面向场景变化的动态自适应同时定位与地图构建

史殿习^{1,2,3}, 童哲航¹, 杨绍武¹, 张拥军^{2*}, 易晓东^{1,2*}

1. 国防科技大学计算机学院, 长沙 410073;

2. 军事科学院国防科技创新研究院, 北京 100166;

3. 天津(滨海)人工智能军民融合创新中心, 天津 300457

* E-mail: yixiaodong@nudt.edu.cn; zhang_vic@263.com

收稿日期: 2018-06-13; 接受日期: 2018-10-25; 网络版发表日期: 2018-11-21

国家重点研发计划(编号: 2017YFB1001901)和国家自然科学基金(批准号: 91648204)资助项目

摘要 在动态变化的复杂环境中, 如何使同时定位与地图构建(SLAM)系统根据场景变化持续可靠地进行定位和地图构建, 成为其能否走向实际应用的关键所在, 也是目前SLAM研究领域重点研究和待解决的问题之一. 针对这一问题, 本文提出了一个可扩展的、面向场景变化的同时定位与地图构建框架—SceneSLAM, 支持场景检测和SLAM算法的有效复用和自动化调度; 根据当前场景检测结果, 动态自适应地调用依赖不同传感器数据的SLAM算法, 从而提高SLAM系统适应环境变化和持续稳定工作的能力; 在SceneSLAM框架基础之上, 提出了能够有效检测室内、室外、黑暗场景的场景检测模型, 以及应对光线变化的室内、室外场景的动态自适应SLAM模型; 在模型切换时自动地进行坐标转换和尺度转换, 从而获得具备全局一致性和尺度一致性的定位与地图构建结果. 最后, 基于ROS平台实现了一个动态适应场景变化的SLAM原型系统, 实验中利用Turtlebot机器人, 在室内、室外、黑暗场景中进行实验, 验证了本文所提出方法的有效性.

关键词 同时定位与地图构建, 场景检测, 动态自适应, ROS

1 引言

同时定位与地图构建SLAM(Simultaneous Localization and Mapping), 指搭载传感器的机器人在陌生环境中移动时, 在没有类似于环境地图这种先验信息的前提下, 利用传感器对环境的观测, 对当前环境以一种增量的方式动态构建一个地图模型, 同时在地图构建过程中, 确定自身在地图中的位置. SLAM^[1]是机器人领域的经典问题, 是机器人实现自主行为能力的最

核心的关键技术之一. SLAM问题第一次被正式提出是在1995年^[1], 从此之后, SLAM的研究一直是机器人领域的热点. 目前, 比较成熟和流行的SLAM系统有基于激光的GMapping^[2]、基于单目视觉的ORB-SLAM^[3]、基于RGBD相机的RGBD-SLAM^[4]等. 这些SLAM系统大多都是依赖单一的传感器数据, 当运行环境满足相应的条件时, 这些SLAM系统可以很好地完成定位与地图构建任务. 但是, 当SLAM系统需要长时间运行时, 特别是在运行过程中要面对场景变化时,

引用格式: 史殿习, 童哲航, 杨绍武, 等. 面向场景变化的动态自适应同时定位与地图构建. 中国科学: 技术科学, 2018, 48: 1373–1391

Shi D X, Tong Z H, Yang S W, et al. Dynamic adaptive simultaneous localization and mapping technique for scene change (in Chinese). Sci Sin Tech, 2018, 48: 1373–1391, doi: 10.1360/N092018-00197

非常容易失效. 比如, 当机器人从明亮场景进入黑暗场景时或者从室内场景进入室外场景时, 依靠单一传感器的SLAM系统往往无法处理这种场景的变化问题^[5]. 而在机器人的实际应用场景中, 这种场景的变化是非常普遍的. 虽然基于激光的SLAM系统可以运行在大多数的场景中, 但是由于激光提供的感知信息有限, 只能提供最基本的定位与地图构建功能, 不具备语义扩展能力. 目前, 很多研究通过多传感器数据的融合来提升SLAM系统的精确度和鲁棒性^[6,7]. 与单一传感器数据相比, 多传感器数据来自多个相同种类或不同种类的传感器, 并且传感器安装的位置互不相同, 从而使数据具备冗余性和互补性. 因此, 通过多传感器的数据融合, 可以降低数据的不确定性, 增强数据的置信度和精确度, 能够有效提高系统决策、规划的可靠性和鲁棒性, 从而克服单一传感器数据的局限性. 例如, OKVIS^[6]通过融合视觉传感器和IMU惯性导航传感器数据, 实现了一个高精度的SLAM系统. Hess等人^[7]通过融合激光雷达、IMU和里程计数据实现了一个可靠稳定的SLAM系统. 但是, 这些SLAM系统依旧忽略了不同场景对传感器数据可靠程度的影响, 在所有场景下几乎采用相同的融合方式. 因此, 即使采用了多传感器的数据融合方法, 当场景发生变化时, 现有的SLAM系统并不能有效地持续稳定的工作.

在动态变化的复杂环境中, 如何使SLAM系统根据场景变化持续可靠地进行定位和地图构建, 成为目前SLAM研究领域重点研究和待解决的问题之一. 针对这一问题, 本文提出了一种基于多传感器数据、面向不同场景的地图构建技术, 其核心是通过检测机器人当前所在场景类型, 动态自适应地调用依赖不同传感器数据的SLAM算法, 增强SLAM系统适应环境变化的能力, 从而使其更好地持续稳定的进行工作. 本文的主要工作和贡献包括以下几点: (1) 提出了一个可扩展的、面向场景变化的同时定位与地图构建框架—SceneSLAM, 其特点是能够有效支持不同场景检测和SLAM算法的复用和动态自适应调度; (2) 针对室内、室外、黑暗场景检测问题, 提出了一个基于卷积神经网络和贝叶斯滤波优化的场景检测模型, 通过贝叶斯滤波方法为卷积神经网络的分类结果添加图像之间的关联信息, 在检测室内、室外、黑暗场景时具有

良好的准确率和稳定性; (3) 针对场景的动态变化, 提出了一个动态自适应SLAM模型, 设计实现了用于应对黑暗场景的激光SLAM模型和用于应对室外场景和室内场景的视觉SLAM模型; (4) 针对模型切换, 提出了一个全局一致性算法, 实现坐标系和尺度信息的自动转换, 从而得到具备全局一致性的定位和地图信息; (5) 基于ROS平台实现了一个动态适应场景变化的SLAM原型系统, 实验中利用Turtlebot机器人, 在室内、室外、黑暗场景中进行实验, 验证了本文所提出方法的有效性; (6) 对本文实现的SLAM系统进行了开源¹⁾, 以促进提升SLAM系统可靠性和环境变化适应能力等方面的进一步研究.

本文的组织如下: 第2节对SLAM系统环境适应性问题以及场景检测的相关工作进行了分析总结; 第3节描述了具有可扩展能力的SceneSLAM框架; 第4节描述了能够有效检测室内、室外、黑暗场景的场景检测模型; 第5节描述了面向场景变化的自适应SLAM模型; 第6节描述了利用Turtlebot机器人, 在室内、室外、黑暗场景中进行实验及实验结果, 最后, 对全文进行了概括总结.

2 相关工作

2.1 SLAM系统环境适应性问题

目前的SLAM系统能够有效运行, 通常是建立在环境条件长时间不发生变化的假设之上. 因此, 目前的SLAM系统常常由于其所处环境的场景动态变化, 造成系统的崩溃或者定位和地图构建的失败, 其核心是环境适应能力差所导致的, 其原因主要涉及算法和传感器两个层面的问题.

