

动态环境下基于语义信息与几何约束的视觉SLAM系统

李嘉铭, 解明扬, 张民, 王从庆

(南京航空航天大学自动化学院, 江苏 南京 211106)

摘要: 目前同时定位与地图构建技术多假设外部环境是静态的, 忽略了动态目标对SLAM系统的影响, 这一假设很大程度上影响了无人系统自主导航的定位精度与鲁棒性。针对这一问题, 提出了一种结合目标检测语义信息与多视图极线几何约束的动态SLAM系统, 根据运动概率来判别与剔除系统中的动态特征点, 并在TUM数据集及实机平台上进行了算法性能测试。实验结果表明在高动态环境下, 该系统相较于ORB-SLAM2在绝对轨迹误差上至少可减小94%以上, 相对位移轨迹和旋转角度误差分别至少降低了41%和40%, 表明该算法能够有效剔除动态特征点, 提高了动态环境下视觉SLAM系统的定位精度和鲁棒性。

关键词: 动态SLAM; 深度学习; 目标检测; 极线几何

中图分类号: TP242.6

文献标志码: A

doi: 10.11959/j.issn.2096-6652.202342

Visual SLAM based on semantic information and geometric constraints in dynamic environment

LI Jiaming, XIE Mingyang, ZHANG Min, WANG Congqing

College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China

Abstract: Most existing visual SLAM systems assume that the external environment is static, ignoring the influence of dynamic objects on the SLAM system. This assumption largely affects the accuracy and robustness of autonomous navigation. To address this issue, a dynamic SLAM system was proposed, which combined semantic information based on object detection and geometric information from multi-view geometry constraints by defining and discriminating the dynamic feature points in the system based on the moving probability. Experiment results on the public TUM dataset and our robot in real environment showed that, when comparing with ORB-SLAM2, the absolute trajectory error could be reduced larger than 94%, and the average relative position and attitude errors were reduced at least 41% and 40%, respectively, in high dynamic environments. It means that the proposed SLAM system effectively removes dynamic feature points, thus improving the localization accuracy and robustness of the visual SLAM system within high dynamic environments.

Key words: dynamic SLAM, deep learning, object detection, epipolar geometry

0 引言

同时定位与地图构建 (simultaneous localization

and mapping, SLAM)^[1]是指智能体通过搭载的传感器对自身定位的同时, 实现对周围环境地图的构建。而视觉SLAM^[2]得益于视觉传感器成本低廉以

收稿日期: 2023-08-14; 修回日期: 2023-10-11

通信作者: 解明扬, myxie@nuaa.edu.cn

基金项目: 启元实验室创新基金项目 (No.S20210201102); 南京航空航天大学科研与实践创新计划 (No. xcxjh20220342)

Foundation Items: The Qiyuan Lab Innovation Fund Project (No.S20210201102), Research and Practice Innovation Program of NUAU (No.xcxjh20220342)

及信息采集的丰富性,适合在复杂的室内环境下使用,获得了国内外研究者的广泛关注,涌现出许多经典的视觉SLAM算法,如ORB-SLAM系列^[3-5],RTAB-MAP^[6]和LSD-SLAM^[7]。在静态环境下,上述算法都能获得良好的定位精度,解决动态环境下SLAM技术中特征点误匹配问题,目前主要通过提前识别并剔除动态点以防止误匹配。近年来,许多学者提出了面向动态环境下基于深度学习的动态目标检测方法,并取得了一定的进展^[8]。例如,Yu等人提出了DS-SLAM^[9],通过语义分割网络Seg-Net^[10]识别出环境中的动态目标,进一步利用运动一致性检测滤除动态特征,提升定位精度。Bescos等人提出了DynaSLAM^[11]系统,使用多视图几何和Mask R-CNN^[12]实例分割网络来检测动态对象,并通过修复被动态目标遮挡的背景来构建静态地图。该作者随后又提出了DynaSLAM II^[13],在原先的基础上增加了多目标跟踪模块,拓展了系统的使用范围。Xie等人^[14]通过深度学习检测动态目标,采用K-均值方法与多视图约束相结合的方法来分割和剔除图像中的动态区域,在获得相当定位精度的同时,运行速度比DynaSLAM快了近一倍。Yuan等人^[15]将特征点划分为静态、动态、静态可移动和由动态变为静态4种状态,并根据特征点的不同状态设计了不同的特征跟踪及建图算法。Liu等人^[16]在ORB-SLAM3的基础上提出了RDS-SLAM算法,利用分割图像的掩膜为每个地图点设置和更新运动概率来判断其移动的状态,这种方法不需要等待语义线程的结束就能进行后续的位姿跟踪,提高了系统运行的实时性。以上方法均采用了基于语义分割网络的动态目标提取,能够获得像素级分割精度,但普遍存在计算量大、算法实时性差等不足。Zhong等人^[17]提出Detect-SLAM,使用单阶段目标检测网络SSD^[18]对关键帧进行动态目标检测,获得关键帧中特征点动态概率,在普通帧中则通过运动概率传递确定特征点的动态概率,该方法采用运动概率来判断特征点是否为动态点,提升了SLAM系统鲁棒性的同时,还提高了目标检测的精度。Crowd-SLAM中^[19]通过采用专门训练的YOLOv3^[20]模型检测人群目标以改善系统的定位精度。

