

ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual-Inertial, and Multimap SLAM

Carlos Campos , Richard Elvira , Juan J. Gómez Rodríguez , Graduate Student Member, IEEE, José M.M. Montiel , Member, IEEE, and Juan D. Tardós , Senior Member, IEEE

摘要

本文介绍了 ORB-SLAM3，这是第一个能够使用针孔和鱼镜头模型，用单眼、立体和 RGB-D 相机执行视觉、视觉-惯性和多地图 SLAM 的系统。第一个主要的创新点是一个紧密集成的视觉-惯性 SLAM 系统，它完全依赖于最大后验（MAP）估计，甚至在 IMU 初始化期间也是如此，从而在小型和大型、室内和室外环境中实现实时稳健运行，比以前的方法准确 2 到 10 倍。第二个主要的创新是一个多地图系统，它依赖于一种新的地点识别方法，该方法具有更好的召回率，使 ORB-SLAM3 能够在长时间的视觉信息不佳的情况下生存：当它迷路时，它启动一个新的地图，当重新访问这些地图时将与以前的地图无缝合并。与只使用最后几秒钟的信息的视觉测绘系统相比，ORB-SLAM3 是第一个能够在所有的算法阶段重新使用来自高视差共视关键帧的所有先前信息的系统，即使它们在时间上相隔很远或来自以前的测绘会议，以提高准确性。我们的实验表明，在所有的传感器配置中，ORB-SLAM3 与文献中最好的系统一样稳健，并且明显更准确。值得注意的是，我们的立体惯性 SLAM 在 EuRoC 无人机上实现了 3.5 厘米的平均精度，在 TUM-VI 数据集的房间里，在快速手持运动下实现了 9 毫米的平均精度，这是 AR/VR 情景的代表。

关键词：计算机视觉；惯性导航；同步定位和制图

1 引言

即时定位与地图构建（Simultaneous Localization and Mapping，简称 SLAM），广泛应用于机器人，AR 眼镜上，分为激光 SLAM 和视觉 SLAM。在视觉 SLAM 中，往往难以兼顾精度和速度，而 orb-slam3 算法是这两者较好的折中，既保证了实时性，又不会让精度太低。然而 orb-slam 在一些硬件比较差的设备上运行的效果不够好，无论是帧率还是定位的准确度都比较低。为了进一步提高 orb-slam3 的速度和精度，对其源码进行一些改进。

2 相关工作

在原来的 orb-slam3 上，为了进一步提高算法的精度和速度，尝试对源码进行部分改进，并通过实验验证结果。在 orb-slam3 中，进行定位与建图的前提是帧间的特征点匹配，而图像之间进行特征匹配时会有一些误匹配的点，这些误匹配点会影响后面的位姿计算，应尽可能剔除掉。源码中有一块是使用“角度投票”的方法剔除误匹配点，但在实验过程中发现这一块有点问题，因此尝试使用设置角度阈值的方法进行改进，改进后进行实验对比，发现精度比改进前提高。

