****

本科生毕业设计(论文)附件

外文文献原文和译文

复杂环境下移动机器人

题 目： \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

即时定位和地图构建研究

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

赵岩

作 者： \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

U202143381

学 号： \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

高等工程师学院

学 院： \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

机器人工程

专 业： \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2025 年 03 月

目 录

[附件A：外文文献的中文译文 1](#_Toc4764743)

[附件B：外文文献的外文原文 35](#_Toc4764744)

附件A：外文文献的中文译文

|  |
| --- |
| **外文文献原文的文献著录信息：**  Li J, Luo J. Approach to 3D SLAM for Mobile Robot Based on RGB-D Image with Semantic Feature in Dynamic Environment[J]. Journal of Intelligent & Robotic Systems, 2023, 109(1): 15. |
| **外文文献著录信息的中文翻译：**  Li J, Luo J. 动态环境下基于RGB-D图像语义特征的移动机器人三维SLAM方法[J]. 智能与机器人系统杂志, 2023, 109(1): 15. |

摘 要

针对移动机器人在动态环境中定位精度低和建图一致性差的问题，本文提出了一种基于RGB-D（红绿蓝和深度）图像与语义特征的三维同步定位与建图（SLAM）算法。前端使用Mask R-CNN（基于区域的卷积神经网络）进行语义分割，并结合特征点的深度和距离信息，通过随机采样获取具有较高置信度的静态特征点。然后，通过匹配相邻帧的动态特征点得到相对静态点，并将其加入当前帧的跟踪和优化中，最终获得最小化的重投影误差和机器人的最优位姿。为了适应动态特征点变化的不确定性，本文提出了一种高性能的关键帧选择方法。后端结合语义信息和词袋模型（BOW）构建了闭环检测策略。根据是否形成闭环，采用基于图优化和因子图优化的组合优化策略，提高了算法的实时性。最后，利用语义分割后获得的带有语义标签的图像，构建环境的语义八叉树地图。通过一系列仿真与实验验证了所提算法的优越性能。

关键词： 动态环境、移动机器人、SLAM、语义分割、语义八叉树地图

1. 引 言

目前，视觉同步定位与建图（SLAM）在动态环境中已经成为智能机器人领域的前沿研究热点。然而，大多数视觉SLAM算法都集中在静态环境下，例如KinectFusion[1]、Large-Scale Direct Monocular SLAM[2]、ORB-SLAM2[3]、Direct Sparse Odometry[4]等。在实际环境中，会有不同种类的动态物体，如行人、车辆等。如果将这些物体提取的动态特征点应用于SLAM算法，将会导致机器人位姿估计出现显著的误差积累，且构建的地图与实际环境之间存在不一致。因此，基于RGB-D（红绿蓝和深度）相机的动态SLAM算法应运而生。随着深度学习和机器人技术的发展，一些研究者已经开展了将深度神经网络与视觉SLAM结合的相关工作。

Zhao 等[5]人采用语义网络对运动物体进行分割，并将分割结果与物体的实际边界融合，从而获得更高精度的掩膜，并有效地去除动态特征点。Xie 等人[6]使用掩膜修复方法来克服语义分割的不足，并利用Lucas-Kanade（LK）光流法动态识别被动移动物体，最终实现了分离动态物体的效果。Yuan 等人[7]通过利用深度和语义信息改进后端优化算法，找出动态特征点，从而提高了动态环境中的位姿估计精度。Li 等人[8]采用语义分割网络和极坐标几何约束，利用贝叶斯模型迭代更新动态特征点，获得高置信度的动态特征点。Wen 等人[9]通过综合考虑重投影误差、光学误差和深度误差的权重，利用语义信息构建语义地图。Cheng 等人[10]结合语义信息和几何约束去除动态特征点，并在跟踪过程中记录动态物体的运动轨迹，这有助于避障等应用研究。

尽管这些方法都具有较高的语义分割精度，但它们都需要较好的实时性。因此，其他研究者在提升算法实时性方面做了一些工作。为了减少环境中动态物体的干扰，Yu 等人[11]将语义分割网络（SegNet）与运动一致性检查方法结合，构建了基于语义信息的语义八叉树地图。Vincent 等人[12]提出了一种使用轻量级YOLACT（You Only Look At Coefficients）[13]实例分割方法，结合卡尔曼滤波实现动态环境下的定位与建图，不仅提高了算法的精度，还满足了实时性的要求。考虑到语义分割的实时性问题，Liu 等人[14]不仅对关键帧进行语义分割，并使用光流法进行预测，从而提高了算法的运行速度，还利用速度估计减少了动态物体的影响。Zhang 等人[15]使用轻量级网络，利用点和线特征对特征点进行描述，随后通过K-means算法过滤动态点，并通过重投影误差和深度信息优化模型，从而显著提高了算法的实时性和精度。Yan 等人[16]提出了一种基于多项式残差模型和语义分割模块的方法，该方法通过提取潜在动态物体的语义帧而非关键帧，能够更准确地去除动态物体。

然而，实时语义分割过于依赖GPU（图形处理单元）。为了克服这一限制，一些研究者借鉴了贝叶斯网络模型、静态加权等方法去除动态物体，使得动态SLAM算法在CPU（中央处理单元）上也能满足实时性要求，并保持较好的精度。

*Li* 等人[17]根据深度信息获取边缘点，并通过*t*分布静态加权估计静态点，利用与强度相关的迭代优化进行加权，从而有效地分离动态特征点和静态特征点。*Cheng* 等人[18]利用相邻帧之间的相似性和当前帧与参考帧之间的差异，采用贝叶斯框架生成最优后验概率，从而检测和去除动态区域。*Cheng* 等人[19]设定了物体动态的先验信息，使用动态权重选择动态区域，然后通过贝叶斯模型更新这些区域，以获得具有高置信度的动态特征点。*Liu* 等人[20]提出了一种双*K-means*聚类方法来获取动态区域的特征点，并通过为每个特征点分配静态加权来增强*RANSAC（*随机采样一致性）算法，从而提高算法的鲁棒性。*Liu* 等人[21]采用无监督学习进行分割，结合多视角几何和人类检测能力，能够在室内动态环境中追踪动态物体。

通过上述分析，可以看出语义分割模型能够准确地识别图像中的物体，从而区分动态物体和静态物体。然而，它需要大量的计算资源，降低了算法的实时性能。因此，一些动态*SLAM*算法往往舍弃时间开销较大的语义分割模型，选择更轻量的模型来提高算法的实时性。在此基础上，一些只使用CPU并满足实时性和精度要求的算法应运而生，但这不利于构建可以用于导航和路径规划的地图。此外，大多数动态*SLAM*算法未能充分考虑相邻帧之间动态特征点的相关性，以及关键帧选择中的不足之处。

本文的主要贡献如下：

1）采用*Mask R-CNN*（基于区域的卷积神经网络）对图像帧进行语义分割，并结合特征点的深度和距离信息，得到更高置信度的静态特征点集。

2）在相邻帧匹配过程中引入了相对静态点，提高了*SLAM*算法的鲁棒性。

3）针对动态环境中变化频繁的动态特征点设计了一种高性能的关键帧选择策略，并将语义像素添加到词袋模型（*BOW*）的相似性计算中，从而提高了闭环检测的精度。

4）提出了一种结合图优化和因子图优化的组合优化策略，提升了*SLAM*算法的实时性和定位精度。

5）考虑到移动物体可能遮挡一些静态物体，开发了基于分区的语义八叉树地图构建方法，确保了建图的一致性。

这篇论文剩余的组织结构如下：第二部分介绍了我们在动态环境中进行三维 *SLAM*的算法框架、改进策略和实现过程。第三部分提供了典型的实验结果和比较分析，包括一系列使用公共数据集的仿真实验和移动机器人案例研究。第四部分总结了论文并展望了未来的研究方向。

1. 方法描述
   1. 算法框架

为了满足移动机器人在动态场景中的自主导航需求，本文的主要思路是建立一个*3D SLAM*框架，包括语义分割、特征点分离、姿态估计、关键帧选择、闭环检测、组合优化和语义地图构建，如图1所示。

输入端通过深度相机进行环境感知和数据采集，获得*RGB-D*图像序列。前端首先提取*RGB*图像帧的特征点，并利用*Mask R-CNN*语义分割模块获得二值掩膜和具有语义信息的图像。二值掩膜用于分离动态和静态特征点，语义图像则用于进一步构建语义地图。接着，通过结合深度信息和距离信息，利用随机采样进一步过滤特征点，获得具有更高置信度的静态特征点。通过匹配相邻帧的动态特征点，得到相对静态点，并将其加入当前帧的优化过程中，以获得最小的重投影误差和机器人的最优位姿。根据动态场景的特点，本文提出了一种高质量的关键帧选择策略，将选出的关键帧传入后端模块进行进一步处理。

后端将语义信息与词袋模型结合，利用语义相似性进行相似度补偿，从而提高闭环检测的精度。同时，根据是否形成闭环，采用图优化和因子图优化进行全局和局部优化，从而提升算法的实时性并保证姿态估计的精度。

最终，输出端提供了完整的机器人轨迹和全局语义八叉树地图，作为移动机器人自主导航的基础。接下来，我们将详细描述三维 *SLAM*的各个部分及其实现过程。

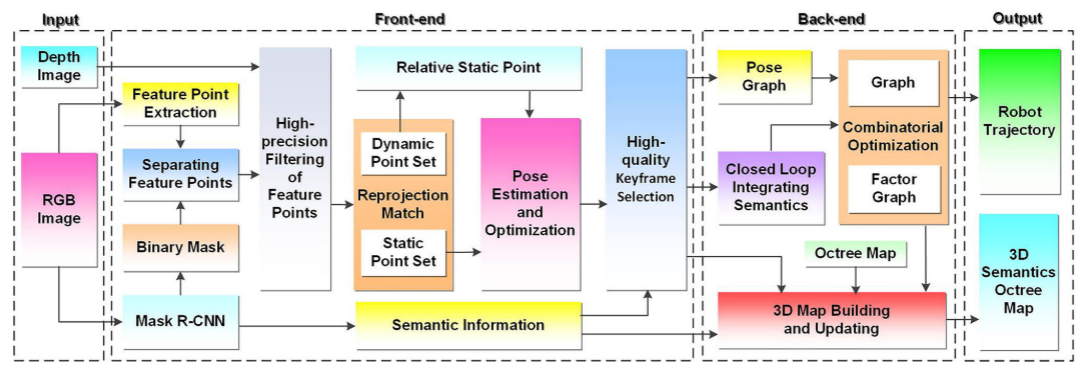


图 1基于RGB-D图像和语义特征的移动机器人3D SLAM方案在动态环境中的应用

* 1. 语义分割

本文使用的语义网络是Mask R-CNN[22]，该网络在Faster R-CNN[23]的基础上进行了改进，具有较高的精度，是目前最好的实例分割网络之一。它不仅可以检测图像中的物体并获得其类别标签，还能提供像素级的预测。