值得注意的是,目标检测和语义分割均会导致特征点数量减少进而导致系统性能下降。近年来研究人员在特征点法的基础上引入额外特征来弥补特征点法的不足,如Zhang等人^[21]在采用语义分割判

别去除动态物体后,对图像进行了线特征的提取,并同时使用点、线特征进行跟踪定位,获得了良好的效果。Wang等人^[22]在点、线特征基础上进一步引入了面特征以及几何特征,保证了去除动态点后特征的丰富性。

为了提高室内动态场景下ORB-SLAM2系统的稳定性和鲁棒性,本文提出了一种结合语义信息与几何约束的动态视觉SLAM算法。首先,给出了改进ORB-SLAM2系统的框架;然后,阐述了基于深度学习的动态目标检测与基于极线几何的动态点判别方法,进一步介绍了基于运动概率传递的动态特征点滤除算法;最后,在TUM数据集以及实机移动机器人上进行了系统性能的实验验证。

1 动态视觉SLAM

1.1 系统结构

本文采用ORB-SLAM2的RGB-D模式作为基础框架,主要对其跟踪线程进行了改进,通过添加YOLO目标检测网络模块以及动态特征点滤除模块,减少SLAM前端中的错误数据关联;局部建图以及回环检测线程保持不变。系统整体结构框架如图1所示。

本文所提动态特征点判别方法结合了几何模型与目标检测网络,并将几何模型与目标检测的结果转换为运动概率,其中将特征点属于动态对象的概率定义为运动概率。

如图2所示,根据运动概率将特征点划分为4类状态点,SLAM系统将在去除所有概率大于0.5的动态点之后进行姿态估计。值得注意的是,由于目标检测网络的预测框(bounding box)在动态环境下存在不稳定跳变现象,为了反映实际系统运行的连续性,本文采用基于运动概率的传递机制,以确保帧间特征点运动概率不发生突变。

1.2 基于YOLOv5的动态目标检测

随着深度学习技术的不断发展,卷积神经网络被广泛应用于图像中动态目标位置的识别与检测^[23]。为了兼顾目标识别准确率与系统实时性,本文采用基于单阶段目标检测网络YOLOv5^[24]来获取动态目标的语义信息,YOLO使用单卷积神经网络来同时预测单张图像内的多个预测框以及预测框的类别置信度,并根据阈值判断预测的准确率。

本文采用YOLOv5m6版本,并使用NCNN框架进行神经网络加速及部署。该神经网络的参数量

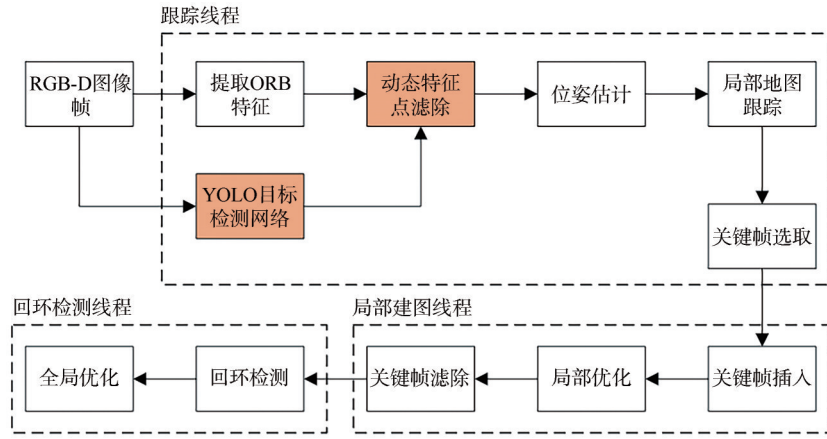


图1 系统整体结构框架

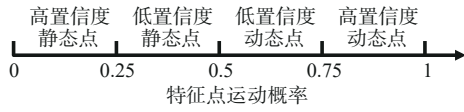


图2 以运动概率区分的特征点的4种状态

为35.7 m，在MS COCO数据集上能够获得51.3% mAP 50-95精度，通过使用NCNN框架加速，在本文所用的3 060 Laptop显卡上能够达到45 FPS的运行速度，满足系统运行的实时性要求。由于存在动态对象判别精度误差，生成的预测框往往比实际动态区域更大，使得预测框边缘往往包含静态特征点。为了识别错误分类的特征点，本文采用二项逻辑回归模型计算先验动态目标预测框内每个特征点的语义动态概率，预测框中动态概率 $P < 0.75$ 的特征点更容易被错误分类，这些特征点往往位于静态背景上，尽管它位于动态目标的预测框内。

