基于目标检测网络的动态场景下视觉 SLAM 优化

方 娟，方振虎
（北京工业大学信息学部，北京 100124）

摘 要：为了降低动态环境对同时定位与建图（simultaneous localization and mapping，SLAM）位姿估计的干扰，提出一种将目标检测网络与 ORB-SLAM2 系统结合的方法. 在帧间估计阶段，使用目标检测网络获取当前帧的语义信息，得到潜在可移动物体边界框，结合深度图像并根据最大类间方差算法分割出边界框内前景，把落在前景中的动态特征点剔除，利用剩下的特征点估计位姿. 在回检测阶段，利用边界框构建图像语义特征，并与历史帧比较，查询相似关键帧，与视觉词袋法相比，该方法查询速度快，内存占用少. 在 TUM Techni 数据集上进行测试，结果表明该方法可以有效提高 ORB-SLAM2 在高动态场景中的性能.

关键词：同时定位与建图；动态环境；目标检测；图像分割；回环检测；位姿估计
中图分类号：TP 242 **文献标志码**：A **文章编号**：0254－0037(2022)05－0467－10
doi：10.11936/bjutxb2021020005

Vision SLAM Optimization in Dynamic Scene Based on Object Detection Network

FANG Juan, FANG Zhenhu
(Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China)

Abstract: To reduce the interference of dynamic environment on the pose estimation of vision SLAM, a method to combine object detection network with ORB-SLAM2 system was proposed. In the inter frame motion estimation stage, the object detection network was to obtain the semantic information of the current frame, the bounding box of potential movable objects was obtained, combined with the depth image and according to the maximum between-class variance algorithm, the foreground in the bounding box was segmented, the dynamic feature points in the foreground was deleted, and the remaining feature points used to estimate the pose. In the loop closure detection stage, the bounding box used to construct image semantic features, and query similar key frames compared with historical frames. Compared with Bag of Visual Word, the method has faster query speed and less memory consumption. We evaluate our method on TUM dataset was evaluated, and the results show that the proposed method can effectively improve the performance of ORB-SLAM2 in high dynamic scene.

Key words: simultaneous localization and mapping (SLAM); dynamic scene; object detection; image segmentation; loop closure detection; pose estimation

同时定位与地图构建（simultaneous localization and mapping，SLAM）在移动机器人、自主驾驶、增强现实等领域有着广泛的应用. SLAM 的目标是在没有任何先验知识的情况下，根据传感器数据实时构

收稿日期：2021-02-02；修回日期：2021-03-25
基金项目：国家自然科学基金资助项目(61202076)；北京市自然科学基金资助项目(4192007)
作者简介：方 娟(1973—)，女，教授，主要从事计算机体系结构、高性能计算方面的研究，E-mail:fangjuan@ bjut. edu. cn

建周围环境地图,同时根据这个地图推测自身的定位,实现自主导航.

近年来,有许多学者研究 SLAM,并取得了显著的成果,如 ORB-SLAM2[1]、基于直接法的大范围单目同时定位和地图构建方法(large-scale direct monocular SLAM,LSD-SLAM)[2]、稀疏直接法里程计(direct sparse odometry,DSO)[3]等方法. 然而,大多数方法无论是基于特征的方法还是直接法都是基于静态场景的假设,而真实场景中动态对象的存在是不可避免的[4]. 如果动态对象具有较强的纹理信息,系统会从动态对象上提取大量的特征,当跟踪到不稳定的特征点时,会严重影响姿态估计,造成较大的轨迹误差甚至跟踪丢失. 此外,视觉词袋(bag of visual word,BoVW)[5]方法虽然在开源 SLAM 框架中取得了不错的效果,但对场景中的光照条件和动态物体等参数十分敏感,容易造成误匹配[6-7].

随着深度学习算法的发展和计算机性能的提高,目标检测和语义分割方法取得了很大的进展[8-9]. 研究人员逐渐意识到这些方法可能有助于解决上述 SLAM 问题. 传统视觉 SLAM 技术与基于深度学习的语义分割、目标检测方法结合可以大大提高动态环境下 SLAM 系统的鲁棒性和准确性[10].

Bescos 等[11]提出 DynaSLAM,它利用 Mask-RCNN[12]获得语义分割结果,进而判断可能移动的特征点. 同时使用多视图几何的方法检测图中语义分割未检出的动态物体,然后将 2 个检测结果合并. 对于一个特征点,只要 2 个检测结果中有一个是动态的,就认为该特征点是动态的,并将其删除. 但是,由于使用 Mask-RCNN 网络,加上背景修复功能,系统难以实时运行.

Dynamic-SLAM[13]使用 SSD(single shot multibox detector)[14]网络检测潜在的动态对象,把出现在目标边界框内部的所有特征点删除,然后采用基于相邻帧速度不变特性的补偿算法来处理目标检测缺少、遗漏的情况. 但是,目标的姿态不同,检测框的大小也不同,这样导致剔除的特征点过多,容易跟踪失败.

针对动态环境下 SLAM 系统的研究,大部分工作都集中在提高机器人在视觉里程计定位精度上,但是所提出的模型很难满足移动机器人的实时性,而且对于光线变化环境下如何同时去优化回环检测几乎没有研究. 动态场景下 SLAM 的优化可以从 2 个方面出发:一个是相机姿态估计的准确性,另一个是回环检测的准确性.

