

基于线流特征的视觉惯性SLAM算法

纪 周

上海理工大学光电信息与计算机工程学院, 上海

收稿日期: 2025年2月28日; 录用日期: 2025年3月20日; 发布日期: 2025年3月31日

摘 要

本文研究了一种基于点和线特征的高效视觉惯性同时定位与建图(SLAM)方法。通过短线融合、线特征均匀分布和自适应阈值提取来改进传统的线检测模型, 以获得用于构建SLAM约束的高质量线特征。基于灰度不变性假设和共线性约束, 提出了一种线光流跟踪方法。实验结果表明, 算法提高了线特征检测与匹配的效率 and 定位精度。

关键词

同时定位与建图(SLAM), 自适应, 灰度不变性假设, 光流跟踪

Visual Inertia SLAM Algorithm Based on Point Line Flow Features

Zhou Ji

School of Optoelectronic Information and Computer Engineering, University of Shanghai for Science and Technology, Shanghai

Received: Feb. 28th, 2025; accepted: Mar. 20th, 2025; published: Mar. 31st, 2025

Abstract

This paper investigates an efficient visual inertial simultaneous localization and mapping (SLAM) method based on point and line features. Improve traditional line detection models by short-term fusion, uniform distribution of line features, and adaptive threshold extraction to obtain high-quality line features for constructing SLAM constraints. A line optical flow tracking method is proposed based on the assumption of grayscale invariance and collinearity constraints. The experimental results show that the algorithm improves the efficiency and positioning accuracy of line feature detection and matching.

Keywords

Simultaneous Localization and Mapping (SLAM), Adaptive, Gray Scale Invariance Assumption,

Optical Flow Tracking

Copyright © 2025 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

同步定位和绘图(SLAM)技术[1]在自动驾驶、无人机、机器人和增强现实中至关重要。不同传感器之间的内在特性使得优势互补的多传感器融合 SLAM 技术成为主导的研究方向。视觉惯性 SLAM 导航系统 (VINS)相对于其他多传感器融合方案具有低成本、轻量化等优点,受到了研究人员的广泛关注[1]-[3]。这些基于点的 VINS 方法难以在一些人工场景中找到足够数量的可靠点特征,例如稀疏或重复纹理,最终导致整个系统位姿估计精度降低甚至失效[4]。人工场景中有丰富的线特征,一维线特征比零维点特征[3]包含更多的信息[5]。因此,当场景中没有足够的点特征时,线特征[6]-[10]是很好的补充。虽然线特征可以提高 SLAM 在挑战性环境下的精度和鲁棒性[4],但其缺点是线特征比点特征更难跟踪,这使得基于线的 SLAM 方法不如基于点的 SLAM 方法流行。主要存在三个问题:错误匹配、遮挡[11]和实时性。重复纹理下线段的错误匹配问题严重,线段提取容易受到遮挡或外观模糊的影响。线流是打破当前 SLAM 中线特征瓶颈的较好方向。利用时空相干性的线流可以减少帧间的错误匹配问题。此外,没有线段描述的线流可以提高线特征跟踪的实时性。然而,这一领域的工作做得很少。由此提出了一种可以高效完成一帧检测和多帧跟踪的线光流,以解决线特征匹配中的误差匹配和耗时问题。

2. 系统概述

所提出的单目视觉惯性 SLAM 系统概述如图 1 所示。算法的基本结构类似于 VINS-Mono 和 PL-VINS [9]。侧重于通过一种新颖的线光流跟踪算法和一种构造线特征残差的新方法来提高最先进算法的效率和精度。需要注意的是,与基于 VINS-Mono 的其他点线融合 VINS,如 PL-VIO [12]、PL-VINS、PLF-VINS 相比,算法摒弃了耗时的 LSD [4]和 LBD [9]组合,采用了改进版的 EDLines [13]检测和新的线光流跟踪方法,大大提高了运行速度。对于线特征的测量模型,构建了残差,避免了遮挡和短线匹配的情况。