预测框中特征点 p_i 距离预测框边缘的距离定义为：

$$\text{dist}(p_i, B) = \min_{b_j \in B} \|p_i - b_j\|_2 \quad (1)$$

其中， b_j 表示预测框 B 的各个边界。

利用二项逻辑回归模型，定义特征点 p_i 的语义动态概率为：

$$P(p_i \in d|s) = \frac{1}{\exp(-\alpha \cdot \text{dist}(p_i, B)) + 1} \quad (2)$$

其中， α 是平滑检测影响因子，在实验中设为0.1。

1.3 基于极线几何约束的动态点判别

由于YOLO是使用COCO数据集训练的，并且只能确定80类左右目标，因此仍然存在一些运动目标无法检测到的问题。针对这些问题，本文提出了在连续帧间使用极线几何约束来进一步识别动

态关键点。属于移动对象的匹配特征点不满足极线几何约束，因此，如果匹配特征点到其极线的距离大于某一阈值，则认为该匹配特征点为动态点。

如图3所示， I_1 和 I_2 表示连续两帧图像的像平面， C_{t-1} 和 C_t 表示两帧对应的相机光心， p_1 和 p_2 是特征匹配后的特征点对。假设该特征点对为静态点，则连线 $C_{t-1}p_1$ 与 $C_t p_2$ 会在三维空间中相交于 P 点，此时， C_{t-1} ， C_t ， P 三点可以确定一个平面，称为极平面； $C_{t-1}C_t$ 连线与像平面 I_1 ， I_2 的交点分别为 e_1 与 e_2 ，称为极点； $C_{t-1}C_t$ 称为基线，连线 $p_1 e_1$ 与 $p_2 e_2$ 则称为极线。

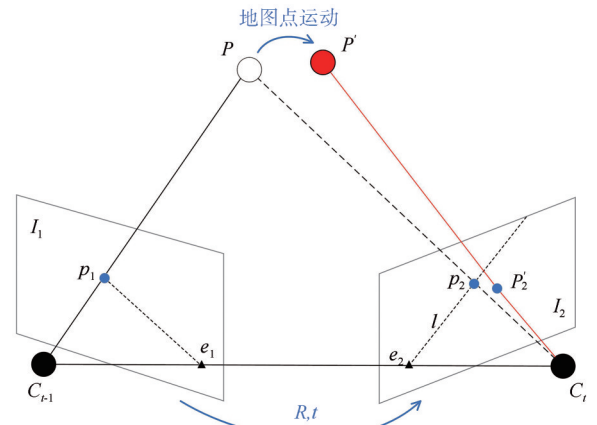


图3 极线几何约束

在 C_{t-1} 坐标系下，空间点 P 三维坐标定义为：

$$P = [X, Y, Z] \quad (3)$$

根据针孔相机模型，可以得到像平面内像素点 p_1 和 p_2 的齐次坐标：

$$\begin{cases} p_1 = [x, y, 1] = KP \\ p_2 = [x', y', 1] = K(RP + t) \end{cases} \quad (4)$$

其中, $\mathbf{K}$ 表示相机内参矩阵, $\mathbf{R}$ 表示旋转矩阵, $\mathbf{t}$ 表示平移向量。

设 x_1, x_2 为像素 p_1 点和 p_2 在归一化平面上的坐标, 即:

$$\begin{cases} x_1 = \mathbf{K}^{-1} p_1 \\ x_2 = \mathbf{K}^{-1} p_2 \end{cases} \quad (5)$$

根据式 (4) 和式 (5), 极线几何约束为:

$$p_2^T \mathbf{K}^{-T} \mathbf{t}^T \mathbf{R} \mathbf{K}^{-1} p_1 = 0 \quad (6)$$

定义如下本质矩阵 (essential matrix) $\mathbf{E}$ 和基础矩阵 (fundamental matrix) $\mathbf{F}$:

$$\mathbf{E} = \mathbf{t}^T \mathbf{R}, \mathbf{F} = \mathbf{K}^{-T} \mathbf{E} \mathbf{K}^{-1} \quad (7)$$

则极线几何约束式 (6) 可进一步转写成:

$$x_2^T \mathbf{E} x_1 = p_2^T \mathbf{F} p_1 = 0 \quad (8)$$

式 (8) 表明, 若某一像素特征点为另一像素特征点的正确匹配, 其必在其图像平面的极线上, 从而可估算出相对准确的帧间基本矩阵。然而估算出的基本矩阵往往存在误差, 导致实际计算中特征点不能落在所计算的极线上, 因此, 本文采用每个特征点到各自极线的距离来衡量匹配点对的可靠性。

根据基本矩阵 $\mathbf{F}$ 和 p_1 , 可以计算像平面 I_2 上的极线 l 方程为:

$$l = \begin{bmatrix} \mathbf{A} \\ \mathbf{B} \\ \mathbf{C} \end{bmatrix} = \mathbf{F} p_1 = \mathbf{F} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (9)$$

式 (9) 中, $\mathbf{A}, \mathbf{B}, \mathbf{C}$ 代表极线方程向量。 P' 在像平面 I_2 内的投影点 p_2' 与极线 l 的距离为:

$$\text{dist}(p, l) = \frac{|p_2'^T \mathbf{F} p_1|}{\sqrt{\|\mathbf{A}\|^2 + \|\mathbf{B}\|^2}} \quad (10)$$

