基于深度学习的动态场景视觉SLAM算法

王晓栋[1],陈援峰[1],杨伟高[2]
(1.广州城市职业学院 智能制造学院,广东 广州 511300;
2.广州城市职业学院 设备处,广东 广州 511300)

摘　要: 同时定位与地图构建(SLAM)技术是无人机或机器人在未知复杂环境中进行自主探索的关键研究方向。借助该技术,机器人能够通过其传感器获取的数据,实时计算自身的位姿,同时构建外部环境的高精度地图。基于这些信息,机器人不仅能够制定路径规划,还可以动态修正位姿误差,从而显著提升在未知环境中导航的准确性和稳定性。在使用视觉传感器的SLAM系统中,位姿解算通常依赖于几何算法和特征匹配技术。这些方法通常假设外部环境由静止的物体构成,即基于场景静态化的前提。然而,在实际应用中,动态物体如行人和车辆经常出现,这对系统的性能和鲁棒性提出了严峻的挑战。因此,引入深度学习技术与视觉SLAM算法相结合,在现有的ORB-SLAM2算法框架上新增一个动态目标检测线程,用于识别动态物体,并在里程计的计算中剔除动态点,以减少动态目标对系统定位精度的影响。实验结果表明,该方法能够显著降低绝对轨迹误差,提升SLAM算法在动态场景下的适用性和稳定性。

关键词: 视觉SLAM;动态场景;深度学习;目标检测
中图分类号: TP391.4　　**文献标志码:** A　　**文章编号:** 1674-0408(2025)03-0095-06

SLAM问题的起源可以追溯到20世纪80年代。这一时期,概率统计方法首次被引入机器人和人工智能领域,使两个领域的研究逐渐交叉融合。大量研究人员开始探索如何将估计理论应用于定位与地图构建的问题中,从而推动了SLAM技术的初步发展。其中,扩展卡尔曼滤波(EKF)是最早被应用于SLAM的算法之一,而粒子滤波则作为一种专门为SLAM设计的创新技术,具有重要意义。在2000年,Murphy等人[1]首次将Rao-Blackwellized粒子滤波(RBPF)应用于SLAM算法当中,为该领域的发展奠定了基础。2007年,基于扩展卡尔曼滤波算法(Extended Kalman Filter,EKF)的MonoSLAM[2]被提出,作为首个实现实时效果的单目视觉SLAM系统,成为滤波SLAM算法中的经典之作。MonoSLAM的核心在于为每一帧图像建立状态变量,通过构建状态方程,并结合运动模型与相机观测,利用卡尔曼滤波(KF)基于上一时刻的状态推算当前状态的最优估计。同年,Klein[3]等人提出了并行跟踪和映射(Parallel Tracking and Mapping,PTAM)。该方法另辟蹊径,采用非线性优化理论而非当时主流的滤波方案,首次引入关键帧的概念。同时,PTAM将跟踪定位和建图拆分为两个独立线程进行并行优化,显著降低了计算成本,为后续视觉SLAM的发展开创了全新的方向。2015年,西班牙科研团队推出了基于ORB特征点的ORB-SLAM[4]算法。该算法将SLAM系统分为三个线程:跟踪定位、局部地图构建与优化,以及回环检测与全局优化。通过提取ORB特征点进行匹配,该算法具备计算速度快、旋转不变性强以及鲁棒性高的优势。在优化阶段,ORB-SLAM利用图优化模型实现局部与全局位姿的最优解,显著提升了系统的精度和整体性能。随着相机传感器

收稿日期: 2025-04-18
基金项目: 广东省普通高校重点领域专项(高端装备制造)"基于视觉引导的宏微操作机器人设计"(编号:2024ZDZX3094);广东省"数字化智能工厂场景应用产教融合创新平台"(编号:2023CJPT011);广州市"智能制造开放型区域产教融合实践中心"(编号:2024QTGG014);广州市"工业机器人技术专业群"(编号:2023GSPZYQ002)。
作者简介: 王晓栋,男,广州城市职业学院智能制造学院讲师。

