

doi:10.19306/j.cnki.2095-8110.2021.01.006

USME: 统一 SLAM 度量与评测技术研究

屈桢深, 张启航, 杨志伟, 董鸿宇

(哈尔滨工业大学, 哈尔滨 150001)

摘要: 同时定位与建图(SLAM)技术近年来得到迅速发展,但由于缺乏在统一框架下对算法的度量和比较,对SLAM的客观评估和应用造成障碍。提出了统一SLAM度量与评测(USME)框架,从指标体系、数据集及评测方法三个维度为各种SLAM方法的性能度量及比较研究提供基准。针对不同场景,建立了包括长时间运行漂移量,闭环检测能力,存在相机遮挡、光照变化和运动物体时SLAM方法的鲁棒性,以及多体协同性能等的综合性能指标体系。基于三维仿真平台,以指标体系为基准建立了合成数据序列及对应数据集,以对性能指标进行度量与评估。还建立了平均指标均值的数据处理与评测方法,以综合评价不同参数选择对方法性能的影响。通过典型SLAM方法验证了上述方法的可行性。

关键词: SLAM性能评估框架;指标体系;仿真数据集;多体协同性能

中图分类号: TP391

文献标志码: A

开放科学(资源服务)标识码(OSID):

文章编号: 2095-8110(2021)01-0055-11



USME: Unified SLAM Measurement and Evaluation

QU Zhen-shen, ZHANG Qi-hang, YANG Zhi-wei, DONG Hong-yu

(Harbin Institute of Technology, Harbin 150001, China)

Abstract: The technology of simultaneous localization and mapping (SLAM) has developed rapidly in recent years, but the lack of a unified framework under which the measurement and comparison of algorithms can be carried out objectively has become a major obstacle to the objective evaluation and application of SLAM. A unified SLAM measurement and evaluation (USME) framework is proposed, which provides benchmarks for the performance measurement and comparative research of various SLAM methods in terms of metrics system, dataset and evaluation method. To adjust to different scenarios, we establish a comprehensive performance index system encompassing long-term drift, closed-loop detection capabilities, the robustness of the SLAM method when camera occlusion, illumination changes and moving objects exist, and multi-body coordination performance. Based on a three-dimensional simulation platform, synthetic data sequences and corresponding datasets based on the metrics system are established to measure and evaluate performance. We also present a data processing and evaluation method for the average value of the average metrics to comprehensively evaluate the impact of different parameter choices on the performance of the method. The feasibility of the above scheme is verified by a typical SLAM method.

Key words: SLAM performance evaluation performance framework; Metrics system; Simulation dataset; Multi-body collaborative performance

收稿日期: 2020-09-21; 修订日期: 2020-11-15

基金项目: 国家自然科学基金青年基金(61703123)

作者简介: 屈桢深(1973-), 男, 副教授, 主要从事制导控制方面的研究。E-mail: ocicq@126.com

0 引言

随着智能无人系统的不断发展,同时定位与建图(Simultaneous Localization and Mapping, SLAM)技术作为无人系统感知自身状态与周围环境的关键技术,已经成为了当前科技研究的热点领域。针对现存SLAM算法,如何建立一致、定量的评估框架,以及用什么样的具体评估指标集和测试方法来实现评估,从而既满足不同算法统一评估的需求,又可针对特定应用场景进行专项评估,成为了当前待研究的重要课题。

SLAM算法研究初期,大多论文仅针对具体测试数据给出测试指标,不同方法之间缺乏一致性。2015年,Nardi等提出了SLAMBench^[1]评估框架,该框架的主要工作是以不同的方式实现KinectFusion算法,然后通过ICL-NUIM数据集对其进行评估,比较不同方法实现时SLAM系统的性能、精度和能耗。2018年,在其后续的工作SLAMBench2^[2]中,Bodin等在SLAMBench的基础上增加了框架的适用性,具有与数据集无关,支持即插即用的特点。评价指标方面,在计算速度、算法精度和功耗之外增加了SLAM系统内存使用的评估指标,以全面评估算法在硬件平台上的性能。该方法还支持使用多种数据集(包括ICL-NUIM、TUM RGB-D和EuRoC等)对多种SLAM算法进行评估。2019年,Zhao等提出了新的SLAM框架GSLAM^[3],该框架不仅能够提供性能评估的功能,还开发了对应的工具包,以便于定制化SLAM系统的开发。GSLAM将输入数据集、SLAM算法实现、可视化以及SLAM应用全部置于一个统一的框架中。同年,Li等针对增强现实中的单目视觉惯性SLAM算法提出了一种评测方法^[16],使用手机在真实场景中采集数据制作数据集,并通过跟踪精度及鲁棒性、初始化质量和重定位时间对SLAM算法进行评测。这些评估框架可以实现不同SLAM算法在同一框架下进行比较,但在性能评估方面还存在评价指标较为单一的问题。

对于测试SLAM算法性能的数据集主要包含两种,一种是在仿真环境中获取,而另一种是在实际场景中进行采集。针对第一种,2014年,Handa等利用POVRay作为渲染软件为RGB-D SLAM生成两个评测序列^[24]。该数据集只含有用于定位和局部深度图的真实值,而且图像的数量和场景的复

杂度都十分有限。2016年,Ros等提供了多个含有不同天气、季节和光照条件的序列用于评测SLAM的性能^[25],还提供了多视角相机的图片数据和标定数据。但是该数据集仅以5fps的帧率进行录制,无法对实时SLAM系统进行评估,而且图片的质量也较低。针对第二种在实际场景中采集的数据集,现有的典型SLAM数据集,如KITTI数据集^[4]、TUM数据集^[5]和EuRoC数据集^[6]等,通过使用不同的传感器在不同的环境中采集数据,为SLAM系统的评估提供了较好的参照,但它们各自也存在场景不够丰富、性能评估指标比较单一等问题。