算法层面主要包含数据关联问题和参数调整问题. 数据关联是指将传感器观测数据与场景中的实际物体进行关联, 多个传感器观测数据与环境中同一个物体建立了关联关系之后, 即可以对机器人的位姿进行估计. 数据关联错误大致可以分为两种, 即短时间数据关联错误和长时间数据关联错误. 短时间数据关联错误发生在利用上一时刻的地图和定位结果, 对当前时刻的机器人位姿进行估计的时候, 由于一些环境条件的变化(比如光线变化)或者物体位置的变化(比如

1) <https://github.com/zhehangT/SceneSLAM>

人的走动), 导致针对环境中的同一个物体, 在多次传感器观测中得到不一致的感知数据. 数据关联算法把这些互不相同的感知数据作为来自不同的物体对待, 从而造成数据关联的错误, 进而导致定位与地图构建的失败. 目前, 关于动态环境中数据关联问题, 重点研究和解决的是SLAM系统如何检测、消除或者跟踪动态变化的物体等问题. 主流的方法是通过消除环境中的动态部分的方法^[8]、或者把动态部分作为模型的一部分的方法^[9-11]等. 如果环境变化是周期性的, 则采用SLAM系统对环境构建多个地图^[12,13], 或者基于时间参数进行地图表示^[14,15]等方法. 长时间的数据关联错误主要表现在回环检测问题上, 回环检测用于提高地图和定位结果的全局一致性. 错误的数据关联可能导致建立回环的约束条件是不正确的, 基于这种不正确的约束条件得到的非线性优化结果往往是错误的, 从而造成定位和地图的整体漂移^[16]. 为了减少回环检测错误, 可以通过在执行优化之前, 首先进行错误回环检测并剔除错误, 从而避免产生错误的结果^[17]; 或者在优化期间, 通过验证某些约束条件产生的误差大小的方法, 防止产生错误的回环约束^[18]. 参数调整问题^[5]其一定程度上也是数据关联的具体形式, 因为很多参数设置, 本质上是为了能够得到比较好的数据关联. 比如在ORB-SLAM2中, 对于ORB特征点的提取和匹配有特定的参数配置, 相机内参矩阵的设置是为了建立图像与环境中物体之间的投影约束关系. 一个合适的参数配置, 可以使SLAM系统在特定环境中更加可靠的运行^[19]. 从这个层面上讲, 如果SLAM系统能够自动调整参数, 则能够在一定程度上提高SLAM系统的鲁棒性和适应环境的能力, 文献^[20]在相机的参数调整设置上做了初步的探索与研究.

传感器层面主要包括传感器老化以及传感器工作原理的局限性等方面的问题. 一方面, 由于传感器通常都是由电子元器件组成, 随着传感器的长时间运行, 电子元器件会出现老化现象, 从而导致通过传感器获得的感知数据误差增大, 使得传感器数据异常. 传感器本身出现问题可能已经超出了SLAM研究本身的范围, 但是一个鲁棒的SLAM系统至少可以检测出传感器数据的异常, 从而避免定位和地图构建结果完全错误的情况, 甚至导致系统崩溃的可能性. 因此, SLAM系统如何检测传感器数据异常也是一个有待于深入研究的课题. 另一方面, SLAM系统在工作过程中, 都是利用

传感器获取的周围环境数据进行定位和地图构建, 但是由于受传感器本身工作原理的限制, 机器人携带的传感器在某些场景下会产生失效问题. 例如, 单目相机无法在弱光环境中获得图像, 因为图像是通过感知环境中的光线进行曝光得到的, 当光线过暗时, 整个曝光过程就无法完成; RGBD相机(如Kinect之类的消费级RGBD相机)在室外这种宽阔并且光线强度大的环境中, 往往无法正确获取深度图像. 因此, 受传感器本身工作原理和工作条件的限制, 并不能保证在不同的环境条件下, 传感器都能正常有效地工作, 从而导致依赖相应传感器的SLAM系统, 无法在不同的环境中都能够持续稳定的工作, 甚至崩溃或失效. 在实际应用中, 一个鲁棒的SLAM系统应该具备应对场景变化(如室内、室外、黑暗)的能力, 从而体现出适应环境变化的能力, 目前的SLAM系统主要通过增加数据源的方式来提升对环境变化的适应能力^[21,22].

2.2 场景检测

场景检测通常可以分为基于静态图像的图像分类和基于动态视频流的动态场景分类^[23]. 静态图像分类是指基于单张图像的场景分类问题. 如果智能系统能够理解拍摄图像时的场景类型(如家居、办公、海滩等), 将极大提升机器对图像信息的理解能力, 为理解更复杂的图像内容提供一定的先验信息. 因此静态场景分类被广泛应用于图像检索、物体识别、动作识别等计算机视觉任务. 静态场景分类技术主要经历了三个技术发展阶段^[24].

首先, 是基于底层特征的表示方法. 底层特征通常是基于像素点提取的图像属性, 着重描述图像细微的颜色信息和纹理信息. 常用的底层特征有SIFT特征^[24]、HOG特征^[25]、CENTRIST特征^[26]等. 其次, 是基于高层特征的表示方法. 利用底层特征设计组合模型, 可以提取出图像的高层特征, 这些高层特征蕴含丰富的图像语义信息. 常见的高层特征表示方法有LPR图像特征^[27]、BOP图像特征^[28]和OB图像特征^[29]等. 近年来, 随着面向场景分类问题的大规模数据集的构造^[30,31], 以及基于卷积神经网络^[32]的深度学习方法的广泛应用, 极大地提升了计算机在解决场景分类问题上的能力. 文献^[33]证明, 利用卷积神经网络提取出的视觉特征, 相比于传统的手工特征更具优势. 特别是, 文献^[34]利用卷积神经网络, 在2012年ImageNet^[35]

(Large Scale Visual Recognition Challenge)比赛中, 以远远超过手工特征的成绩获得冠军。在目前最复杂的场景分类数据集Place365^[31]上, ResNet^[36]在测试集上以Top-1正确率54.74%和Top-5正确率85.07%的成绩, 成为目前正确率最高的静态图像场景分类器。

早期的图像场景分类都是基于单张图像, 但是随着移动摄像头广泛应用, 视频流数据正在飞速地产生和积累, 随之产生了基于视频流图像的动态图像场景分类问题。在静态图像分类问题中, 图像与图像之间通常是独立不相关的。与静态图像信息不同的是, 动态图像分类问题中的分类对象是一系列连续图像, 图像和图像之间具有相关的时间信息和空间信息。SA-CNN^[37]首先通过卷积神经网络提取连续图像的特征, 然后计算这些特征的统计聚合(Statistical Aggregation)信息, 得到连续图像的特征描述子(Feature Descriptor), 通过这种方式为单个图像的深度卷积特征添加时间关联信息和空间关联信息, 使其在动态场景分类数据集YUPenn^[38]上取得了最高准确率。文献[39]用卷积神经网络在动态图像场景分类上做了探索性工作, 机器人在环境中移动时, 利用相机接收到的图像是视频流图像。因此, 机器场景检测问题和动态图像场景分类问题十分相似。文献[40]利用卷积神经网络对场景进行分类, 同时通过SLAM算法得到的位置和地图, 构造了带场景标签的语义地图。

3 SceneSLAM框架

为有效支持SLAM系统在不同场景下持续正确稳定地运行, 本文提出了一个可扩展的、面向场景变化的同时定位与地图构建框架—SceneSLAM, 如图1所示。

SceneSLAM框架由传感器数据获取机制、场景检测模型、地图定位与构建模型、场景检测算法与SLAM算法兼容机制、SLAM算法动态自适应调度机制以及地图融合与可视化显示等构成。其中, 传感器数据获取机制根据需求, 获取机器人搭载的各类传感器的数据; 场景检测模型负责对机器人所运行的场景进行实时检测, 并将检测结果传递给自动化调度机制; 地图定位与构建模型负责机器人在运动过程中的自主定位和地图构建; 场景检测算法兼容机制及SLAM算法兼容机制分别为场景检测算法和SLAM算法的复用