在动态环境中，无论是室内还是室外，运动物体的状态非常不稳定，因此我们打算对运动物体进行语义切割，首先要排除它们对视觉里程计的影响。具体来说，首先使用Mask R-CNN网络对RGB图像中的运动物体进行分割，然后根据分割结果进一步进行二值化，最终生成大小为m \* n \* 1的二值掩膜，如图2所示，使用RPN（区域生成网络）提取候选框。然而，语义分割网络无法完全分割动态物体，尤其是动态物体的边缘区域。总会有一些区域无法分割，这些区域中的特征点也会影响相机的定位。为了获得精确的位姿估计，必须进行进一步的优化。需要注意的是，尽管前端使用Mask R-CNN实例分割网络，但为了高效地分离动态物体并构建语义地图，我们主要利用语义标签来构建语义地图。因此，在下面的内容中，在地图中不同椅子的颜色将显示相同。

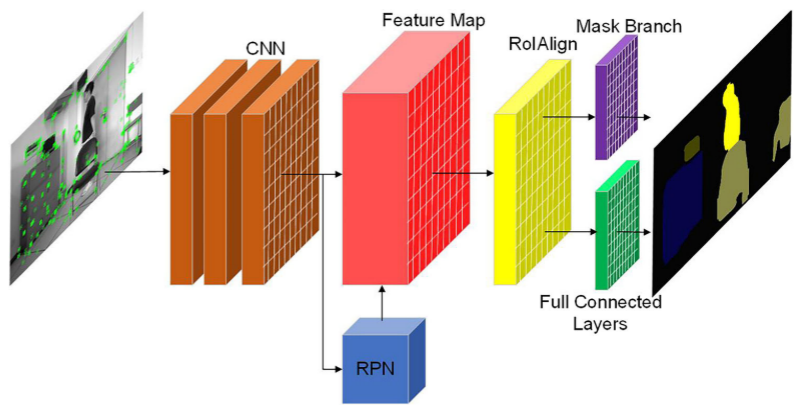


图 2 Mask R-CNN架构

* 1. 深度值融合

一般情况下，物体上每个特征点的深度值不会有太大差异，对于同一目标来说，动态物体可以获得大量动态特征点。经过语义切割后，我们考虑将这些动态特征点的信息与它们的深度值融合，以重新判断这些边缘处误判为静态特征点的点是否为动态点。

图3展示了同一动态物体上特征点的分布，其中，蓝色点为误判为静态的动态物体特征点，红色点为动态特征点，矩形框代表同一物体，τ为一定的距离阈值。

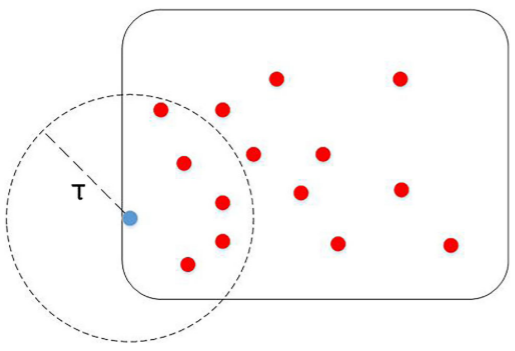


图 3 动态物体上的局部特征点分布

在同一图像中，可能会有几个物体处于相同的深度，但它们的距离相隔很远。为了减弱其他处于相同深度物体的影响，我们特别设置了距离阈值。此外，虽然设置距离阈值可以减少远处物体的影响，但一些靠近动态物体的静态点也可能符合距离阈值和深度差阈值的范围。由于动态物体包含大量的动态特征点，因此对于同一静态点，如果满足条件的动态点数量大于一定值，则该静态点会被判断为动态点。

根据上述描述，深度和距离的约束确保特征点的比较是针对同一物体的，通过设置阈值来确保识别的准确性。

设静态特征点集为动态特征点集为。其中N和M分别表示静态特征点和动态特征点的总数。首先，特征点应该满足一定范围内的深度差值约束，如下所示：

 (1)

其中表示特征点的深度，σ为定义的深度差阈值，设σ为0.2。此外，为了避免相同深度下其他物体的干扰，在计算特征点之间的距离时，需要满足以下条件：

 (2)

其中，表示两个点之间的欧氏距离，τ为定义的距离阈值，设τ为30像素。同时，设num表示满足上述第1和第2式约束条件的动态特征点的数量，那么通过设置阈值，可以判断静态特征点是否为动态点，如下所示：

 (3)

其中ϵ为一定的阈值，设ϵ为10。如果num大于ϵ，则静态点判断为动态点。否则,对于静态点的进一步筛选，需降低σ、num和ϵ的阈值。因为并非每次语义分割得到的动态物体都会包含大量特征点，当动态物体位于图像的边缘或离相机较远时，物体上的动态特征点较少，因此需要降低阈值进行二次筛选。

对于满足条件但的少数特征点，我们称这些特征点为可疑特征点。对于这些点，它们与动态特征点的深度和距离差异较小，但涉及到的动态特征点较少，因此可以通过随机采样已知的动态特征点集来区分可疑特征点。本文通过对已知的动态特征点集进行随机抽样，进一步区分可疑特征点。首先，从动态特征点集随机选择两个动态特征点，并计算它们的深度方差作为算法的估计模型。然后，判断可疑特征点与该模型的差异是否满足阈值条件。如果满足，则判断为动态点；否则，继续进行迭代。在进行k次迭代后，未被分类为动态特征点的可疑特征点将被判定为静态点。方差S²的计算公式为：

 (4)

其中xi是样本数据，μ是样本的均值，n是样本的总数。由于动态特征点集远大于可疑特征点集，这两个集可以近似为一个整体，因此选择动态点的概率可以表示为：

 (5)

其中表示选择动态点的概率，为动态点的数量，N为样本总数。假设每次采样获得n个样本，且这些样本是动态点的概率为。然后，至少有一个点是静态点的概率为，且n个点都不是动态点的概率为k，其中k为迭代次数。因此，以下结果成立：

 (6)

其中P是在迭代期间动态点的概率。那么，迭代次数k可以表示为：

 (7)

通过以上公式可知，进行的迭代过程中的*k*会根据的变化逐渐减少。我们方法中的随机采样主要针对由动态特征点和可疑特征点组成的集合。其中，动态特征点是已知的，而可疑特征点相对较少，也就是说，在大多数样本已知的情况下进行采样判断。因此，与传统的*RANSAC*算法相比，它的迭代次数较少。图4展示了在融合深度信息前后特征点提取的结果。可以看出，在深度信息融合之前，动态物体边缘上的一些动态特征点会被提取出来，而在融合深度信息之后，静态特征点可以准确地提取出来。



(a)没有融合深度信息 (b)融合深度信息后

图 4 深度信息融合前后提取到的特征点对比

* 1. 重投影匹配

基于上述处理，动态特征点和静态特征点可以很好地分离，接着，将分离出来的静态特征点用于相机自身的定位。对于一般情况，通过对当前帧与上一帧或参考帧之间的特征点进行重投影匹配，符合匹配条件的特征点将用于相机位姿的优化。

假设当前帧的特征点为 ，其对应的3D坐标为。那么，空间点与投影像素之间有如下关系[24]：

 (8)

其中，是像素点的深度值，K是相机内参矩阵，) 表示旋转矩阵𝑅和平移向量𝑡的李代数表示。由于存在噪声干扰，公式中会产生误差𝑒，该误差可以表示为：

 (9)

公式9可进一步简化为：

 (10)

表示从3D空间点映射到2D图像平面的映射函数，是对应的信息矩阵。

在本文中，除了对静态特征点进行重投影匹配外，还对动态特征点进行重投影匹配，为了获取相对静态点及其对应的3D空间点，进一步用于当前帧的位姿优化。如图5所示，红色点为上一帧的动态特征点，蓝色点为当前帧对应的动态特征点。当点移动到点的位置时，发现它的位置非常接近，然后将对应的3D空间点的重投影与点进行匹配。如果我们将点与点关联，说明点在相邻两帧中是相对静止的，可以称其为相对静态点。对于点和点也一样。此外，考虑到一些静态点可能误分类为动态点，这些误分类的静态点也会参与匹配。通过将相对静态点添加到跟踪和图优化中，可以进一步优化位姿。因此，重投影误差的优化可以表示为：

 (11)

其中和分别表示相对静态点的2D坐标和3D空间点，是相对静态点的信息矩阵， 。通过将相对静态点加入到优化过程中，重投影误差可以通过图优化持续优化，从而获得最小的重投影误差和最优的机器人位姿。

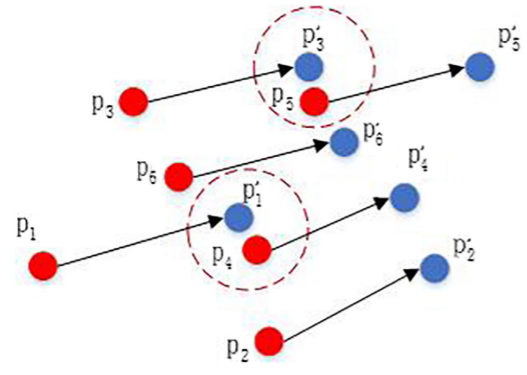


图 5相邻帧的相对静态点

在某些情况下，例如当前帧的动态特征点被去除后，剩余的静态特征点数量不足以满足跟踪条件，因此我们在跟踪和图优化中加入相对静态点。如果当前帧和上一帧之间匹配的静态特征点不足，则会考虑相对静态点，以增加匹配点的数量，从而提高算法的鲁棒性。