本文针对现有评估框架和数据集存在的问题,提出了一类统一SLAM度量与评测(Unified SLAM Measurement and Evaluation, USME)框架。首先该框架针对SLAM性能从定位精度、地图构建以及鲁棒性等方面建立了指标体系。之后针对现有数据集存在的场景单一、测试方法缺乏针对性等问题,提出了数据集生成方式,并对典型SLAM算法长时间运行漂移、闭环检测能力和算法鲁棒性等方面的性能进行评估。最后建立了多体协同SLAM子系统性能、全局状态以及协同SLAM效率的评估方法。

1 指标体系

SLAM系统通过搭载环境感知传感器,利用环境观测信息实现了实时定位与建图^[7]。所以为了对SLAM系统进行全面准确的评估,本文分别从SLAM的定位精度、地图构建、鲁棒性等方面对SLAM系统进行评估。在定位精度评估指标中,通过绝对轨迹误差和相对位姿误差对SLAM算法进行评估;在建图方面,通过点云的重构误差以及3D重构能力、点云稀疏程度和噪声的比例来综合评判SLAM所建的地图;在鲁棒性方面,则是通过跟踪的鲁棒性和跟踪丢失时重定位的时间进行评估。最后通过初始化时间、内存的使用和能耗对SLAM系统的其他方面进行评估。其主要框架如图1所示。

1.1 SLAM算法定位精度评估指标

无人系统通过SLAM定位,计算未知环境中自身的位置与姿态。SLAM定位精度也是SLAM性能最直接的评价指标之一,可通过绝对轨迹误差和相对位姿误差进行测算。

(1)绝对轨迹误差

在已知真实轨迹的情况下,绝对轨迹误差是直

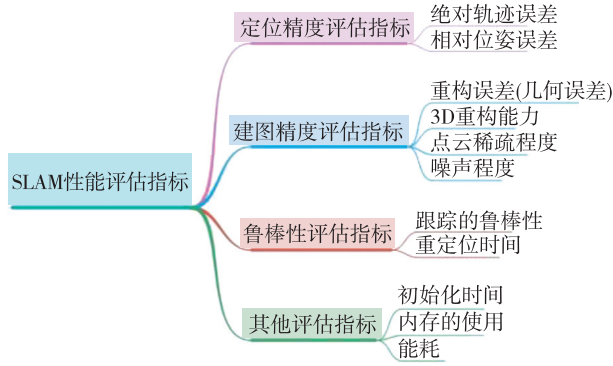


图 1 SLAM 算法性能指标

Fig. 1 SLAM algorithm performance metrics

接测量真实轨迹点和估计轨迹点之间的差值。首先通过时间同步机制,在统一的时间度量下将估计轨迹点与真实轨迹点关联起来;之后将轨迹进行对齐,过程如图 2 所示;最后计算每个点对的差值,并输出它们的均方根作为最后的绝对轨迹误差。该指标可直观地反映全局的位姿误差,同时也可以很好地反映 SLAM 算法的性能。

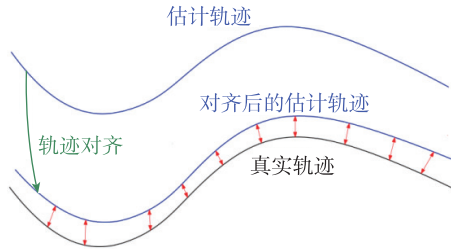


图 2 绝对轨迹误差示意图

Fig. 2 Schematic diagram of absolute trajectory error

对于一个单一的状态,其估计值 \hat{x}'_i 和真实值 x_i 的误差 Δx_i 满足

$$\Delta x_i = x_i - \Delta \mathbf{R}_i \hat{x}'_i \quad (1)$$

所以整条轨迹的绝对轨迹误差可以定义为

$$ATE = \left(\frac{1}{N} \sum_{i=0}^{N-1} \|\Delta x_i\|^2 \right)^{\frac{1}{2}} \quad (2)$$

其中, $\Delta \mathbf{R}_i$ 表示估计轨迹状态与真实轨迹状态之间的旋转矩阵; N 表示轨迹上所有状态的数量。

绝对轨迹误差通过最直观的方式给出了整条轨迹的误差,其优点是易于比较。但它对误差发生的时间比较敏感,误差发生在轨迹刚开始时会比发生在结束时产生更大的误差。因此除了绝对轨迹误差,相对位姿误差也被广泛使用,从而提供了更多信息。

(2) 相对位姿误差

相对位姿误差是计算位姿变化量的差值。通

过时间同步机制,在统一的时间度量下计算真实轨迹与估计轨迹每隔一段时间位姿的变化量,然后对该变化量做差,以获得相对位姿误差,该指标主要用于估计系统的漂移量。

相对位姿误差的基本思想是当 SLAM 里程计在没有全局参考时,通过测量不同时刻状态之间的相对关系来评估轨迹的质量,其过程如图 3 所示。

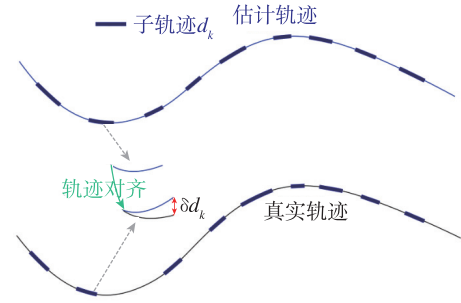


图 3 相对位姿误差示意图

Fig. 3 Schematic diagram of relative pose error

沿着轨迹选择 K 个状态对 d_k 作为一个集合 I

$$I = \{d_k\}_{k=0}^{K-1}, d_k = \{x_s, x_e\} \quad (3)$$

其中, $e > s$, 对于每一段 d_k 的相对位姿误差 δd_k 为

$$\delta d_k = \left(\frac{1}{N_k} \sum_{i=0}^{N_k-1} \|x_{ik} - \Delta \mathbf{R}_{ik} \hat{x}'_{ik}\| \right)^{\frac{1}{2}} \quad (4)$$

其中, N_k 表示 d_k 子轨迹上所有状态的数量; x_{ik} 和 \hat{x}'_{ik} 表示在 $\{x_s, x_e\}$ 之间状态的真实值和估计值; $\Delta \mathbf{R}_{ik}$ 则表示在 $\{x_s, x_e\}$ 之间估计轨迹状态与真实轨迹状态之间的旋转矩阵。