图1 SceneSLAM框架结构

和集成提供支撑; 算法自动化调度机制根据场景检测模型的检测结果(如室内、室外、黑暗等), 动态自适应地调用相应SLAM算法进行定位和地图构建; 地图融合及可视化显示机制针对不同场景下、不同SLAM算法构建的地图, 采用坐标一致性算法和尺度一致性算法对地图进行转换和融合, 从而形成全局一致的地图并进行展示。

SceneSLAM框架的特点在于: 既支持对现有SLAM算法和场景检测算法的快速复用, 亦支持新算法对旧算法的替换与扩展, 从而有效地实现对场景检测能力和SLAM能力的提升与扩展; 同时, 基于场景检测结果实现SLAM算法的动态自适应调度, 以达到在不同场景下调用不同SLAM算法进行定位与地图构建的目的。动态自适应调度与场景检测算法和SLAM算法的具体实现弱相关、松耦合, 只与调度关系的配置强相关, 在对场景检测算法和SLAM算法进行替换与升级时(比如提升场景检测算法的精确度), 不影响算法之间的调度关系; 在对场景检测算法和SLAM算法进行扩展时(比如支持更多类型的场景检测), 只需要修改相应的配置信息, 即可实现新的调度关系。通过这种模型调度与模型实现分离的方式, 可以提升整个地图构建系统的灵活性和可复用性。

SceneSLAM框架具有上述特点的关键在于其所提供的面向SLAM算法与场景检测算法的兼容机制, 以及基于场景检测的自动化调度机制这两个核心机制。下面分别对这两个机制进行详细描述。

3.1 SLAM算法与场景检测算法兼容机制

SLAM算法与场景检测算法兼容机制的核心是实现已有SLAM算法和场景检测算法的复用和调用, 并支持新算法对旧算法的升级与扩展, 从而极大地提升SceneSLAM框架的灵活性和可扩展能力. 要实现不同场景下调用不同SLAM算法进行地图构建, 面临如下三个方面的问题.

(1) SLAM系统的模块化问题. 目前, 大多数SLAM算法均以完整的、可以独立运行的SLAM系统的方式开源. 要实现对开源SLAM算法和场景检测算法的统一调度, 以弱相关、松耦合的方式运行在SceneSLAM框架之中, 存在着如何将原本完整的SLAM系统模块化, 以及如何实现场景检测算法的模块化问题.

(2) SLAM算法的调用问题. 不同的SLAM系统对于系统的启动、停止、数据获取和地图可视化显示都有着各自不同的实现方式, 这对SceneSLAM框架实现对不同的SLAM算法以相同的方式进行调用带来阻碍.

(3) 场景检测算法的调用问题. 目前的场景分类算法大多以分类器的方式存在, 应用到机器人平台时, 缺少统一的感知数据获取和返回场景分类结果的实现方式. 要使场景分类算法能运行在SceneSLAM框架之中, 并支持新算法的替换、升级与扩展功能, 需要设计相应的范式模型和模型接口.

针对以上三个问题, 本文分别提出了地图构建范式模型、场景检测范式模型以及基于pluginlib的插件化机制, 并设计实现了相应的SLAM模型接口和场景检测模型接口, 灵活地支持不同算法的动态调用.

3.1.1 地图构建与场景检测范式模型

不同的SLAM系统对系统的启动、停止、数据获取和地图显示都有着各自不同的实现方式. 比如Cartographer使用Rviz来显示地图和轨迹, 而ORB-SLAM2采用Pangolin库来显示地图和轨迹. 这种实现方式上的差异, 导致在对SLAM系统进行插件化封装和设计时, 存在地图构建模型系统的启动、停止、数据获取和地图显示上的差异. 这些差异使得SceneSLAM框架难以对地图构建模型系统进行无差异化调度. 为此, 我们提出了一个抽象的地图构建范式模型来解决这些差异性. 地图构建范式模型主要包括感知数据获取、SLAM算法、地图可视化展现以及模型接口等几

个部分, 如图2所示. 该范式模型是一个规范标准, 定义了对已有SLAM算法的封装方式和需要提供的标准化接口, 在具体实现时将其映射为以插件化方式实现的SLAM模型.

根据范式模型定义的规范标准, 本文将实现面向激光的SLAM模型, 基于Cartographer实现(如图10), 以及面向视觉的SLAM模型(如图11), 具体的实现过程将在5.1节中给出. 基于地图构建范式模型的规范标准, 可以构建多个具备不同能力的SLAM模型, 这些SLAM模型都能够有效地运行于SceneSLAM框架中, 从而实现SceneSLAM框架对不同SLAM算法的无差异化调用和调度.

场景检测范式模型的设计与地图构建范式模型类似, 即实现对场景检测模型的无差异化调度. 场景检测范式模型定义了对场景分类算法的统一封装方式. 基于不同检测算法实现的场景检测模型, 按照场景检测范式模型的约定进行开发, 即能够有效地运行在SceneSLAM框架中. 场景检测模型不需要进行地图显示, 因此当不同的场景检测算法应用到机器人平台时, 只存在感知数据获取方式和检测结果返回方式上的不同. 为此, 场景检测范式模型包含了感知数据获取和场景检测模型接口两部分. 场景检测范式模型的感知数据获取与地图构建范式模型中的感知数据获取类似, 主要是为了从传感器中读取数据, 然后提供给场景检测算法, 实现场景检测的功能, 本文中具体实现的场景检测模型见第4节.

3.1.2 基于pluginlib的插件化机制

基于ROS中的pluginlib插件化机制, 能够有效地实现已有SLAM系统的模块化以及场景检测算法的模块

图2 地图构建范式模型

化. ROS以其开源、松耦合、功能丰富的特性, 得到越来越多机器人研究者的青睐, 特别是其传感器数据获取发布模块、可视化模块以及程序开发工具等. 因此, 借助于ROS可以消除不同的SLAM系统对于数据获取和地图可视化显示的差异. pluginlib包是ROS中的一个C++开发库, 它提供了一系列的C++开发工具, 使得ROS程序可以动态地加载或者卸载软件插件. 这里的软件插件一般是指具备特定功能的C++动态库. 相同类型的软件插件功能往往是相似的, 但是达到目的算法却不一定相同, 通过装载不同的软件插件, 可以使程序获得不同的性能表现. 为方便插件的开发和使用, pluginlib允许程序根据软件插件名称, 实现对软件插件的动态加载. 这一过程不需要对整个程序重新编译, 只需要单独对软件插件进行重新编译. 因此, 通过设计配置文件和相关机制, 可以很方便地实现只更改配置文件中的插件名, 达到使用不同算法的目的. 基于pluginlib的基本工作原理, 针对SLAM算法和场景检测算法的模块化运行问题, 本文提出了基于pluginlib的插件化机制, 其核心是基于地图构建与场景检测范式模型中定义的统一封装方式, 采用框架总线+软件插件的方式, 将感知数据获取、场景识别算法、SLAM算法以及地图展示等机制实现为软件插件; 并通过标准化接口, 将这些插件插入到SceneSLAM框架中. 在运行过程中, 根据应用需求和场景变化, 动态加载和动态自适应地调用, 提升SLAM系统的动态适应环境变化的能力.

基于pluginlib实现算法兼容机制的核心包括两个方面: 首先, 基于pluginlib定义SLAM模型和场景检测模型的接口, 利用这些接口对实际的SLAM算法和场景检测算法进行功能调用, 从而实现对开源SLAM算法和场景检测算法的统一调用. 基于pluginlib实现功能接口和具体算法实现的分离, 只通过插件名进行调用的工作方式, 可以实现SLAM模型和场景检测模型以弱相关、松耦合的方式在SceneSLAM框架之中运行, 并实现模型调度和模型具体实现相分离. 其次, 基于pluginlib通过修改、替换或增加软件插件等多种方式, 实现算法性能的提升和扩展. 当插件算法更新时, 只需单独对软件插件进行重新编译, 而无需对整个程序重新编译. 利用这种机制实现SLAM模型和场景检测模型, 可以有效支持对SLAM模型和场景检测模型的替换升级与优化扩展, 从而方便开发人员基于Sce-

neSLAM框架实现更具鲁棒性、性能更强的SLAM系统.