* 1. 关键帧选择

动态特征点的不稳定性不仅影响图像中静态特征点的数量，而且对算法的定位精度产生负面影响。因此，动态场景中的关键帧选择应认真考虑两个方面：1）可以用什么信息来衡量场景变化，通过场景信息的变化选择候选关键帧；2）如何去除候选关键帧之间的冗余，提取最终的关键帧。对于在动态环境中不断变化的动态特征点，本文设计了以下关键帧选择条件：

1）静态特征点的数量小于动态特征点的数量。

2）连续两帧之间动态特征点的数量与总特征点的比率大于前帧的平均比值，并且动态特征点与总特征点的比值需要大于某个阈值。

3）连续三帧之间的相机平移或旋转变化大于前帧的平均变化。相机的位姿变化计算公式如下[16]：

 (12)

其中，和分别表示时间t 时的平移和旋转，是上一帧的旋转矩阵的转置，表示矩阵的迹。

4）当前帧中跟踪的静态点与参考关键帧中的静态特征点比率低于某个阈值。

5）所选关键帧属于区间。

具体来说，条件1防止因静态点过少导致跟踪失败或定位不准确。条件2表示当动态物体进入相机视场时，导致动态和静态特征点发生变化，需要及时选择关键帧。在本研究中，。条件3旨在防止相机频繁或大幅度变化导致跟踪失败，本次研究中n=5。条件4旨在解决当前帧的特征点与参考关键帧差异过大导致的跟踪失败或定位不准确问题，阈值。条件5中的下界是为了防止在一般场景中频繁选择关键帧导致冗余，设置为0来及时选择关键帧，提高资源利用率。上限用于防止长时间没有选择关键帧导致的跟踪失败，设置为30。在此基础上，如果满足上述五个条件中的任何一个，都允许插入一个关键帧。

* 1. 闭环检测融合语义特征

由于动态物体的干扰，在动态环境中仅依靠词袋模型无法保证相似度计算结果的可靠性。因此，本文将语义信息与词袋模型结合，计算图像的相似度。在词袋模型中，采用词向量表示图像，然后通过比较词向量之间的距离来表达图像之间的相似性。词向量与物体的空间位置和排列顺序无关。因此，当相机略微移动时，只要物体仍然出现在视场内，词向量就会保持不变。词向量之间的相似度可以通过以下公式计算[25]：

 (13)

其中，表示共视图图中的一个关键帧的词向量，表示当前帧的词向量。

尽管前端已去除动态特征点，但被动态物体遮挡的静态特征点无法同时获取。因此，动态物体的出现会使得两帧之间的特征向量差异变大，导致相似度降低。为了解决这个问题，我们采用语义信息构建语义相似度，从而弥补词袋模型的不足。

在不同角度和距离下，通过语义网络分割的语义物体的像素区域大小也不同，但图像的总体大小保持不变。因此，通过比较两帧之间语义物体的像素比例，可以确定两帧之间的相似度。语义像素之间的相似度可以通过以下公式计算：

 (14)

其中，*N* 是共视图图中的所有关键帧数，*M* 是关键帧中静态物体的语义类别数，是共视图图中关键帧所有带语义标签物体的语义像素比例，是当前帧中所有带语义标签物体的语义像素比例，是所有带语义标签物体的像素数，是整体图像的像素数。

语义相似度仅考虑两帧之间共享的静态语义信息，从而减少了动态物体的影响。结合语义相似度和BOW相似度，总体相似度*S*可以通过以下公式计算：

 (15)

其中，是相似度的权重，。由于动态物体对词向量的影响较大，因此语义相似度的可靠性高于词袋模型。

该方法充分利用语义信息，并将语义像素添加到相似性计算中。通过使用语义相似度来补偿相似度，最终的相似度计算结果具有更高的置信度，从而提高了闭环检测的精度。

* 1. 后端组合优化

随着移动机器人不断移动，用于后端优化的位姿和地标数量在不断增加。对于图优化，算法的时间复杂度会随着优化节点数量的增加而增加，从而降低算法的实时性。虽然我们仅对关键帧进行优化以提高算法的实时性，但对于本文提出的关键帧选择策略，动态环境中的关键帧数量将高于静态环境，显然传统的图优化无法很好地适应本文的算法。

鉴于位姿和地标数量的增加，我们将图优化的全局优化与基于因子图的局部优化结合，构建了以下组合优化策略：

 (16)

其中，和分别表示图优化和因子图优化的求解方程。一旦检测到闭环，使用图优化进行全局优化；否则，使用因子图优化进行局部优化。

在现实中，如果所有相邻的关键帧都参与图优化，优化速度会变慢，且优化结果不会显著改善。为加速图优化的求解，我们进一步采用了关键帧子区间优化策略。即，对于普通帧序列，设每*n*帧为一个子区间，由于关键帧选择策略的影响，每个子区间内的关键帧数量不同，但子区间内的关键帧之间可以保持一定的相关性。然后，选择子区间中静态特征点观察数量最多的关键帧 加入图优化。选择策略如下：

 (17)

其中，是子区间内唯一满足条件的关键帧，表示子区间内静态特征点观察数量最多的关键帧数量，和分别表示子区间内最左和最右满足条件的关键帧，是和之间的帧数，是最接近子区间中心的关键帧。

根据上述规则，如果只有一个关键帧满足最大观察条件，则该帧将被加入到优化中。如果满足最大观察条件的关键帧数量大于或等于两个帧，则需要判断满足最大观察条件的最左和最右关键帧之间的差异是否大于或等于*n/2*，如果是，则将两个帧都加入到优化中；否则，将加入距离区间中心最近的关键帧进行优化，在此，。

通常，最大后验估计（MAP）𝑋\*可以写成如下形式：

 (18)

其中，表示因子函数，*𝑋i*是一个变量节点。

为了将因子图与图优化结合，需要将因子图的MAP问题进行转换。在假设只有视觉约束和高斯噪声的情况下，公式18可以近似为最小二乘问题：

 (19)

其中，A和分别表示雅可比矩阵和信息矩阵，𝑏是观测误差。

对于雅可比矩阵𝐴在[26]和[27]的工作中采用了QR分解。如果有新的观测值，只需更新分解后的矩阵𝑅来减少优化时间。然而，雅可比矩阵𝐴并不能始终增量平滑。为了确保因子图优化的准确性，有必要定期维护雅可比矩阵𝐴。Kaess 等人[26]采用了固定的N-step更新策略，有效地解决了矩阵更新问题，但缺乏灵活性。针对这个问题，Kaess 等人[27]提出了贝叶斯树的结构，但随着树的深度增加，维护树的成本也会增加。因此，我们提出了以下更新策略：

 (20)

其中，*A’* 是更新后的雅可比矩阵，采用列近似最小度排序算法[28]获得，*aT*是新添加的观测向量的转置，*CoV(.,.)*表示两个帧之间的共视点数量，是新添加的关键帧。更新后被赋值给​。

当 和之间的共视点数量少于*m*时，需要更新矩阵*A*。否则，继续更新增量，此时，*m=15*。利用共视图之间的相关性可以确保矩阵*A*的增量平滑。当新关键帧和矩阵*A*之间的差异过大时，可以及时更新矩阵*A*，使其在优化中更加灵活。此外，这样也更容易容纳稀疏映射点并减少成本。

我们采用图优化进行全局优化，并借鉴因子图优化实现局部优化，从而提高算法的实时性能并确保位姿估计的准确性。

* 1. 语义地图构建

通常，语义地图有两种方式：基于语义像素的或基于语义对象的。考虑到语义分割网络可能无法将图像中的所有物体进行分割，因此本文采用基于语义像素的方法。在这种方法中，与图像像素点对应的3D点云可以表示为以下形式：

 (21)

其中表示二维到三维的投影，和分别是RGB图像和对应的语义图像中的像素值，的值由像素的语义信息决定。当像素在语义图像中有语义信息时，；否则，。是像素的深度值。类似地，在构建地图时，动态物体需要被去除，因此的定义如下：

 (22)

为了确保地图的一致性，在映射过程中会排除距离动态物体*h*像素范围内的部分像素，本文中*h*设为15像素。

在本文中，我们通过语义分割获得的语义标签完成了语义八叉树地图的构建。与点云地图相比，八叉树地图具有更好的灵活性和压缩性。同时，八叉树地图可以随时更新，因此在动态场景下具有更好的性能和应用效果。在构建地图时，地图中没有语义信息的部分会用原始颜色填充，而有语义信息的部分则用其语义标签的颜色填充。

为了减少动态物体对地图构建的影响，我们首先将具有语义信息的物体与没有语义信息的物体分开进行映射。然后，将当前的语义物体点云与上一关键帧的全局语义物体点云融合，形成全局语义物体点云。最后，将当前帧的全局语义物体点云与非语义物体点云进行融合，生成全局语义地图。融合的公式如下：

 (23)

其中，和分别表示全局语义物体和当前语义物体的点云，是从开始到上一帧的全局语义物体点云，是当前帧的变换矩阵，的值为：

 (24)

该公式表示，在两点云融合时，如果它们之间的距离小于设定的阈值，则放弃融合，否则进行融合。这里，阈值设为1。

在动态环境中，移动物体会遮挡一些静态物体，导致无法获得该物体的语义信息，这会导致构建地图时丧失语义信息。针对这种情况，本文采用上述分区构建方法，确保静态物体的一致性。语义地图的场景如图6所示。

