

动态环境下的语义地图构建

摘要: 三维语义地图在移动机器人导航、路径规划等任务中有着关键的作用,如何实时地构建三维语义地图尤为重要。当前同时定位和地图构建(SLAM)算法在一些场景下能够达到良好的性能,但是在动态环境下如何处理动态物体获得较高的定位精度,以及理解周围场景中存在物体及其位置信息等问题仍然没有很好的解决。本文提出了一种可在动态环境下构建语义地图的算法。本算法在 ORB-SLAM2 上进行改进,在跟踪线程中加入动静点检测算法,剔除检测为动态的特征点,提高了动态环境下的定位精度,添加目标检测线程对关键图像进行目标检测,在地图构建线程中加入 Octo-Map 稠密地图构建,同时根据检测结果构建 3D 目标数据库,在数据集和实验室中对算法进行测试,结果表明算法具有较高的精度和实时性。

关键词: 语义地图构建; 动态点检测; 目标检测; 视觉 SLAM

中图分类号: TP249

文献标识码: A

Semantic Mapping in Dynamic Environments

Abstract: Three-dimensional semantic map plays a key role in mobile robot navigation, path planning and other tasks. So how to construct 3D semantic maps in real time is especially important. At present, simultaneous localization and mapping (SLAM) algorithms can achieve good performance in some scenarios. However, how to deal with dynamic objects to obtain high positioning accuracy in the dynamic environment, and understand the existence of objects and their location information in the surrounding scenes are still not well solved. This paper presents an algorithm for constructing semantic maps in dynamic environments. This algorithm is improved on ORB-SLAM2. The dynamic and static point detection algorithm is added to the tracking thread to eliminate the detection as dynamic feature points, which improves the positioning accuracy in dynamic environment. Object detection threads are added to detect key images. Octo-Map dense map is added to the mapping threads. At the same time, 3D object database is constructed according to the 2D detection results. The algorithm is tested in the laboratory and dataset, and the results show that the algorithm has high accuracy and real-time performance.

Keywords: Semantic Mapping; Dynamic Point Detection; Object Detection; VSLAM

1 引言(Introduction)

机器人技术代表了当今世界高新技术的前进方向。机器人对周围环境的感知、建模和理解的能力是决定其智能水平的关键因素,因为只有机器人对自身和周围环境正确建模后,机器人才可能完成如导航、抓取等任务。在实际研究中,机器人常常装载着相机、GPS、激光雷达、超声波、IMU 等传感器。这些装置为机器人提供最原始的数据,不过还不能为机器人提供更加结构化、层次化的信息,需要经过多种算法的处理才能获取较好的结果。

在移动机器人技术领域,地图构建技术是感知、建模、规划与理解的基础与核心。机器人同时定位与地图构建技术(Synchronous Localization and Mapping)解决的就是如何利用传感器的数据进行自定位,与此同时构建出周围环境的地图。视觉 SLAM 是以相机为主要外部传感器的 SLAM 系统,有时还会结合其他传感器的信息,如 IMU 信息,可应用于机器人、无人驾驶和增强现实(AR)等领域,是近年来热门的研究方向。目前大多数的视觉 SLAM 算法是在静态场景这一假设下运行的,获得的信息中不包含高层的语义信息,仅仅是一些几何点信息。通过 SLAM 算法机器人可以精确定位,但无法对环境中的物体进行识别和建模,这将导致机器人无法根据环境中的高层语义信息进行智能操作。随着室内外移动机器人技术、自动驾驶和增强现实技术等研究及应用的发展,室内外场景中语义理解能力的重

要性愈加显著。

近几年,深度学习技术快速发展,为解决上述问题提供了新的思路。考虑快速性与实时性而提出的 SSD^[4]目标检测框架在保证检测精度的同时提高了速度。为了提高神经网络的运行速度,以 Mobile Net^[7-8]为代表的小型化网络被提出,这些网络通过巧妙设计网络结构、简化卷积核等方法使得网络的运算量降低。ORB-SLAM2^[1-2]算法是近年来基于特征点法的视觉 SLAM 算法中较为出色的一个算法框架。输入视觉 SLAM 算法的图像信息流中常常包含了各种物体,结合目标检测网络的语义信息提取优势与 SLAM 算法获取的精确的几何信息,使得机器人可以从周围环境中获取包含更加结构化、语义化、层次化的地图信息。近年来有一些研究根据目标检测结果的先验信息和动态点检测算法的测量信息来剔除相机视野中的动态点来提高算法定位精度。