3 本文方法

3.1 本文方法概述

源码中“角度投票”的原理如图 1所示：orb-slam3 中特征点有角度，如果是正确的特征匹配，那么匹配的特征点角度差应该约等于图像旋转的角度。

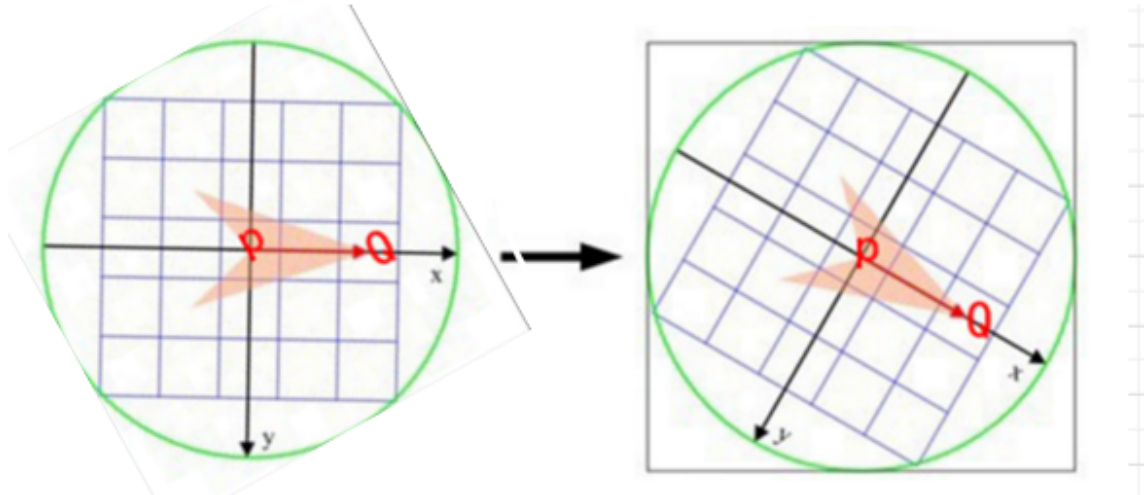


图 1: “角度投票”原理

具体的实现流程如 2所示，将匹配好的特征点对按照特征点对间的角度差分配到 30 个不同的区间，最后选出特征点数量最多的三个不同区间，认为这三个区间是主流旋转角度，也是图像的旋转角度，剔除掉除这三个区间外所有区间的特征点。

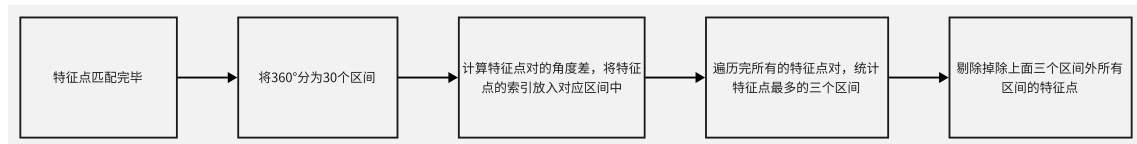


图 2: “角度投票”流程

但是在运行 EuRoc 数据集中的 MH01-MH05 5 个视频流时发现一个相同的问题，如图 3, 4所示，如果最大三个区间是 0、1、29 的话数值会比较正常，说明当图像顺时针或逆时针旋转一个小角度的时候这段代码是有效的，但是如果最大三个区间不是这 0、1、29 的话，数值会很小，说明图像之间几乎匹配不到特征点，这一段代码几乎是无效的。于是想到两种解决方法：一是把区间划的更细，如果划分更多的区间也许能够提高定位和建图的精度，但是代价是损失一定的运行速度；二是设置一个角度阈值取代掉原来的代码，因为原来代码只在图像旋转角度较小时有效，而且剔除掉的也是大角度的区间，所以设置一个角度阈值，如果匹配点对角度差大于这个阈值的直接剔除掉。由于第二种方法既提升了精度又不会增加太多操作，所以选第二种。