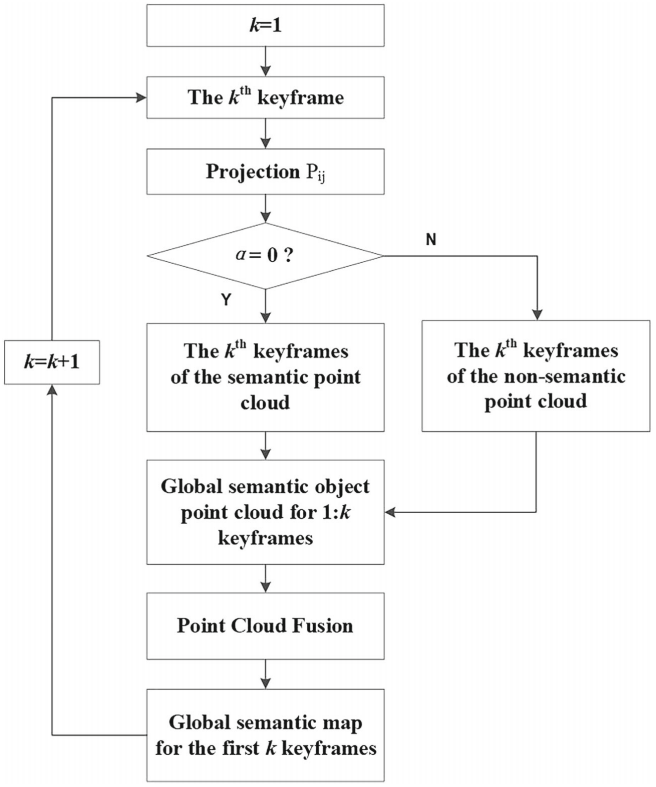


图 6 语义映射的场景

1. 仿真实验与比较分析

为了验证所提出算法的性能，本节给出了仿真实验和移动机器人在实际场景中的案例研究。仿真实验使用了TUM（慕尼黑工业大学）RGB-D公共数据集，该数据集主要在一个有动态物体的办公场景中进行，目的是测试算法在动态环境中的鲁棒性和精度。所有实验均在一台配置为*Intel i5-7200U*处理器、8GB RAM的移动计算机上运行，操作系统为Ubuntu 18.04。

* 1. TUM数据集仿真实验

本研究主要集中在TUM行走数据集的比较分析上。为了验证所提算法的优越性，除了与经典的ORB-SLAM2进行比较外，我们还将算法与最先进的动态SLAM算法进行了比较，包括DS-SLAM[11]（面向动态环境的语义视觉SLAM）、KMOP-SLAM[21]（基于RGB-D相机的动态视觉SLAM）、DOTMASK[12]（动态物体跟踪和掩膜）。同时，采用绝对轨迹误差（ATE）和相对位姿误差（RPE）作为评估指标，均方根误差（RMSE）是评估绝对轨迹误差和相对位姿误差的主要指标。为确保实验结果的可靠性和有效性，我们进行了多次独立实验。

表1比较了在采用ORB-SLAM2和本文提出的关键帧选择策略的情况下，算法在30次实验中的RMSE值。可以看出，在*fr3\_walking\_static*数据集中，RMSE值的改善并不显著。这是因为在该数据集中，本文算法已经能够很好地进行轨迹估计。然而，在动态环境中，本文提出的关键帧选择策略能够在动态物体快速移动和相机频繁旋转与平移时及时插入关键帧，从而确保了算法的稳定性和精度。

表 1 不同关键帧选择策略下绝对轨迹误差（ATE/m）RMSE的比较

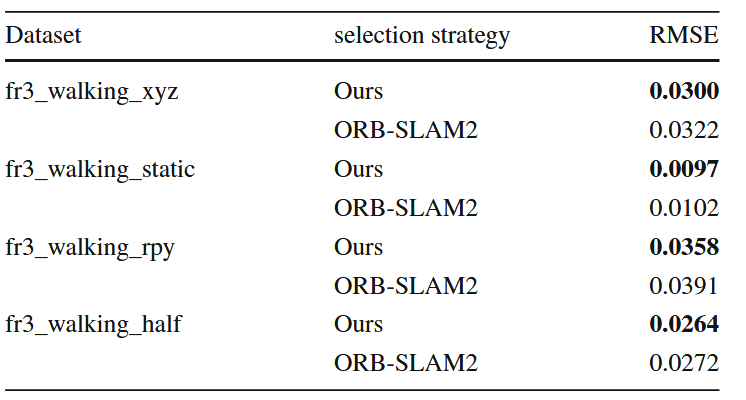
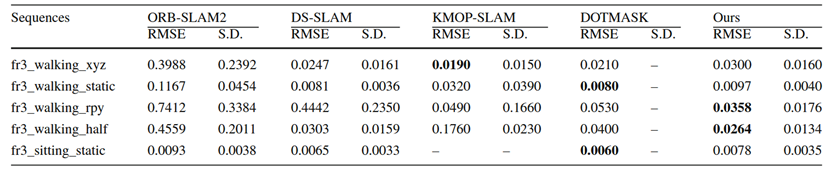


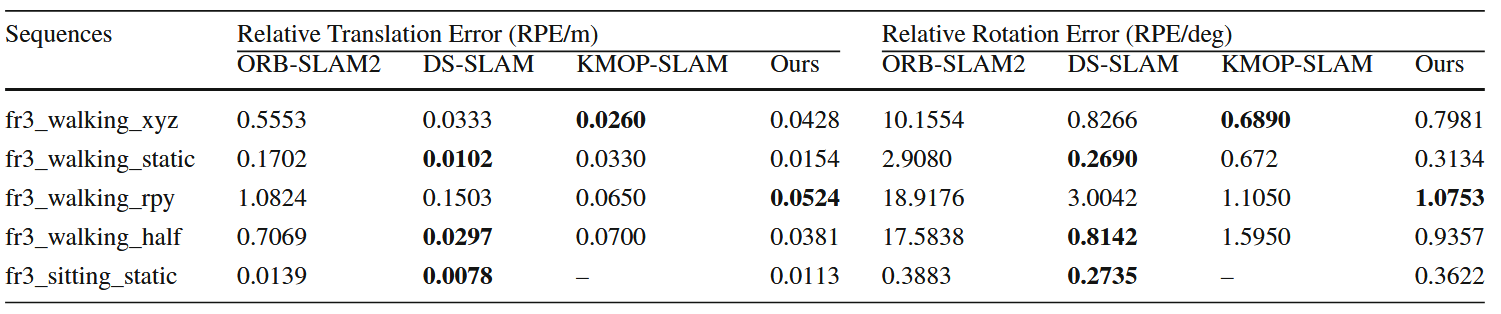
表2和表3分别比较了不同算法的绝对轨迹误差（ATE）和相对位姿误差（RPE）的均方根误差(RMSE)。从表2可以看出，在行走数据集的高动态场景中，动态特征点较多。ORB-SLAM2算法在位姿优化过程中将动态点当作静态点，因此产生了大量的误差积累。相比之下，所提出的算法可以有效去除位姿估计中的动态特征点，从而提高了位姿估计的准确性。此外，所提算法估计的轨迹与DS-SLAM的均方根误差(RMSE)差异不大。但由于DS-SLAM在前端采用极坐标几何方法来分离动态和静态特征点，因此在相机频繁旋转时，极坐标线难以找到，导致准确度下降。而在我们的算法中，动态特征点的过滤依赖于深度值。当相机频繁旋转时，深度值变化较小，因此算法在*walking\_rpy*和*walking\_half*数据集上的表现较好。然而，在相机平移或静止的情况下，所提出的算法会误将靠近动态物体且深度值相似的静态特征点分类为动态特征点，从而导致准确性比DS-SLAM更差。从表2可以看出，由于KMOP-SLAM和DS-SLAM算法在前端采用了多视角几何方法，因此在*walking\_half*和*walking\_rpy*数据集上的表现较差。而DOTMASK仅在前端利用语义分割去除动态特征点，尽管能够满足实时性能要求，但动态特征点去除不够精确，导致了显著的定位误差。

表 2 不同算法的绝对轨迹误差（ATE/m）RMSE比较



从表3可以看出，由于所提出的算法在前端获得了更高置信度的静态特征点集，并将其与相对静止点一起加入位姿估计中，因此漂移较小，相对位置误差明显优于ORB-SLAM2。然而，在某些情况下，算法的表现不如其他两个算法，但由于后端采用了基于语义信息的闭环检测和组合优化策略，能够有效补偿本文算法中产生的相对位置误差。

表 3 不同算法的相对位姿误差（RPE）RM差异比较



此外，根据表2中的ATE RMSE，我们可以定量分析不同算法在动态环境中与ORB-SLAM2相比的精度提升。表4展示了不同算法在动态环境中相对于ORB-SLAM2的轨迹估计精度提升率，提升率𝜂的计算公式如下：

 (25)

其中，𝛾和𝜙分别表示其他算法和ORB-SLAM2的轨迹估计RMSE。从公式25可以看出，𝛾越大于𝜙，得到的𝜂就越高，即算法相对于ORB-SLAM2的性能越好。

我们可以直观地看到，本文算法在高动态环境下相对于ORB-SLAM2的的定位精度提升超过90%，其中在*w\_xyz*数据集中为91.93%，在*w\_rpy*数据集中为91.69%，在*w\_half*数据集中为94.72%，在*w\_static*数据集中为94.19%。但在低动态环境中的算法精度提升的并不明显。这是因为低动态环境中的干扰对ORB-SLAM2的影响较小。此外，尽管本文算法在所有高动态场景中的表现并不是最优的，但其性能稳定。由于算法性能稳定，平均提升率达到了93.39%，而其他算法只有在某些情况下有显著提升，整体提升并不是最佳的。