当该距离过大时, 将该特征点判断为动态点, 并将其去除。具体过程为: 首先, 通过计算当前帧中的光流金字塔来获得匹配的关键点, 若匹配点离图像边缘太近或以匹配点为中心的 3×3 图像区域像素差太大, 则去除匹配点对; 其次, 使用随机采样一致性 (random sample consensus, RANSAC) 得到基本矩阵 $\mathbf{F}$; 最后, 计算匹配的特征点与极线的距离, 如果距离大于阈值 ε ($\varepsilon=0.75$), 则认为特征点为外点。与第 1.2 节中一样, 使用二项逻辑回归模型估算特征点 p_i 的几何动态概率为:

$$P(p_i \in d|g) = \frac{1}{\exp(-\beta \cdot \text{dist}(p_i, l)) + 1} \quad (11)$$

其中, β 为平滑检测结果影响因子, 实验中设为 1。

1.4 基于运动概率传递的动态点去除算法

传统基于几何或语义的视觉里程计只在一帧中消除动态点, 忽视了动态目标运动的连续性。本文研究了面向连续帧的视觉里程计的动态点剔除问题, 提出了基于运动概率传递的动态点剔除算法, 该算法结合了几何模型与深度学习技术, 利用过去与现在的信息来评估和消除动态关键点。

算法 1. 基于运动概率传递的动态点去除

输入: 当前帧 I_t 特征点集合 $L_t = \{p_1, \dots, p_n\}$ 及上一帧 I_{t-1} 特征点集合

$L_{t-1} = \{p'_1, \dots, p'_m\}$ 。

输出: 判别的动态点集合 U 。

```

1: for each  $p_i$  in  $L_t$  do
2:   if  $p_i$  在 YOLO 提取预测框内 do
3:     计算  $P(p_i \in d|s)$ ;
4:     if  $P(p_i \in d|s) > 0.75$  do
5:        $N_s++$ ;
6:     end if
7:   end if
8:   计算  $\text{dist}(p_i, l)$  和  $P(p_i \in d|g)$ ;
9:   if  $P(p_i \in d|g) > 0.75$  do
10:     $N_g++$ ;
11:   end if
12:   计算  $p_i$  的运动概率  $P_t(p_i \in d)$ ;
13:   if  $P_t(p_i \in d) > 0.5$  do
14:     将  $p_i$  加入  $U$ ;
15:   end if
16: end for

```

首先, 根据特征匹配找到上一帧中与当前帧特征点集合 L_t 相匹配的特征点集合 L_{t-1} , 其中 $n \geq m$ 。对于当前帧特征点 p_i , 如果其在 YOLO 检测到的动态目标预测框内, 则计算其语义动态概率。其次, 根据 L_t 和 L_{t-1} 计算基本矩阵 $\mathbf{F}$, 并计算特征点 p_i 的几何动态概率。

根据语义动态概率以及几何动态概率, 定义联合概率为:

$$P_t(c) = \omega P_t(p_i \in d|s) + (1 - \omega) P_t(p_i \in d|g) \quad (12)$$

其中, ω 由式 (13) 获得:

$$\omega = \frac{N_s}{N_s + N_g} \quad (13)$$

其中, N_s 表示当前帧由目标检测网络判别的语义外点数, N_g 表示由极线几何约束判别的几何外点数。

最后, 基于运动概率传递的当前帧 I_t 中特征点 p_i 为动态点的概率为:

$$P_t(p_i \in d) = \lambda P_t(c) + (1 - \lambda) P_{t-1}(p_i \in d) \quad (14)$$

其中, P_{t-1} 表示上一帧中特征点 p_i 为动态点的概

率,称为先验动态概率,计算式如下:

$$P_{t-1}(p_i \in d) = \begin{cases} P_{t-1}(p'_i \in d), i \leq n \\ 0.5, \text{否则} \end{cases} \quad (15)$$

如果特征点 p_i 在上一帧中存在相匹配的特征点 p'_i ,即 $i \leq n$ 时,有 $P_{t-1}(p_i \in d) = P_{t-1}(p'_i \in d)$ 。如果 p_i 是当前帧新提取的特征点时,有 $P_{t-1}(p_i \in d) = 0.5$ 。 λ 是平滑当前帧检测结果影响因子,值越高,表示对当前帧检测结果越敏感,值越低,表示多视图的连续性越强。在实验中,为了权衡动态点判别方法的准确性与动态目标在多视图中运动的连续性,将 λ 设置为0.5。在获得基于运动概率传递的动态点概率之后,根据图2来区分动态点状态,将概率大于0.5的动态点剔除,仅使用剩余的静态特征点进行位姿跟踪。