本文以复杂的动态的室内实验室环境为背景,探索动态环境下构建语义地图的方法,结合基于 RGBD 相机的视觉 SLAM 系统和基于回归预测的多尺度预测的深度卷积神经网络 SSD 框架,设计算法剔除动态物体点、融合位姿信息和语义信息以及构建语义目标数据库,实时构建高层次语义地图。本文所提出的方法在动态环境下语义地图构建等方面具有一定的参考价值,可以帮助机器人实现更为智能的导航任务。

2 动态信息检测和 3D 语义信息获取 (Dynamic Information Detection and 3D Semantic Information Acquisition)

2.1 动态信息检测

当前的 SLAM 技术中，大多是根据场景是静态的这个假设来设计的，如果有运动物体进入相机的视野，其会影响相机自身位姿的估计，其二运动物体的轨迹将出现在点云图中，并且在将点云图转换为其他类型时将保留动态轨迹信息。包含动态对象信息的地图不能直接用于导航，因为我们无法判断移动机器人是否能够安全通过。在图像序列中寻找动态对象的区域，并消除这些区域的影响，是一项具有挑战性的工作。

我们基于二维视觉中的光流检测算法来判断相机视野中哪些像素对应的物体在运动，不过对于移动的相机来说，需要考虑相机自身运动带来的光流场变化。为剔除相机运动带来的影响，其一我们可以设置一个光流幅值较大的阈值，来滤除噪声和相机运动带来的光流，这样也有一个缺点对于相机视野中运动量过小的物体就不能检测到，但是这样处理的速度比较快，同时可以排除掉一些运动量较大的物体。其二我们可以对相机的运动建模，利用相邻两帧的特征点对来求解出相机的单应变换矩阵 H ，然后对当前图像进行反变换后，再和上一帧图像一起求解光流场，这样可以有效剔除掉一些相机的运动，不过在相机视角变化太大时，效果也变差了。可以使用 4 对点对联立线性方程组使用奇异值分解求解矩阵 H ，由于匹配点对数量较多，我们需要使用 RANSAC 方法得到更加鲁棒的结果，该方法如图 1 所示。

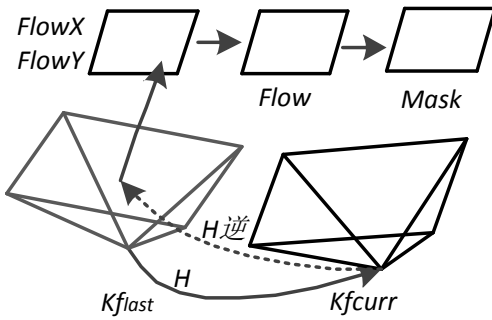


图 1 基于光流的动态点检测算法

Fig.1 Dynamic Point Detection Based on Optical Flow

实验中整幅图像光流场的计算时间较长，我们可以先对两副图像做下采样，然后再计算光流场，之后对光流图像进行上采样得到原始图像尺寸的光流场图像，这样可以大幅度提高计算速度，不过光流场的精度稍微降低一些。得到光流场之后我们可以通过水平和垂直光流值计算光流幅值 f_m ，然后使用一个阈值对光流场进行二值化，之后再对二值形式的光流场 f_{mask} 进行腐蚀和膨胀操作，降低一些噪声，阈值是一个实验中获得的数据，相机静止不动时，取值为 3~4 之间，相机运动时取值为 6~9 之间，也可以根据相机的运动量大小，设置一个自适应的阈值。该值取值过大，就会对一些运动量小的物体不

灵敏，取值过小就会被过多的噪声影响。

上述光流法动静点检测算法在相机运动时会产生大量的误判断，虽然通过单应变换矩阵进行校正以及自适应的阈值处理，在相机运动剧烈时会产生大面积的误判断，一旦提取的关键点数量过少，系统定位模块就容易丢失，需要重新定位。我们还尝试了另外一种基于三维视觉中的多视角几何的动静点检测算法^[5]。主要思路是，局部地图中的点投影到当前相机视角下，其测量的深度值应该和投影的深度值相差不大，否则就可能是有动态物体遮挡了当前的部分视野，其原理如图 2 所示。