表 4 不同算法相对于ORB-SLAM2的绝对轨迹误差（ATE/m）RMSE提升率

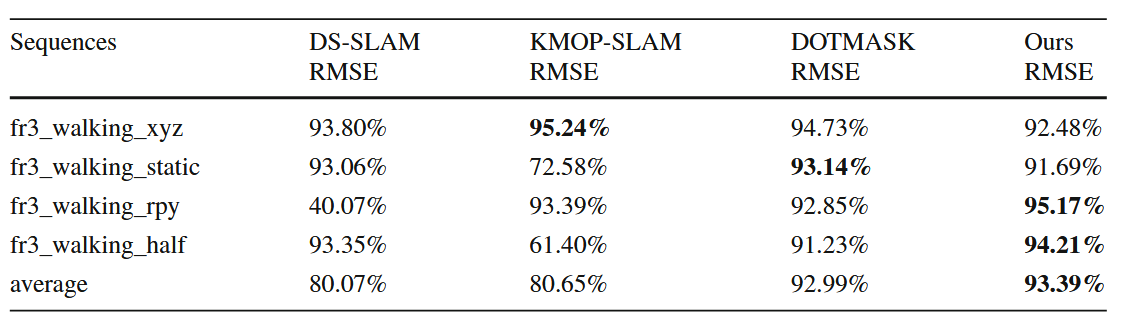


图7展示了在*w\_xyz*、*w\_rpy*、*w\_half*和*w\_static*数据集上，ORB-SLAM2与所提出算法之间绝对轨迹误差比较。可以发现，本文算法估计的轨迹与真实轨迹的吻合度更高。与ORB-SLAM2相比，本文算法去除了动态特征点，使得在动态环境中的轨迹估计误差更小。另一方面，本文提出的关键帧选择策略使得算法在不断变化的动态环境中具有更好的适应性，从而提高了算法的准确性，这也为后端优化和闭环检测提供了可靠的支持。尽管本文算法使用了组合优化策略来缩短优化过程中的时间，但与语义分割所消耗的时间相比，其优势较为不显著。因此，所提出的算法有效提高了移动机器人在动态场景中的定位精度。

图8比较了在TUM的*walking\_xyz*数据集下，DS-SLAM、Zhang等人[30]算法和我们方法的局部语义八叉树地图结果。其中，DS-SLAM仅构建了语义八叉树地图，而[30]中的工作进一步增强了语义地图的完整性，但仍然未考虑语义静态物体的不可变性。由于椅子在构建地图的过程中持续移动，我们主要使用计算机类物体来说明不同方法的效果。在实验中，地图中所有非语义部分用灰色表示，去除的动态物体部分用白色表示，计算机用深绿色表示。我们发现，所提出的方法相比另外两种算法具有更完整的语义信息。其主要原因是，DS-SLAM中的映射阶段没有优化，使得映射过程容易受到动态物体的影响。尽管[30]中提出的算法减少了动态物体对映射的影响，但忽视了静态语义物体的不可变性。我们的算法更新了每个关键帧的语义信息，使得最终的映射结果更加一致。

图9进一步展示了在去除动态物体后，基于[30]算法和我们方法构建的全局语义八叉树地图在*fr3\_walking\_xyz*数据集上的对比。图9(a)展示了*fr3\_walking\_xyz*的一帧。从图9(b)可以看到，尽管地图中的计算机和椅子的语义特征较为明显，但不够完整，且垃圾桶的语义未被突出显示。这是因为垃圾桶只出现在少数几帧中，并且已经被包含在全局映射中，此外，算法忽略了静态语义物体的不可变性。相比之下，图9(c)中的语义地图展示了更清晰、更完整的语义特征。

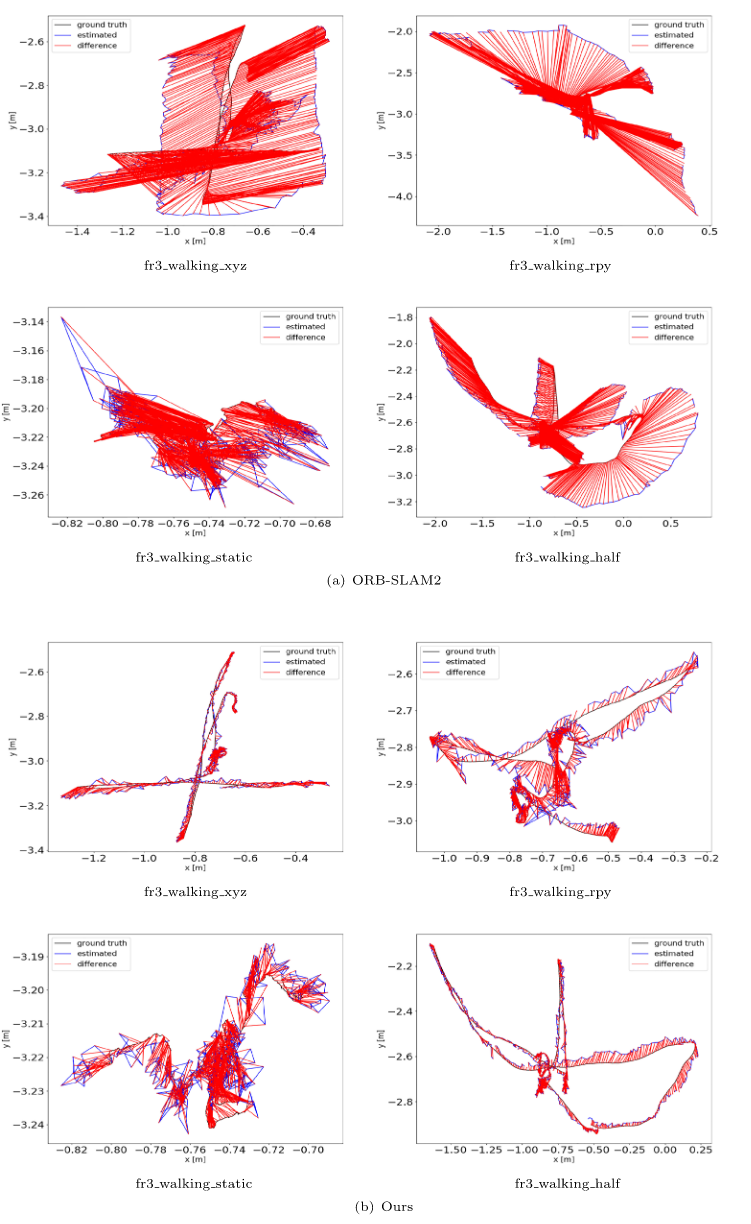


图 7不同算法的轨迹对比

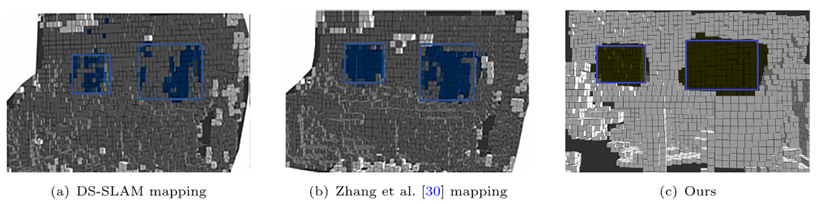


图 8 不同算法在TUM walk\_xyz数据集下的局部语义八叉树地图结果

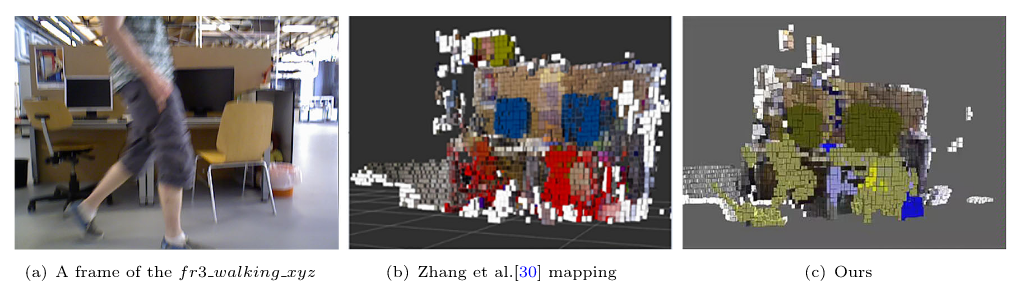


图 9 不同算法在TUM walk\_xyz 数据集下的全局语义八叉树地图结果

为了评估语义地图的性能，我们采用了类似于[30]论文中的标准来评估我们的算法：语义完整性和语义波动性。完整性表示每个地图中，代表某一类型物体的语义网格占某种类型网格的百分比。波动性表示在整个语义映射过程中，动态物体的连续运动对静态物体的影响程度。

从图10、图11和表5可以看出，我们的方法在语义完整性方面优于其他两种方法。这是因为我们采用了部分映射的方法，忽略了动态物体对映射过程的影响。然而，在*fr3\_walk\_half\_sphere*和*fr3\_walk\_rpy*数据集下，我们方法的语义波动率相对较大。这是因为本文中的映射性能受到了语义分割精度的影响。具体来说，当相机频繁旋转时，语义分割容易产生错误，这可能导致在语义融合后错误积累，进而减少了分割错误物体的语义完整性，从而影响了整体的语义波动性。

为了验证所开发的闭环检测方法的综合性能，使用相同的前端在*fr3\_walking\_half\_sphere*数据集上进行跟踪和定位。然后，分别加入传统的词袋模型和带有语义信息的词袋模型进行对比实验。通过使用精确度-召回率（P-R）曲线进一步评估不同闭环检测算法的性能。图12展示了在fr3\_walking\_half\_sphere数据集下，两个具有高相似度的图像。在仿真中，动态物体从左到右移动，依次遮挡了计算机显示器、椅子等物体。

精确度*P*和召回率*R*的计算公式如下：

 (26)