为了实现对SLAM算法和场景检测算法的统一调用, 并支持SLAM算法和场景检测算法插件化, 本文定义了标准化的SLAM模型接口, 接口定义如图3所示. 其中, Activate接口用于启动SLAM模型和场景检测模型, 使其开始订阅传感器数据, 并根据接收到的传感器数据执行相应的功能; Shutdown接口用于关闭模型, 取消传感器数据的订阅, 并释放资源. SceneSLAM框架支持两种SLAM模型切换模式: 模型间切换和模型内切换; Change接口用于支持模型内切换这种切换模式, 只有当一个SLAM模型支持两种及两种以上不同的传感器类型进行地图构建时(比如既支持单目相机又支持RGBD相机), 才会用到模型内切换; GetPose和SetPose接口为实现全局坐标一致性提供和设置必要的位姿信息, 每个SLAM模型都有各自的局部坐标系, 要将局部坐标系下构建的地图显示在全局坐标系中的正确位置形成全局一致性地图, 需要进行局部坐标系和全局坐标系之间的转换.

场景检测模型接口是为了支持SceneSLAM框架对场景检测模型进行无差异化调度. 场景检测模型接口具体定义如图4所示. 其中, Activate接口和Shutdown接口与SLAM模型中的类似. 而GetScene是场景检测模型中最为重要的接口, 返回scene变量作为场景检测的结果, SceneSLAM框架通过该接口获取场景检测的结果, 并根据该结果对地图构建算法进行动态调度.

3.2 基于场景检测的自动化调度机制

面向开源SLAM算法与场景检测算法的兼容机制, 保证了场景检测模型和SLAM模型都能以软件插件的方式在SceneSLAM框架中运行. 但是要应对复杂环境中的场景变化问题, SceneSLAM框架还应该具备在不

图3 SLAM模型接口

<<Interface>> SceneDetectionInterface
+ scene :String
+ Activate(String) :void + Shutdown() :void + GetScene() :String

图 4 场景检测模型接口

同场景下调度不同SLAM模型(算法)的能力,即根据场景检测模型所返回的检测结果,对不同的SLAM模型实现自动化调度.要使SceneSLAM框架具备这样的能力,需要解决两个方面的问题:首先,要解决的问题是如何建立场景类型与SLAM模型之间的调度关系.这种调度关系作为一种先验信息,主要包含两个内容:一是在当前SceneSLAM框架中,应该使用哪一个场景检测模型,完成场景检测任务;二是在得到场景检测结果时,该调用哪一个SLAM模型(算法).为此,本文采用外部描述文件的方式为SceneSLAM框架提供先验信息,其原因主要出于两个方面.

(1) 描述文件易于修改与扩展.当用新的场景检测模型替换旧的场景检测模型,或者因为增加场景类型需要建立新的调度关系时,只需要修改描述文件即可完成先验信息的更新,而不需要修改和重新编译SceneSLAM框架.这种实现方式提升了基于SceneSLAM框架进行开发的灵活性.

(2) 描述文件易于理解与编写.利用描述文件可以非常清晰地展现当前SceneSLAM框架正在使用哪一个场景检测模型,以及根据场景检测结果该调度哪一个SLAM模型(算法),从而降低了为SceneSLAM框架建立先验信息的复杂性.

在SceneSLAM框架建立的场景类型与SLAM模型之间的调度关系基础上,SceneSLAM框架如何利用建立好的调度关系完成模型调度,是自动化调度机制要解决的第二个问题.如果SceneSLAM框架能够按照调度关系完成对SLAM模型的调度,并且场景类型与SLAM模型之间的调度关系合理有效,则可以避免在某些场景下SLAM算法失效问题.因此,本文提出了一个基于场景检测结果的自动化调度算法,该算法根据先验信息描述文件提供的先验信息,通过SceneSLAM框架提供的场景检测模型和SLAM模型接口,实现基于场景检测结果的SLAM模型的自动调度.自动化调

度算法主要利用一个有限状态机线程,使其能够根据场景类型与SLAM模型(算法)之间的调度关系,对SLAM模型执行实例化、启动与停止等操作.自动化调度算法主要分为如下三个步骤:首先,解析先验信息描述文件,获得场景识别模型的名称,以及场景类型与地图构建模型系统之间的调度关系;其次,根据场景识别模型的名称实例化场景识别模型,通过Activate接口启动场景识别模型执行场景检测,然后通过GetScene接口获取场景检测的结果;最后,启动有限状态机(Finite-state machine, FSM)线程,使其根据场景类型与地图构建模型之间的调度关系,对地图构建模型执行实例化、启动与停止等操作.有限状态机具有五种不同的系统状态,分别为初始化、开始、运行、切换和结束,如图5所示.在有限状态机的运行过程中,会处于这五种状态中的其中一种状态.在每一种状态下,有限状态机会执行不同的动作,并根据执行动作的结果重新设置状态.因此,在有限状态机的运行过程中,其状态并非保持不变,而是在多个状态之间动态切换.借助于有限状态机思想,可以将场景检测、SLAM模型的启动、停止与切换映射为有限状态机的动作,再结合描述文件中的先验信息,可以有效地实现对场景检测模型和SLAM模型的动态自适应调度.

4 面向机器人的自适应场景检测模型

基于SceneSLAM框架实现的SLAM系统能否应对机器人在室内、室外、黑暗场景中切换时的失效问题,实现对SLAM模型(算法)的调度,很大程度上依赖于场景检测模型对室内、室外、黑暗三种场景进行检测的实时性和准确性.针对室内、室外、黑暗场景的场景检测问题,本文提出了一个面向机器人的自适应场景检测模型如图6所示.该模型基于卷积神经网络的场景分类算法,实现对室内、室外、黑暗三种场景进行高效准确的检测,其中,基于卷积神经网络的场景分类算法是执行场景检测功能的主体.

近年来,卷积神经网络已经在计算机视觉中的分类问题上,展现出比基于传统手工特征分类更高的性能优势.在图像分类、物体检测、目标检测、目标跟踪问题的各项基准数据集(Benchmark)上,目前的前沿方法均为基于卷积神经网络的深度学习方法.但是,卷积神经网络在实际的应用中依旧存在诸多问题.

图 5 (网络版彩图)有限状态机

图 6 场景检测模型

首先是卷积神经网络的结构选择问题。随着卷积神经网络展现出强大的图像特征表示能力，卷积神经网络的结构越来越复杂，神经网络的层数越来越深。从早期的AlexNet的8层网络到被广泛使用的ResNet的152层网络，其图像特征的表示能力越来越强，但是，这也同样伴随着其计算复杂度在逐步提升的问题。特别是，当卷积神经网络应用到机器人平台时，受限于机器人平台的计算能力和计算资源，计算复杂度会极大地影响场景检测的效率和实时性。如果机器人平台无法实时并且低功耗地完成场景检测，即使检测的准确率再高，这样的检测模型也并不能满足实际需求。由于室外、室内、黑暗场景的图像分类问题相对上千种

的分类问题来说相对简单，且目前机器人平台的计算能力和计算资源都受限，因此应该在保证分类准确率的前提下，尽可能地降低分类算法的计算开销和计算时间。文献[41]对比了目前主流的卷积神经网络结构在相同计算平台下执行分类计算时的计算速度，结果表明分类能力最强的是Inception-v4和ResNet-152，但它们完成每一次前向传播的耗时达到了100 ms级别，而AlexNet完成一次前向传播耗时不到20 ms。可以看到AlexNet在计算速度上具有明显的优势。为此，本文采用AlexNet作为实现室内、室外、黑暗场景的分类器。

其次是训练数据问题。要使卷积神经网络具备强大的分类能力，必须为卷积神经网络提供大规模的训练数据。ImageNet和Place365这样的基准数据集提供了百万级别的带标签的图像训练数据，为卷积神经网络及其变体的训练提供了前提条件。但是，针对本文所要解决的室内、室外、黑暗场景的分类问题，目前并不存在完全适用的基准数据集。因此，需要对卷积神经网络从零开始训练，得到具备高性能的室内、室外、黑暗场景的分类器。针对缺少室内、室外、黑暗场景分类相关数据集的问题，本文采取了大样本数据预训练、小样本数据微调的方法。文献[33]证明了基于卷积神经网络学习得到的图像特征，可以作为计算机视觉任务中的通用特征。利用大样本数据集进行预训练，实际上就是学习区分物体通用特征的过程；利

用小样本数据集微调,本质上就是基于大样本数据学到的通用特征,再基于小样本数据进行二次训练.这种神经网络的训练模式,在缺少大规模图像数据的领域(如医疗图像领域)被证明是有效的^[42].