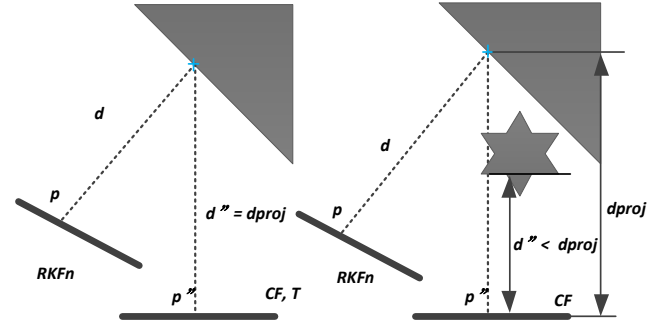


图 2 基于多视角几何的动态点检测算法

Fig.2 Dynamic Point Detection Based on Multi-view geometry

上图中左图所示为无动态遮挡物体下的局部地图，三角形表示参考帧的局部地图点，根据参考帧和当前帧的位姿关系可将局部地图点反投影到当前帧下，投影点的投影深度值和当前帧的测量值差不多，但是在右图所示有动态物体遮挡的情况下，测量深度小于投影深度（一定阈值，实验中取值为 0.4m），这种情况下可以将对应的点标记为动态点。

2.2 2D 目标检测

考虑到直接对点云进行 3D 语义分割的复杂性以及低实时性，我们采用 2D 目标检测结合点云的方式获取 3D 目标信息。为了保证目标检测算法的实时性，我们用 Mobile Net 网络作为 SSD 目标检测网络的前端特征提取网络，其局部结构如图 3 所示。

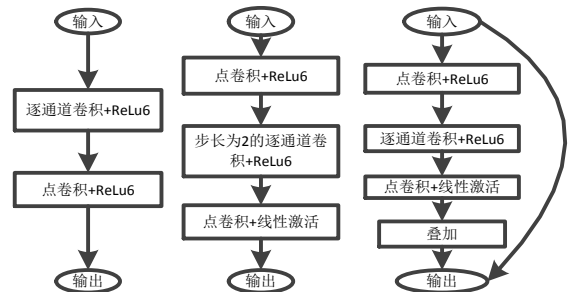


图 3 Mobile Net 结构

Fig.3 Mobile Net Structures

上左图为 Mobile Net-v1 结构，右边两个图为 Mobile Net-v2 的带下采样和无下采样的两个版本，核心思想是使用逐通道卷积代替普通卷积降低运算量，提高运算速度。本文结合 Mobile Net 和 SSD 形成的 Mobile Net-SSD 目标检测网络可以在普通 PC 机上使用 NCNN 平台近似实时运行。整体网络结构

The diagram illustrates the MobileNet architecture. It starts with a 'Mobile Net 骨架网络' (MobileNet backbone network) block. This is followed by a series of five '卷积下采样' (Convolutional downsampling) blocks, each progressively smaller in size. Arrows indicate the flow of data from the backbone network through each downsampling block. Additionally, skip connections are shown as arrows that bypass the downsampling blocks, connecting the output of the backbone network directly to the input of each subsequent downsampling block. The final output of the last downsampling block is fed into a '边框回归与分类' (Bounding box regression and classification) block, which then leads to a '非极大值抑制' (Non-maximum suppression) block.

Fig.4 Mobile Net-SSD Object Detection Network

我们使用 2.2 小节目标检测算法得到的 2D 检测结果和 RGBD 相机捕获的点云来获取 3D 语义目标信息。两种思路，一是以 2D 检测信息为主在有序点云上框取对应的点云，经过滤波等处理获取点云团的位置和空间尺寸信息；二是以点云为主，对点云进行分割后获取有序点云团投影在图像平面上的 2D 框，再和目标检测的 2D 框比对，根据交并比计算相似性进行匹配。前者根据 RANSAC 方法在对应检测框内选取目标物体点云的平均深度，根据平均深度对 2D 框内的点云进行滤波之后计算中心点、3D 包围框尺寸等信息，具体方法如图 5 所示。