其中，TP是真阳性的概率，FP是假阳性的概率，FN是假阴性的概率。

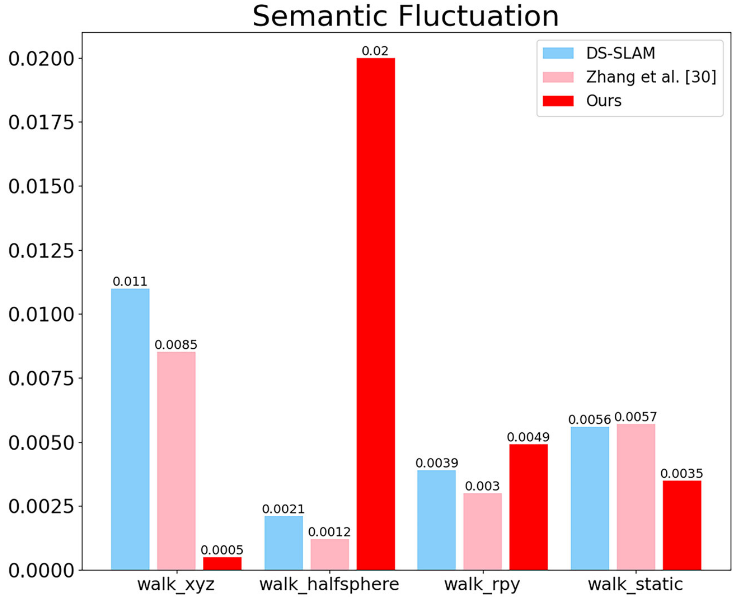
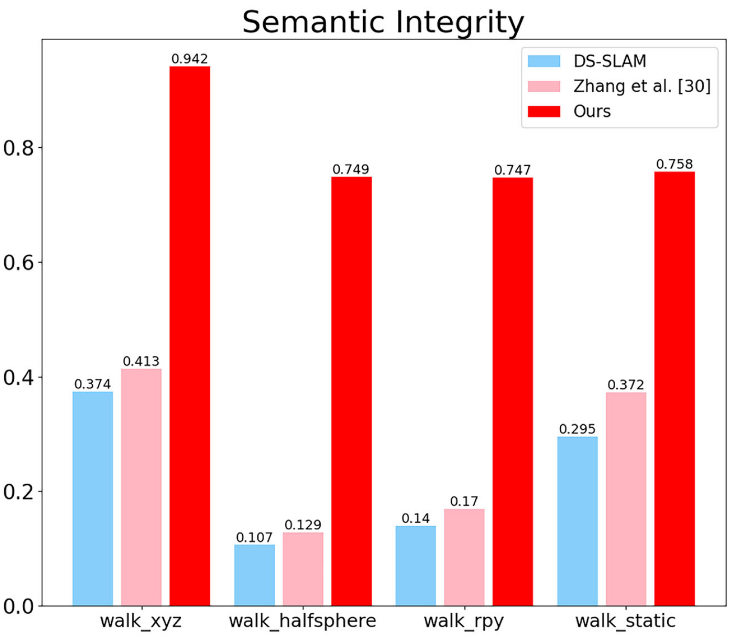


图 10不同算法在TUM数据集下的语义完整性

图 11不同算法在TUM数据集下的语义波动性

表 5 不同算法在TUM数据集下的语义完整率和波动率

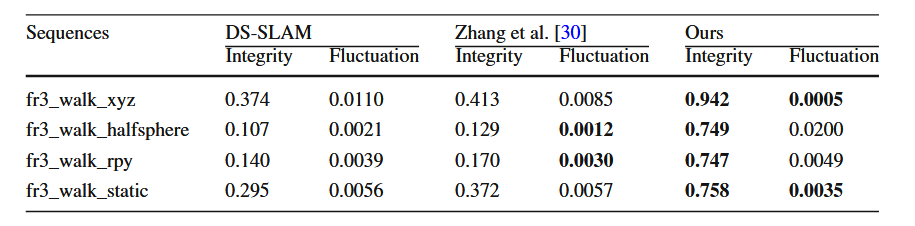


图 12 在fr3\_walking\_half\_sphere数据集下，具有高相似度的两张图像

图13展示了在*fr3\_walking\_half\_sphere*数据集下，不同闭环检测方法的P-R曲线。在图中，蓝线和红线分别代表传统词袋模型和带有语义信息的词袋模型。显然，“BOW+语义”方法的准确性高于传统词袋模型。这是因为如果仅采用词袋模型的检测策略进行判断，提供的相似度较低，甚至可能产生假阳性。然而，我们的方法利用静态语义信息去除动态物体的干扰，从而减少动态物体对词袋模型相似度计算的影响。因此，减少了误计算的可能性，并提高了闭环检测的真正阳性。还可以发现，在我们的方法中，P-R曲线下降得太快，这是因为提出的关键帧选择策略增加了闭环检测中的关键帧数量，导致使用词袋模型的闭环检测方法的真正阳性减少。

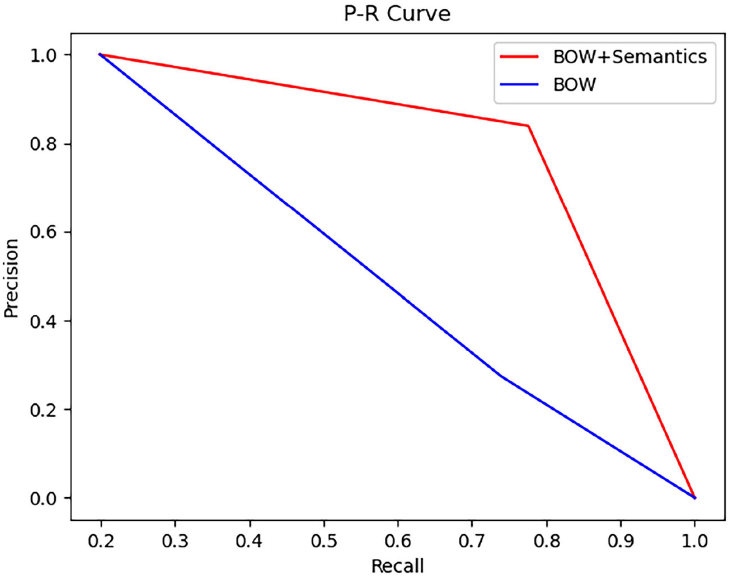


图 13 在fr3\_walking\_half\_sphere数据集下，不同闭环检测方法的P-R曲线

3.2移动机器人案例研究

为了进一步验证本文算法的可行性和有效性，我们使用装载了“Letv Pro Xtion”深度相机的“Sanbao”移动机器人，在实际动态场景中进行跟踪并遵循参考轨迹，分析算法的定位精度、地图构建结果和实时性能。实验场景为一个办公室，办公室内有人来回走动，作为动态物体，如图14和图15。

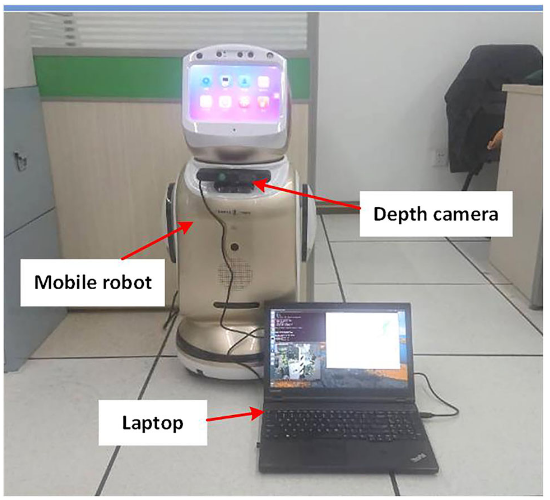


图 14 移动机器人

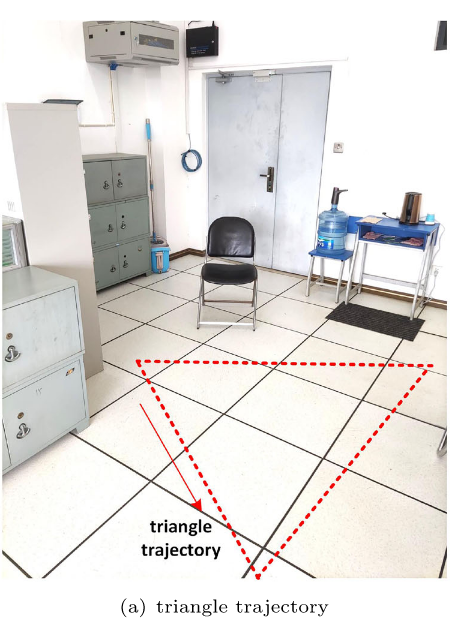
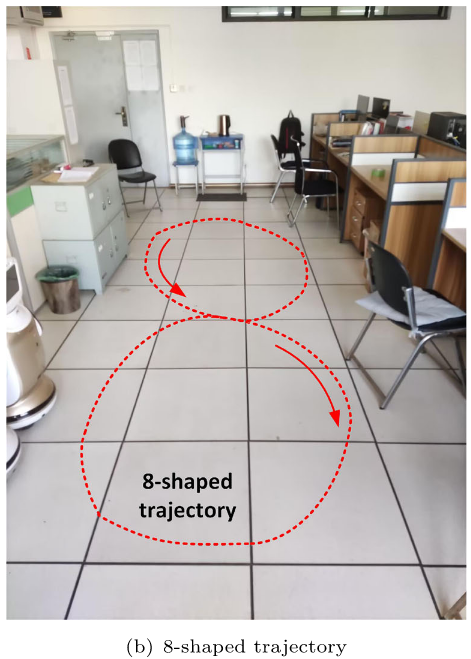
 

图 15 实验场景和行走轨迹

在实验中，一些输入帧包括带有ORB特征点的灰度图像和对应的语义图像，如图16所示。从图16中可以看到，本文方法有效去除了运动物体（如人）的动态特征点，保留了静态物体（如椅子、桌子、计算机等）的静态特征点。在显示实验场景的语义信息时，为了区分不同的物体，给椅子、桌子、电脑和垃圾桶等设置了不同的语义标签，同时让一个人随机走动，模拟动态场景。

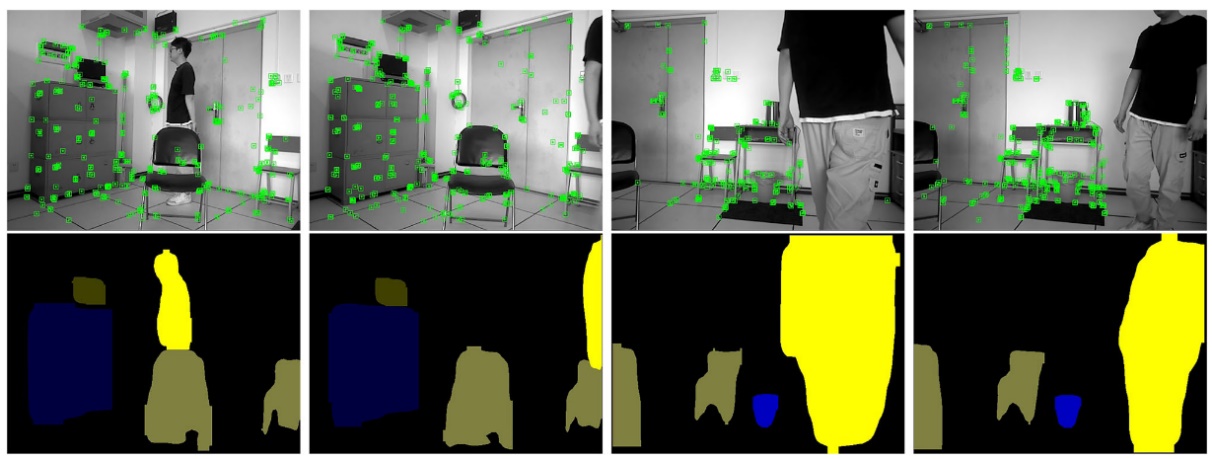


图 16第一行是带有特征点的灰度图像，第二行是语义图像

图17展示了通过不同算法提取的灰度图像和稀疏地图。可以得出结论，ORB-SLAM2在实际场景中提取了大量的动态物体特征点（如人），并未去除高动态特征点，而本文方法有效地去除了运动物体的特征点，为后续的语义地图构建提供了可靠支持。