大样本数据预训练、小样本数据微调主要分为三个步骤.

首先,是预训练阶段.本文采用基于ImageNet图像数据集对AlexNet进行预训练. ImageNet图像数据集是目前世界上最大的带标签的图像识别数据库,总计超过100万张图片,共有1000个不同的类别.基于ImageNet图像数据集进行预训练,可以充分学习用于图像分类问题的通用特征.

其次,是小样本数据收集阶段.在本文工作中,我们控制机器人在室内、室外、黑暗场景中运动,并通过ROS中的ROS Bag工具采集图像数据.机器人接收到的是视频流图像,通常一秒之内就有几十帧,因此,在一秒之内采集到的图像是高度类似的.这对于卷积神经网络的训练将带来不利影响.为此,在采集图像的过程中,我们采用一秒一张图像的方式进行图像收集,以保证图像与图像之间具有一定的差异性.在完成图像数据采集之后,根据其运动的场景对图像数据进行标签标记,总共标记约1000张带标签的图像数据.室内、室外、黑暗场景的图像数目分别为400张、400张、200张.图像数据的样本如图7所示.室外图像采集于校园、道路、球场等位置;室内图像采集于实验室、教室、图书馆、室内健身房等位置;黑暗场景图像采集于晚上光线不足的时刻,包括室外以及没有开

灯的室内位置.这1000张带标签的图像被随机打乱,并以3:1的比例划分为训练集、验证集.至此,完成小样本室内、室外、黑暗场景图像数据集的构建.

最后,是微调阶段.由于预训练阶段基于ImageNet数据集训练的卷积神经网络用于解决1000个物体的分类问题,因此,网络的最后一层是共有1000个输出的softmax层.针对室内、室外、黑暗场景的分类问题只有3个类别,需要将最后一层修改为具有3个输出的softmax层,并将该层的权重参数重新初始化.利用收集与标注的室内、室外、黑暗场景图像数据集对修改后的softmax层进行重新训练.在这个过程中,神经网络中的其他层的权重参数始终保持不变,只对修改后并且重新初始化的softmax层的权重参数进行训练.从微调的学习过程上看,经过预训练的卷积神经网络在小样本上进行二次训练时,会快速达到收敛效果.如图8所示,横坐标表示迭代次数,绿色线条表示训练集在训练过程中的损失值,红色线条表示训练集在训练过程中的正确率,黄色线条表示验证集在训练过程中的正确率.

为了验证Alexnet分类器的性能,将微调后的Alexnet在测试集上进行测试,得到的正确率如表1所示.在917张场景图片的测试集上进行测试,总正确率达到了94.9%.其中,室内场景图像正确率为96.8%,室外场景图像正确率为93%,黑暗场景图像正确率为94.4%.

为了有效地将Alexnet分类器集成到ScenseSLAM框架中,使其能够在框架中运行,我们将Alexnet分类器封装成具有标准接口的软件插件(具体接口见3.1

图7 室外、室内、黑暗场景图像数据集样本

图 8 迭代次数与训练集验证集的关系

表 1 Alexnet在测试集上的性能

测试集类别	图片张数	正确率(%)
室内	310	96.8
室外	483	93
黑暗	124	94.4
总计	917	94.9

节). 通过框架总线提供的统一接口将其插入到框架当中; 运行时通过标准的加载接口进行加载, 并通过GetScene方法获取场景检测结果, 为SceneSLAM框架根据场景检测结果调用相应的SLAM算法进行同时定位和地图构建提供支持.

5 面向场景的动态自适应SLAM模型

基于RGBD相机的SLAM算法具备提供绝对尺度、无初始化问题、对地图进行语义扩展的能力等诸多优点; 受限于RGBD相机的工作原理, 基于RGBD相机的SLAM算法无法在室外环境正常运行; 基于单目相机的SLAM算法虽然同样具备语义扩展的能力, 但其存在单目初始化、尺度漂移等问题, 并且无法在黑暗场景中运行; 基于激光的SLAM算法可以适用于大多数的应用场景, 但激光传感器只能提供距离信息, 信息量单一, 通常还是二维平面距离, 因此其只能保证最低限度的定位与地图构建能力; 考虑到室内、室外、黑暗三种场景类型以及基于不同传感器的SLAM算法的优缺点, 本文构建了一个面向场景变化的动态自适应SLAM模型, 如图9所示. 该模型是对第3节中SceneSLAM框架中的地图构建模型的具体化和进一步细化. 面向场景变化的动态自适应SLAM模型将场景检测结果与不同模式的SLAM模型有机地结合起来, 在将不同模式的SLAM模型有效集成的同时, 又能够根据场景变化动态自适应地调用不同的SLAM模型(算法). 本文针对室内、室外、黑暗三种场景, 进一步构建和实现了面向激光传感器的SLAM模型和面向视觉传感器的SLAM模型. 视觉SLAM模型用于应对光线充足的场景, 视觉SLAM模型提供两种模式: 基于RGBD相机的深度模式和基于单目相机的单目模式, 分别用于应对室内场景和室外场景的地图构建与定位任务. 激光SLAM模型用于应对黑暗场景, 满足最基本

图 9 基于场景的动态自适应SLAM模型

的地图构建和定位需求。

5.1 面向激光和面向视觉的SLAM模型

目前比较优秀的基于激光的SLAM系统有hector_slam、Gmapping、Cartographer等。Cartographer作为最新开源的激光SLAM系统, 相比于其他基于激光的SLAM系统, 其优势在于能够利用回环检测, 消除地图构建过程中产生的累积误差, 进而提升了基于激光的SLAM系统的准确性和可靠性。通过Cartographer构建的地图理论上的误差只有5 cm。考虑到Cartographer所具有的优越性, 本文基于Cartographer来构建激光地图构建模型, 激光地图构建模型如图10所示, 面向激光的SLAM模型是第3.1.1节中的地图构建范式模型的具体映射和实现。

视觉SLAM模型用于应对光线充足环境下的地图构建与定位任务。在光线充足环境下采用基于视觉传感器的SLAM算法, 可以充分利用视觉图像提供的丰富信息, 实现对三维空间的地图构建和定位, 并提供语义扩展的应用前景。ORB-SLAM2基于传统的图像特征匹配方法进行地图构建与定位, 在各个视觉SLAM基准数据集上都表现出良好的精确度和实时性, 并且支持单目相机、双目相机和RGBD相机三种视觉传感器。考虑到ORB-SLAM2的优越性以及其支持传感器类型的丰富性, 本文基于ORB-SLAM2实现面向视觉传感器的SLAM模型, 并提供基于单目相机和基于RGBD相机的两种工作模式, 分别应对室外场景和室内场景, 如图11所示。