所以整条轨迹的相对位姿误差可以记作

$$RPE = \{d_k\}_{k=0}^{K-1} \quad (5)$$

由于相对位姿误差不是一个单一的量,而是满足某些标准的所有子轨迹的误差集合,因此可以通过计算中值和平均值等统计数据,这与绝对轨迹误差相比能够提供更多的信息。

1.2 SLAM 算法地图构建评估指标

地图构建作为 SLAM 系统的另一个重要功能,地图的质量直接影响了后续路径规划或者导航等其他功能的成功率。针对 3D 点云地图^[8],本节基于点云的几何误差、3D 重构能力和稀疏程度等对 3D 点云地图进行评估。

(1) 点云的几何精度

由于设备的精度、人为操作或者环境因素等影响,SLAM 系统输出的点云数据中不可避免地会出

现一些离群点。为了能够更加精确地计算点云的几何精度,在进行点云配准之前,需要对这些点进行剔除。剔除通常使用统计异常值去除(Statistical Outlier Removal, SOR)滤波器。其主要思想是计算每一个点邻域内所有临近点的平均距离 μ 和标准差 σ ,然后筛选掉 $\mu \pm \alpha\sigma$ 以外的点,此处的 α 取决于所分析邻域的大小。剔除离群点后使用最近点迭代(Iterative Closest Points, ICP)^[9]算法进行点云的配准^[10]。ICP算法是一种点集对点集的配准方法,基于最小二乘的最优匹配,重复进行确定对应关系点集到计算最优刚体变换的过程

$$\min_{R, T} J = \frac{1}{2} \sum_{i=1}^n \|(p_i - (Rp'_i + t))\| \quad (6)$$

其中, J 为构建的代价函数; p_i 和 p'_i 分别为估计点云与实际点云中的点; n 为所选择点的数量; R 和 t 则表示需要求得的旋转矩阵和平移向量。

当点云配准之后,通过计算点云之间的距离评估SLAM构建地图的几何精度。最常用的方法是点云到点云的直接计算(Cloud-to-Cloud, C2C)。利用该方法比较3D点云地图是最简单直接的,不需要对点云进行分割或者特征提取。针对每一个需要比较的点,该算法都会在真实环境点云中寻找最近的点,同时计算它们的欧几里得距离,如图4所示。

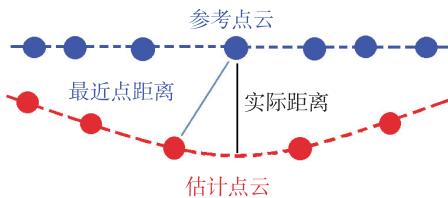


图4 点云距离计算示意图

Fig. 4 Point cloud distance calculation schematic diagram

在计算估计点云与真实点云之间的距离时,可能不是点对点之间的距离计算,而是一个点集对一个点集之间的计算。因此在C2C算法中,采用豪斯多夫距离(Hausdorff distance)计算点云之间的距离。点集 A 与点集 B 之间的豪斯多夫距离为

$$H(A, B) = \max_{a \in A} \{ \min_{b \in B} \{ d(a, b) \} \} \quad (7)$$

其中, a 和 b 为点集 A 和 B 中的点,而 $d(a, b)$ 表示 a 和 b 之间的欧几里得距离。式(7)的含义是在 A 中选择一个点 a ,然后计算它和 B 中所有点的欧几里得距离,并选择最小的一个;之后遍历 A 中的点,选择结果最大的一个作为豪斯多夫距离。这

种方法只依赖于3D点云数据中点的密度变化,在生成点云时,如果可以使点云具有相似的局部密度,则可以最小化距离测量的误差。

(2) 点云地图的其他评估指标

通过计算点云距离可以得到点云的几何精度。但是针对点云地图,只通过一个几何精度并不能全面地评判其好坏。所以为了更客观全面地评估SLAM算法的地图构建能力,在几何精度的基础上又提出了三个指标,分别为点云的噪声、点云稀疏程度以及3D重构能力。

当通过滤波器对点云进行过滤后,依然会存在一些无关点,这些点可以通过视觉上的检查直接进行剔除,噪声点的百分比就是清理掉点的数量与原始点云中点的数量的比。所以点云的噪声可以定义为

$$\alpha_c = \frac{N_n}{N_c} \quad (8)$$

其中, N_n 和 N_c 分别代表清除掉的无关点数量和点云的总点数,同时 N_c 也可以反映点云的稀疏程度。像一些点云准确率很高的算法如ORB-SLAM^[8,11],由于太过稀疏,几乎不能应用于导航任务^[12];反而像LSD-SLAM^[13]这种准确率中等,但可以提供半稠密点云地图的算法更适用于导航任务。对于3D重构这项指标,通过视觉上观察就可以对其进行评价。

在对SLAM系统生成的3D点云地图进行评估时,并不能仅仅只是关注精度的问题,还需综合考虑该SLAM算法的应用场景。例如应用于导航或者路径规划时,一个高精度的稀疏点云地图几乎没有任何帮助,但一个精度较低的半稠密或者稠密点云地图往往会有更好的效果^[14]。

1.3 SLAM算法鲁棒性评估指标

当SLAM系统运行时,可能会受到各种各样来自于外界的干扰,如光照的变化、移动物体等。一个鲁棒性较好的SLAM系统可以对这些干扰做出正确的判断,并尽可能地减少这些干扰对其产生的影响,从而保证系统的稳定运行^[15]。本节则通过跟踪的鲁棒性以及重定位的时间对SLAM系统的鲁棒性进行评估。

(1) 跟踪的鲁棒性

在极端情况下,视觉轨迹可能会出现跟丢的情况。一个鲁棒性较好的系统应该尽可能快地从丢失状态中恢复,并且可以将新的位姿与之前的轨迹相结合。假设在跟踪时,发生了一小段丢失的情