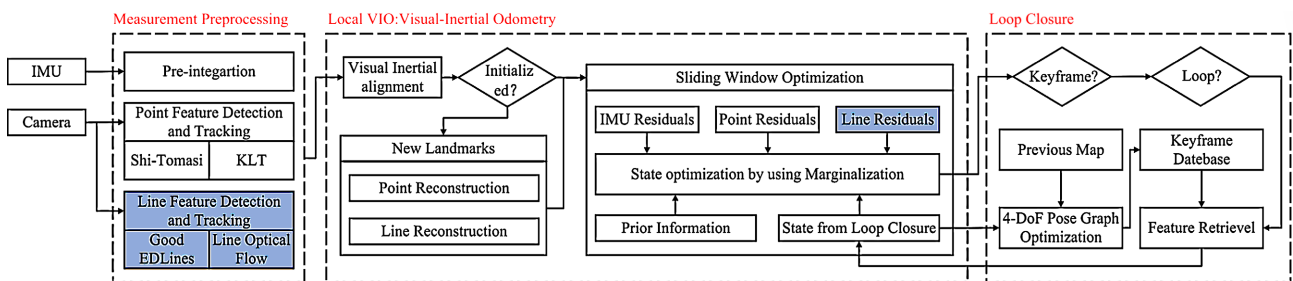


Figure 1. System framework diagram

图 1. 系统框架图

3. 线特征测量模型

3.1. 直线表示

一条线在空间中有四个自由度。本文使用了两种形式的线表示。plucker 表示由 6 个参数组成, 分别

用于初始化、变换和投影。标准正交表示由 4 个参数组成，用于优化。

plucker 表示：三维空间线可以表示为：

$$\mathcal{L} = (\mathbf{n}^T, \mathbf{v}^T)^T \quad (1)$$

式中， $\mathbf{v} \in \mathbb{R}^3$ 表示直线方向向量， $\mathbf{n} \in \mathbb{R}^3$ 表示由 L 和摄像机坐标系原点确定的平面 π 的法向量。

标准正交表示：plucker 坐标虽然可以简洁地表示变换过程，但由于其过度参数化，不容易直接优化。因此，使用由四个参数组成的标准正交表示来进行优化。三维空间线 L 可以表示为：

$$\mathcal{L} = (\mathbf{U}, \mathbf{W}) \quad (2)$$

其中 $\mathbf{U} \in SO(3)$ ， $\mathbf{W} \in SO(2)$ 。plucker 坐标系到正交坐标系的方程可以表示为：

$$\mathbf{U} = [\phi_1 \quad \phi_2 \quad \phi_3] = \begin{bmatrix} \frac{\mathbf{n}}{\|\mathbf{n}\|} & \frac{\mathbf{v}}{\|\mathbf{v}\|} & \frac{\mathbf{n} \times \mathbf{v}}{\|\mathbf{n} \times \mathbf{v}\|} \end{bmatrix} \quad (3)$$

$$\mathbf{W} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} = \frac{1}{\sqrt{\|\mathbf{n}\|^2 + \|\mathbf{v}\|^2}} \begin{bmatrix} \|\mathbf{n}\| & -\|\mathbf{v}\| \\ \|\mathbf{v}\| & \|\mathbf{n}\| \end{bmatrix} \quad (4)$$

因此，标准正交表示也可以表示为：

$$\mathcal{O} = (\phi_1 \quad \phi_2 \quad \phi_3 \quad \theta). \quad (5)$$

3.2. 观测模型

在先前的研究中，如 PL-VIO 和 PL-VINS，线重投影误差是由观测线端点与重投影线端点之间的距离来构建的。残差的大小取决于线端点的精度。然而，在不同视角下，空间中的线特征在相机图像平面中观察到的线的长度是不一致的。此外，当线受到其他扰动(如遮挡)时，线特征检测方法可能会将长线划分为几个短线段。这些会导致端点位置发生显著变化，进而影响残差值。因此，提出了一种简洁的方法来构建独立于线段端点的残差。

定义平面 $\tilde{\pi}$ 和平面 π ，在相机坐标系的原点 C_i 和归一化平面上的观测线之间形成平面 $\tilde{\pi}$ ，在摄像机坐标系的原点 C_i 与归一化平面上的投影线之间形成平面 π 。直线重投影误差可以定义为：

$$r(\mathbf{z}_L^i, \chi) = 1 - \frac{\tilde{\mathbf{n}}^{c_i} \cdot \mathbf{n}^{c_i}}{\|\tilde{\mathbf{n}}^{c_i}\| \|\mathbf{n}^{c_i}\|} \quad (6)$$