为了有效地将Cartographer集成到ScenseSLAM框架中, 我们将Cartographer封装成一个具有标准接口的软件插件(具体接口见3.1节), 将其编译注册为软件插件, 通过框架总线提供的统一接口将其插入到框架总线当中。Cartographer运行时通过标准的加载接口进行加载, 并且根据场景检测结果对其进行调用。与激光SLAM模型类似, ORB-SLAM2也被封装成一个具有标准接口的软件插件。由于视觉SLAM模型支持两种工作模式, 因此除了Activate、Shutdown、SetPose和GetPose接口外, 还实现了Change接口、执行模型内切换(具体接口见3.1节)。在此基础上, 通过框架总线提供的统一接口将其插入到框架总线当中。运行时通过标准的加载接口对其进行加载, 根据场景检测结果对其进行调用。

图10 激光地图构建模型

5.2 动态自适应地图全局一致性融合

机器人在同时定位和地图构建过程中, 根据场景变化, 激光地图构建模型和视觉地图构建模型之间发生模型间切换时, 会导致坐标系不一致; 同时, 基于单目视觉构造的地图和定位信息不具备绝对尺度。上述两个问题将最终导致地图信息与定位信息在全局坐标系下的不一致性问题。针对这一问题, 本文提出了一个地图全局一致性算法, 包括坐标系一致性算法和尺度一致性算法两个部分, 通过地图全局一致性算法实现不同模型系统构建地图的融合。

5.2.1 坐标一致性转换算法

不同模式的SLAM算法都是在各自的局部坐标系中进行地图构建与定位, 激光模式SLAM和视觉模式SLAM均有各自的地图坐标系, 称为局部坐标系。因为缺少全局信息, 当直接对地图与定位信息在全局坐标系下进行可视化显示时, 会存在坐标不一致问题。为保证地图在全局视图下具备全局一致性, 本文提出了

图11 视觉地图构建模型

一个坐标转换算法, 使得每个SLAM模型在局部坐标系中构建的地图, 能够正确转换到其在全局坐标系下的位置. 坐标一致性算法借助于GetPose和SetPose接口实现, 其基本原理是在每一次执行模型切换时, 重新计算机器人在全局坐标系下的位姿. SLAM模型根据启动该模型时机器人在全局坐标系下的位姿, 可以将每一时刻机器人在局部坐标系中的位姿, 转换为其在全局坐标系中的位姿. 不妨用 $\mathbf{T}_{\text{conversion}}$ 表示在执行模型切换时, 机器人在全局坐标系下的位姿. $\mathbf{T}_{\text{conversion}}$ 表示一个变换矩阵(Transform Matrix), 变换矩阵是描述刚体运动常用的表达方式. 描述一个刚体运动需要两方面的信息: 旋转信息和平移信息. 通过变换矩阵 $\mathbf{T}_{\text{conversion}}$ 可以恢复出旋转矩阵 $\mathbf{R}_{\text{conversion}}$ 和平移向量 $t_{\text{conversion}}$. 假设通过SLAM模型得到在局部坐标系下的机器人位姿为 $\mathbf{T}_{\text{local}}$ 及其坐标为 c_{local} , 其中 c_{local} 即为 $\mathbf{T}_{\text{local}}$ 中的平移向量. 因此, 通过 $\mathbf{T}_{\text{conversion}}$ 即可计算在全局坐标系下的机器人位姿 $\mathbf{T}_{\text{global}}$ 和坐标 c_{global} . 计算公式如式(1)所示. SceneSLAM框架将通过SetPose接口为每一个SLAM模型设置 $\mathbf{T}_{\text{conversion}}$.

$$\mathbf{T}_{\text{global}} = \mathbf{T}_{\text{conversion}} \cdot \mathbf{T}_{\text{local}}. \quad (1)$$

在SceneSLAM框架因为场景变化, 需要执行模型间切换时, 对变换矩阵 $\mathbf{T}_{\text{conversion}}$ 进行更新. 初始状态下, 机器人的全局坐标为原点坐标, 因而可以设置为单位矩阵. 因此, 第一个被调用的SLAM模型的局部坐标系等同于全局坐标系. 当检测到场景变化并且SceneSLAM框架执行模型间切换时, 进行以下几个操作: 首先, 在停止当前的SLAM模型前, 通过GetPose接口获取机器人此时在局部坐标系下的位姿. 其次, 根据该位姿对 $\mathbf{T}_{\text{conversion}}$ 进行更新, 从而得到此时机器人在全局坐标系下的位姿. 当前SLAM模型停止时, 机器人在全局坐标系下的位姿, 恰好是下一个SLAM模型启动时, 机器人在全局坐标系下的位姿. 最后, 通过Shutdown接口和Activate接口, 关闭当前SLAM模型, 启动新的SLAM模型, 完成模型间切换, 并将此时机器人在全局坐标系下的位姿通过SetPose接口传递给新启动的SLAM模型.

5.2.2 尺度一致性算法

在单目模式SLAM初始化过程中, 通过单应矩阵或本质矩阵求解得到表示机器人运动的变换矩阵时,

具有尺度等价性, 即当变换矩阵乘以任意非零常数时, 矩阵求解依旧成立. 这个问题被称为单目视觉的尺度不确定性问题. 但是, 在基于激光和RGBD相机的SLAM算法中, 由于其传感器可以直接测量机器人与环境中物体之间的绝对距离, 因此不存在尺度不确定性问题. 这会直接导致在视觉SLAM模型单目模式下构建的地图, 与在深度模式下构建的地图或者激光SLAM模型构建的地图之间存在尺度不一致的问题, 从而导致定位与地图信息在全局坐标系下的不一致. 针对这个问题, 本文提出了一种尺度一致性算法, 尺度一致性算法的基本原理是: 当从视觉SLAM模型的单目模式切换到深度模式时, 即机器人的工作环境从室外场景进入到室内场景, 此时, RGBD相机已经可以正常工作并获取深度图像. 因此, 在停止单目模式之前, 根据当前的深度图像(具备绝对尺度)以及从单目图像中计算得到的地图三维特征点坐标(相对尺度), 计算两者之间的尺度因子. 通过该尺度因子, 对之前基于单目图像计算获得的所有地图三维特征点坐标和关键帧坐标进行尺度缩放, 使其统一到绝对尺度之下. 从单目模式切换到激光SLAM模型时, 其原理类似, 主要区别在于需要对激光数据和单目图像数据进行空间标定.

6 原型实现与实验结果

基于Turtlebot机器人硬件平台, 我们以C++语言和ROS实现了一个能够有效应对室内、室外、黑暗三种场景类型, 并根据场景类型调用不同SLAM模型的原型系统, 其架构如图12所示.

实验中的Turtlebot配备了Kinect v1 RGBD相机传感器, 搭载了一台运行ROS Indigo的笔记本电脑, 处理器为i7-4710MQ, 内存为12 GB. 原型系统实验平台的详细软硬件配置如下: Turtlebot 机器人包括Microsoft Kinect v1, kobuki移动底座(包括里程计、IMU), 支架; 笔记本电脑包含: CPU(i7-4710MQ), RAM(金士顿DDR3L, 8G+4G), 硬盘(64G SSD), OS(Ubuntu14.04), 开发环境(robware)等.

本文实验中使用的Turtlebot机器人如图13(a)所示. 实验过程中控制Turtlebot机器人在室内、室外、黑暗场景中移动, 室内场景、室外场景和黑暗场景如图13(b)所示.