况。其中 s_1 和 s_2 分别代表丢失前后的两段轨迹, 这两段轨迹相对于真实轨迹之间会分别存在一个旋转矩阵和平移向量

$$s_1 \rightarrow \{R_1, t_1\}, s_2 \rightarrow \{R_2, t_2\} \quad (9)$$

对于一个鲁棒性较好的 SLAM 系统, 子轨迹 s_1 和 s_2 应该能够较好地与真实轨迹进行对齐, 因此 $\{R_1, t_1\}$ 和 $\{R_2, t_2\}$ 应该比较接近; 否则, 估计轨迹将会与真实轨迹存在较大的偏差, 从而造成全局的不一致性, 如图 5 所示。

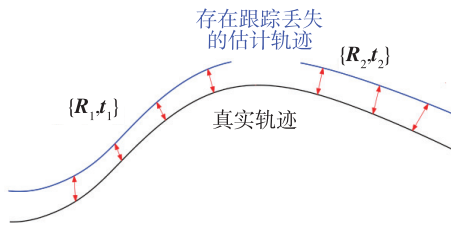


图 5 SLAM 系统跟踪鲁棒性示意图

Fig. 5 SLAM system tracking robustness schematics diagram

通过 SLAM 系统获得一个跟踪结果时, 其中可能发生多段丢失的情况。首先根据这些丢失的情况将轨迹进行分段, 然后使用上述方法将每一段估计轨迹和真实轨迹进行对齐。通过比较两个临近子轨迹的变化来计算重定位误差

$$E_{RL} = \sum_{i=1}^{n-1} \|\log_{\text{Sim}(3)}(\xi_i^{-1}\xi_{i+1})\|_2^2 \quad (10)$$

其中, n 代表总共子轨迹的数量; ξ_i 和 ξ_{i+1} 则代表相邻两段子轨迹与其对应真实轨迹之间的相似变化。

一个好的 SLAM 系统应该可以从丢失状态中快速恢复, 并且回到正确的位置。同时丢失的时间越短, 重定位的误差越小, SLAM 系统的鲁棒性也就越好。

(2) 重定位时间

跟踪的鲁棒性是用来反映 SLAM 系统稳定性的性能指标, 同时在轨迹发生丢失后, SLAM 系统完成重定位的时间也十分重要。通过强迫系统进入重定位阶段可以准确地获取重定位时间, 为此设计了一种测试方式。首先制作一个测试序列, 其具体方式是: 围绕一个有纹理的物体平稳旋转几圈。在绕过第一圈之后, 将后面的一些帧换成黑色帧, 其中黑色帧的时长定为 τ 。在第一圈时, 一个 SLAM 系统可以将其所看到的视觉信息存储起来, 当黑色帧到来时, 它被迫进入重定位阶段; 在当黑

色帧结束后, 会有一段时间的原始帧数据, SLAM 系统可以在这段时间内进行重定位。通过手动添加黑色帧的方式可以准确地计算出重定位的时间, 即黑色帧结束到开始输出估计姿态的时间差。

假设将 t_{s_i} 作为第 i 次黑色帧结束的时间, t_{e_i} 是在第 i 次黑色帧结束后输出的第一个有效位姿的时间。如果一共插入了 N 次黑色帧, 那么平均的重定位时间可以定义为

$$t_{RL} = \frac{1}{N} \sum_{i=1}^N (t_{e_i} - t_{s_i}) \quad (11)$$

1.4 SLAM 算法的其他评估指标

(1) 初始化时间

对于 SLAM 系统, 一个快速准确的初始化可以带来很好的用户体验。但是不同的 SLAM 系统可能使用不同的初始化策略。原始的 PTAM 算法需要用户的交互, 基于滤波的 MSCKF 需要设备保持静止一段时间, 而 ORB-SLAM2 则能够自动地选择两帧用于初始化。如果使用 SLAM 系统传回来的第一个有效位姿作为初始化结束的标志, 其后面的估计位姿可能不是一个很好的估计结果。因此, 通过判断运动的估计尺度是否稳定来确定初始化是否完成^[16]。

在 t 时刻, 将 SLAM 系统估计轨迹的尺度定义为 $s(t)$, 当 SLAM 系统完成初始化时, 需要达到一个稳定的尺度。所以在时间 $[t, t + \Delta t]$ 内定义最大相对尺度 $r(t)$ 为

$$r(t) = \frac{\max_{\tau \in [t, t + \Delta t]} \{|s(\tau) - s(t)|\}}{s(t)} \quad (12)$$

当 $r(t)$ 小于某个阈值 r_{init} 时, 就可以认为 SLAM 系统估计轨迹的尺度已经稳定, 系统已经完成初始化, 此时的 t 记作 t_{init} , 表示 SLAM 系统的初始化时间。

除了初始化时间, 在初始化时估计轨迹尺度的准确性也非常重要。计算在 t_{init} 时的对称相对尺度误差^[16]为

$$\epsilon_{scale} = \frac{1}{2} \left(\left| \frac{s(t_{init})}{s_g} - 1 \right| + \left| \frac{s_g}{s(t_{init})} - 1 \right| \right) \times 100\% \quad (13)$$

其中, s_g 代表每条轨迹的全局尺度。对于视觉 SLAM 系统, 可以通过将所有结果与真实轨迹进行对齐来获得 s_g ; 对于视觉惯性 SLAM 系统^[17, 26], 则需要将 s_g 设为 1, 因为对其来说, 使用准确的尺度进行初始化是非常重要的。

(2)内存的使用及能耗

内存的使用及能耗是指 SLAM 系统在工作时所消耗的内存及能量。不同的使用者对于 SLAM 系统的需求可能不同,他们所使用的硬件平台也不尽相同。有一些可能使用一些大型的移动设备,它们拥有较高的功耗和性能较好的处理器;而有一些则可能使用一些小型的移动载体^[18],如小型的无人机,像这样的载体设备对于功耗的需求和处理器的性能就会较为敏感。然而,SLAM 系统,往往会消耗大量的内存,但是对于内存的消耗通常不会作为评估的标准。所以为了更好地帮助使用者选择适合他们的 SLAM 系统,应该将功耗以及内存的使用作为两个评估 SLAM 系统的重要指标。目前,SLAM 算法的动态内存消耗可以在每一帧的水平上进行精确测量,同时也可以计算整体的内存消耗。而对于 SLAM 系统能耗指标的测量,可以采用一种常用的传感器应用程序编程接口(Application Programming Interface, API)对其进行测量。