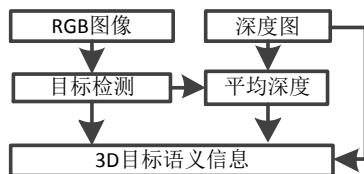


Fig.5 3D Semantic Information Acquisition Based on Depth Filtering

```
graph TD; A[深度图] --> B[多平面分割]; B --> C[欧氏距离聚类]; C --> D[反投影到RGB平面]; E[RGB图像] --> F[目标检测]; F --> G[2D矩形框]; D --> H[相似度关联]; G --> H; H --> I[中心点+3D包围框等信息];
```

Fig.6 3D Semantic Information Acquisition Based on Point Cloud Segmentation

在 SLAM 构建地图的过程中,如果有运动物体进入摄像机的视野,并产生较大的运动,这种情况会影响相机自身位姿的估计,同时运动物体的轨迹将出现在点云图中,并且在将点云图转换为其他类型时将保留信息。包含动态对象信息的地图不能直接用于导航,因为我们无法判断移动对象是否能够安全通过。如何在图像序列中寻找动态对象的区域,

The diagram illustrates the overall architecture of the proposed algorithm, organized into several main functional blocks:

- Input and Initial Processing:**
 - 单目/双目/RGBD 传感器等** (Monocular/Binocular/RGBD sensors, etc.) provide data to **特征提取** (Feature extraction).
 - 标定参数 图像数据 相关模型** (Calibration parameters, image data, correlation model) also feed into **特征提取**.
 - 特征提取** leads to **动静点检测** (Dynamic/static point detection).
- Tracking and Keyframe Management (跟踪):**
 - 动静点检测** leads to **帧间初始化位姿估计/重定位** (Inter-frame initialization pose estimation/relocation).
 - This leads to **姿态优化** (Pose optimization).
 - 姿态优化** leads to **检索新关键帧** (Retrieve new keyframes).
 - 检索新关键帧** leads to **初始关键帧** (Initial keyframe).
- Target Detection and Map Building:**
 - 初始关键帧** leads to **SSD目标检测线程** (SSD target detection thread).
 - SSD目标检测线程** leads to **目标数据库构建** (Target database construction).
 - 初始关键帧** also leads to **局部优化建图** (Local optimization map building) and **Octomap动态建图** (Octomap dynamic map building).
- Map Building and Optimization (建图):**
 - 局部优化建图** includes: **添加关键帧** (Add keyframes), **局部束束调整** (Local bundle adjustment), and **关键帧优化** (Keyframe optimization).
 - Octomap动态建图** includes: **点云分割** (Point cloud segmentation), **更新占有概率** (Update occupancy probability), and **插入octomap** (Insert octomap).
- Target Database (目标数据库):**
 - Includes: **3D目标信息** (3D target information), **目标库检索** (Target library search), and **目标库更新** (Target library update).
- Loop Closure (回环检测):**
 - Includes: **优化本质图** (Optimize essential graph), **回环检测/融合** (Loop detection/fusion), **相似变换计算** (Similarity transformation calculation), and **候选帧检测** (Candidate frame detection).

Fig.7 Semantic Map Construction Algorithms in Dynamic Environment

Octo-Map 将地图按八叉树结构进行存储, 最小的叶子结点尺寸由分辨率参数决定, 由于环境中动态物体的存在, 有时同一空间是被占有的, 一会又变为空闲状态, 所以一个叶子的占有概率应该由多次观测的联合概率决定, 我们对联合概率进行如式 1 的 *logit* 变换, 可以得到概率更新的简化方式。

上述 *logit* 变换的反变换为 *sigmoid* 函数形式, 如式 2 所示。

概率更新的简化方式如式 3 所示。

式中 $L(n | Z_{1:T-1})$ 为前面时刻占有格的 α 值, 加上最近一次的 α 值 $L(n | Z_T)$ 可以得到当前综合的 α 值, 再经过式 2 的反变换可以得到占有格的占有概率值, 其中 $L(n | Z_T)$ 由 4 式决定。

$$L(n|z_T) = \begin{cases} l_{occ}, & \text{节点}n\text{被观测到} \\ 0, & \text{节点}n\text{未被观测到} \\ l_{free}, & \text{地面等不考虑} \end{cases} \quad (4)$$

式中, l_{occ} 与 l_{free} 是预定义的值, 这里我们取 $l_{occ} = 0.85$, $l_{free} = -0.4$ 分别对应占有概率值 0.7 和 0.4, 而 0 对应的概率值为 0.5, 如果节点 n 经常被观测到, 其占有概率会上升, 那些未被观测到的以及不在考虑中的点 (点云分割获取地面以及动态点) 的占有概率会相对减小。