技术的进步，具备更强信息采集能力的双目相机和RGB-D相机逐渐被广泛应用于各类视觉SLAM系统中。2017年，Mur-Artal[5]等人在原有系统的基础上进行了扩展，增加了对双目相机和RGB-D相机的支持，优化了多线程架构，提出了ORB-SLAM2，该系统在鲁棒性和系统结构的完善性方面均有显著提升。这些改进使ORB-SLAM2成为学术界通用且高效的视觉SLAM算法之一，并进一步促进了视觉SLAM技术的发展。同一课题组的Campos[6]等人进一步提出了ORB-SLAM3。该系统融合了惯性测量单元（Inertial Measurement Unit, IMU）传感器，增加了对鱼眼相机的支持，并引入了多地图机制，在各大数据集上表现出更优异的性能。此外，ORB-SLAM系列完全开源，其完善的系统架构和高质量的代码设计将基于特征点的视觉SLAM方案推向了新的高度，成为现代SLAM系统中的里程碑之作。

在现实生活或工业生产环境中，动态场景占据了大量比例。然而，目前多数视觉SLAM系统仍基于应用场景为静态的假设进行设计，这对其在实际动态环境中的适用性构成了限制。近几年来，研究人员[7-12]利用目标检测网络识别环境中的动态目标（如人和车辆）的区域，并对这些区域中的特征点进行过滤，从而提高位姿估计的精度。同时，为了减少SSD网络带来的系统延迟，通过传播关键点的运动概率，无需对每一帧都进行检测，从而提升了系统的实时性。

一、基于Yolov5的动态对象检测

现有的动态SLAM系统通常依赖于复杂的语义分割或光流分析技术，例如Mask R-CNN和稠密光流。这些方法虽然在检测精度上表现优异，但由于计算复杂度较高，难以满足实际场景对实时性的要求。因此，本文在ORB-SLAM2的框架上引入轻量级目标检测网络YOLO[12]来进行动态对象检测。

（一）ORB-SLAM2系统架构

ORB-SLAM是一种以特征点为基础的实时单目SLAM系统，能够在大规模或小规模的场景中运行，无论是室内还是室外环境均表现出色。该系统对快速或剧烈的运动具有很高的鲁棒性，同时支持宽基线条件下的闭环检测和重定位，并能够实现全自动化的初始化过程。ORB-SLAM系统包含了所有SLAM系统的核心模块，包括跟踪（Tracking）、局部建图（LocalMapping）及回环检测（Loop Closing），从而为精确的地图构建和定位提供了强有力的技术支持。

（二）YOLOv5网络结构

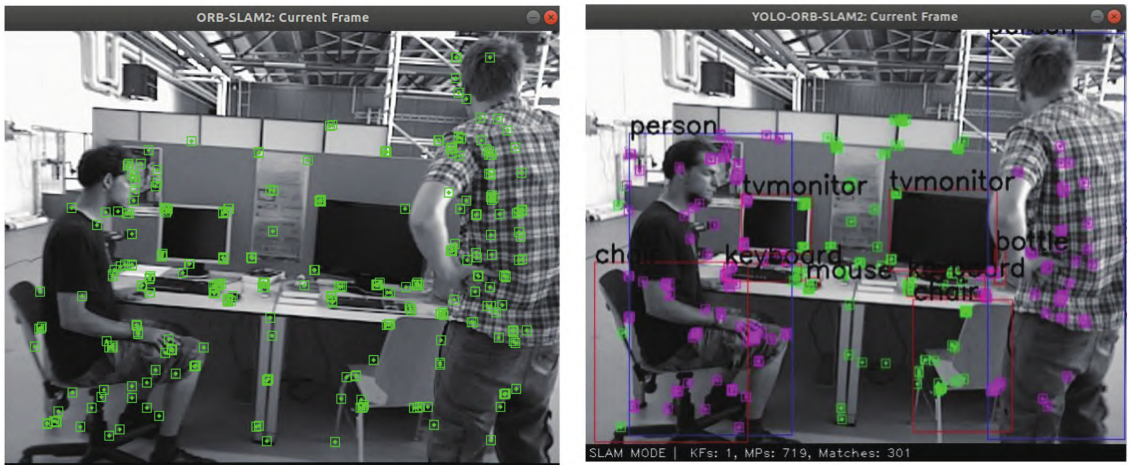
YOLOv5是一种快速且高效的目标检测算法，特别适用于对实时性要求较高的动态SLAM任务。相较于Mask R-CNN[9]，它采用了更轻量的网络结构，并在性能上进行了优化，提升了检测精度和推理速度。鉴于SLAM系统对实时性的需求以及后续在移动平台上的部署要求，本文选择了YOLOv5 s作为前端的动态对象检测网络。