图 12 原型系统总体架构

为了验证SceneSLAM框架和本文实现的场景检测模型以及自适应SLAM模型在应对室内、室外、黑暗场景变化时,根据场景检测结果对SLAM模型进行调度的能力,实验中控制Turtlebot机器人分别在室内、室外、黑暗场景下移动,并利用本文实现的原型系统执行地图构建与定位任务。

首先,控制Turtlebot机器人在室内场景下移动,并启动原型系统执行地图构建与定位任务。此时,视觉SLAM模型的深度模式被调用,即基于RGBD相机的感知数据进行地图构建与定位,机器人的运动轨迹和构建得到的地图如图14(a)所示,机器人所在的环境如图14(b)所示。通过视觉SLAM模型的深度模式,构建了当前场景的三维特征点地图,即图14(a)中的红色点与黑色点。红色点表示在当前图像中可看到的三维特

征点,黑色点表示在当前图像中看不到的三维特征点,绿色线条表示当前机器人的运动轨迹。

其次,控制Turtlebot机器人在室外场景下移动。此时,视觉SLAM模型的单目模式被调用,即基于单目相机的感知数据进行地图构建与定位,机器人的运动轨迹和构建得到的地图如图15(a)所示,机器人所在的环境如图15(b)所示。与视觉SLAM模型的深度模式类似,单目模式下构建了当前场景的三维特征点地图,黄色线条表示当前机器人的运动轨迹。

最后,控制Turtlebot机器人在黑暗场景下移动。此时,激光SLAM模型被调用,即基于激光数据进行地图构建与定位。在实验过程中,利用深度图像对激光数据进行模拟,同时利用ROS提供的相关驱动模块,将深度图像数据转化为激光数据。在黑暗场景中移动时,机器

图 13 Turtlebot机器人以及实验环境。(a) Turtlebot; (b) 实验环境

图 14 在室内场景移动时构建的地图与轨迹. (a) 深度模式下构建的地图与轨迹; (b) 室内场景

人的运动轨迹和构建得到的地图如图16(a)所示, 机器人所在的环境如图16(b)所示, 黑暗场景为晚上没有开灯的房间. 通过激光SLAM模型构建得到的是二维的栅格地图, 地图显示的白色越明显, 表示当前的二维平面没有障碍物的概率越高; 地图显示黑色时, 表示当前位置有障碍物; 蓝色线条表示当前机器人的运动轨迹.

SceneSLAM框架不仅具备不同场景下调用不同

SLAM模型的能力, 还具备在检测到场景切换时, 对SLAM模型进行切换的能力. 为了验证SceneSLAM框架具备的这种能力, 本文通过控制Turtlebot机器人从室内光线明亮场景进入到黑暗场景进行实验验证, 机器人所在的实验环境, 如图17所示. 机器人正前方没有开灯的房间为黑暗场景. 在进入到没有开灯的房间之前, 场景检测模型检测到室内场景, 此时调用视觉SLAM模型的深度模式执行地图构建与定位任务. 因

图 15 在室外场景移动时构建的地图与轨迹. (a) 单目模式下构建的地图与轨迹; (b) 室外场景

图 16 在室内场景移动时构建的地图与轨迹. (a) 激光SLAM模型构建的地图与轨迹; (b) 黑暗场景

此, 在进入到黑暗房间之前, 原型系统产生的是三维特征点地图, 绿色线条为此时视觉SLAM模型产生的机器人轨迹. 由于此时视觉SLAM模型构建的地图是初始地图, 因此其局部坐标系即为当前的全局坐标系. 在进入到黑暗房间之后, 场景检测模型检测到黑暗场景, 此时SceneSLAM框架执行模型间切换, 即停止视觉SLAM模型, 启动激光SLAM模型. 模型间切换时, 会产生局部地图无法显示在全局坐标系中的问题, 因此SceneSLAM框架执行坐标一致性算法, 使得激光SLAM模型的局部坐标系正确的显示在全局坐标系中. 切换到激光SLAM模型后, 开始构建二维的栅格地图, 并产生蓝色轨迹. 机器人的运动轨迹与地图构建结果如图18所示, 从图18中可以看出, 利用视觉SLAM模型和激光SLAM模型, 构建了一个混合地图, 并得到了具备全局一致性的机器人运动轨迹.

当SceneSLAM框架因为场景变化执行SLAM模型切换时, 会发生地图和定位信息不一致的问题, 地图的不一致包含坐标系不一致和尺度不一致两个方面. 坐标系不一致问题主要发生在模型间切换时, 如图19所示. 视觉SLAM模型与激光SLAM模型在进行地图构建与定位时都有各自的局部坐标系, 如果不将局部坐标系转化为全局坐标系, 会导致每次启动一个地图构建模型, 其构建的地图和定位结果都是基于坐标原点, 因而丢失了全局信息. 坐标系一致性算法的作用是保留机器人在模型切换时的全局位置信息, 将局部坐标系转换到全局坐标系下, 坐标系一致性算法的有效性如图20所示. 从视觉SLAM模型切换到激光SLAM模型时, 视觉SLAM模型的结束位置即为激光SLAM模型的开始位置, 坐标一致性算法利用这个位置信息将激光SLAM模型的局部坐标系转换到全局坐标系下, 形成了具备全局一致性的地图和机器人运动轨迹, 通过两个SLAM模型构建的地图和轨迹实现了全局坐标

系下的融合.

基于单目相机的SLAM系统具有尺度不确定性, 即在视觉SLAM模型的单目模式下, 构建的地图和定位信息不具备绝对尺度. 图21体现出尺度不确定性问题, 图21(a)利用深度模式对机器人所处环境进行地图构建与定位, 图21(b)对同一环境用单目模式进行地图

图 18 室内明亮场景切换到黑暗场景时的地图与轨迹

图 19 坐标系不一致问题

图 20 坐标一致性全局地图

图 17 室内明亮场景与黑暗场景切换时的实验环境

图 21 (a) 深度模式地图构建; (b) 单目模式地图构建

构建与定位. 对比图21(a)和(b)可以发现, 单目模式和深度模式对相同的环境进行地图构建与定位时, 存在明显的尺度上的差异. 因此在停止单目模式之前, 利用可以获取的深度图像(具备绝对尺度), 计算相对尺度与绝对尺度之间的关系, 即尺度因子; 根据计算得到的尺度因子, 对所有的三维特征点的坐标和关键帧的坐标进行缩放, 使其统一到绝对尺度之下, 如图22所示. 对比图21(a)和图22可以得出, 尺度一致性算法应用到单目模式构建的地图和定位信息之后, 地图和定位信息的尺度已经接近于绝对尺度.

图 22 尺度一致性地图构建

7 结论

本文提出了一种面向场景变化的同时定位与地图构建方法, 通过检测机器人当前所在的场景类型, 调用依赖不同传感器数据的SLAM模型, 有效地避免了SLAM系统在某些场景下的失效问题, 从而有效提升了SLAM系统在复杂场景下的可靠性和适应环境的能力. 概括起来, 本文提出了一个面向场景变化的同时定位与地图构建框架即SceneSLAM框架、面向机器人的场景检测模型以及面向场景变化的动态自适应

SLAM模型, 并基于ROS和Turtlebot硬件平台实现了原型系统. 通过在室内、室外、黑暗场景下反复实验, 验证了本文所提方法的有效性. 由于机器人实际工作环境的复杂性, 本文的研究工作还存在一些不足, 有待进一步的研究: 探索混合地图在定位与导航中的进一步应用; 对场景自适应的混合SLAM系统的性能进行测试和优化, 尤其是对混合模式与单独模式的性能差异进行对比测试; 进一步探索模型间的回环检测和位姿优化, 从而提升系统的性能等.