第1次角度投票

第一个角度区间位置及大小: 0-----497

第二个角度区间位置及大小: -1-----18446708891310326212

第三个角度区间位置及大小: -1-----18446708891310326212

第2次角度投票

第一个角度区间位置及大小: 0-----378

第二个角度区间位置及大小: -1-----18446708891310326212

第三个角度区间位置及大小: -1-----18446708891310326212

第3次角度投票

第一个角度区间位置及大小: 0-----333

第二个角度区间位置及大小: -1-----18446708891310326212

第三个角度区间位置及大小: -1-----18446708891310326212

第4次角度投票

第一个角度区间位置及大小: 0-----168

第二个角度区间位置及大小: -1-----18446708891310288836

第三个角度区间位置及大小: -1-----18446708891310288836

第5次角度投票

第一个角度区间位置及大小: 0-----247

第二个角度区间位置及大小: 29-----33

第三个角度区间位置及大小: 1-----32

第6次角度投票

第一个角度区间位置及大小: 0-----392

第二个角度区间位置及大小: 1-----51

第三个角度区间位置及大小: 29-----49

第7次角度投票

第一个角度区间位置及大小: 0-----317

第二个角度区间位置及大小: 1-----54

第三个角度区间位置及大小: 29-----54

第8次角度投票

第一个角度区间位置及大小: 0-----385

第二个角度区间位置及大小: 29-----73

第三个角度区间位置及大小: 1-----66

图 3: “角度投票”区间具体数值 (1)

```

第1026次角度投票
第一个角度区间位置及大小: 2-----1
第二个角度区间位置及大小: 5-----1
第三个角度区间位置及大小: -1-----18446709049946348692
第1027次角度投票
第一个角度区间位置及大小: 3-----1
第二个角度区间位置及大小: 5-----1
第三个角度区间位置及大小: 12-----1
第1028次角度投票
第一个角度区间位置及大小: 3-----1
第二个角度区间位置及大小: 9-----1
第三个角度区间位置及大小: 23-----1
第1029次角度投票
第一个角度区间位置及大小: 3-----2
第二个角度区间位置及大小: 1-----1
第三个角度区间位置及大小: 2-----1
第1030次角度投票
第一个角度区间位置及大小: 5-----1
第二个角度区间位置及大小: 6-----1
第三个角度区间位置及大小: 7-----1
第1031次角度投票
第一个角度区间位置及大小: 7-----2
第二个角度区间位置及大小: 29-----1
第三个角度区间位置及大小: -1-----18446709049946348692
第1032次角度投票
第一个角度区间位置及大小: 8-----1
第二个角度区间位置及大小: 20-----1
第三个角度区间位置及大小: -1-----18446709049946348692
第1033次角度投票
第一个角度区间位置及大小: 4-----1
第二个角度区间位置及大小: 10-----1
第三个角度区间位置及大小: 13-----1

```

图 4: “角度投票”区间具体数值 (2)

4 复现细节

4.1 与已有开源代码对比

将源码中所有关于“角度投票”的代码注释掉，然后在剔除特征点对时增加一个判断，大于一个角度阈值的特征点对直接从候补特征点对中剔除掉，原来代码在 2 个小区间内是正常的，因此角度阈值暂且设为 -24° 和 24° ，修改完成后运行 EuRoc 数据集 MH01-MH05, 并用 evo 工具测评改进前和改进后的精度。

4.2 实验环境搭建

Linux:Ubuntu 18.04 OpenCV:version 3.4.0 Pangolin Eigen: version 3.1.0

4.3 界面分析与使用说明

执行代码

```
./Monocular/mono_euroc ../Vocabulary/ORBvoc.txt ./Monocular/EuRoC.yaml /home/klwa/project/ORB_SLAM3-master/datasets/MH01 ./Monocular/EuRoC_TimeStamps/MH01.txt dataset-MH01_mono
```

evo 工具计算估算的轨迹和实际轨迹的绝对误差

```
evo_traj tum f_dataset-MH01_mono.txt --ref=MH01_gt.tum -p --plot_mode=xy -as
```

evo 工具计算估算轨迹和实际轨迹的各种误差（标准差、和方差、均方跟等）

```
evo_ape tum MH01_gt.tum f_dataset-MH01_mono.txt -p --plot_mode=xy -as
```