（三）YOLOv5部署与线程构建

YOLOv5是一种高效的目标检测算法，其开发基于Python语言，并依赖于PyTorch这一功能强大的深度学习框架进行构建和训练，具有良好的可扩展性和灵活性。而ORB-SLAM2作为一种成熟的视觉SLAM算法，则采用C++语言编写，强调运行效率和系统的实时性。由于两者在开发语言和运行环境方面存在差异，为了实现YOLOv5在以ORB-SLAM2为核心的系统中的集成与协同工作，必须解决其跨平台部署问题。以确保目标检测与SLAM模块之间的数据交互高效且稳定。在系统实现的初始阶段，首先需要加载已经在COCO数据集上完成训练的YOLOv5 s模型。在完成YOLOv5 s模型的加载后，接下来可以借助YOLO官方提供的转换工具和接口函数，将原始的PyTorch格式模型转换为TorchScript格式。通过这种方式，模型在部署环节中能够表现出更高的兼容性和移植性，特别适用于边缘设备或嵌入式系统中的实时目标检测任务。

首先定义一个动态对象类别，包括行人、猫、狗、车等，然后创建一个专门用于YOLO目标检测的线程，对SLAM系统输入的图像数据进行处理，并将检测结果实时可视化显示在窗口中，从而直观地呈现处理效果。使用训练好的yolov5 s.torchscript.pt模型文件来对每一帧输入图像进行动态对象推理，效果如图1所示：

从图像结果中可以清晰地观察到，在原始的ORB-SLAM2系统中，通过特征提取所得到的所有关键点以绿色标记，这些特征点原本未经过任何动态性筛选，因而包含了来自静态背景和动态目标的混合信息。然而，在系统中引入YOLOv5 s模型并进行图像推理后，动态物体被准确识别，并由检测框框选出来。在这些检测框内部的特征点随后被标记为紫色，用以区分它们来源于被识别出



(a)ORB-SLAM2原有效果　　(b)加入YOLOv5s进行动态对象推理

图1　图像检测效果

的动态区域。这些紫色特征点代表了潜在的不稳定因素,若直接用于SLAM建图与定位,可能会导致系统精度下降。为此,在ORB-SLAM2的特征提取模块中,系统会主动将这些动态特征点从候选特征集合中剔除,仅保留那些处于静态背景区域、仍以绿色显示的特征点用于后续的姿态估计与地图构建。该策略有效提升了系统在动态场景下的鲁棒性和准确性。

二、运动一致性检测

在引入YOLOv5模型进行动态目标检测之后,图像中剔除动态对象后的剩余特征点会被保留。为了进一步提高特征点的匹配精度,对这些保留的特征点匹配进行运动一致性检测。这一过程能够有效排除由于误匹配或其他干扰因素引起的错误特征点,从而确保匹配的准确性,为后续的位姿估计和地图构建提供更加可靠的输入数据。

使用对极几何约束来对保留的特征点进行运动一致性检测,对极几何模型如图2所示,在该模型中,O_1和O_2表示相机的光心,P_1和P_2分别表示相机在前一帧与当前帧中观测到的特征点。虚线L_1和L_2表示两帧图像的极线。特征点P_1和P_2的齐次坐标表示为:

$$P_1=[x_1,y_1,1]、P_2=[x_2,y_2,1] \tag{1}$$

其中,x,y分别表示特征点在图像像素坐标系中的坐标。通过基础矩阵F可以计算出当前帧中的极线L_2。基本矩阵F描述了两帧图像之间的几何约束关系,通常由相机的内参、外参以及两帧之间的相对运动计算得到。极线L_2的表达式为:

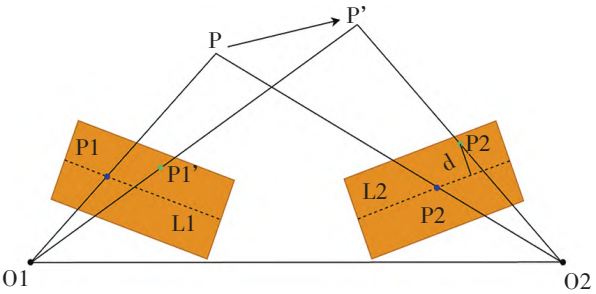


图2　对极几何

$$L_2=\begin{bmatrix}X\\Y\\Z\end{bmatrix}=FP_1=F\begin{bmatrix}x_1\\y_1\\1\end{bmatrix} \tag{2}$$

根据特征点P_2与极线L_2的几何关系,可以判断特征点的状态。当特征点P_2不位于极线L_2上时,空间点P可能是动态特征点,但同时也可能由于系统误差导致静态点未满足约束关系。因此,需计算点到直线之间的距离d,其公式为:

$$d=\frac{P_2^T l_2}{\sqrt{X^2+Y^2}} \tag{3}$$

如果计算得到的距离d大于通过实验和统计分析确定的阈值d_α,则可以将这些特征点判定为动态特征点。该阈值d_α通常根据具体应用场景的误差范围和动态检测需求设定,以平衡误差检测的灵敏性和鲁棒性。

三、实验对比与分析

实验采用TUM数据集来进行对比与分析,TUM数据集来自慕尼黑工业大学,TUM数据集中的fr3系列是一个经典的动态场景序列集,广泛用

于评估视觉SLAM系统在动态环境中的性能表现。该数据集通过Kinect V1相机采集，具有较高的时间同步精度，提供彩色RGB图像和精确的真实运动轨迹（Ground Truth），为研究者提供了可靠的实验数据支持。其中，“walking”序列特别展示了一个人在桌子旁行走的动态场景，可有效用于测试算法在处理动态目标时的鲁棒性和准确性。此外，fr3系列涵盖了多种动态场景，是视觉SLAM以及动态点剔除研究的重要基准数据集。因此使用TUM数据集的动态SLAM序列freiburg3_walking_xyz与freiburg3_walking_static来进行实验，将采用ORB-SLAM2作为基准，与本文提出的算法进行实验对比。通过对比分析，可以评估本文算法在动态场景下的性能优势与改进效果。

通过EVO工具来对本文提出的算法与ORB-SLAM2来进行轨迹绘制并且进行实验对比，估计轨迹与真实轨迹对比如图3、4所示：

图3(a)和(b)分别展示了本文提出的方法与原始ORB-SLAM2算法在动态场景下的轨迹估计结果。

其中，灰色虚线表示相机的真实运动轨迹（Ground Truth），蓝色实线代表算法估计的轨迹。从图3中可以明显看出，本文提出的方法在动态场景中显著提高了轨迹估计的精度。尤其是在复杂动态目标区域，蓝色轨迹与真实轨迹的吻合度明显优于原始ORB-SLAM2算法。

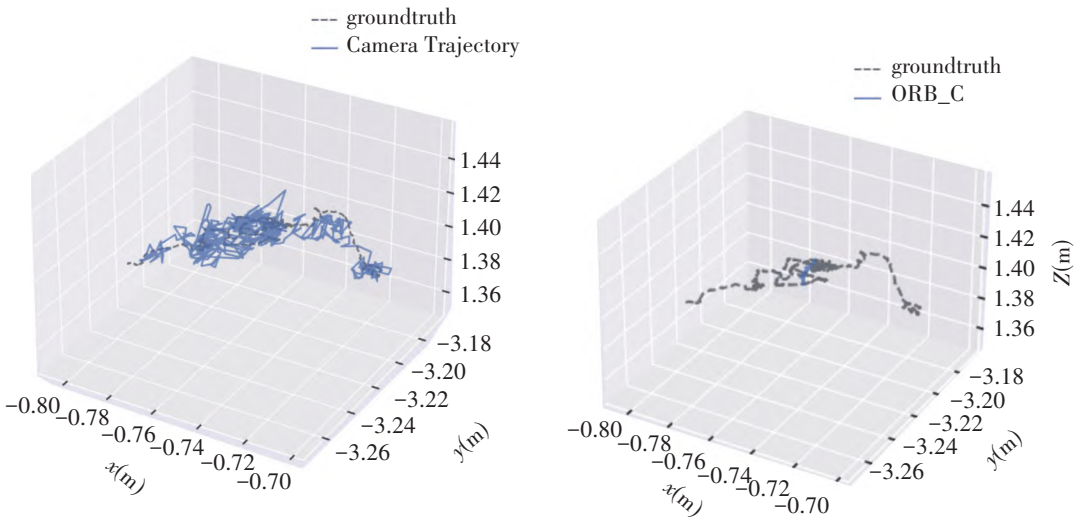
在图4(a)中，本文方法在三维轨迹（xyz）估计中与真实轨迹（Ground Truth）高度贴合，特别是在复杂动态场景下展现了较强的鲁棒性与精确性，整体表现优于原始ORB-SLAM2算法。而ORB-SLAM2在动态场景下偏差明显，轨迹明显偏离真实值，这主要归因于动态目标对特征点匹配的干扰影响了位姿估计。