在目标数据库构建过程中首先根据 2.2 节和 2.3 节生成每一帧观测中存在的 3D 目标物体, 然后加入到数据库中。当下一次再检测到相同的类别时, 首先根据位置信息判断是否是同一个物体, 如果是同一个物体, 更新该物体的空间坐标等信息, 否则记录为新物体。语义目标数据库构建流程如下图 8 所示。

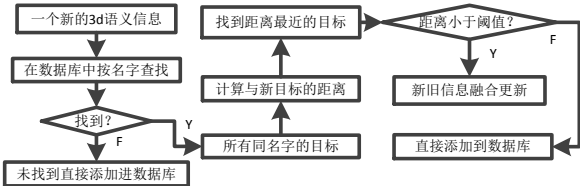


图 8 语义目标数据库构建流程

Fig.8 Semantic Object Database Construction Process

4 实验 (Experiment)

4.1 动态点检测

TUM RGB-D^[6]数据集是使用 Microsoft Kinect 传感器在不同室内场景下录制的包含 39 个序列的数据集, 包含 RGB 图像和深度图像以及由高精度运动捕捉系统得到的真实的相机轨迹。我们在一个名为 walking(w)的序列中进行实验, 其中场景是有两个人从后方走到前方并坐到椅子上。基于光流算法的动态点检测算法检测结果如图 9 所示。

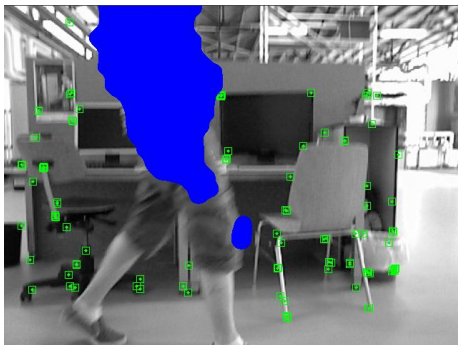


图 9 基于光流的动态点检测算法

Fig.9 Dynamic Point Detection Result Based on Optical Flow

图中标记为深蓝色的区域为检测到的动态点区域, 在提取关键点以及地图构建过程中应当剔除, 以增强系统位姿估计的鲁棒性以及地图的质量。多视角几何动静点检测算法结果如图 10 所示, 图中标记为深蓝色的区域为检测到的动态点区域, 同样在提取关键点以及地图构建过程中应当剔除。



图 10 基于多视角几何的动态点检测算法结果

Fig.10 Dynamic Point Detection Result Based on Multi-view geometry

我们将原版 orb-slam2 算法、加入光流动点检测以及多视角几何动点检测的相机定位结果进行对比, 如图 11 所示。

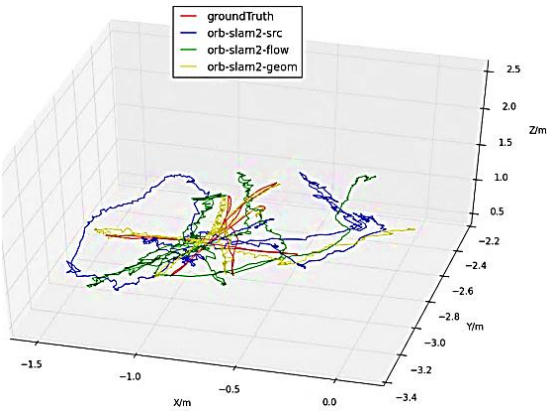


图 11 各种算法结果对比

Fig.11 Comparisons of Algorithm Results

图中, ground truth 代表的红色曲线是运动捕捉设备记录的相机真实轨迹, orb-slam2-src 代表的绿色曲线是原版 ORB-SLAM2 算法获得的相机轨迹, orb-slam2-flow 代表的蓝色曲线是使用光流算法剔除动态点的 ORB-SLAM2 算法获得到相机轨迹, orb-slam2-geom 代表的黄色曲线是使用多视角几何算法剔除动态点的 ORB-SLAM2 算法获得到相机轨迹, 算法结果误差对比分析如表 1。