其中 $\tilde{\mathbf{n}}^{c_i}$ 和 $\mathbf{n}^{c_i}$ 分别表示平面 $\tilde{\pi}$ 和平面 π 的法向量， χ 为滑动窗口中的完整状态变量， $\mathbf{z}_L^i$ 表示第 i 个相机帧观察到的线特征 L。对应的雅可比矩阵 $\mathbf{J}_n$ 可由链式法则得到：

$$\mathbf{J}_n = \frac{\partial \mathbf{r}_n}{\partial \mathbf{n}^{c_i}} \frac{\partial \mathbf{n}^{c_i}}{\partial \mathcal{L}^{c_i}} \left[\frac{\partial \mathcal{L}^{c_i}}{\partial \delta x^i} \quad \frac{\partial \mathcal{L}^{c_i}}{\partial \mathcal{L}^w} \frac{\partial \mathcal{L}^w}{\partial \delta \mathcal{O}} \right] \quad (7)$$

$$\frac{\partial \mathbf{r}_n}{\partial \mathbf{n}^{c_i}} = \left[\frac{\partial \mathbf{r}_n}{\partial \mathbf{n}_1^{c_i}} \quad \frac{\partial \mathbf{r}_n}{\partial \mathbf{n}_2^{c_i}} \quad \frac{\partial \mathbf{r}_n}{\partial \mathbf{n}_3^{c_i}} \right]^T = \begin{bmatrix} n_1^{c_i} \frac{\tilde{\mathbf{n}}^{c_i} \cdot \mathbf{n}^{c_i}}{\|\tilde{\mathbf{n}}^{c_i}\|^{\frac{3}{2}} \|\mathbf{n}^{c_i}\|^{\frac{3}{2}}} - \frac{\tilde{n}_1^{c_i}}{\|\tilde{\mathbf{n}}^{c_i}\| \|\mathbf{n}^{c_i}\|} \\ n_2^{c_i} \frac{\tilde{\mathbf{n}}^{c_i} \cdot \mathbf{n}^{c_i}}{\|\tilde{\mathbf{n}}^{c_i}\|^{\frac{3}{2}} \|\mathbf{n}^{c_i}\|^{\frac{3}{2}}} - \frac{\tilde{n}_2^{c_i}}{\|\tilde{\mathbf{n}}^{c_i}\| \|\mathbf{n}^{c_i}\|} \\ n_3^{c_i} \frac{\tilde{\mathbf{n}}^{c_i} \cdot \mathbf{n}^{c_i}}{\|\tilde{\mathbf{n}}^{c_i}\|^{\frac{3}{2}} \|\mathbf{n}^{c_i}\|^{\frac{3}{2}}} - \frac{\tilde{n}_3^{c_i}}{\|\tilde{\mathbf{n}}^{c_i}\| \|\mathbf{n}^{c_i}\|} \end{bmatrix} \quad (8)$$

4. 线特征检测与跟踪模型

4.1. 改进的 EDLines 检测方法

选择 EDLines 作为线段检测的基本算法，该算法具有丰富的可调参数。然而，原始 EDLines 检测到的片段在空间分布上并不适合 SLAM 约束。因此，提出了一种改进的 EDLines 检测方法。对三个部分做了改进分别是：短线融合、线特征均匀分布和自适应阈值提取。

短线融合：由于部分遮挡经常将一条线分割成碎片，图像噪声可能导致线端点[11]不稳定，在这种情况下，EDLines 可能会将一条完整的线提取成几条短线。这可能会使线流跟踪和优化变得不稳定，所以需要合并短线。判断标准：两条线共线，两条线端点之间的距离足够小。

线特征均匀分布：特征在空间中的分布越均匀，特征匹配就越能准确地表示空间中的几何关系。在这种方法中，在当前图像中检测到的线的位置添加了一个宽的掩码，不在掩码上提取新的线。

自适应阈值提取：在上述两种策略之后，对提取的线条进行计数。假设提取的行数小于事先设定的阈值。在纹理较弱、灰度变化较小的情况下，通过降低 EDLines 的梯度阈值参数，再次进行线特征提取，获得足够的线特征。