2 数据集

对 SLAM 算法进行评估时,往往需要使用数据集^[19]。一个通用准确的数据集对于视觉 SLAM 系统的发展是非常重要的。目前,数据集的种类主要有两种,一种是在实际环境中采集的数据集,现有的典型数据集有 TUM RGB-D 数据集、KITTI 数据集和 EuRoC MAV 数据集。它们各自的特点如表 1 所示。

表 1 典型 SLAM 数据集的特点
Tab. 1 Features of typical SLAM datasets

数据集	数据集采集场景	相机的类型	数据集所包含数据	采集平台
TUM RGB-D 数据集	室内场景,包含办公室场景和工业厂房场景	RGB-D 相机	彩色图片 深度图片 真实轨迹的位置与姿态	手持相机或安装在移动机器人上拍摄
KITTI 数据集	室外场景,高速公路、中等城市和社区的道路	灰度和彩色的双目相机	双目相机拍摄图片 IMU 数据 激光雷达数据	自动驾驶平台采集
EuRoC MAV 数据集	室内场景,工业厂房和普通的房间	灰色的双目相机	双目相机拍摄图片 IMU 数据 激光扫描 3D 点云图	通过无人机进行拍摄

同时 TUM 数据集和 KITTI 数据集也提供了

轨迹评估工具,可以通过绝对轨迹误差和和相对位姿误差对估计轨迹进行评估。但在真实环境中采集数据集也存在很多困难:

1)相机的位置和姿态难以获取。准确地测量相机在每一帧的 6 自由度位姿是一项艰巨的任务。

2)准确的校准参数获取较为复杂。大多数视觉 SLAM 系统都需要对相机的内参进行标定,如果是视觉惯性^[20-21] SLAM 系统,还需要对惯性测量单元(Inertial Measurement Unit, IMU)以及相机和 IMU 的外参进行标定。

3)无法模拟失真场景。实际场景中相机的畸变参数是固定的,并且人为不可控,所获得的畸变效果不一定可以满足使用的需求。

4)标签标注较为困难。在一些结合语义信息的 SLAM 系统需要对每一张图片进行语义的标注,但通过手动的方式对图片进行标注需要大量的人力,同时也无法保证在每个物体的边缘都有像素级的精度。

而另一种数据集是在仿真环境中进行采集。在真实环境中采集数据集存在以上困难,而在仿真环境中这些问题都可以轻松地解决。首先在仿真环境中,可以准确地获取相机的运动轨迹、任意时刻的姿态和相机的参数;同时相机的参数还可以根据自己的需求进行更改,从而实现了获取的图片中含有运动模糊等畸变效果。另外,针对环境的语义分割结果也可以实现像素级的精度,同时在仿真环境中,使用者可以使用任何载体搭载相机来获取数据;除此之外,在仿真环境的另一个优势就是可以根据使用者的需求任意选择场景。

在仿真环境中采集数据集具有对场景的选择更加灵活、成本更低、获取数据更加丰富准确的优点,因此通过建立仿真环境来生成数据集,并使用仿真数据集对 SLAM 算法进行评估。本文基于 Unreal Engine 游戏开发平台,使用由微软开发的 AirSim^[22] 插件来采集数据。同时为了对 SLAM 系统进行更加客观真实的评测,选择城市交通、居民住宅、工业厂房和房间等场景对 SLAM 算法进行测试。其中部分场景如图 6 所示。

住宅区场景和城市场景均为室外场景,其中住宅区场景的特点是含有较多的植被,在该环境下采集数据时,由于影子的原因会产生很多明暗变化的效果;同时该场景中的马路较为规则,很方便形成多个闭环,可以用于测试 SLAM 系统回环检测的能



(a) 住宅区



(b) 城市场景



(c) 工厂厂房



(d) 房间场景

图 6 部分 SLAM 测试场景

Fig. 6 Part of the SLAM test scenarios

力。而在城市场景中,存在很多运动的车和行人,可以用于测试 SLAM 系统应对移动物体时的鲁棒性;同时该场景中的马路交错复杂,所以可以选择合适的路线,在没有闭环的情况下测试 SLAM 系统长期工作时的漂移量。

工厂厂房和房间场景均为室内场景,其中工厂厂房场景较大,而房间场景就相对狭窄。在工厂厂房中有很多移动的机器人,还有很多特征较为相似的货架,这些因素对 SLAM 系统来说具有一定的挑战性。在房间场景中,由于空间比较狭窄,相机移

动时很有可能遇到白墙等特征点较少的情况,这对基于特征点匹配的 SLAM 算法也具有一定的挑战性。同时在房间中光线变化较为明显,这也可以对 SLAM 系统应对室内光照变化的情况进行评测。

在文中生成的仿真数据集中,包含在仿真环境中采集的 RGB-D 图片、深度图片、语义分割图片,还包含相机的真实位姿,其中包括每个采样时间所对应的三维位置和用四元数表示的旋转。

3 测试方法

通过测试方法确定测试场景及如何开展测试。本文选择 8 个典型数据集序列,针对不同情况开展指标测试。它们分别是在室外的住宅区和城市场景以及室内的工厂厂房和普通的房间场景中进行采集,然后针对在室内室外场景、是否存在闭环、是否含有运动模糊等一系列不同的情况,对 SLAM 系统进行测试。其中具体的测试方法如图 7 所示。



图 7 数据集所包含的主要测试方法

Fig. 7 The primary test methods contained in the dataset

在实验中,使用上述仿真环境生成的数据集对 SLAM 算法进行测试。为了对 SLAM 不同方面的性能进行评估,通过使用不同的场景以及不同特点的轨迹对其进行测试。具体的测试方式如下:

1) 选择不同的室内场景与室外场景进行测试。在所选的测试数据集中,室内场景大部分的光线强弱比较均匀,同时还存在一些像墙角和货架等形状规则的结构,这些结构有利于特征的提取及匹配。在室外场景中,由于存在一些植被和建筑,它们的影子会产生明暗变化的效果,所以可以通过这些序列测试 SLAM 算法对光照变化的鲁棒性。