2 实验结果及分析

本文主要采用TUM数据集^[25]进行算法性能评估,在TUM的walking序列中,两个人在场景中来回走动,场景变化较大,因此本文选用该序列进行高动态运动性能测试,并在搭载Intel Realsense D435深度相机的Turtlebot3移动机器人平台上进行了真实环境实验。本文测试实验是在一台处理器为AMD Ryzen7 5 800H@3.20 GHz,显卡为RTX 3 060 Laptop,内存为16 GB,操作系统为ubuntu 18.04的笔记本计算机上进行的,所有测试均进行了5次,结果取平均值。

2.1 动态特征点剔除实验

采用walking_xyz序列RGB-D图像作为实验输入,结果如图4所示,其中,图4(a)为不使用动态点剔除方法时的ORB特征提取结果,可以看出在左侧坐下与右侧站立的人身上都提取了很多特征点;图4(b)为YOLOv5目标检测结果,用来检测潜在的动态目标(在室内环境中主要是人);图4(c)为

采用本文所提动态点剔除方法后的ORB特征点提取结果,可以看到,左侧坐下的人由于没有运动,在其身上提取了ORB特征,而右侧站立的行人由于在前后帧中进行了移动,因此没有在其身上提取ORB特征。

为了验证本文所提算法在实际环境中的效果,设计了3组基于Turtlebot3移动机器人的实验,如图5所示。实验一为:移动机器人保持不动,单人在相机前多次往返实验,动态点去除效果如图5(a)所示,位姿估计结果如图5(d)所示。实验二为:移动机器人在地面上匀速直线运动,同时单人在移动机器人面前多次往返走动,以模拟正常工作环境。动态点去除效果如图5(b)所示,位姿估计结果如图5(e)所示。实验三为:移动机器人在桌面上匀速直线运动,同时多人在移动机器人面前多次往返运动,以模拟复杂工作环境。动态点去除效果如图5(c)所示,位姿估计结果如图5(f)所示。

实验结果表明,本文所提动态点剔除方法能够分辨具体动态的目标,在保证去除动态特征点的前提下,保留了潜在动态目标上的静态特征点,解决了因特征点过少影响定位精度的问题。

2.2 改进SLAM系统性能实验

为了验证本文SLAM系统性能,首先,将本文所提系统(Our System)与ORB-SLAM2、只使用目标检测进行动态点剔除的系统(YOLO-Only)进行了性能对比。之后,将本文所提系统与现有知名动态SLAM系统在TUM数据集上进行性能对比。

图6为绝对轨迹误差可视化效果,可以看出,高动态环境下ORB-SLAM2出现了严重的漂移现象,不能很好地进行位姿跟踪;只使用目标检测进行动态点剔除在一定程度上减少了位姿漂移,但估计值与真实值有较大偏差;基于本文所提动态点去

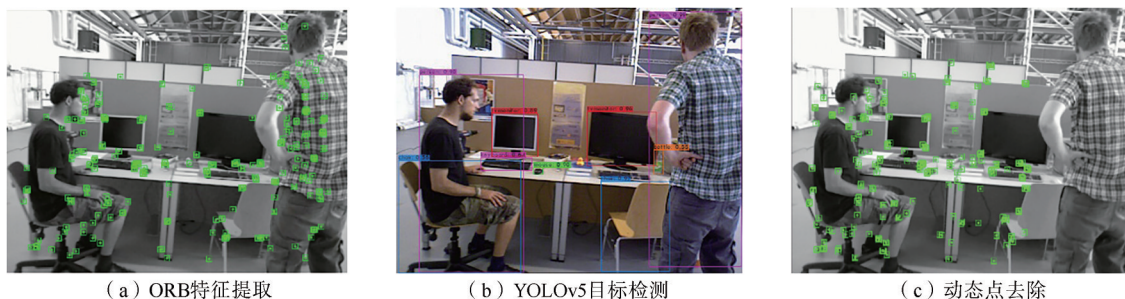


图4 TUM数据集动态特征点去除结果

除方法的 SLAM 系统的绝对轨迹误差较小, 能够很好地估计高动态环境下的相机位姿。

表1~表3分别为绝对轨迹误差、相对位移轨迹误差和相对旋转角度误差的实验结果, 采用均

方根误差 (root mean square error, RMSE) 和标准差 (standard deviation, STD) 分别表征系统的定位精度与稳定性。表1~3中的提升值 Improvement 计算方式为:

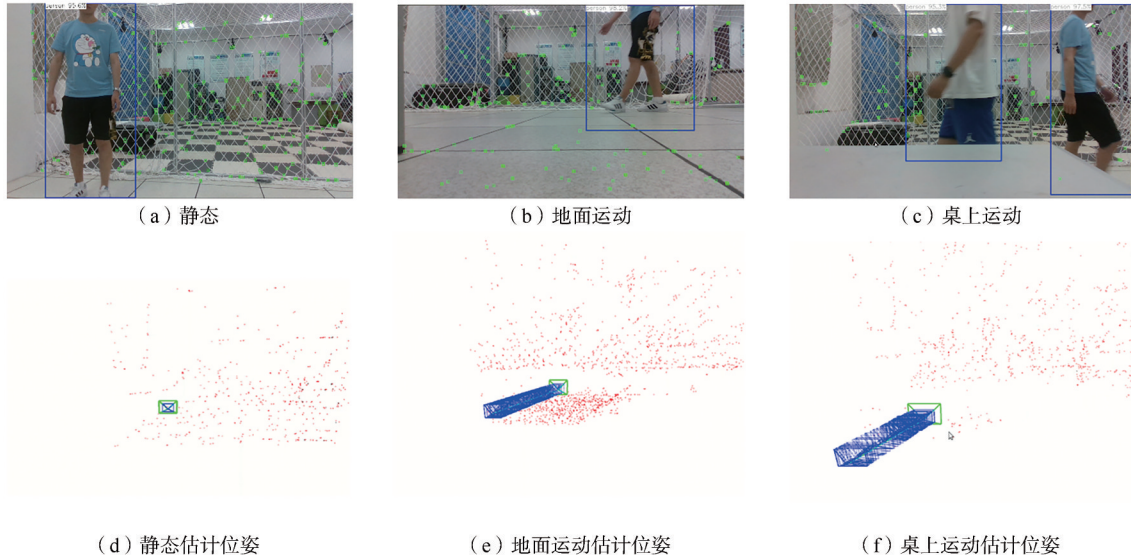


图5 实机环境下动态点去除算法效果

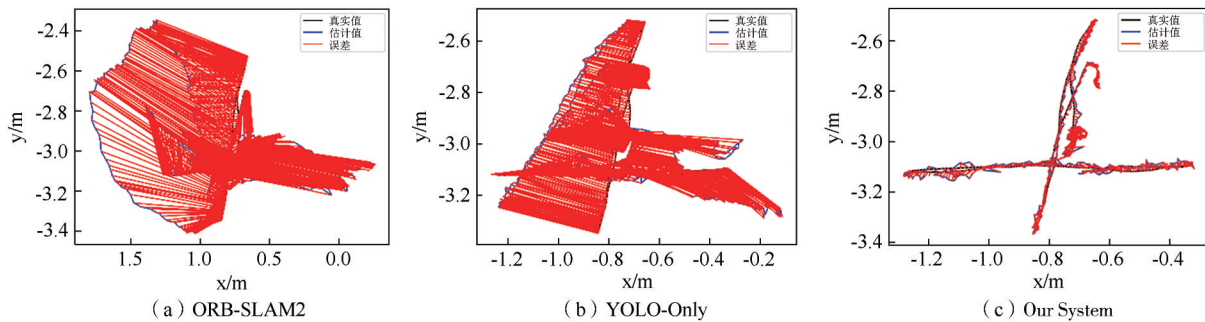


图6 绝对轨迹误差可视化效果 (walking_xyz序列)

表1 绝对轨迹误差 (ATE) 实验结果对比

序列	ORB-SLAM2/m		YOLO-Only/m		Our System/m		Improvement(O)/%		Improvement(Y)/%	
	RMSE	STD	RMSE	STD	RMSE	STD	RMSE	STD	RMSE	STD
walk_half	0.5763	0.2952	0.1304	0.0267	0.0305	0.0169	94.71	94.28	76.61	36.70
walk_xyz	0.7405	0.3623	0.2584	0.0874	0.0163	0.0079	97.80	97.82	93.69	90.96
walk_static	0.3779	0.1537	0.0082	0.0036	0.0069	0.0028	97.99	97.59	7.32	22.22
walk_rpy	0.7021	0.3425	0.2013	0.1378	0.0213	0.0166	96.97	95.15	89.42	87.95

表2 相对位移轨迹误差 (RPE) 实验结果对比

序列	ORB-SLAM2/m		YOLO-Only/m		Our System/m		Improvement(O)/%		Improvement(Y)/%	
	RMSE	STD	RMSE	STD	RMSE	STD	RMSE	STD	RMSE	STD
walk_half	0.0280	0.0190	0.0183	0.0117	0.0163	0.0101	41.78	46.84	10.93	13.68
walk_xyz	0.0438	0.0344	0.0150	0.0093	0.0120	0.0076	72.60	77.91	20.00	18.28
walk_static	0.0184	0.0146	0.0063	0.0035	0.0057	0.0031	69.02	78.77	9.52	11.43
walk_rpy	0.0416	0.0339	0.0303	0.0208	0.0216	0.0144	48.08	57.52	28.71	30.77

表3 相对旋转角度误差(RPE)实验结果对比

序列	ORB-SLAM2/度		YOLO-Only/度		Our System/度		Improvement(O)/%		Improvement(Y)/%	
	RMSE	STD	RMSE	STD	RMSE	STD	RMSE	STD	RMSE	STD
walk_half	0.6849	0.4271	0.4740	0.2689	0.4072	0.2064	40.55	51.67	14.09	23.24
walk_xyz	0.9306	0.7095	0.4245	0.2942	0.3927	0.2733	57.80	61.48	7.49	7.10
walk_static	0.3863	0.2827	0.1793	0.0943	0.1735	0.0873	55.09	69.12	3.23	7.42
walk_rpy	0.9246	0.7079	0.6277	0.3828	0.4839	0.2900	47.66	59.03	22.91	24.24

$$\text{Improvement} = \left(1 - \frac{\tau}{\Gamma}\right) \times 100\% \quad (16)$$