为了提高室内动态环境下 SLAM 系统的鲁棒性,本文提出了一种基于语义边界框和深度图的动态环境下 SLAM 系统位姿优化方法. 该系统基于 ORB-SLAM2 架构,采用 RGB-D 相机,首先使用目标检测网络检测当前帧所有可识别物体,得到物体语义信息与边界框,根据语义信息区分潜在移动物体,如“人”,结合边界框与当前帧对应的深度图,分割出前景物体,剔除物体上的特征点,用剩下的特征点来计算相机位姿信息,同时使用可识别物体的语义信息和边界框构造图像特征,然后与历史帧比较,检测是否发生回环. 实验结果表明,本方法可以有效提高 ORB-SLAM2 在高动态场景中的性能.

1 系统概述

本文提出的 SLAM 系统的结构见图 1,在经典开源系统 ORB-SLAM2 基础上,新增了目标检测线程,如图中绿色部分,将目标检测结果嵌入到视觉里程计与回环检测模块. 本系统视觉传感器采用 RGB-D 相机,该相机不仅能获得场景彩色图像,而且还能得到彩色图像中每个像素对应的深度值,这个深度值的大小对应着该像素距离相机的远近.

首先,采用 ORB(oriented FAST and rotated, BRIEF)[15]特征提取模块提取当前帧的特征点,同时通过目标检测模块检测场景语义信息,得到物体的语义边界框,根据语义信息区分潜在动态物体与静态物体,然后把潜在动态语义边界框送入边界框内前背景分割模块分割出前景信息,与 ORB 特征点相结合,根据特征点坐标是否落在分割区域来确定是否剔除该特征点,然后使用剩下的特征点计算相机位姿. 同时,如果当前帧中不存在潜在可移动物体,那么就把该帧送入回环检测模块,根据图像中物体边界框信息构建语义特征图,与历史特征图比较,确定是否检测到回环. 得到帧间估计结果或回环检测结果后,需要经过后端优化模块进行全局轨迹优化,最后建立地图.

目前,主流的目标检测算法可分为一阶段方法和两阶段方法[16-17]. 两阶段检测算法在精度和准确度上都有较大的优势,但在检测速度上,一阶段检测算法具有更好的性能. SLAM 系统对目标检测的实时性提出了较高的要求,一阶段检测算法省去了候选区域的生成步骤,在同一个卷积神经网络中实现了特征提取、目标分类和回归,将目标检测过程简化为一个端到端的回归问题,大大提高了基于深度学习的目标检测算法的速度. 本文目标检测算法采用

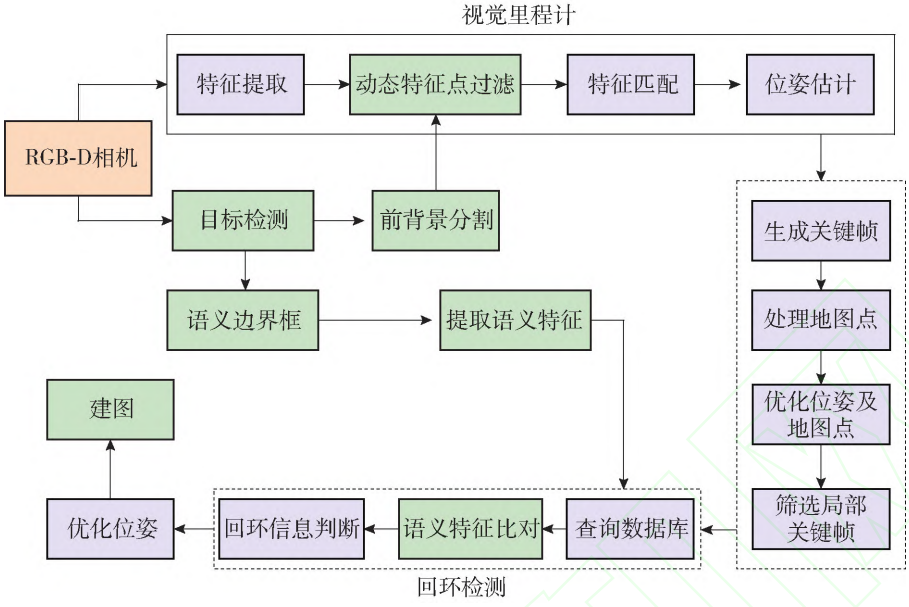
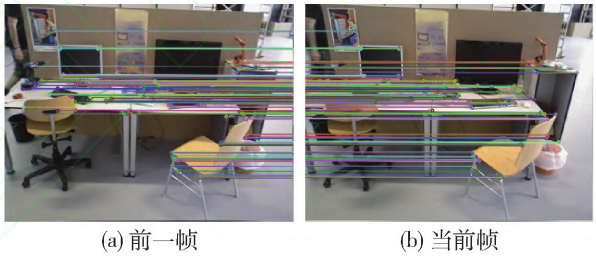


图 1　本文算法框架

Fig. 1　Framework of the proposed method

YOLOv4[18]，YOLOv4 算法在保持高处理帧速率的同时具有最先进的精度，在 MS-COCO 数据集上的预测精度达到 43.5%.

本文在低功耗 GPU MX150 上测试，YOLOv4 推理速度约为 10 帧/s，而 MASK-RCNN 推理速度不到 1 帧/s. 对于目标检测网络，高精度不再是唯一要求，希望模型能在移动设备上持久运行. 因此，用低功耗的硬件实时处理输入图像，对于移动机器人变得很重要.



(a) 前一帧　　(b) 当前帧

图 2　特征匹配

Fig. 2　Feature point matching

2　剔除潜在动态特征点的视觉里程计

目前，帧间估计主流的计算方法有特征点法与直接法. 由于特征点对光照、运动、旋转比较不敏感，相机运动较快也能跟踪成功，鲁棒性好，因此，基于特征点法的视觉里程计目前比较成熟[19]. ORB 特征则是目前非常具有代表性的实时图像特征. 提取 ORB 特征点后，接下来需要进行特征匹配. 特征匹配解决了 SLAM 中的数据关联问题，即确定当前看到的路标与之前看到的路标之间的对应关系. 其特征匹配效果如图 2 所示.