4.2. 光流跟踪

线流模型基于灰度不变性理论。在时刻 t 位于 (u, v) 的像素在时刻 $t + dt$ 移动到 $(u + du, v + dv)$ 。可以将这个像素的灰度值定义为：

$$I(u + du, v + dv, t + dt) = I(u, v, t) \quad (9)$$

式(1)左侧的泰勒展开式为：

$$I(u + du, v + dv, t + dt) \approx I(u, v, t) + \frac{\partial I}{\partial u} du + \frac{\partial I}{\partial v} dv + \frac{\partial I}{\partial t} dt. \quad (10)$$

合并式(1)和式(2)：

$$\frac{\partial I}{\partial u} \frac{du}{dt} + \frac{\partial I}{\partial v} \frac{dv}{dt} = -\frac{\partial I}{\partial t} \quad (11)$$

其中 $\frac{\partial I}{\partial u}$ 和 $\frac{\partial I}{\partial v}$ ，记为 I_u 和 I_v ，分别表示该点图像在 u 和 v 方向上的灰度梯度。 $\frac{\partial I}{\partial t}$ 记为 I_t ，表示图像灰度值随时间的变化量。化简为

$$\begin{bmatrix} I_u & I_v \end{bmatrix} \begin{bmatrix} \frac{du}{dt} \\ \frac{dv}{dt} \end{bmatrix} = -I_t \quad (12)$$

式(4)在图像中的任何像素点上都有效，因此它在线段上成立，而直线上的点也必须满足共线性约束。定义这条线上的点的集合：

$$l = \{(u_1, v_1), (u_2, v_2), \dots, (u_n, v_n), \dots, (u_m, v_m)\} \quad (13)$$

其中 (u_1, v_1) 和 (u_m, v_m) 分别表示直线的起点和终点。 l_n 表示 (u_1, v_1) 和 (u_m, v_m) 之间的欧几里得距离。 l' 表示 l 的下一帧位置，连续两帧之间的直线对应点之间的关系为：

$$\begin{cases} u'_n = u_n + g_1 + l'_n \cos(\alpha + g_3) - l'_n \cos \alpha \\ v'_n = v_n + g_2 + l'_n \sin(\alpha + g_3) - l'_n \sin \alpha \end{cases} \quad (14)$$

其中 g_1, g_2, g_3 分别表示起始点的位置变化和围绕起始点 (u_1, v_1) 的旋转变化。在两个连续的帧之间, g_3 是一个微小的变化, $l_n \approx l'_n$ 。因此, (6)可以简化为:

$$\begin{cases} u'_n = u_n + g_1 - g_3(v_n - v_1), \\ v'_n = v_n + g_2 + g_3(u_n - u_1). \end{cases} \quad (15)$$

由(7), 我们可以得到:

$$\begin{cases} \frac{du}{dt} = \frac{d(u'_n - u_n)}{dt} = \frac{dg_1}{dt} - (v_n - v_1) \frac{dg_3}{dt}, \\ \frac{dv}{dt} = \frac{d(v'_n - v_n)}{dt} = \frac{dg_2}{dt} + (u_n - u_1) \frac{dg_3}{dt}. \end{cases} \quad (16)$$

由式(8), 定义 $\frac{dg_1}{dt}, \frac{dg_2}{dt}, \frac{dg_3}{dt}$ 为 G_1, G_2, G_3 , 由此公式(4)可表示为

$$\begin{bmatrix} I_u & I_v \end{bmatrix} \begin{bmatrix} 1 & 0 & -(v_n - v_1) \\ 0 & 1 & (u_n - u_1) \end{bmatrix} \begin{bmatrix} G_1 \\ G_2 \\ G_3 \end{bmatrix} = -I_t. \quad (17)$$

直线上的每个点 (u_n, v_n) 都满足上述方程。为了提高线流跟踪能力, 我们将 Lucas-Kanade 特征跟踪器的 pyra-midal 实现扩展到线流。丢弃两个端点附近的一个小直线段来构建线流跟踪的约束, 并在跟踪完成后通过判断端点灰度梯度是否与线段一致来扩展它。通过迭代的收敛性来判断每条线跟踪的质量。将跟踪成功的线加入到优化中。