其中, τ 表示本文所提系统的实验测量值, Γ 表示比较系统的实验测量值。分别比较了与 ORB-SLAM2 系统的精度提升 Improvement(O) 以及只使用 YOLO 的系统的精度提升 Improvement(Y)。

从表1可以看出, 本文所提系统相较于标准 ORB-SLAM2 系统定位精度明显提升, 至少提升 94% 以上; 相较于 YOLO-Only 方法, 在除 walk_static 序列外均有较大的精度提升, 在序列 walk_static 中, 由于相机基本没有运动, 因此特征点数量对系统定位精度的影响较小, YOLO-Only 的实验结果与本文系统相近。表2和表3分别为相对位移轨迹误差和相对旋转角度误差实验结果, 可以看出, 相较于 ORB-SLAM2, 本文所提系统在平移误差以及角度误差性能上均有较大提升, 其中, 相对位移轨迹误差至少有 42% 的性能提升, 相对旋转角度误差至少有 40% 的平均精度提升。相较于只使用目标检测方法, 本文所提系统也有平均 15% 的性能提升。

表4为本文所提系统与现有动态 SLAM 系统在 TUM 数据集上的性能对比, 所有数据均来自源论

文。表5为各动态 SLAM 系统所用硬件条件与算法运行效率。由表4和表5可以看出, 相较于同样使用目标检测网络的 Detect-SLAM, 本文所提系统在位姿估计的精度以及算法运行效率两方面都有较大提升; 与基于语义分割的动态 SLAM 系统相比, 本文系统的位姿估计精度领先于使用 SegNet 的 DS-SLAM 和 RDS-SLAM。

3 结束语

本文提出了一种结合目标检测语义信息与极线几何约束信息的动态视觉 SLAM 系统, 通过使用运动概率传递, 同时考虑了 SLAM 系统对动态目标判别的准确性以及实际系统运行环境的连续性。在 TUM 数据集以及实验室实机环境进行了算法性能对比实验验证, 相较于 ORB-SLAM2 算法, 本文所提改进的动态 SLAM 系统的绝对轨迹误差降低了 94% 以上, 相对位移轨迹误差降低了 41% 以上; 相较于只使用目标检测, 本文所提系统也有较大程度的性能提升。此外, 本文同现有主流动态 SLAM 系统进行了性能对比, 结果显示, 本文所提系统较好地平衡了定位精度以及运行效率。最后, 将本文所提系统部署在实际机器人系统中并进行了测试,

表4 动态 SLAM 系统绝对轨迹误差对比 (单位: m)

序列	DS-SLAM		RDS-SLAM		Detect-SLAM		Our System	
	RMSE	STD	RMSE	STD	RMSE	STD	RMSE	STD
walk_half	0.0303	0.0159	0.0807	0.0454	0.0514	—	0.0305	0.0169
walk_xyz	0.0247	0.0161	0.0571	0.0229	0.0241	—	0.0163	0.0079
walk_static	0.0081	0.0036	0.0206	0.0120	—	—	0.0069	0.0028
walk_rpy	0.4442	0.2350	0.1604	0.0873	0.2959	—	0.0213	0.0166

表5 平均计算时间对比

系统	平均每帧处理时间/ms	选用神经网络	硬件平台
ORB-SLAM2	23	—	AMD Ryzen7 5 800H CPU
DS-SLAM	76.5	SegNet	Nvidia Quadro P4 000
RDS-SLAM	50-65	SegNet	Nvidia RTX 2 080Ti
Detect-SLAM	>340	SSD	Nvidia GTX 960M
Our System	62	YOLOv5m6	Nvidia GTX 3 060Laptop

实验结果表明, 本文所提动态 SLAM 系统在高动态环境下具有良好的定位精度以及鲁棒性。下一步的研究工作将聚焦在改进特征点提取方法上, 提高在动态目标占据大规模视野情况下系统的稳定性。

参考文献:

- [1] SMITH R C, CHEESEMAN P. On the representation and estimation of spatial uncertainty[J]. The International Journal of Robotics Research, 1986, 5(4): 56-68.
- [2] 王霞, 左一凡. 视觉 SLAM 研究进展[J]. 智能系统学报, 2020, 15(5): 825-834.
WANG X, ZUO Y F. Advances in visual SLAM research[J]. CAAI Transactions on Intelligent Systems, 2020, 15(5): 825-834.
- [3] MUR-ARTAL R, MONTIEL J M M, TARDÓS J D. ORB-SLAM: a versatile and accurate monocular SLAM system[J]. IEEE Transactions on Robotics, 2015, 31(5): 1147-1163.
- [4] MUR-ARTAL R, TARDÓS J D. ORB-SLAM2: an open-source SLAM system for monocular, stereo, and RGB-D cameras[J]. IEEE Transactions on Robotics, 2017, 33(5): 1255-1262.
- [5] CAMPOS C, ELVIRA R, RODRÍGUEZ J J G, et al. ORB-SLAM3: an accurate open-source library for visual, visual-inertial, and multi-map SLAM[J]. IEEE Transactions on Robotics, 2021, 37(6): 1874-1890.
- [6] LABBÉ M, MICHAUD F. Online global loop closure detection for large-scale multi-session graph-based SLAM[C]//Proceedings of 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. Piscataway: IEEE Press, 2014: 2661-2666.
- [7] CARUSO D, ENGEL J, CREMERS D. Large-scale direct SLAM for omnidirectional cameras[C]//Proceedings of 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Piscataway: IEEE Press, 2015: 141-148.
- [8] 郑南宁. 人工智能新时代[J]. 智能科学与技术学报, 2019, 1(1): 1-3.
ZHENG N N. The new era of artificial intelligence[J]. Chinese Journal of Intelligent Science and Technology, 2019, 1(1): 1-3.
- [9] YU C, LIU Z X, LIU X J, et al. DS-SLAM: a semantic visual SLAM towards dynamic environments[C]//Proceedings of 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Piscataway: IEEE Press, 2019: 1168-1174.
- [10] BADRINARAYANAN V, KENDALL A, CIPOLLA R. SegNet: a deep convolutional encoder-decoder architecture for image segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(12): 2481-2495.
- [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 M, GKIOXARI G, DOLLÁR P, et al. Mask R-CNN[C]//Proceedings of 2017 IEEE International Conference on Computer Vision (ICCV). Piscataway: IEEE Press, 2017: 2980-2988.
- [13] BESCOS B, CAMPOS C, TARDÓS J D, et al. DynaSLAM II: tightly-coupled multi-object tracking and SLAM[J]. IEEE Robotics and Automation Letters, 2021, 6(3): 5191-5198.
- [14] JI T T, WANG C, XIE L H. Towards real-time semantic RGB-D SLAM in dynamic environments[C]//Proceedings of 2021 IEEE International Conference on Robotics and Automation (ICRA). Piscataway: IEEE Press, 2021: 11175-11181.
- [15] YUAN X, CHEN S. SaD-SLAM: a visual SLAM based on semantic and depth information[C]//Proceedings of 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Piscataway: IEEE Press, 2020: 4930-4935.
- [16] LIU Y B, MIURA J. RDS-SLAM: real-time dynamic SLAM using semantic segmentation methods[J]. IEEE Access, 2021, 9: 23772-23785.
- [17] ZHONG F W, WANG S, ZHANG Z Q, et al. Detect-SLAM: making object detection and SLAM mutually beneficial[C]//Proceedings of 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). Piscataway: IEEE Press, 2018: 1001-1010.
- [18] LIU W, ANGUELOV D, ERHAN D, et al. SSD: single shot MultiBox detector[M]//Computer Vision-ECCV 2016. Cham: Springer International Publishing, 2016: 21-37.
- [19] SOARES J C V, GATTASS M, MEGGIOLARO M A. Crowd-SLAM: visual SLAM towards crowded environments using object detection[J]. Journal of Intelligent & Robotic Systems, 2021, 102(2): 1-16.
- [20] REDMON J, FARHADI A. YOLOv3: an incremental improvement[EB]. 2018, arXiv preprint, arXiv:1804.02767.
- [21] ZHANG C Y, HUANG T, ZHANG R C, et al. PLD-SLAM: a new RGB-D SLAM method with point and line features for indoor dynamic scene[J]. ISPRS International Journal of Geo-Information, 2021, 10(3): 163.
- [22] WANG Y N, XU K, TIAN Y B, et al. DRG-SLAM: a semantic RGB-D SLAM using geometric features for indoor dynamic scene[C]//Proceedings of 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Piscataway: IEEE Press, 2022: 1352-1359.
- [23] LECUN Y, BENGIO Y, HINTON G. Deep learning[J]. Nature, 2015, 521(7553): 436-444.
- [24] JOCHER G, CHAURASIA A, STOKEN A, et al. Ultralytics/YOLOv5: v7.0-YOLOv5 sota realtime instance segmentation[Z]. Zenodo, 2022.
- [25] STURM J, ENGELHARD N, ENDRES F, et al. A benchmark for the evaluation of RGB-D SLAM systems[C]//Proceedings of 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems. Piscataway: IEEE Press, 2012: 573-580.

[作者简介]



李嘉铭 (1999-), 男, 南京航空航天大学自动化学院硕士研究生, 主要研究方向为移动机器人自主导航。



解明扬 (1988-), 男, 博士, 南京航空航天大学自动化学院副研究员, 主要研究方向为智能机器人技术。



张民（1973- ），男，博士，南京航空航天大学自动化学院副研究员，主要研究方向为多无人机/编队协同控制。



王从庆（1960- ），男，博士，南京航空航天大学自动化学院教授，主要研究方向为智能机器人与智能控制。