对 2 幅 RGB-D 图像进行了特征匹配之后，得到了特征点之间的对应关系，假设有一组配对好的 3D 点

$$P=\{p_1,p_2,\cdots,p_n\},Q=\{q_1,q_2,\cdots,q_n\} \tag{1}$$

现在需要求解一个欧氏变换 R、t，使得

$$\forall i,p_i=Rq_i+t \tag{2}$$

成立. 先定义第 i 对点的误差项

$$e_i=p_i-(Rq_i+t) \tag{3}$$

然后，构建最小二乘问题，求使误差平方和达到极小的 R，t，公式为

$$\min_{R,t}\frac{1}{2}\sum_{i=1}^{n}\|(p_i-(Rq_i+t))\|_2^2 \tag{4}$$

如果定义 2 组点的质心为

$$p=\frac{1}{n}\sum_{i=1}^{n}(p_i),\ q=\frac{1}{n}\sum_{i=1}^{1}(q_i) \tag{5}$$

那么目标函数可以简化为

$$\min_{R,t}J=\frac{1}{2}\sum_{i=1}^{n}\|p_i-p-R(q_i-q))\|^2+\|p-Rq-t\|^2 \tag{6}$$

如果有大量特征点在移动，那么估计出的位姿信息会有较大偏差，如果在位姿计算前，剔除动态特征点，那么就能有效地降低误差. 如图 3 所示，可以看见图中有 2 个人，其中一位坐着，另一位在走动. 系统从其中一个人身上采集了大量的特征点，如果把这些特征点删除掉，那么剩下的特征点就有利于计算位姿. 图 3(b)是删除后的效果.

(a) 删除前 (b) 删除后

图 3 删除潜在动态特征点

Fig. 3 Delete potential dynamic feature points

本文使用已训练的目标检测网络预测每一帧图像,建立先验语义信息,然后结合语义信息与深度图,分割出可移动物体,剔除该物体上的特征点.

对于目标检测,目标周围的边界框不能完全拟合目标的实际边界,不可避免地包含一些背景信息.

在这种情况下,判断特征点是否在目标对象上并不容易. 基于深度学习的实例分割虽然在分割效果上有很好的表现,但需要花费较多的计算时间. 因此,需要一种快速分割方法来提取物体边界框中的前景. 由于目标检测框与深度图结合,得到的图像区域内只存在单个前景,所以本文考虑单目标前景区域提取问题,使用最大类间方差算法. 最大类间方差法是一种自适应的阈值选取的算法,它是按图像的灰度特性,将图像分成背景和目标两部分. 背景和目标之间的类间方差越大,说明构成图像的两部分的差别越大,当部分目标错分为背景或部分背景错分为目标都会导致两部分差别变小[20].

RGB-D 相机提供的深度图存在像素的深度值为 0 的情况,可能是该点的深度值超出了相机量程,或者没有检测到深度. 在计算深度分割阈值之前,应该首先过滤掉深度图中深度值为 0 的像素. 因此,修改后的最大类间方差算法如下.

对于图像 $I(x,y)$,假设图像的大小为 $W\times H$,将前景(即目标)和背景的分割阈值记作 T,图像中灰度值小于阈值 T 的像素个数记作 N_0,像素灰度大于等于阈值 T 的像素个数记作 N_1,像素为 0 的个数记作 E,属于前景的像素点数占整幅图像的比例记为 ω_0,则

$$\omega_0=\frac{(N_0-E)}{(W\times H-E)} \tag{7}$$

其平均灰度为

$$\mu_0=\frac{\sum_{i}^{N_0-E} I(x_i,y_i)}{N_0-E} \tag{8}$$

将背景像素点数占整幅图像的比例记为 ω_1,则

$$\omega_1=\frac{N_1}{(W\times H-E)} \tag{9}$$

其平均灰度为

$$\mu_1=\frac{\sum_{j}^{N_1} I(x_j,y_j)}{N_1} \tag{10}$$

将图像的总平均灰度记为 μ,则

$$\mu=\omega_0\times\mu_0+\omega_1\times\mu_1 \tag{11}$$

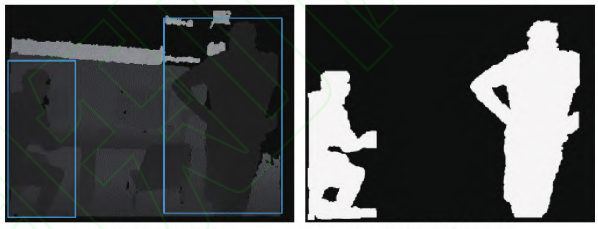
将类间方差记为 g,则

$$g=\omega_0\times(\mu_0-\mu)^2+\omega_1\times(\mu_1-\mu)^2 \tag{12}$$

将式(11)代入式(12),得到等价公式为

$$g=\omega_0\times\omega_1\times(\mu_0-\mu_1)^2 \tag{13}$$

最后,边界框内前背景分割的结果如图 4 所示.