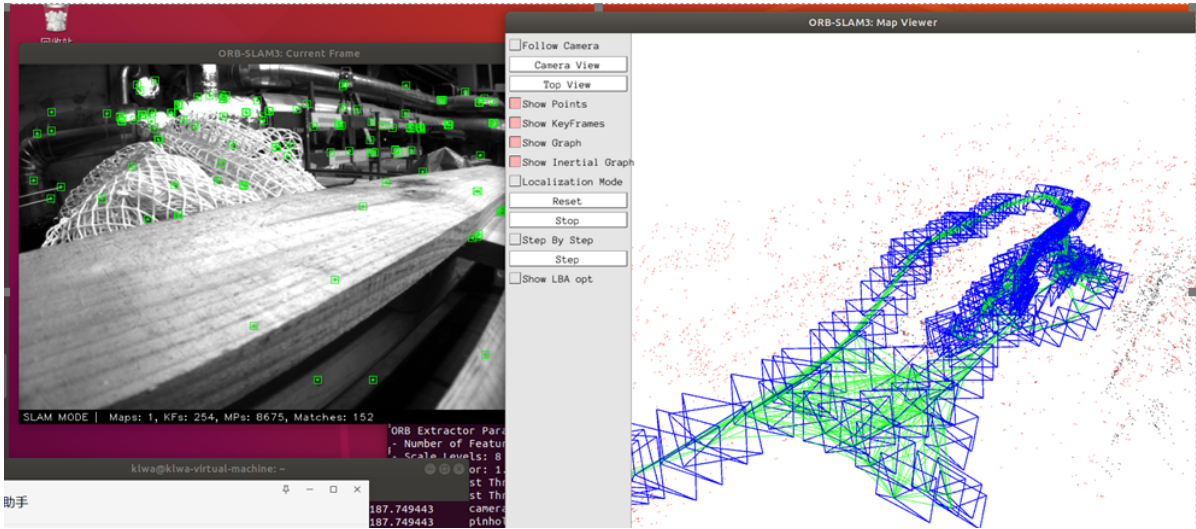


图 5: orb-slam3 运行界面示意

4.4 创新点

主要是发现了源代码运行时的一个问题，针对这个问题进行改进，从而得到更好的效果。

5 实验结果分析

将 evo 测试所得的结果汇总成表格如图 6所示，红色部分代表误差比源码更小的数据，可以看到整体上来说误差比原来小，精度提高，如果以 RMSE 作为精度标准的话则误差减小约 6.28%。

	原来								改进后						
	max	mean	median	min	rmse	sse	std		max	mean	median	min	rmse	sse	std
MH01	0.115948	0.036286	0.028588	0.001252	0.043007	6.729004	0.023086		0.10601	0.03185	0.02562	0.002383	0.03847	5.38462	0.02159
MH02	0.25849	0.028399	0.021397	0.002283	0.034725	3.617469	0.019983		0.09963	0.028827	0.024676	0.002289	0.03289	3.24503	0.01583
MH03	0.111537	0.032684	0.029019	0.001024	0.036893	3.571439	0.017112		0.10903	0.03127	0.02813	0.002059	0.03578	3.359083	0.017381
MH04	0.243413	0.080828	0.078943	0.004603	0.086615	14.70416	0.031128		0.378616	0.05673	0.05233	0.00444	0.06784	9.0569	0.037199
MH05	0.308686	0.133992	0.132727	0.013071	0.146432	46.46524	0.059062		0.14064	0.04251	0.0374	0.00201	0.04813	4.84179	0.02257

图 6: 实验结果示意

6 总结与展望

通过实验可以看到通过加入角度阈值的方法替代“角度投票”能够提高 orb-slam3 的精度。orb-slam3 是目前视觉、视觉-惯性和多地图 SLAM 中最完整的开源库，包含单目、立体、RGB-D、针孔和鱼眼相机。但仍然有许多需要完善的地方，ORB - SLAM3 的主要失效案例是低纹理环境, 直接方法对低纹理更加鲁棒，但仅限于短期^[1]和中期^[2]的数据关联。最近在用卷积神经网络（CNN）从单一图像进行深度估计方面取得的进展为可靠的、真实尺度的单眼 SLAM 提供了良好的前景^[3]，至少在 CNN 被训练的同一类型的环境中是如此。

参考文献

- [1] J. ENGEL V K, CREMERS D. Direct sparse odometry[J]. IEEE Trans. Pattern Anal. Mach. Intell., 2018, 40(03): 611-625.
- [2] J. ZUBIZARRETA I A, MONTIEL J M M. Direct sparse mapping[J]. IEEE Trans. Robot., 2020, 36(04): 1363-1370.
- [3] N. YANG L V, CREMERS D. D3VO : Deep depth, deep pose and deep uncertainty for monocular visual odometry[J]. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020: 1281-1292.