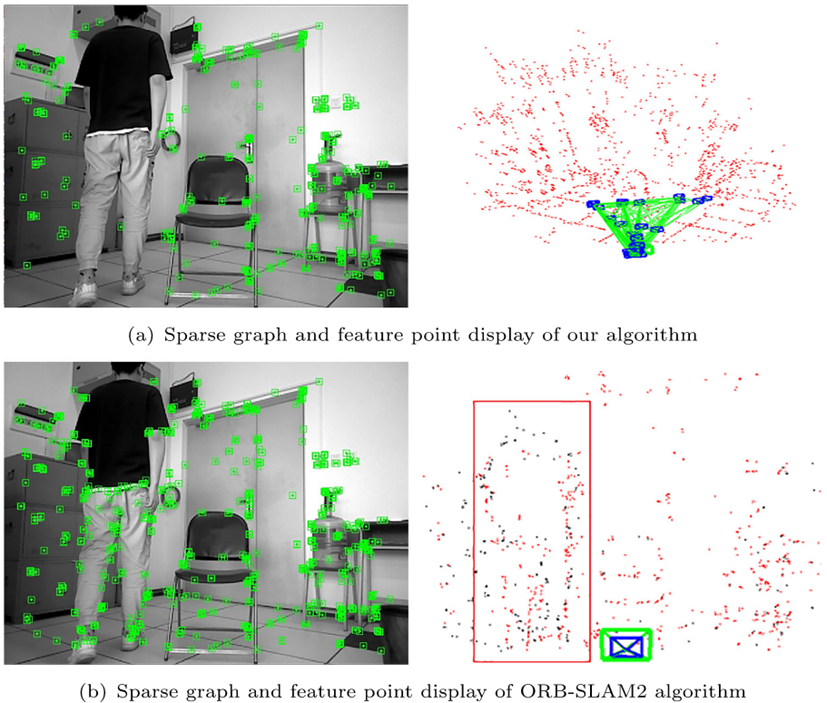


图 17 不同算法在全局地图构建中的效果

为了展示所提出的闭环检测方法在实际场景中的应用效果，我们让移动机器人按照8字形的参考轨迹进行移动，并通过比较估计的轨迹来验证不同算法的性能。从图18中可以看到，传统的BOW方法未能正确检测闭环，而本文提出的BOW结合语义信息的方法能够正确检测闭环。这可能是由于闭环位置有一个移动物体，干扰了BOW模型相似度的计算，导致闭环检测失败。

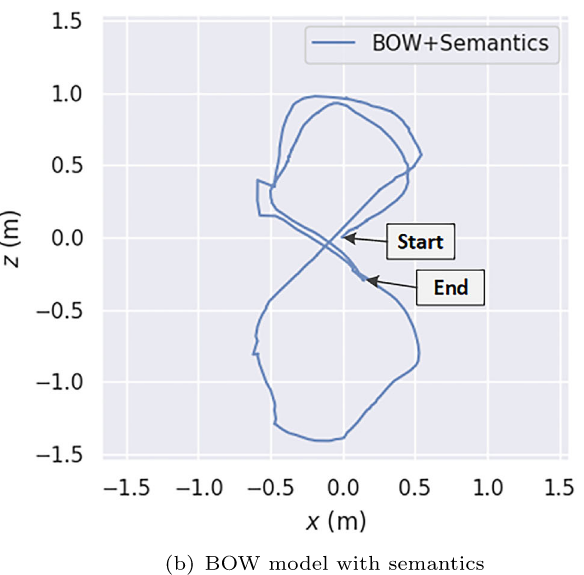
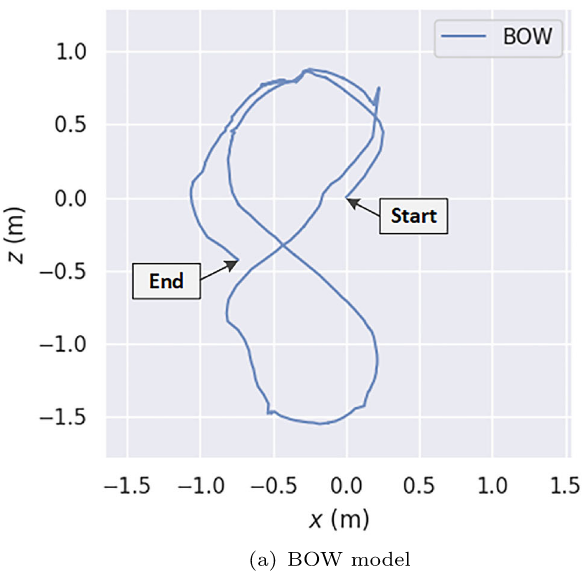


图 18不同闭环检测算法的8字形轨迹比较

图19进一步展示了闭环检测过程中两张相似图像的对比。移动物体从右到左进入相机的视场，然后从左到右走动，导致静态物体在移动机器人视场中的遮挡。这使得仅使用BOW模型计算相似度时，产生了大量的误差，无法正确检测闭环。然而，本文方法通过集成语义信息，将动态物体的区域从相似度计算中排除，从而有效弥补了BOW模型的不足。



图 19闭环检测过程中的一组相似图像

在实验中，表6给出了两种闭环检测算法的综合性能比较。显然，两个算法都没有出现感知偏差，即没有错误地将非闭环识别为闭环（FP=0）。传统BOW模型出现了感知偏差，即同一位置被错误地认为是不同的位置（FN=1），而我们的方法没有这种情况发生（FN=0）。在召回率方面，两种方法均达到了100%，但在精确度方面，传统BOW模型的精确度仅为50%（TP=1），而我们的方法达到了100%（TP=2），表示两次闭环均被正确检测。

表 6不同闭环检测算法的综合性能比较

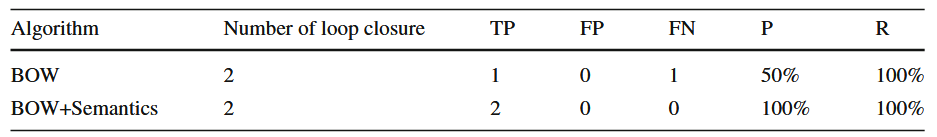


图20展示了ORB-SLAM2与本文方法的轨迹估计结果比较。由于ORB-SLAM2无法检测并去除动态特征点，当人出现在机器人视场中时，机器人位姿估计会产生大量的误差积累，导致估计轨迹与参考轨迹之间存在较大误差。相比之下，本文方法能够有效去除动态特征点，估计的轨迹与参考轨迹的相似度较高，因此定位精度较ORB-SLAM2更高。

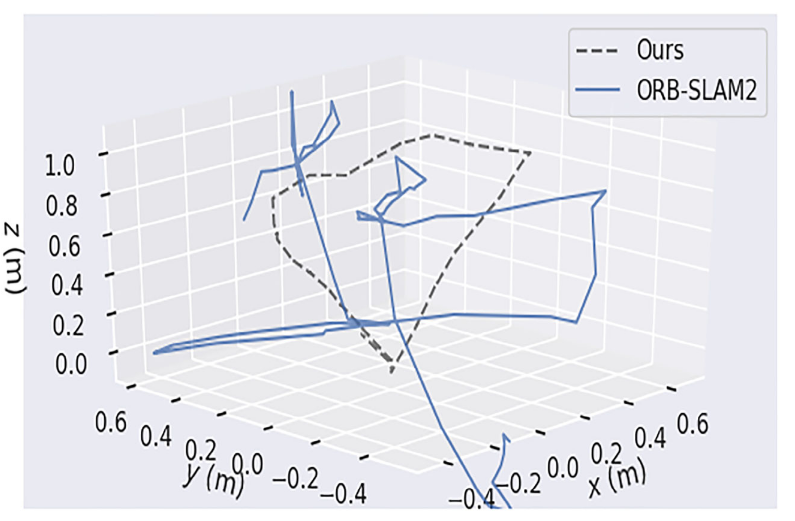


图 20不同算法在实际场景中三角形轨迹的比较

图21展示了在实际场景中构建的语义八叉树地图。从图中可以看出，运动物体（如行人）已被排除在外，场景中静态物体（如柜子、椅子、门等）的轮廓清晰可见，没有失真和变形，从而满足了地图一致性的要求。

此外，由于实验中语义网络和计算机配置的影响，本文方法的平均每帧处理时间为0.98秒。与ORB-SLAM2相比，本文方法在语义分割和语义地图构建过程中所花费的时间较长，但基本满足实时性要求。尽管如此，它获得了更高的定位精度和更一致的地图，且系统运行流畅。