使用EVO工具进一步对轨迹进行了绝对轨迹误差统计，结果如图5和表1。

图5对比了ORB-SLAM2原始算法和本文改进算法在Freiburg3_walking_xyz数据集上的绝对轨迹误差（APE）。右图显示，ORB-SLAM2误差波动较大，且误差分布分散。左图表明，改进算法误差显著降低，误差波动小，标准差范围明显缩小，展现了更高的轨迹估计精度和鲁棒性。

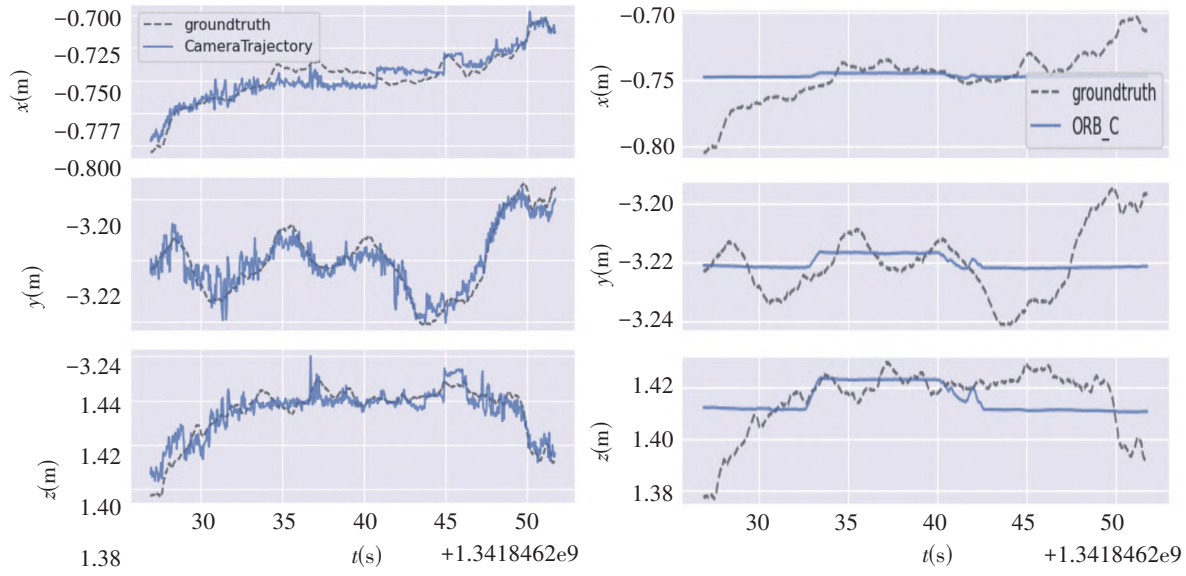
从表1可以明显看出，本文所提出的方法在绝对轨迹误差（Absolute Trajectory Error,ATE）方面相较于原始的ORB-SLAM2系统表现出显著的性能提升。这表明，本文方法在应对动态环境中的视觉SLAM任务时，能够有效过滤不稳定的动态特征，从而提升轨迹估计的准确性与系统的稳定性。