2) 在住宅区场景中,本文通过录制两个相似的长序列,它们的主要区别是一个不存在闭环,而另

一个只是在最后形成一个大的闭环,分别用于测试 SLAM 系统长时间运行的漂移量和检测闭环的能力。

3)同样是在住宅区场景中,本文则录制了一个存在三个闭环的长序列,并且在其基础上加入运动模糊生成另一个与其相似的序列,主要目的是测试视觉 SLAM 系统在应对相机失真情况时的性能表现。

4)在工厂厂房和城市交通场景中存在很多运

动物体,这些运动物体对 SLAM 系统的定位与建图会造成一定的干扰,所以可以通过这些序列对 SLAM 系统应对动态环境时的性能表现进行测试。

5)最后还有一些序列,它们的特点是存在相机的快速移动和旋转,借此可以测试 SLAM 系统在快速移动和旋转情况下跟踪的稳定性。同时还在其中一个序列中插入一段黑色帧,主要目的是测试 SLAM 系统从传感器短暂中断中恢复过来的能力。

每个数据集序列的具体特点如表 2 所示。

表 2 仿真测试数据集
Tab. 2 The simulation test dataset

序列名称	序列特点
NH_NOLOOP	在住宅区场景中最外环的马路环绕一周,但不形成闭环,用于测试 SLAM 系统长时间工作的漂移量
NH_WITHLOOP	与上一个序列相似,在住宅区场景最外环环绕一周,但在开始和结束的时候形成闭环,用于测试 SLAM 系统闭环检测的能力。同时可与 NH_NOLOOP 数据集形成对照组
NH_NOBLUR	在住宅区场景中,围绕内环与外环分别环绕一圈,整个轨迹共有三处存在闭环。同时数据集中的每张图片均为清晰图片。该序列可与 NH_WITHBLUR 数据集形成对照组
NH_WITHBLUR	与上一序列相似,只不过在该数据集中的图片存在运动模糊。该序列主要用于测试 SLAM 系统在存在运动模糊时的性能表现
NH_WITHCOVER	该序列与 NH_NOBLUR 序列相同,只不过在序列中的一部分用与原图像相同大小的黑色图片代替。该序列主要用于测试 SLAM 系统的传感器短暂中断对其性能的影响
FACTORY	在室内工厂场景中,围绕工厂内部进行环绕。同时在该场景中,存在一些移动机器人可以用于测试在室内场景中移动物体对 SLAM 性能的影响
ROOM	在房间场景中,围绕房间进行环绕。在房间中存在明显的光照强弱的变化。同时由于房间比较狭窄,所以存在一些相机纯旋转的情况
CITY	在室外城市场景中,存在较多的建筑物与移动的车辆和行人,相机在其中穿梭,最终形成一个大的闭环。主要用于测试 SLAM 系统在室外场景中移动物体对其性能的影响

目前被广泛使用的 SLAM 系统有很多,而 ORB-SLAM 与其他 SLAM 系统相比,其前端采用 ORB 特征进行提取与跟踪,具有较好的鲁棒性;后端则是基于图优化进行全局位姿的优化,同时还采用词袋模型实现了较为准确的回环检测。因此,本文选择使用 ORB-SLAM 对数据集进行测试,其输出结果如图 8 所示,图中分别表示在住宅区场景、城市交通场景、工厂厂房场景和房间场景中 ORB-SLAM 输出的估计轨迹及点云地图。通过实验结果验证了仿真数据集在实际使用过程中的可行性。

4 多体协同 SLAM 性能

相对于单体 SLAM 系统而言,多体协同 SLAM 系统能够适应更加难以预先判断的动态环境,同时在 SLAM 系统出现故障或者环境中干扰较大时具有更好的鲁棒性^[23]。大多数的多体协同 SLAM 系统采用一种集中式的结构,主要包括由各个子端组

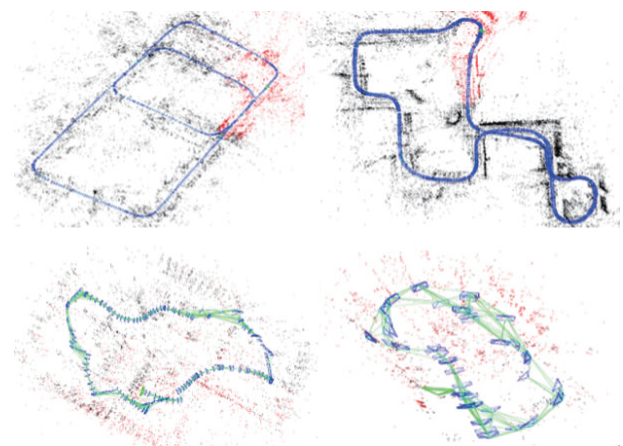


图 8 在不同场景下运行 ORB-SLAM2 得到的输出轨迹
Fig. 8 The output trajectories obtained by running ORB-SLAM2 in different scenarios

成的前端、服务器后端和多体协同工作决策系统。为了对多体协同 SLAM 系统性能进行准确和全面的评估,可以对每个子系统的性能、经过服务器后

端融合优化后的全局状态以及多体协同 SLAM 系统的工作效率三个方面进行评估。

针对不同的子端和服务端优化后的全局状态的评估可以沿用前文提到的单体 SLAM 性能评估指标,但对于它们各自的侧重点有所不同。其中各个子端是整个协同 SLAM 系统的前端,对其进行准确性、稳定性和实时性的评估是十分必要的。针对定位的准确性,可以使用绝对轨迹误差和相对位姿误差进行评估,同时根据其跟踪的稳定性和实时性测试整个系统的鲁棒性。

对于多体协同 SLAM 系统的后端,负责接收各个子端的关键帧信息,并对每个子系统的位姿估计进行优化,对局部点云地图进行融合、去噪与优化,并作为 SLAM 系统的输出。因此,对其整体的定位精度和地图构建结果进行评估也十分重要。