5. 实验

5.1. 线流分析

运行时间: 我们测量了 EuRoC、TUM VI 和 KAIST VIO 数据集上的线特征检测和跟踪运行时间, 如图 2 所示。实验结果表明, 与 PL-VINS 相比, MED 和 LOFT 模块在线条特征检测和匹配方面都有显著提高。为了进行更全面的比较, 我们评估了算法在 EuRoC 数据集完整序列上的每个线程的平均运行时间, 如表 1 所示。在所有线程上, VINS-Fusion 略快于 VINS-Mono。由于在优化中加入了线特征残差, 因 VINS-Mono 相比, PL-VINS、E-PL-VINS、PL-F-VINS 和 EPLF-VINS 在线程 3 上的时间增加。对于螺纹 2, E-PL-VINS、PL-F-VINS 和 EPLF-VINS 的速度分别比 PL-VINS 提高 28.7%、18.7%和 32.9%。

5.2. 定量比较

实验测量 EuRoC 数据集的完整序列, 该数据集由机器大厅和 Vicon 室的 11 个序列组成, 具有结构信息。表 2 给出了绝对轨迹误差(ATE)的均方根误差(RMSE)。在无闭环约束条件下, EPLF-VINS 在平均平移值上比 VINS-Mono 提高 35.83%, 比 PL-VINS 提高 28.54%。PL-F-VINS 显示出一定程度的精度提高。在闭环约束条件下, EPLF-VINS 比 VINS-Mono 精度提高 32.27%, 比 PL-VINS 精度提高 13.61%。实验结果表明, 与 PL-VINS、E-PL-VINS 和 PL-F-VINS 相比, 我们的系统线特征测量模型进一步提高了测量精度。

5.3. 定性评价

TUM VI 基准是一个序列, 围绕大学建筑的中央大厅散步, 包含丰富的线条特征。与 PL-VINS 进行了线特征检测与跟踪的可视化实验, 如图 3(a)所示。绿色提取线条, 蓝色追踪图 3(a)中的线在空间上的分

布比红线更为合理。合理的线特征检测和残差测量模型使得后端映射更加精确和准确，如图 3(b)所示。

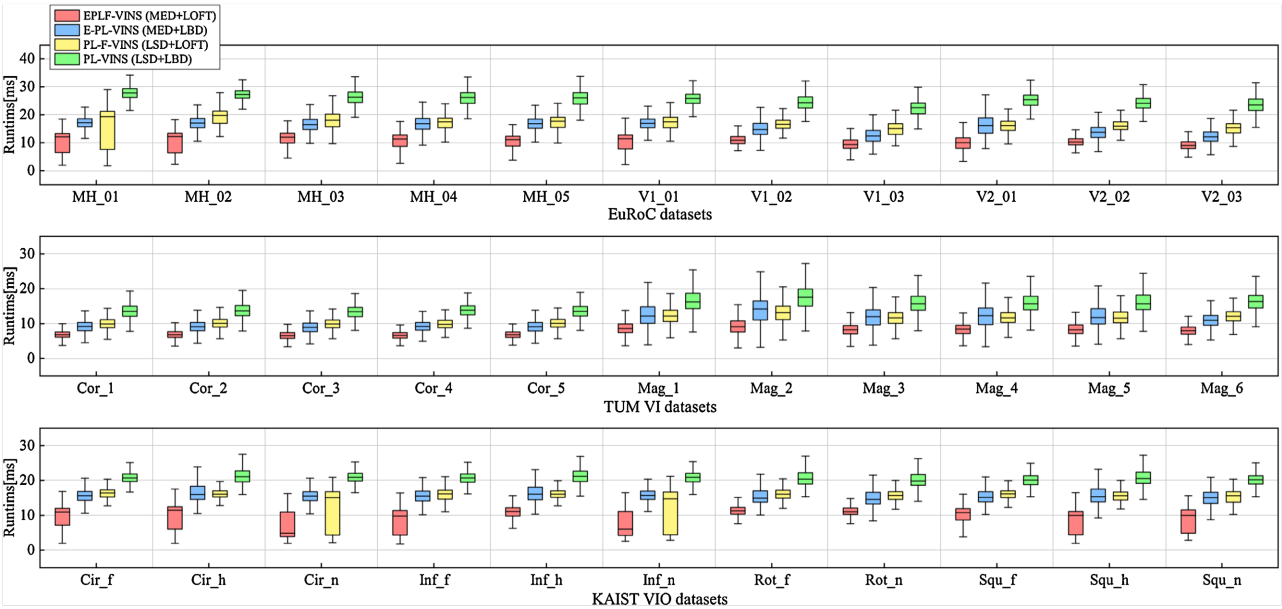
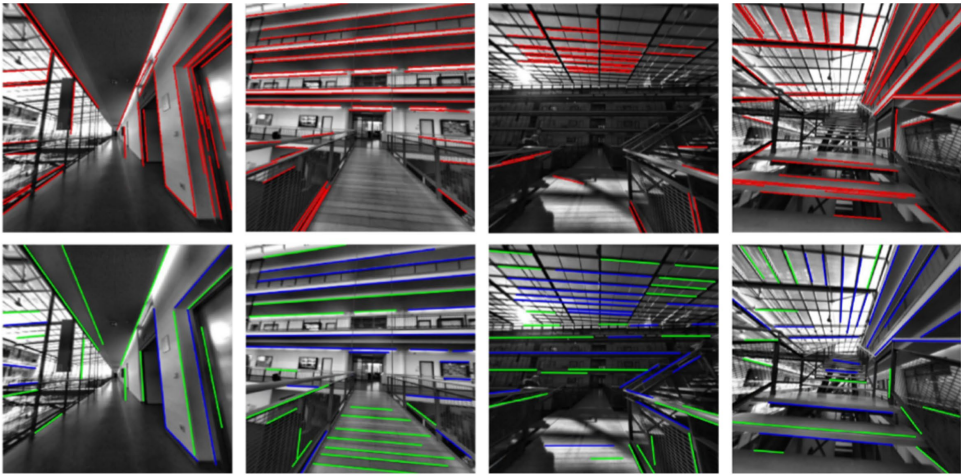