(a) 语义边界框 (b) 前景分割结果

图 4 边界框内图像分割

Fig. 4 Image segmentation in bounding box

3 基于语义边界框的回环检测

只有相邻关键帧数据时,视觉里程计无从消除累积误差. 回环检测模块能够给出除了相邻帧之外的一些时隔更加久远的约束. 回环检测的实质是图像匹配问题,但对实时性要求很高,需要能满足机器人快速搜索历史帧数据. 目前,一些成熟的视觉 SLAM 系统,如 ORB-SLAM2,使用了 BoVW 方法来检测回环. 该方法将场景的一些视觉信息存储为一个视觉词典,利用尺度不变特征变换(scale-invariant feature transform,SIFT)、ORB 等特征点构造图像特征,从而计算图像之间的相似度.

虽然 BoVW 方法在开源 SLAM 框架中取得了显著的效果,但是由于可移动物体的存在、光照条件的变化,BoVW 方法识别真实回环的效率不高,容易出现误匹配问题. 此外,BoVW 中的词汇量越大,占用的内存就越多. 本文专注于在线特征数据库,尝试通过使用不同于传统 BoVW 的方法来减少存储特征的内存使用,同时通过使用图像的深层和更抽象的特征而不是手工制作的特征来改进回环检测. 使用 BoVW 方法,不同光线下 ORB 特征提取与匹配结果如图 5 所示.

(a) 较暗光线　　(b) 较亮光线

图 5　不同光线下 ORB 特征提取与匹配

Fig. 5　ORB feature extraction and matching under different light conditions

图 5 中的 2 张图光线不同，左图室内的灯只开了一半，可以看见后面的墙壁比较暗，右图室内的灯全开了，可以看见图像顶部的电灯开了，后面的墙壁比较亮. 然后对这 2 张图像提取 ORB 特征，每张图都提取了 200 个特征，匹配成功的只有 60 个，特征点匹配成功数量显著减少.

用该图像测试目标检测，结果如图 6 所示. 图 6(a)共检测到 12 个物体，图 6(b)共检测到 15 个物体，检测到的物体类别没有变化，数量上有些差异，每个物体的置信度可能不同. 图 6(a)比图 6(b)少的的 3 个物体分别为 cup_0.45、chair_0.50 和 tv_0.41. 这里的 0.45、0.50、0.41 表示的是置信度，也就是说去掉那些置信度比较小的物体，那么留下的物体在不同光线下差异就会很小，因此，考虑使用这些边界框信息描述整幅图像.

(a) 开启一半灯光目标检测结果

(b) 开启全部灯光目标检测结果

图 6　不同光线下目标检测结果

Fig. 6　Object detection results in different light

回环检测前需要提取图像特征，下面详细介绍特征提取方法. 从图 6 的图像中可以得到当前帧中边界框数量和每个边界框的坐标 B_{cd}、面积 B_{ar}、类别 B_{ty}、置信度 B_{cf}.

环境亮度不同，目标检测得到的物体数量、类别、和置信度可能就不同，但是置信度越高在不同亮度下检测到的可能性就越大. 因此，首先遍历所有的边界框，去掉置信度小于 T_1 的，再统计边界框的数量 B_{ct}. 然后，按照边界框的面积把边界框从大到小排序，得到面积最大的边界框 M_{ab} 和与之对应的类别 M_{at}，累加所有边界框的面积得到 T_{ba}. 如果大于 T_2，则丢掉该图像，目的是为了减小边界框面积太大的物体对特征比对阶段的影响. 之后，根据 B_{ty} 构建特征表 B_{ta}. B_{ta} 特征结构如表 1 所示.

表 1　特征结构

Table 1　Feature structure

类别	面积比	位置	置信度
TV	0.383	(413, 1)(640, 271)	0.890
	0.257	(216, 2)(442, 184)	0.810
keyboard	0.182	(144, 185)(368, 315)	0.920
plant	0.147	(0, 0)(142, 166)	0.670
mouse	0.020	(329, 283)(395, 331)	0.910
cup	0.010	(228, 110)(258, 158)	0.627

从表中可以看出，共有 5 种类型，6 个物体，其中面积最大的类别 M_{at} 为 TV，该类别下有 2 个实例，并按面积大小做了排序，最大面积比为 0.383. 最后把 M_{at}、B_{ta} 组合一起，得到 F_{map}. 特征提取后，进行特征比对，算法流程如算法 1 所示. 在算法 1 中，首先根据 M_{at} 加快匹配速度，之后为了过滤不匹配的图像，使用了交并比(intersection over union, IOU)[21]，即 2 个矩形框面积的交集和并集的比值，记为 I. 假设 2 个边界框 A 和 B，坐标已知，面积为 S_A、S_B，那么

$$I_{A,B}=\frac{S_A\cap S_B}{S_A\cup S_B} \tag{14}$$

算法 1　特征比对

输入：当前帧 F_{map}，历史帧 F_{list}

　　阈值 T

输出：是否检测到回环

For F_{map_i} in F_{list}

　$B_{ty\_i}=F_{map\_i}->M_{at}$

判断特征表中最大面积元素是否相同

IF $F_{map}->M_{at}!=B_{ty\_i}$:

continue

$B_{ta}=F_{map}->B_{ta}$

$B_{ta\_i}=F_{map\_i}->B_{ta}$

$R_1=\text{calcIou}(B_{ta}0],B_{ta\_i}[0])$

计算 2 个特征表中最大面积元素交并比

IF $R_1<0.8$:

continue

$R_2=\text{calcAllBoxIou}(B_{ta},B_{ta\_i})$

IF $R_2>T$

return True

return False

因此,根据式(14)可以得到 R_1,本文考虑到边界框面积越大的物体越容易检测到,而且在现实生活中位置改变的可能性就越小,比如水杯、鼠标会经常改变位置,但是显示器、桌子就不会经常改变位置,因此,采用加权 IOU 的方式来计算整幅图像的特征相似度,calcAllBoxIou 计算方式如下:

1) 保留当前帧特征表 B_{ta} 与历史帧特征表 B_{ta_i} 中有相同类型的边界框.