表 1 算法误差分析

Tab.1 Experimental Data Analysis

算法	均方根/m	均值/m	中值/m	标准差/m	最小值/m	最大值/m
orb-slam2	0.70	0.58	0.52	0.40	0.08	1.48
flow	0.39	0.32	0.20	0.22	0.07	1.10
gem	0.15	0.07	0.04	0.13	0.003	0.89

由于真实轨迹录制时的坐标系和算法一开始的坐标系存在差异, 所以算法估计的相机轨迹和真实轨迹之间存在一个欧式变换, 所以对估计值和真实值进行配准后, 需要求解真实值和匹配的估计值之间的一个欧式变换。表 1 是对估计值进行欧式变换后再与真实值计算三维点距离差值, 对比了误差的均方根、均值、中值等信息, orb-slam2-geom 算法效果最好, 不过速度慢的多, orb-slam2-flow 算法在保持实时性的同时相对原版本 orb-slam2 精度提高

将近一倍。

4.2 三维目标信息获取

根据2.3小节3D语义信息获取的两种算法进行实验。基于深度信息滤波的方法处理结果如图 12 所示。



图 12 基于深度滤波的算法结果

Fig.12 Algorithm Results based on Depth Filtering

可以看出这种算法由于目标检测算法获取的 2D 框包含了较多其他非目标的区域,容易得到很多噪声点云,获取的 3D 框和位置信息不是太精确。第二种基于点云分割的算法,只要目标物体放置的不太紧密都能获得较好的分割结果,如图 13 所示。



图 13 基于点云分割的算法结果

Fig.13 Algorithm Results based on Point Cloud Segmentation

图中一个人坐在椅子上,与椅子有重叠,不过在与椅子靠背有间隙的情况下也能获得较好的分割结果,可以看出只要不是物体紧密接触,该方法也能获取较为精确的识别和定位精度。

4.3 动态环境下语义地图构建

在日常实验室场景下进行动态语义地图的地图构建实验,实验中有部分人在走动,同时有椅子被移动的情况发生,地图构建的结果如图 14 所示,使用基于光流法的动静点检测算法提高位姿估计精度,同时对关键图像使用目标检测算法进行 2D 目标检测,使用基于点云分割的 3D 目标检测框生成算法生成 3D 目标信息,使用 3D 目标信息构建目标数据库,同时根据点云分割信息更新 Octo-Map 地图(地面和动态点云概率降低,非地面点云概率升高,可结合目标检测信息),使用 0.8 的占有概率来更新显示 Octo-Map 地图。

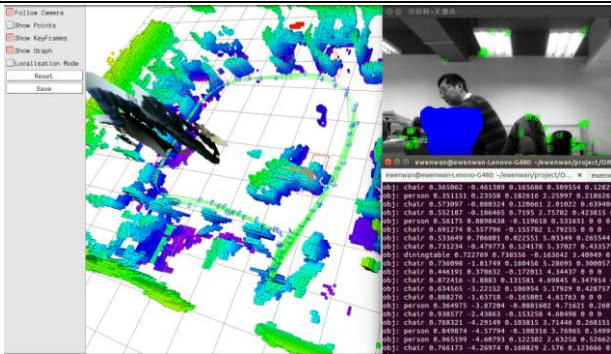


图 14 动态环境下语义地图构建结果

Fig.14 Semantic Map Construction Result in Dynamic

Environment

从建立的语义目标数据库中截取部分有代表性的值,如表 2 所示。

表 2 目标数据库结果

Tab.2 Object database results

类别	概率	X/m	Y/m	Z/m	长/m	宽/m	高/m
per	0.62	-0.31	-0.034	2.042	0.489	0.816	0.38
chair	0.37	-0.46	0.166	0.51	0.128	0.093	0.02
per	0.35	0.24	0.183	2.26	0.219	0.71	0.11
cha	0.73	-0.48	0.124	5.37	0.434	0.55	0.29
din	0.72	0.74	-0.164	3.41	0	0	0
cha	0.87	-3.89	0.132	4.7	0.348	0.38	0.39
per	0.37	-3.87	-0.089	4.72	0.26	0.3	0.3
cha	0.94	-2.44	-0.153	4.61	0	0	0
per	0.56	0.09	-0.12	0.53	0	0	0