Figure 2. Runtimes of line feature detection and tracking.
图 2. 线特征检测和跟踪的运行时间

Table 1. RMSE ATE [m] comparison results
表 1. RMSE ATE [m]比较结果

	MH_01	MH_02	MH_03	MH_04	MH_05
VINS-Mono	0.2023	0.1879	0.2281	0.3697	0.2968
PL-VINS	0.2159	0.1933	0.2177	0.2471	0.3254
E-PL-VINS	0.2245	0.1968	0.2195	0.2476	0.3248
PL-F-VINS	0.2059	0.1812	0.1624	0.1622	0.2185
EPLF-VINS	0.1404	0.0879	0.1137	0.1826	0.1817



(a)

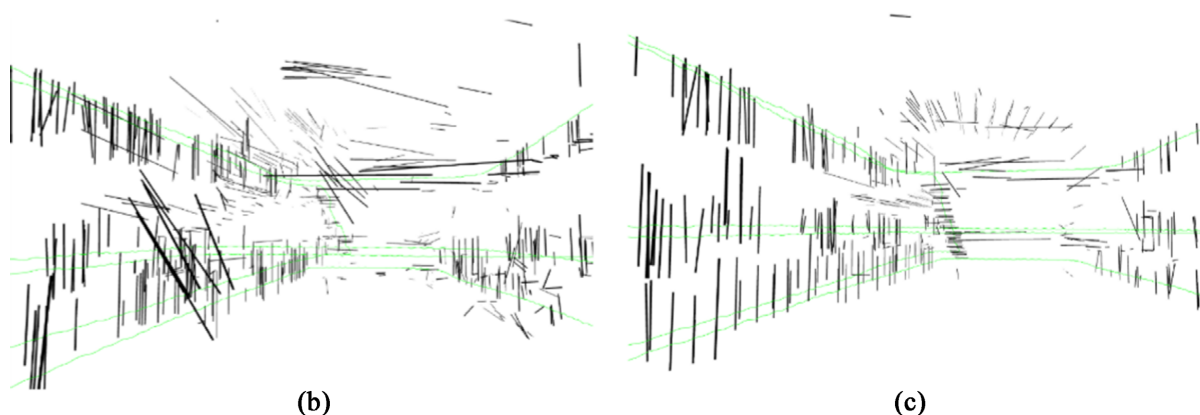


Figure 3. Experimental results on the TUM VI benchmark dataset. (a) The red line in the middle represents the line detection and matching visualization of PL-VINS. (a) The green line (detection line) and blue line (tracking line) on the lower side represent the results of our algorithm. (a) Comparison results of line feature extraction and tracking with PL-VINS. (b) PL-VINS line chart. (c) EPLF-VINS line chart