2) 根据 B_{ta} 中记录的面积比,计算两幅图的加权 IOU,得到 R_3.

3) 根据 B_{ta_i} 中记录的面积比,计算两幅图的加权 IOU,得到 R_4.

4) 返回 $(R_3+R_4)/2$.

其中计算加权 IOU 的方式为

$$O_{A,B}=\sum_{i=0}^{N}\sum_{j=0}^{C_i}S_{i,j}\times I_{A,B} \tag{15}$$

式中:$O_{A,B}$ 为 A 与 B 的加权 IOU;N 为相同类别数量;C_i 为当前类别对应的边界框数量;$S_{i,j}$ 为该边界框的面积比.

4 实验结果

把改进的 SLAM 算法称为 DyOD-SLAM,使用 TUM RGB-D 数据集和实验室内采集的不同光线相同场景数据集,从 2 个方面来评估提出的视觉 SLAM 优化方法的性能. 实验中 TUM 数据集一共选取 3 个序列:freiburg3_walking_halfsphere(w_half),freiburg3_walking_xyz (w_xyz)是 2 个动态场景视频序列,用来验证帧间估计的准确性;freiburg1_room 用来验证回环检测的实时性和内存消耗. 这些序列包含 640×480 像素的 8 位 RGB 图像和 640×480 像素的 16 位深度图像. 此外,真实相机移动轨迹通过具有 8 个高速跟踪摄像机的高精度运动捕获系统获得.

这 2 个选定的动态序列是在“办公桌”场景中拍摄的,2 个人正在走路或坐着. “w_xyz”序列表示 2 个人走过办公桌前的场景,并且将相机向着 *xyz* 方向移动,同时保持角度不变. “w_half”序列表示 2 个人正走过办公室,并且摄像机在直径约 1 m 的小半球上移动. 这些序列旨在评估视觉 SLAM 算法对移动动态对象的鲁棒性.

4.1 相机位姿误差实验

对于位姿误差评估,本文采用绝对路径误差(absolute trajectory error, ATE)和相对位姿误差(relative pose error, RPE)作为依据. ATE 表示真实轨迹点坐标与 SLAM 系统定位点坐标的误差,可以通过对比估计轨迹与真实轨迹的重合度来判断. RPE 表现了 SLAM 系统得到的帧间位姿变换值与真实值的误差,可以用两者差值均方根的形式进行表示,即

$$E_{RPE}=\sqrt{\frac{1}{n}\sum_{i,j}(\delta_{i,j}-\tilde{\delta}_{i,j})^2} \tag{16}$$

式中:E_{EPE} 为相对位姿误差;$\tilde{\delta}_{i,j}$ 为 SLAM 系统估计的 i、j 时刻间相对位姿变换值;$\delta_{i,j}$ 为对应的真实位姿变化值. 本文与 ORB-SLAM2 做对比实验,得到相机估计轨迹,通过与真实轨迹比较,计算出 E_{RPE} 用来评估 SLAM 系统. 实验结果如图 7 ~ 9 所示.

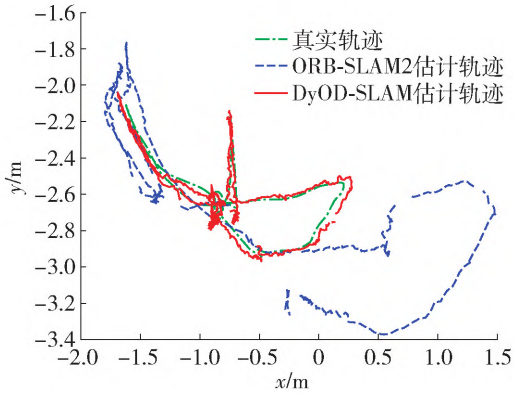


图 7 w_half 数据集下绝对路径轨迹

Fig. 7 Absolute path trajectories in w_half datasets

从图 7、8 可以看出,DyOD-SLAM 在 w_half、w_xyz 数据集中估计的运动轨迹与真实轨迹基本重合,相比较 ORB-SLAM2 有较大的提升.

图 9 中横坐标 t 表示系统运行的时间. 可以看出,ORB-SLAM2 在 w_half 序列前 20 s 估计的 RPE 波动较大,后续比较平稳,而 DyOD-SLAM 仅在第 8 s