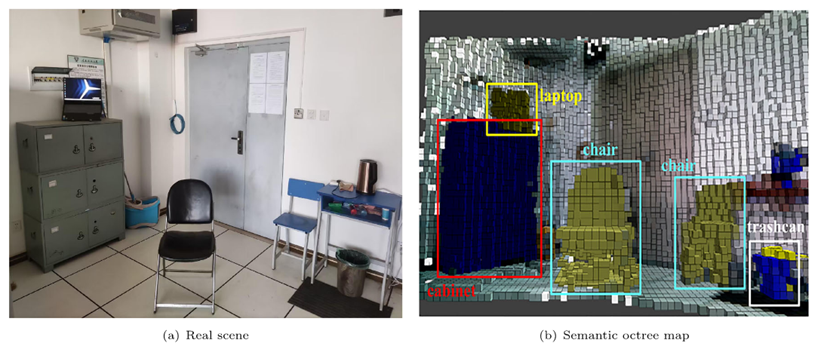


图 21在实际场景中构建的语义八叉树地图

值得注意的是，在实际应用中，除了动态物体的干扰外，当移动机器人穿越崎岖的地形或进入不平坦的环境时，自身振动引起的不确定性尤为显著，这会对SLAM系统的稳定性产生不利影响。然而，作为一种典型的欠驱动系统，轮式移动机器人无法通过任何连续不变状态反馈控制器实现稳定控制。如果移动机器人的视觉系统能够在动态场景中实现稳定控制，机器人将变得更加智能，更好地适应环境变化。因此，未来的工作可以将Chen等人[31–33]提出的视觉伺服策略结合进来，通过反馈控制机器人运动，减少SLAM视觉模块的预测位姿误差，使机器人能够到达指定位置或跟踪设定轨迹。

1. 结论和未来展望

本文主要关注移动机器人在动态环境中3D SLAM算法的低定位精度和差地图一致性问题。为此，前端通过丢弃动态特征点作为离群值来实现精确的定位结果。为了增强算法的鲁棒性，我们进一步结合了动态特征点之间相邻帧的相关性，获得了相对静态点，并将其加入到局部跟踪和优化中。此外，我们设计了一种高性能的关键帧选择策略，为地图构建和位姿优化提供了保障。在后端，为充分利用Mask R-CNN获得的语义特征，我们将语义像素融合到词袋模型相似性计算中，从而提高了闭环检测的精度。同时，采用图优化进行全局优化，并结合因子图优化实现局部优化，从而提高了SLAM算法的实时性，并确保了位姿估计的精度。最后，提出了基于分区的语义八叉树地图构建方法，为机器人的自主导航提供支持。仿真和实验结果表明，所提出的方法在动态环境中具有良好的精度和鲁棒性。然而，值得注意的是，所提方法在相机频繁旋转时表现较好，但在相机平移和静止时略显不足。此外，语义波动问题需要在相机频繁旋转的情况下进一步提高。

未来的工作将致力于采用更轻量的语义分割网络，以提高算法的实时性。同时，我们计划进一步增强算法在相机平移下的精度，使算法更加鲁棒。研究的另一个方向是同时处理场景中的静态和动态成分，追踪移动物体，并确保它们的特征不被用于位姿估计，以确保系统的鲁棒性；进一步地，估计相机运动和动态物体运动，追踪并重建多个移动刚性物体，并将它们加入背景地图中。

参考文献

1. Newcombe, R.A., Izadi, S., Hilliges, O., Molyneaux, D., Kim,D., Davison, A.J., Kohi, P., Shotton, J., Hodges, S., Fitzgibbon, A.: Kinectfusion: Real-time dense surface mapping and track-ing. In: 2011 10th IEEE International Symposium on Mixed and Augmented Reality, pp .127–136 (2011).
2. Engel, J., Schps, T., Cremers, D.: Lsd-slam: Large-scale direct monocular slam. In: European Conference on Computer Vision,pp. 834–849 (2014). Springer
3. Mur-Artal, R., Tardós, J.D.: Orb-slam2: An open-source slamsystem for monocular, stereo, and rgb-d cameras. IEEE Trans.Robot. 33(5), 1255–1262 (2017).
4. Engel, J., Koltun, V., Cremers, D.: Direct sparse odometry.IEEE Trans. Pattern Anal. Mach. Intell. 40 (3), 611- 625(2018).
5. Zhao, L., Liu, Z., Chen, J., Cai, W., Wang,W., Zeng, L.: A compat-ible framework for rgb-d slam in dynamic scenes. IEEE Access7, 75604–75614 (2019).
6. Xie, W., Liu, P.X., Zheng, M.: Moving object segmentation anddetection for robust rgbd-slam in dynamic environments. IEEETrans. Instrum. Meas. 70, 1–8 (2021).
7. Yuan, X., Chen, S.: Sad-slam: a visual slam based on semantic anddepth information. In: 2020 IEEE/RSJ International Conference onIntelligent Robots and Systems (IROS) 4930–4935 (2020). IEEE
8. Li, A., Wang, J., Xu, M., Chen, Z.: Dpslam: A visual slam withmoving probability towards dynamic environments. Inf. Sci. 556,128–142 (2021)
9. Wen, S., Li, P., Zhao, Y., Zhang, H., Sun, F., Wang, Z.: Semanticvisual slam in dynamic environment. Auton. Robots 45(4), 493–504 (2021).
10. Cheng, J., Wang, C., Mai, X., Min, Z., Meng, M.Q.-H.: Improv-ing dense mapping for mobile robots in dynamic environmentsbasedon semantic information.IEEESens. J. 21(10), 11740–11747(2021).
11. Yu, C., Liu, Z., Liu, X.-J., Xie, F., Yang, Y., Wei, Q., Fei, Q.: Ds-slam: A semantic visual slam towards dynamic environments. In:2018 IEEE/RSJ International Conference on Intelligent Robots andSystems (IROS), pp. 1168–1174 (2018).
12. Vincent, J., Labbé, M., Lauzon, J.-S., Grondin, F., Comtois-Rivet,P.-M., Michaud,F.:Dynamic object tracking andmasking for visualslam. In: 2020 IEEE/RSJ International Conference on IntelligentRobots and Systems (IROS), pp. 4974–4979 (2020).
13. Bolya, D., Zhou, C.,Xiao,F.,Lee,Y.J.:Yolact: Real-time instancesegmentation. In: 2019 IEEE/CVF International Conference onComputer Vision (ICCV), pp. 9156–9165 (2019).
14. Liu,Y.,Miura,J.:Rdmo-slam: Real-time visual slam for dynamicenvironments using semantic label prediction with optical flow. IEEE Access 9, 106981–106997 (2021).
15. Zhang, C., Huang, T., Zhang, R., Yi, X.: Pld-slam: a new rgb-dslam method with point and line features for indoor dynamic scene. ISPRS Int. J. Geo-Inf. 10(3), 163 (2021)
16. Yan, L., Hu, X., Zhao, L., Chen, Y., Wei, P., Xie, H.: Dgs-slam:A fast and robust rgbd slam in dynamic environments combinedby geometric and semantic information. Remote Sens. 14(3), 795(2022)
17. Li, S., Lee, D.: Rgb-d slam in dynamic environments using staticpoint weighting.IEEERobot.Autom.Lett.2(4),2263–2270(2017).
18. Cheng, J., Wang, C., Meng, M.Q.-H.: Robust visual localizationin dynamic environments based on sparse motion removal. IEEE Trans. Autom. Sci. Eng. 17(2), 658–669 (2020).
19. Cheng, J., Zhang, H., Meng, M.Q.-H.: Improving visual localiza-tion accuracy in dynamic environments based on dynamic regionremoval. IEEE Trans.Autom.Sci.Eng.17(3),1585–1596 (2020).
20. Liu, Y., Wu, Y., Pan, W.: Dynamic rgb-d slam based on static prob-ability and observation number. IEEE Trans. Instrum. and Meas.70, 1–11 (2021).
21. Liu, Y., Miura, J.: Kmop-vslam: Dynamic visual slam for rgb-dcameras using k-means and openpose. In: 2021 IEEE/SICE Inter-national Symposium on System Integration (SII), pp. 415–420(2021)
22. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: 2017IEEE International Conference on Computer Vision (ICCV), pp.2980–2988 (2017).
23. Ren, S., He, K., Girshick, R., Sun, J.: Faster rcnn: Towards real-time object detection with region proposal networks. IEEE Trans.Pattern Anal. Mach. Intell. 39(6), 1137–1149 (2017).
24. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bun-dle adjustment–a modern synthesis. In: Vision Algorithms: Theory and Practice: International Workshop on Vision Algorithms Corfu, Greece, September 21–22, 1999 Proceedings, pp. 298–372 (2000).Springer
25. Gálvez-López, D., Tardos, J.D.: Bags of binary words for fast placerecognition in image sequences. IEEE Trans. Robot. 28(5), 1188–1197 (2012)
26. Kaess, M., Ranganathan, A., Dellaert, F.: isam: Incrementalsmoothing and mapping. IEEE Trans. Robot. 24(6), 1365–1378(2008)
27. Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J.J., Del-laert, F.: isam2: Incremental smoothing and mapping using thebayes tree. Int. J. Robot. Res. 31(2), 216–235 (2012)
28. Davis, T.A., Gilbert, J.R., Larimore, S.I., Ng, E.G.: A columnapproximate minimum degree ordering algorithm. ACM Trans.Math. Softw. (TOMS) 30(3), 353–376 (2004)
29. Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: Abenchmark for the evaluation of rgb-d slam systems. In: Proc. ofthe International Conference on Intelligent Robot Systems (IROS)(2012)
30. Zhang, J., Liu, Y., Guo, C., Zhan, J.: Optimized segmentation withimage inpainting for semantic mapping in dynamic scenes AppliedIntelligence, 1–16 (2022)
31. Chen, H., Zhang, J., Chen, B., Li, B.: Global practical stabilizationfor non-holonomic mobile robots with uncalibrated visual param-eters by using a switching controller. IMA J. Math. Control Inf.30(4), 543–557 (2013)
32. Chen, H.: Robust stabilization for a class of dynamic feedbackuncertain nonholonomic mobile robots with input saturation. Int.J. Control Autom. Syst. 12(6), 1216–1224 (2014)
33. Chen, H., Chen, Y., Wang, M.: Trajectory tracking for under-actuated surface vessels with time delays and unknown controldirections. IET Control Theory & Appl. 16(6), 587–599 (2022)

附件B：外文文献的外文原文

|  |
| --- |
| **外文文献原文的文献著录信息：**  Li J, Luo J. Approach to 3D SLAM for Mobile Robot Based on RGB-D Image with Semantic Feature in Dynamic Environment[J]. Journal of Intelligent & Robotic Systems, 2023, 109(1): 15. |
| **外文文献著录信息的中文翻译：**  Li J, Luo J. 动态环境下基于RGB-D图像语义特征的移动机器人三维SLAM方法[J]. 智能与机器人系统杂志, 2023, 109(1): 15. |