图 3. TUM VI 基准数据集上的实验结果。(a)中的红线表示 PL-VINS 的线检测和匹配可视化。(a)下侧的绿线(检测线)和蓝线(追踪线)表示我们的算法结果。(a)与 PL-VINS 的线特征提取与跟踪对比结果。(b) PL-VINS 线形图。(c) EPLF-VINS 线形图

6. 结论

我们提出了一种简洁高效的线流检测和跟踪算法，该算法通过改进的线特征提取策略提高了线特征的质量。线流跟踪器比传统的 LBD 匹配算法更有效。此外，提出了一种新的线特征测量模型来表示残差，以消除线端点对残差估计的影响。我们使用 EuRoC、TUM VI 和 KAIST VIO 数据集提供定量评估和可视化实验。与最先进的算法相比，我们的算法计算效率更高，得到的结果也更准确。

参考文献

- [1] Huang, G. (2019) Visual-Inertial Navigation: A Concise Review. 2019 *International Conference on Robotics and Automation (ICRA)*, Montreal, 20-24 May 2019, 9572-9582. <https://doi.org/10.1109/icra.2019.8793604>
- [2] Mourikis, A.I. and Roumeliotis, S.I. (2007) A Multi-State Constraint Kalman Filter for Vision-Aided Inertial Navigation. *Proceedings 2007 IEEE International Conference on Robotics and Automation*, Rome, 10-14 April 2007, 3565-3572. <https://doi.org/10.1109/robot.2007.364024>
- [3] 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>
- [4] Gomez-Ojeda, R., Moreno, F., Zuniga-Noel, D., Scaramuzza, D. and Gonzalez-Jimenez, J. (2019) PL-SLAM: A Stereo SLAM System through the Combination of Points and Line Segments. *IEEE Transactions on Robotics*, **35**, 734-746. <https://doi.org/10.1109/tro.2019.2899783>
- [5] Zhang, G. and Suh, I.H. (2011) Building a Partial 3D Line-Based Map Using a Monocular SLAM. 2011 *IEEE International Conference on Robotics and Automation*, Shanghai, 9-13 May 2011, 1497-1502. <https://doi.org/10.1109/icra.2011.5979665>
- [6] Smith, P., Reid, I. and Davison, A. (2006) Real-Time Monocular SLAM with Straight Lines. *The British Machine Vision Conference BMVC*, Edinburgh, 4-7 September 2006, 17-26.
- [7] Zhou, H., Zou, D., Pei, L., Ying, R., Liu, P. and Yu, W. (2015) StructSLAM: Visual SLAM with Building Structure Lines. *IEEE Transactions on Vehicular Technology*, **64**, 1364-1375. <https://doi.org/10.1109/tvt.2015.2388780>
- [8] Ruifang, D., Fremont, V., Lacroix, S., Fantoni, I. and Changan, L. (2017) Line-Based Monocular Graph SLAM. 2017 *IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, Daegu, 16-18 November 2017, 494-500. <https://doi.org/10.1109/mfi.2017.8170369>
- [9] Fu, Q., Wang, J., Yu, H., Ali, I., Guo, F. and Zhang, H. (2020) PL-VINS: Real-Time Monocular Visual-Inertial SLAM with Point and Line. <https://arxiv.org/abs/2009.07462>
- [10] Zhao, Y. and Vela, P.A. (2018) Good Line Cutting: Towards Accurate Pose Tracking of Line-Assisted VO/VSLAM.

15th European Conference on Computer Vision, Munich, 8-14 September 2018, 527-543.

https://doi.org/10.1007/978-3-030-01216-8_32

- [11] Wang, Q., Yan, Z., Wang, J., Xue, F., Ma, W. and Zha, H. (2021) Line Flow Based Simultaneous Localization and Mapping. *IEEE Transactions on Robotics*, **37**, 1416-1432. <https://doi.org/10.1109/tro.2021.3061403>
- [12] He, Y., Zhao, J., Guo, Y., He, W. and Yuan, K. (2018) PL-VIO: Tightly-Coupled Monocular Visual-Inertial Odometry Using Point and Line Features. *Sensors*, **18**, Article No. 1159. <https://doi.org/10.3390/s18041159>
- [13] 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>