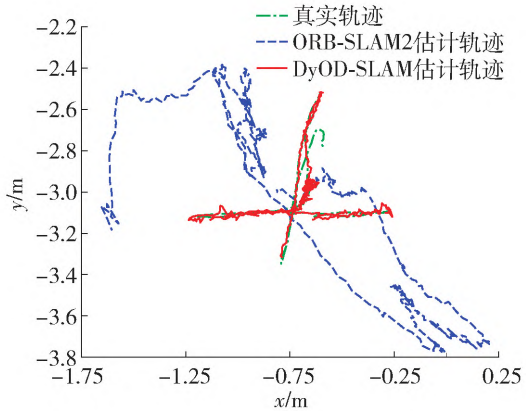
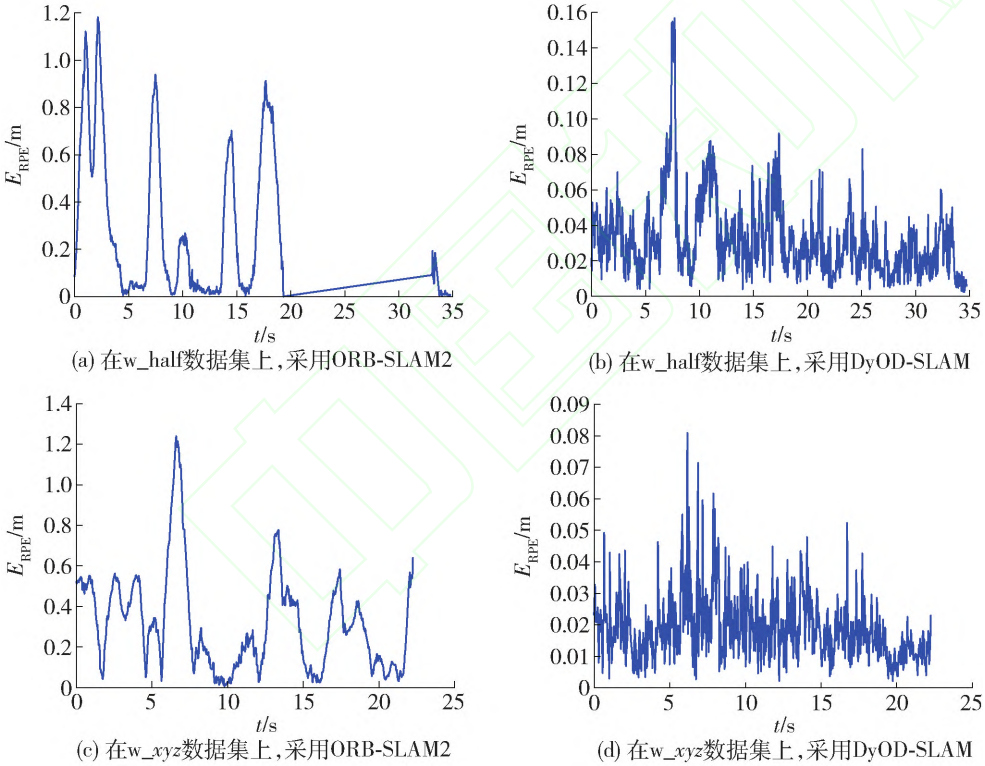


图 8　w_xyz 数据集下绝对路径轨迹

Fig. 8　Absolute path trajectories in w_xyz datasets

左右有较大的波动. 在 w_xyz 序列第 7 s 左右时，两系统 RPE 均出现明显波动. 分析其原因，视野中出现了较大面积的移动目标，动态特征点导致了系统 RPE 增大. ORB-SLAM2 在 w_half、w_xyz 数据集上 RPE 上限为 1.2 m，而 DyOD-SLAM 滤除了大部分动态特征点，在 2 个数据集上误差上限分别为 0.16、0.08 m，远小于 ORB-SLAM2，改进的系统 RPE 波动较低，说明 DyOD-SLAM 更稳定. 定量比较结果见表 2、3，评价指标包括均方根误差（root mean squard error，RMSE）、平均误差、标准差和中值误差.



(a) 在w_half数据集上，采用ORB-SLAM2　(b) 在w_half数据集上，采用DyOD-SLAM

(c) 在w_*xyz*数据集上，采用ORB-SLAM2　(d) 在w_*xyz*数据集上，采用DyOD-SLAM

图 9　ORB-SLAM2 与 DyOD-SLAM 在 2 个数据集上的相对位姿误差

Fig. 9　Relative pose errors between ORB-SLAM2 and DyOD-SLAM on two datasets

表 2　平移漂移结果

Table 2　Results of translation drift　m

算法	数据	RMSE	平均误差	中值误差	标准差
ORB-SLAM2	w_half	0.438	0.299	0.149	0.320
	w_xyz	0.395	0.314	0.277	0.240
DyOD-SLAM	w_half	0.038	0.031	0.026	0.021
	w_xyz	0.021	0.018	0.017	0.010
DS-SLAM	w_half	0.031	0.022	0.022	0.016
	w_xyz	0.026	0.021	0.017	0.015

表 3　旋转漂移结果

Table 3　Result of rotation drift　(°)

算法	数据	RMSE	平均误差	中值误差	标准差
ORB-SLAM2	w_half	9.049	6.236	0.057	6.557
	w_xyz	7.671	5.982	0.081	4.802
DyOD-SLAM	w_half	0.955	0.804	0.011	0.515
	w_xyz	0.589	0.454	0.006	0.375
DS-SLAM	w_half	0.814	0.703	0.013	0.810
	w_xyz	0.826	0.583	0.009	0.582

表 2 表示相机平移相对轨迹误差,表 3 表示相机旋转相对角度误差. 表 4 与表 5 详细地与 DS-SLAM[22]做了比较. DyOD-SLAM 在 w_half 序列,相对于 ORB-SLAM2 的 RMSE 性能分别提升 91.3%、89.4%,在 w_xyz 序列,性能分别提升 94.6%、92.3%.