参考文献

- 1 Durrant-Whyte H, Bailey T. Simultaneous localization and mapping: Part I. *IEEE Robot Autom Mag*, 2006, 13: 99–110
- 2 Grisetti G, Stachniss C, Burgard W. Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Trans Robot*, 2007, 23: 34–46
- 3 Mur-Artal R, Montiel J M M, Tardos J D. ORB-SLAM: A versatile and accurate monocular SLAM system. *IEEE Trans Robot*, 2015, 31: 1147–1163

- 4 Endres F, Hess J, Sturm J, et al. 3-D mapping with an RGB-D camera. *IEEE Trans Robot*, 2014, 30: 177–187
- 5 Cadena C, Carlone L, Carrillo H, et al. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. *IEEE Trans Robot*, 2016, 32: 1309–1332
- 6 Leutenegger S, Lynen S, Bosse M, et al. Keyframe-based visual-inertial odometry using nonlinear optimization. *Int J Robot Res*, 2014, 34: 314–334
- 7 Hess W, Kohler D, Rapp H, et al. Real-time loop closure in 2D LIDAR SLAM. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2016. 1271–1278
- 8 Neira J, Tardos J D. Data association in stochastic mapping using the joint compatibility test. *IEEE Trans Robot Autom*, 2001, 17: 890–897
- 9 Rosen D M, Mason J, Leonard J J. Towards lifelong feature-based mapping in semi-static environments. In: 2016 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2016. 1063–1070
- 10 Walcott-Bryant A, Kaess M, Johansson H, et al. Dynamic pose graph SLAM: Long-term mapping in low dynamic environments. In: 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2012. 1871–1878
- 11 Wang C, Thorpe C, Thrun S, et al. Simultaneous localization, mapping and moving object tracking. *Int J Robot Res*, 2007, 26: 889–916
- 12 Dayoub F, Cielniak G, Duckett T. Long-term experiments with an adaptive spherical view representation for navigation in changing environments. *Robot Auton Syst*, 2011, 59: 285–295
- 13 Konolige K, Bowman J, Chen J, et al. View-based maps. *Int J Robot Res*, 2010, 29: 941–957
- 14 Krajník T, Fentanes J P, Mozos O M, et al. Long-term topological localisation for service robots in dynamic environments using spectral maps. In: 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2014. 4537–4542
- 15 Santos J M, Krajník T, Fentanes J P, et al. Lifelong information-driven exploration to complete and refine 4D spatio-temporal maps. *IEEE Robot Autom Lett*, 2016, 1: 684–691
- 16 Sunderhauf N, Protzel P. Towards a robust back-end for pose graph SLAM. In: 2012 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2012. 1254–1261
- 17 Sabatta D, Scaramuzza D, Siegwart R. Improved appearance-based matching in similar and dynamic environments using a vocabulary tree. In: 2010 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2010. 1008–1013
- 18 Carlone L, Censi A, Dellaert F. Selecting good measurements via l1 relaxation: A convex approach for robust estimation over graphs. In: 2014 IEEE/RSJ International Conference on Robots and Systems (IROS). IEEE, 2014. 2667–2674
- 19 Yang N, Wang R, Gao X, et al. Challenges in monocular visual odometry: Photometric calibration, motion bias and rolling shutter effect. *IEEE Robot Autom Lett*, 2018, 3: 2878–2885
- 20 Keivan N, Sibley G. Online SLAM with any-time self-calibration and automatic change detection. In: 2015 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2015. 5775–5782
- 21 Yousif K, Taguchi Y, Ramalingam S. MonoRGBD-SLAM: Simultaneous localization and mapping using both monocular and RGBD cameras. In: 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2017. 4495–4502
- 22 Wan G, Yang X, Cai R, et al. Robust and precise vehicle localization based on multi-sensor fusion in diverse scenes. In: The Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2017
- 23 Lowe D G. Distinctive image features from scale-invariant keypoints. *Int J Comput Vision*, 2004, 60: 91–110
- 24 李学龙, 史建华, 董永生, 等. 场景图像分类技术综述. *中国科学: 信息科学*, 2015, 45: 827–848
- 25 Dalal N, Triggs B. Histograms of oriented gradients for human detection. In: The Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. San Diego, 2005. 886–893
- 26 J Wu, Rehg J M. CENTRIST: A visual descriptor for scene categorization. *IEEE Trans Pattern Anal Mach Intell*, 2011, 33: 1489–1501
- 27 Sadeghi F, Tappen M F. Latent pyramidal regions for recognizing scenes. In: The Proceedings of European Conference on Computer Vision. Florence, 2012. 228–241
- 28 Juneja M, Vedaldi A, Jawahar C V, et al. Blocks that shout: Distinctive parts for scene classification. In: The Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Portland, 2013. 923–930
- 29 Li L J, Su H, Li F F, et al. Object bank: A high-level image representation for scene classification & semantic feature sparsification. In: The Proceedings of Advances in Neural Information Processing Systems. Vancouver, 2010. 1378–138
- 30 Xiao J X, Ehinger K A, Hays J, et al. Sun database: Exploring a large collection of scene categories. *Int J Comput Vision*, 2016, 119: 3–22

- 31 Zhou B, Lapedriza A, Khosla A, et al. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 2018, 40: 1452–1464
- 32 Lecun Y, Bengio Y, Hinton G. Deep learning. *Nature*, 2015, 521: 436
- 33 Donahue J, Jia Y, Vinyals O, et al. DeCAF: A deep convolutional activation feature for generic visual recognition. In: *The Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2013. 1-647
- 34 Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks. In: *The Proceedings of International Conference on Neural Information Processing Systems*. Curran Associates Inc, 2012. 1097–1105
- 35 Russakovsky O, Deng J, Su H, et al. ImageNet large scale visual recognition challenge. *Int J Comput Vision*, 2015, 115: 211–252
- 36 He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. In: *The Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2016. 770–778
- 37 Gangopadhyay A, Tripathi S M, Jindal I, et al. SA-CNN: Dynamic scene classification using convolutional neural networks. *Computer Science*, 2015, 1287–1295
- 38 Derpanis K G, Lecce M, Daniilidis K, et al. Dynamic scene understanding: The role of orientation features in space and time in scene classification. In: *The Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012. 1306–1313
- 39 Karpathy A, Toderici G, Shetty S, et al. Large-scale video classification with convolutional neural networks. In: *The Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2014. 1725–732
- 40 Sünderhauf N, Dayoub F, McMahon S, et al. Place categorization and semantic mapping on a mobile robot. In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015. 5729–5736
- 41 Canziani A, Paszke A, Culurciello E. An analysis of deep neural network models for practical applications. In: *The Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2016
- 42 Esteva A, Kuprel B, Novoa R A, et al. Dermatologist-level classification of skin cancer with deep neural networks. [Nature](#), 2017, 542: 115–118

Dynamic adaptive simultaneous localization and mapping technique for scene change

SHI DianXi^{1,2,3}, TONG ZheHang¹, YANG ShaoWu¹, ZHANG YongJun² & YI XiaoDong^{1,2}

¹ School of Computer Science, National University of Defense Technology, Changsha 410073, China;

² Research Institute of Defense Technology, Military Academy of Sciences, Beijing 100166, China;

³ Tianjin Artificial Intelligence Innovation, Tianjin 300457, China

How to make simultaneous localization and mapping (SLAM) system to be robust and reliable in complex application scenarios has been widely regarded as a key part of automatic and practical robots. Robustness of SLAM system is one of the research focuses in SLAM field. To tackle this issue, we present multi-scene SLAM technique in this paper, which is a novel extensible SLAM framework by combing different SLAM systems facilitated by a scene detection method. Firstly, we propose a robust and extensible SLAM framework called SceneSLAM to enhance the self-adaptive performance of current SLAM systems. By model structure design, it is easy to utilize open-source SLAM systems to be SLAM models of SceneSLAM and modify the scene detection models to support complex application scenarios. Based on the scene detection results, it can schedule the SLAM models automatically. Secondly, aiming at indoor, outdoor and dark scene detection problems, we present a scene detection model based on convolutional neural network and Bias filter optimization. Experimental results show that the proposed scene detection model has reliable accuracy and stability in detecting indoor, outdoor and dark scenes. What's more, considering sensor failure in indoor, outdoor and dark scenes, we design dynamic adaptive SLAM models to handle the issue. Map fusion is performed to achieve globally consistent localization and mapping when SLAM models scheduled. At last but not least, we build up a prototype system based on SceneSLAM to enhance the reliability of existing SLAM systems when the scene changes between indoor, outdoor and dark scenes. This system runs on a TurtleBot robot equipped with a Kinect sensor. The experimental results show that multi-scene SLAM technique proposed in this paper provides an effective solution to enhance the robustness of existing SLAM systems in dynamic environments.

SLAM, scene detection, dynamic adaptive, ROS

doi: [10.1360/N092018-00197](https://doi.org/10.1360/N092018-00197)