心血和汗水;感谢所有引用文献的作者,正是他们的文章让我有了新的构思和设想;最后感谢在本文写作过程中所有提供帮助的老师、同学、朋友,是你们的帮助让我顺利完成文章,谢谢你们!

参考文献

- [1] Klein, G. and Murray, D. (2007) Parallel Tracking and Mapping for Small AR Workspaces. 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, Nara, 13-16 November 2007, 225-234. <https://doi.org/10.1109/ISMAR.2007.4538852>
- [2] Lowe, D.G. (2004) Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, **60**, 91-110. <https://doi.org/10.1023/B:VISI.0000029664.99615.94>
- [3] Bay, H., Ess, A., Tuytelaars, T., et al. (2008) Speeded-Up Robust Features. *Computer Vision and Image Understanding*, **110**, 404-417. <https://doi.org/10.1016/j.cviu.2007.09.014>
- [4] Rublee, E., Rabaud, V., Konolige, K., et al. (2011) ORB: An Efficient Alternative to SIFT or SURF. 2011 International Conference on Computer Vision, Barcelona, 6-13 November 2011, 2564-2571. <https://doi.org/10.1109/ICCV.2011.6126544>
- [5] Pumarola, A., Vakhitov, A., Agudo, A., et al. (2017) PL-SLAM: Real-Time Monocular Visual SLAM with Points and Lines. 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May-3 June 2017, 4503-4508. <https://doi.org/10.1109/ICRA.2017.7989522>
- [6] Mur-Artal, R., Montiel, J.M.M. and Tardos, J.D. (2015) ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Transactions on Robotics*, **31**, 1147-1163. <https://doi.org/10.1109/TRO.2015.2463671>
- [7] He, Y., Zhao, J., Guo, Y., et al. (2018) PL-VIO: Tightly-Coupled Monocular Visual-Inertial Odometry Using Point and Line Features. *Sensors (Basel)*, **18**, 1159. <https://doi.org/10.3390/s18041159>
- [8] Gomez-Ojeda, R., Briales, J. and Gonzalez-Jimenez, J. (2016) PL-SVO: Semi-Direct Monocular Visual Odometry by Combining Points and Line Segments. 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Daejeon, 9-14 October 2016, 4211-4216. <https://doi.org/10.1109/IROS.2016.7759620>
- [9] Forster, C., Pizzoli, M. and Scaramuzza, D. (2014) SVO: Fast Semi-Direct Monocular Visual Odometry. *IEEE International Conference on Robotics & Automation*, Hong Kong, 31 May-7 June, 2014, 15-22. <https://doi.org/10.1109/ICRA.2014.6906584>
- [10] 谢晓佳. 基于点线综合特征的双目视觉 SLAM 方法[D]: [硕士学位论文]. 杭州: 浙江大学, 2017.
- [11] Fu, Q., Wang, J., Yu, H., et al. (2020) PL-VINS: Real-Time Monocular Visual-Inertial SLAM with Point and Line.
- [12] Qin, T., Li, P. and Shen, S. (2018) VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator. *IEEE Transactions on Robotics*, **34**, 1004-1020. <https://doi.org/10.1109/TRO.2018.2853729>
- [13] Von Gioi, R.G., Jakubowicz, J., Morel, J.M., et al. (2010) LSD: A Fast Line Segment Detector with a False Detection Control. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **32**, 722-732. <https://doi.org/10.1109/TPAMI.2008.300>
- [14] Akinlar, C. and Topal, C. (2011) EDLines: A Real-Time Line Segment Detector with a False Detection Control. *Pattern Recognition Letters*, **32**, 1633-1642. <https://doi.org/10.1016/j.patrec.2011.06.001>
- [15] Zhang, L. and Koch, R. (2013) An Efficient and Robust Line Segment Matching Approach Based on LBD Descriptor and Pairwise Geometric Consistency. *Journal of Visual Communication & Image representation*, **24**, 794-805. <https://doi.org/10.1016/j.jvcir.2013.05.006>
- [16] Chum, O., Matas, J. and Kittler, J. (2003) Locally Optimized RANSAC. In: *Pattern Recognition*, Lecture Notes in Computer Science, Vol. 2781, Springer, Berlin, 236-243. https://doi.org/10.1007/978-3-540-45243-0_31
- [17] 林利蒙, 王梅. 改进点线特征的双目视觉 SLAM 算法[J]. 计算机测量与控制, 2019, 27(9): 156-162.
- [18] Lin, L.M. and Wang, M. (2019) An Improved Stereo SLAM System through the Combination of Points and Line Segments. *Computer Measurement & Control*, **27**, 156-162.
- [19] 王玉良. 基于 RGBD 的室内移动机器人定位算法研究[D]: [硕士学位论文]. 马鞍山: 安徽工业大学, 2018.
- [20] 方琪, 王晓华, 苏杰. 基于分组策略的点线特征融合同步定位与地图构建算法[J]. 激光与光电子学进展, 2021, 58(14): 9.
- [21] Burri, M., Nikolic, J., Gohl, P., et al. (2016) The EuRoC Micro Aerial Vehicle Datasets. *The International Journal of Robotics Research*, **35**, 1157-1163. <https://doi.org/10.1177/0278364915620033>