表 4 与 ORB-SLAM2 相比平移误差的改进

Table 4 Improvement of translation error compared with ORB-SLAM2 %

算法	数据	RMSE	平均误差	中值误差	标准差
DS-SLAM	w_half	92.9	92.6	85.2	95.0
	w_xyz	93.4	93.3	93.8	95.3
DyOD-SLAM	w_half	91.3	89.6	82.5	93.4
	w_xyz	94.6	94.2	93.8	95.8

表 5 与 ORB-SLAM2 相比旋转误差的改进

Table 5 Improvement of rotation error compared with ORB-SLAM2 %

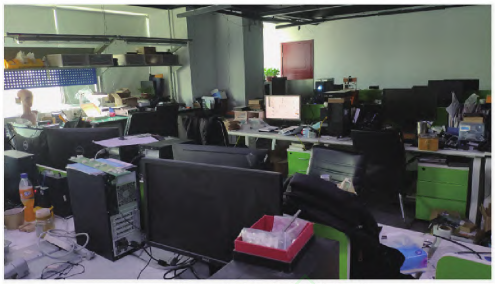
算法	数据	RMSE	平均误差	中值误差	标准差
DS-SLAM	w_half	91.0	88.7	77.1	87.6
	w_xyz	89.2	90.2	88.9	87.9
DyOD-SLAM	w_half	89.4	87.1	80.7	92.1
	w_xyz	92.3	92.4	92.5	92.2

DyOD-SLAM 在 w_xyz 序列相对平移误差要优于 DS-SLAM.

2 种算法在 w_half 序列的 RMSE 和标准差都大于 w_xyz 序列,原因是此序列相机不稳定,图像模糊,有些动态特征点未检测到. 从总体的性能提升可以看出本文的算法 DyOD-SLAM 在动态场景数据集上 E_{RPE}远小于 ORB-SLAM2. 无论是平移,还是旋转,在 RMSE 方面,DyOD-SLAM 精度提升了近 90%,表明了本文算法在动态环境下具有较高的稳定性.

4.2 回环检测实验

这里测试用的数据集来自白天实验室内场景,通过固定相机并按照固定轨迹抓拍得到. 为了得到不同光线下的数据,先开启一半的灯光得到 room_dark_half 数据集,然后开启全部灯光得到 room_light 数据集,如图 10 所示. 使用准确率-召回率曲线来评估结果.



(a) 开启一半灯光的数据集

(b) 开启全部灯光的数据集

图 10 相同数据集、不同亮度的图像对比

Fig. 10 Same dataset, different brightness

用目标检测算法提取 2 个数据集中所有图像的语义边界框,然后过滤置信度小于 T_1 的边界框,计算 2 张对应图像的 IOU 值,得到的结果相加,然后取平均值得到 I_a. 实验中 T_1 从 0 开始,测得当 T_1 为 0.55 时,I_a 最大. 确定 T_1 的值后,接下来要确定 T_2 的值. 假设图像中最大面积的边界框占所有边界框面积的比值为 S_{max},通常情况下这个值越大表示这个图像中小物体越多,在光线变化环境下这些物体被检测到的可能性就比较低,本文选取经验值 0.7 作为 T_2 的值.

实验中提取 room_light 数据集中图像的 ORB 特征、语义边界框特征构建特征数据库. 首先,在 room_light 据集下做回环检测测试;然后,在改变亮度后的 room_dark_half 数据集中遍历图像,提取特征并与特征数据库做比较;最后,得到不同亮度数据集下的准确率-召回率曲线. 不同亮度数据集上的回环检测比较如图 11 所示.

freiburg1_room 数据集含有 1362 张图片,统计了 ORB 特征数据库与语义特征数据库占用磁盘大小和内存大小和所有图像暴力匹配需要的时间,结果如表 6 所示.

由表 6 中可以看出,DyOD-SLAM 占用系统资源非常少,较适合大场景下回环检测,而且实时性更高. 图 11 中,本文改进的方法在原始亮度环境下,相同召回率下准确率明显高于传统方法,特别是当

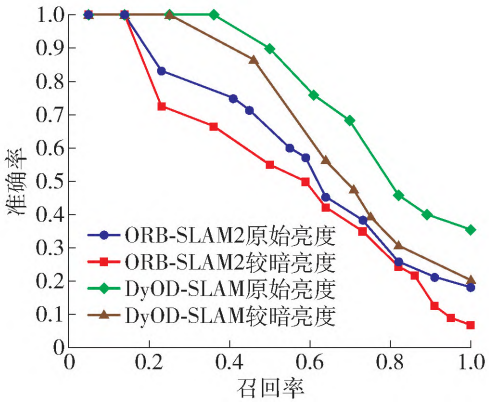


图 11　不同亮度数据集上的回环检测比较

Fig. 11　Comparison of loop closure detection on different brightness datasets

表 6　资源消耗对比

Table 6　Comparison of resource consumption

算法	内存占用/MB	磁盘占用/B	特征比对时间/s
ORB-SLAM2	200	62.000	2.000
DyOD-SLAM	10	0.314	0.005

召回率小于 0.38 时，本文的方法一直保持 100% 的准确率. 随着召回率上升，准确率下降，但是 DyOD-SLAM 的准确率下降速度小于 ORB-SLAM2，当召回率为 0.78 时，2 种方法准确率差距最大. 当亮度发生改变时，2 种方法准确率都有下降，DyOD-SLAM 准确率下降得比较少，在召回率小于 0.6 的情况下，基于语义特征的回环检测方法性能都高于 ORB-SLAM2. 其准确率下降的原因是有些物体未检测到，或置信度过低，发生在小物体比较多的情况下. 实验中发现，如果室内场景中可识别的物体较少，本文方法较容易产生漏匹配问题，因此，后续考虑将图像全局特征与语义特征结合，适应更多场景.

5　结论

1）本文基于 ORB-SLAM2，结合目标检测网络修改帧间估计模块、回环检测模块，提高系统在动态环境下位姿估计的稳定性.

2）在帧间估计阶段，使用目标检测网络确定当前图像是否存在潜在可移动物体，根据最大类间方差算法分割边界框内前背景，过滤潜在动态特征点，相对位姿准确率提升近 90%.