在本文中,使用绝对轨迹误差和各个子系统的相对位姿对融合了各个子端位姿之后的整体定位精度进行评估。绝对轨迹误差是直接测量真实轨迹点和估计轨迹点之间的差值,可以直观地反映全局的位姿误差。对于一个多体协同 SLAM 系统,子系统 i 的第 j 个状态表示为 \hat{x}'_{ij} ,它与对应的真实值 x_{ij} 的误差 Δx_{ij} 满足

$$\Delta x_{ij} = x_{ij} - \Delta \mathbf{R}_j \hat{x}'_{ij} \quad (14)$$

所以,一个完整的系统轨迹的绝对轨迹误差可以定义为

$$ATE = \left(\frac{1}{NM} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \|\Delta x_{ij}\|^2 \right)^{\frac{1}{2}} \quad (15)$$

其中, $\Delta \mathbf{R}_i$ 表示估计值和真实值状态之间的旋转矩阵; N 表示轨迹上所有状态的数量; M 表示子系统的数量。

多体协同 SLAM 的绝对轨迹误差给出了整个系统各个子端的平均误差,便于对系统性能进行最直观测量。

在多体协同定位中,每个子系统与邻近子系统的相对距离信息和方位信息是尤其重要的。因为在使用多体协同 SLAM 系统的过程中,决策算法将尽可能减少系统的冗余,优化分布,提升各个子系统对于空间的利用。而子系统之间的相对位姿关系将是决策算法的重要输入。此外,除了考虑对环境中固定位置的障碍物进行避障,还需避免各个子系统出现动态冲突的问题。此时,准确的相对位姿估计能够为规划系统提供正确的信息,从而有效防止子系统之间出现碰撞。其中各个子系统之间的相对

位姿误差如图 9 所示。为了对各个子系统相对位姿误差进行评估,在系统第 j 个状态时,子系统 i 和子系统 k 的相似变换 $\hat{\xi}'_{ikj}$ 表示为 $\hat{\xi}'_{ikj} = \{\hat{\mathbf{R}}'_{ikj}, \hat{\mathbf{t}}'_{ikj}\}$ 。其中 $\hat{\mathbf{R}}'_{ikj}$ 和 $\hat{\mathbf{t}}'_{ikj}$ 分别代表两个子系统之间的旋转矩阵和平移向量,与对应的系统第 j 个状态的两个子系统的相似变换真实值 ξ_{ikj} 的误差为

$$\epsilon_{ikj} = \left(\|\log_{\text{Sim}(3)}(\hat{\xi}'_{ikj}^{-1} \xi_{ikj})\|^2 \right)^{\frac{1}{2}} \quad (16)$$

所以,一个完整的多体协同 SLAM 系统轨迹的相对位姿误差 RE 可以定义为

$$RE = \frac{1}{2M(M-1)N} \sum_{j=0}^{N-1} \sum_{i=0}^{M-1} \sum_{k=0}^{M-1} \epsilon_{ikj} \quad (k \neq i) \quad (17)$$

其中, M 表示子系统的数量; N 表示运动过程中所有子系统之间的状态数量。

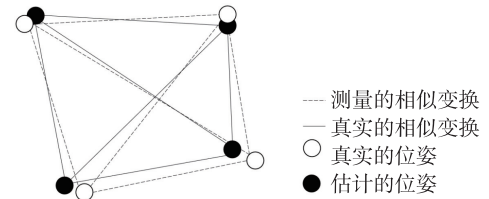


图 9 相对位姿误差状态图

Fig. 9 Relative pose error state map

对于多体协同 SLAM 算法后端融合优化得到的点云地图进行评估时,本文沿用之前对于单体 SLAM 系统 3D 点云地图的指标进行评估,分析了 3D 点云的几何误差、3D 重构能力和稀疏程度等因素。与单体 SLAM 的建图效果进行比较,可得出协同 SLAM 算法的优点与有待改进之处。

最后,提出了对多体协同 SLAM 工作效率的评估方法。相对于单体 SLAM 而言,多体协同系统额外引入了一个组合优化的问题,优化的效率决定了子系统的利用与协同效率。在面对大型或复杂任务时,通常会考虑引入多体协同 SLAM 系统,通过多个子端共同工作,减少了 SLAM 系统运行的时间,从而提高了系统的鲁棒性与任务的执行效率。所以在子系统个数相同的情况下,通过引入运行时间和各个子系统移动的平均路程这两个重要指标来衡量协同 SLAM 系统的任务执行效率。当 SLAM 系统任务完成的时间越短,移动的平均路程越少时,说明此系统具有更高的工作效率。

在时间的计算上,记录多体协同 SLAM 系统开启的时间 t_1 和任务完成的时间 t_2 ,即可得到系统运行的时间

$$T = t_1 - t_2 \quad (18)$$

在路程计算上,为了得到每个系统的行走路程,可以在一段很短的时间 Δt 内近似子系统的 SLAM 运行轨迹为直线。将每个系统的轨迹分成了 N_i 段

$$N_i = \frac{T}{\Delta t} \quad (19)$$

计算每个子端移动的平均距离为

$$\bar{s} = \frac{1}{M} \sum_{i=0}^{M-1} \sum_{j=0}^{N_i-1} \|x_{i,j+1} - x_{i,j}\|^2 \quad (20)$$

其中, M 表示子系统的个数; $x_{i,j}$ 表示子系统 i 的第 j 个状态。

通过对每个子系统性能、后端融合优化后的全局状态以及多体协同 SLAM 系统的工作效率三个方面的评估,可以从系统的精确性、鲁棒性、实时性以及多体协同效率对其做出更加准确和全面的评估。

5 结论

针对 SLAM 领域目前缺少定量准确的评估体系,本文提出了一类 USME 框架,从指标体系、数据集及评测方法三个维度为各种 SLAM 方法的性能度量及比较研究提供基准,主要的贡献如下:

1) 在指标体系中,通过绝对轨迹误差和相对位姿误差对 SLAM 的定位精度进行评估;通过几何精度、3D 重构能力、点云稀疏程度和噪声程度对地图构建进行评估;通过跟踪的鲁棒性和重定位时间对鲁棒性进行评估;以及通过内存使用和能耗等对 SLAM 性能进行评估,从而构成了 SLAM 系统的全面评估体系。

2) 基于 3D 仿真平台生成自己的仿真数据集,以对 SLAM 系统在不同场景中的表现、长时间运行时的漂移量、闭环检测的能力、相机遮挡及光照和运动物体对其性能的影响等方面进行测试。

3) 通过对多体协同 SLAM 系统每个子系统的性能、经过服务器后端融合优化后的全局状态以及多体协同 SLAM 系统的工作效率三个方面进行评估。

希望该评估框架可加快 SLAM 领域在定位精度、地图构建、鲁棒性以及多体协同 SLAM 方向的发展,同时推动视觉导航定位的实际应用。

参考文献

[1] Nardi L, Bodin B, Zia Z, et al. Introducing SLAM-

Bench, a performance and accuracy benchmarking methodology for SLAM[C]// Proceedings of IEEE International Conference on Robotics and Automation (ICRA), 2015.

[2] Bodin B, Wagstaff H, Saeedi S, et al. SLAM-Bench2: multi-objective head-to-head benchmarking for visual SLAM[C]// Proceedings of IEEE International Conference on Robotics and Automation (ICRA), 2018: 1-8.

[3] Zhao Y, Xu S, Bu S, et al. GSLAM: a general SLAM framework and benchmark[C]// Proceedings of 2019 IEEE/CVF International Conference on Computer Vision (ICCV). IEEE, 2019.

[4] Geiger A, Lenz P, Urtasun R, et al. Are we ready for autonomous driving? The kitti vision benchmark suite[C]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2012.

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

[6] 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.

[7] 王常虹, 窦赫暄, 陈晓东, 等. 无人平台 SLAM 技术研究进展[J]. 导航定位与授时, 2019, 6(4): 12-19. Wang Changhong, Dou Hexuan, Chen Xiaodong, et al. Advances in the research of unmanned platform SLAM technology[J]. Navigation Positioning and Timing, 2019, 6(4): 12-19(in Chinese).

[8] Campos C, Elvira R, Rodriguez J J, et al. ORB-SLAM3: an accurate open-source library for visual, visual-inertial and multi-map SLAM [J]. arXiv preprint arXiv: 2007.11898, 2020.

[9] Besl P J, Mckay H D. A method for registration of 3-D shapes[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1992, 14(2): 239-256.

[10] Yang H, Shi J, Carlone L. TEASER: fast and certifiable point cloud registration[J]. arXiv preprint arXiv: 2001.07715, 2020.

[11] 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.

[12] Jatavallabhula M K, Iyer G, Paull L. GradSLAM: dense SLAM meets automatic differentiation [J].

- arXiv preprint arXiv: 1910.10672, 2019.
- [13] Engel J, Schöps T, Cremers D. LSD-SLAM: large-scale direct monocular SLAM[C]// Proceedings of European Conference on Computer Vision. Springer, 2014: 834-849.
- [14] Czarnowski J, Laidlow T, Clark R, et al. Deepfactors: real-time probabilistic dense monocular SLAM[J]. IEEE Robotics and Automation Letters, 2020(99): 1.
- [15] Wen F, Ying R, Gong Z, et al. Efficient algorithms for maximum consensus robust fitting[J]. IEEE Transactions on Robotics, 2019(99): 1-15.
- [16] Li J, Yang B, Cheng D, et al. Survey and evaluation of monocular visual-inertial SLAM algorithms for augmented reality[J]. Virtual Reality & Intelligent Hardware, 2019, 1(4): 386-410.
- [17] Qin T, Li P, Shen S. VINS-mono: a robust and versatile monocular visual-inertial state estimator[J]. IEEE Transactions on Robotics, 2018, 34(4): 1004-1020.
- [18] Reijgwart V, Millane A, Oleynikova H, et al. Voxgraph: globally consistent, volumetric mapping using signed distance function submaps[J]. IEEE Robotics and Automation Letters, 2020, 5(1): 227-234.
- [19] Shi X, Li D, Tian Q, et al. Are we ready for service robots? The OpenLORIS-Scene datasets for lifelong SLAM[J]. arXiv preprint arXiv: 1911.05603, 2019.
- [20] Campos C, Montiel J M M, Tardos J D. Inertial-only optimization for visual-inertial initialization[C]// Proceedings of IEEE International Conference on Robotics and Automation (ICRA), 2020.
- [21] Zou D, Wu Y, Pei L, et al. Structvio: visual-inertial odometry with structural regularity of manmade environments[J]. IEEE Transactions on Robotics, 2019, 35(4): 999-1013.
- [22] Shah S, Dey D, Lovett C, et al. Airsim: high-fidelity visual and physical simulation for autonomous vehicles[J]. Field and Service Robotics, Springer, 2018: 621-635.
- [23] 王璐, 杨功流, 蔡庆中, 等. 多智能体协同视觉 SLAM 技术研究进展[J]. 导航定位与授时, 2020, 7(3): 84-92.
- Wang Lu, Yang Gongliu, Cai Qingzhong, et al. Research progress of multi-agent collaborative visual SLAM technology[J]. Navigation Positioning and Timing, 2020, 7(3): 84-92(in Chinese).
- [24] Handa A, Whelan T, McDonald J, et al. A benchmark for RGB-D visual odometry, 3D reconstruction and SLAM[C]// Proceedings of 2014 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2014: 1524-1531.
- [25] Ros G, Sellart L, Materzynska J, et al. The synthia dataset: a large collection of synthetic images for semantic segmentation of urban scenes[C]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2016: 3234-3243.
- [26] 张礼廉, 屈豪, 毛军, 等. 视觉/惯性组合导航技术发展综述[J]. 导航定位与授时, 2020, 7(4): 50-63.
- Zhang Lilian, Qu Hao, Mao Jun, et al. Overview of the development of visual/inertial integrated navigation technology[J]. Navigation Positioning and Timing, 2020, 7(4): 50-63(in Chinese).

(编辑: 李瑾)