数据库包含的信息有物体类别、类别概率值、位置坐标以及 3D 包围框尺寸,其中类别栏标注为黑体的值是不正确的信息,由于 2D 目标检测模型未使用具体场景下的数据进行训练,有部分识别错误的情况,如果直接根据识别概率来剔除错误物体,也不能达到好的效果,如识别为 dining table 的概率值有 0.72,而其他识别正确的结果概率值有部分比较小,如果仅根据一次的识别结果来剔除很容易将正确的也剔除掉,可以考虑采用采用概率融合的方式更新数据库,如果有两种不同类别的物体其中心坐标很接近,那么该点云团是一个物体,只不过目标检测算法检测错误,出现了两种物体。此时可以根据两种类别物体的概率大小进行选取,选取概率较大的类别作为该点云团的类别。其次应该剔除由于噪声导致的 3D 包围框尺寸为 0 的值。

5 结论 (Conclusion)

5.1 总结

本文的主要工作是基于高精度的 ORB-SLAM2 算法、Mobile Net-SSD 目标检测算法、动态点检测算法、3D 目标信息提取算法以及动态地图更新算法，在动态环境下实时构建语义地图。主要贡献为提出了一种简单的基于光流算法的动态点检测算法与两种 3D 目标信息获取算法。本文将 VSLAM 算法获取的精确定位信息和目标检测网络得到的语义信息有效地结合起来，实现语义信息和空间点云团之间的一一映射，构建一个有效的语义目标数据库，可以为机器人智能导航提供有效的语义地图信息，算法在普通 PC 机上就可以近似实时运行，具有较强的实际应用价值。

5.2 展望

针对算法出现的误差，未来的研究方向有：

1) 如何更有效地解决遮挡问题。本文基于点云分割的 3D 目标信息获取算法在物体放置不紧密的情况下能够获得很高的精度，但是在物体紧密放置情况下，由于欧式距离分割算法不能将物体分割开，导致分割错误。这里可以采用以及颜色凸凹性的超体聚类分割算法实现更精细的分割。

2) 如何有效的检测出动态点。在相机定位过程中，动态关键点存在会干扰相机的定位，如何在相机定位中剔除动态点是提高相机在动态环境中定位精度的关键。本文提出的基于光流算法的动点检测算法速度快，但是精度低，而且还容易产生很多误检测。基于多视角几何的算法精度高，但是速度太慢。

3) 语义数据库动态更新。实验中 2D 目标算法的检出率低，构建的语义数据库不全，同时目标检测算法会检测出错，未考虑目标检测发生错误的情况，如同一个空间中的物体，在两次检测中被识别成了不同的物体，数据库中就会有多个语义物体信息。可以根据位置信息对检测出的物体进行融合。

参考文献 (References)

- [1] Mur-Artal R, Montiel JMM, Tardos JD. ORB-SLAM: a versatile and accurate monocular SLAM system [J]. IEEE Transactions on Robotics, 2015, 31(5): 1147-1163.
- [2] Mur-Artal R, Tardos JD. ORB-SLAM2: an Open-Source SLAM system for monocular stereo and RGB-D cameras [J]. IEEE Transactions on Robotics, 2017, 33(5): 1255-1262.
- [3] Hornung A, Wurm K M, Bennewitz M, et al. OctoMap: an efficient probabilistic 3D mapping framework based on octrees[J]. Autonomous Robots, 2013, 34(3): 189-206.
- [4] Liu W, Anguelov D, Erhan D, Szegedy C, Reed S.E. SSD: single shot multibox detector[C]// European Conference on Computer Vision. Berlin , German:Springer, 2016:21-37
- [5] Bescos B, Facil J M, Civera J, et al. DynaSLAM: Tracking, Mapping, and Inpainting in Dynamic Scenes[C]// IEEE International Conference on Robotics and Automation. ICRA, 2018, 3(4): 4076-4083.
- [6] Sturm J, Engelhard N, Endres F, et al. A benchmark for the evaluation of RGB-D SLAM systems[C] // IEEE International Conference on Intelligent Robots and Systems. IROS, 2012: 573-580.
- [7] Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., et al.: Mobilenets: Efficient convolutional neural networks for mobile vision applications [EB/OL]. [2017-04-04].<https://arxiv.org/pdf/1704.04861.pdf>.
- [8] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: MobileNetV2: Inverted Residuals and Linear Bottlenecks [EB/OL]. [2018-01-04].<https://arxiv.org/pdf/1801.04381.pdf>.