3）在回环检测阶段，使用上个阶段提取的语义边界框信息，构建语义特征，实现特征比对，在召回率小于 0.6 的情况下，相较于 ORB-SLAM2 准确率提升 20%，而且占用系统资源少，查询速度快.

4）本文只是根据语义信息来区分移动物体，没有实际验证是否发生移动，存在遗漏动态特征点的情况.

5）本文的回环检测方法适用于室内，而且需要较丰富的可识别物体，适应场景能力不足，未来将针对此问题进行研究.

参考文献：

[1] MUR-ARTAL R, TARDÓS J D. ORB-SLAM: an open-source slam system for monocular, stereo, and RGB-D cameras[J]. IEEE Transactions on Robotics, 2017, 33(5): 1255-1262.

[2] ENGEL J, SCHÖPS T, CREMERS D. LSD-SLAM: large-scale direct monocular SLAM[C]// European Conference on Computer Vision. Berlin: Springer, 2014: 834-849.

[3] ENGEL J, KOLTUN V, CREMERS D. Direct sparse odometry[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 40(3): 611-625.

[4] YU C, LIU Z, LIU X J, et al. DS-SLAM: a semantic visual SLAM towards dynamic environments[C]// 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Piscataway: IEEE, 2018: 1168-1174.

[5] GARCIA-FIDALGO E, ORTIZ A. IBOW-LCD: an appearance-based loop-closure detection approach using incremental bags of binary words[J]. IEEE Robotics and Automation Letters, 2018, 3(4): 3051-3057.

[6] GAO X, ZHANG T. Unsupervised learning to detect loops using deep neural networks for visual SLAM system[J]. Autonomous Robots, 2017, 41(1): 1-18.

[7] XIA Y, LI J, QI L, et al. Loop closure detection for visual SLAM using PCANet features[C]// 2016 International Joint Conference on Neural Networks (IJCNN). Piscataway: IEEE, 2016: 2274-2281.

[8] 张顺，龚怡宏，王进军. 深度卷积神经网络的发展及其在计算机视觉领域的应用[J]. 计算机学报，2019，42(3)：453-482.
ZHANG S, GONG Y H, WANG J J. The development of deep convolution neural network and its applications on computer vision[J]. Chinese Journal of Computers, 2019, 42(3): 453-482. (in Chinese)

[9] 张政馗，庞为光，谢文静，等. 面向实时应用的深度学习研究综述[J]. 软件学报，2020，31(9)：2654-2677.
ZHANG Z K, PANG W G, XIE W J, et al. Deep learning for real-time applications: a survey[J]. Journal of

Software, 2020, 31(9): 2654-2677. (in Chinese)

[10] 赵洋, 刘国良, 田国会, 等. 基于深度学习的视觉 SLAM 综述[J]. 机器人, 2017, 39(6): 889-896.
ZHAO Y, LIU G L, TIAN G H, et al. A survey of visual SLAM based on deep learning[J]. Robot, 2017, 39(6): 889-896. (in Chinese)

[11] BESCOS B, FÁCIL J M, CIVERA J, et al. DynaSLAM: tracking, mapping, and inpainting in dynamic scenes [J]. IEEE Robotics and Automation Letters, 2018, 3(4): 4076-4083.

[12] HE K, GKIOXARI G, DOLLÁR P, et al. Mask R-CNN [C]// Proceedings of the IEEE International Conference on Computer Vision. Piscataway: IEEE, 2017: 2961-2969.

[13] XIAO L, WANG J, QIU X, et al. Dynamic-SLAM: semantic monocular visual localization and mapping based on deep learning in dynamic environment[J]. Robotics and Autonomous Systems, 2019, 117: 1-16.

[14] LIU W, ANGUELOV D, ERHAN D, et al. SSD: single shot multibox detector[C]// European Conference on Computer Vision. Berlin: Springer, 2016: 21-37.

[15] RUBLEE E, RABAUD V, KONOLIGE K, et al. ORB: an efficient alternative to SIFT or SURF[C]// 2011 International Conference on Computer Vision. Piscataway: IEEE, 2011: 2564-2571.

[16] JIAO L, ZHANG F, LIU F, et al. A survey of deep learning-based object detection[J]. IEEE Access, 2019, 7: 128837-128868.

[17] ZHAO Z Q, ZHENG P, XU S, et al. Object detection with deep learning: a review[J]. IEEE Transactions on Neural Networks and Learning Systems, 2019, 30(11): 3212-3232.

[18] BOCHKOVSKIY A, WANG C Y, LIAO H Y M. YOLOv4: optimal speed and accuracy of object detection [EB/OL]. [2020-10-01]. https://arxiv.org/abs/2004.10934v1.

[19] 张峻宁, 苏群星, 刘鹏远, 等. 一种自适应特征地图匹配的改进 VSLAM 算法[J]. 自动化学报, 2019, 45(3): 553-565.
ZHANG J N, SU Q X, LIU P Y, et al. An improved VSLAM algorithm based on adaptive feature map[J]. Acta Automatica Sinica, 2019, 45(3): 553-565. (in Chinese)

[20] QU Z, ZHANG L. Research on image segmentation based on the improved Otsu algorithm[C]// 2010 Second International Conference on Intelligent Human-Machine Systems and Cybernetics. Piscataway: IEEE, 2010, 2: 228-231.

[21] NOWOZIN S. Optimal decisions from probabilistic models: the intersection-over-union case [C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE , 2014: 548-555.

[22] YU C, LIU Z, LIU X J, et al. DS-SLAM: a semantic visual SLAM towards dynamic environments[C]// 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Piscataway: IEEE, 2018: 1168-1174.

(责任编辑 梁 洁)