从表2中的对比结果可以看出，本文所提出的方法在绝对轨迹误差（Absolute Trajectory Error,ATE）方面相较于已有的DynaSLAM系统，在动态环境下依然展现出良好的精度与稳定性。说明本文算法在处理动态目标干扰方面具备更强的鲁棒性，而在相对静态的fr3_walking_static序列中，DynaSLAM略占优势，但本文方法仍保持在较低误差范围内，整



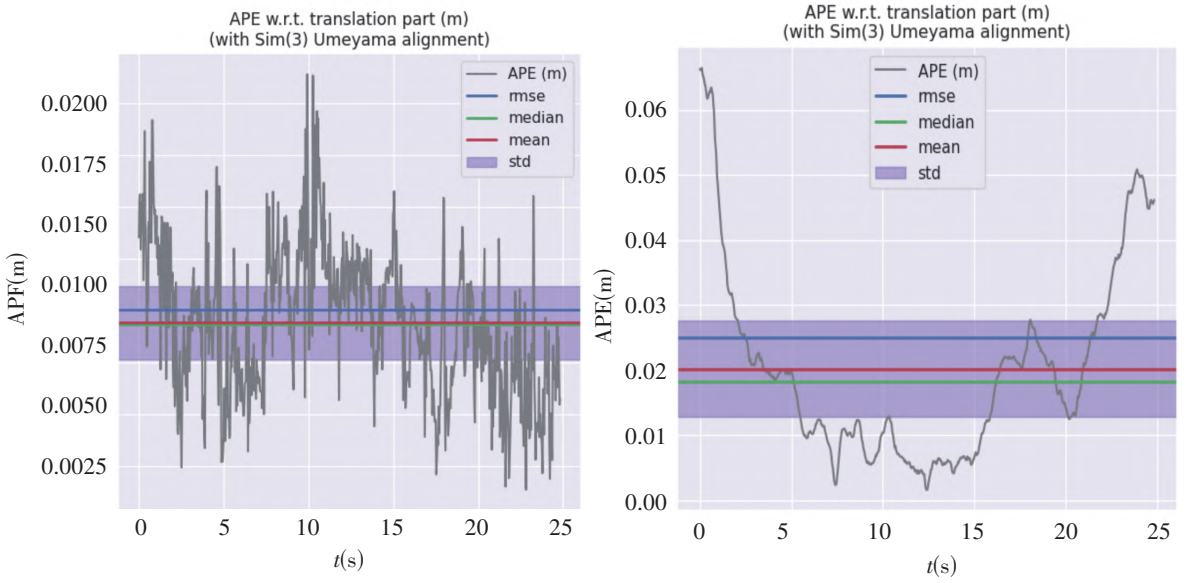
(a)本文的方法　　(b)ORB-SLAM2算法

图3　动态场景下的轨迹



(a)本文的方法　　(b)ORB-SLAM2算法

图4　三维轨迹



(a)本文的方法　　(b)ORB-SLAM2算法

图5　绝对轨迹误差

表1　绝对轨迹误差评估表

序列	绝对轨迹误差/m					
	ORB-SLAM2		本文算法		性能提升/%	
	RMSE	STD	RMSE	STD	RMSE	STD
fr3_walking_xyz	0.2919	0.1291	0.0128	0.0068	95.61	94.73
fr3_walking_static	0.3851	0.1471	0.0098	0.0057	97.46	96.13

体精度较高，证明其不仅适用于动态环境，在静态场景下同样具备良好的通用性与稳定性。

表2 本文算法与DynaSLAM的ATE对比表

序列	本文算法		DynaSLAM	
	RMSE	STD	RMSE	STD
fr3_walking_xyz	0.0128	0.0068	0.0150	0.0086
fr3_walking_static	0.0098	0.0057	0.0060	0.0034

综上所述，本文提出的改进算法在动态场景下展现了更高的轨迹估计精度和鲁棒性，有效减少了动态物体对SLAM系统定位精度的干扰，在静态场景中也保持了可比的性能，体现出其在复杂环境中高效可靠的轨迹估计能力。该方法不仅展现出卓越的动态目标处理能力，还为视觉SLAM技术在复杂环境中的应用提供了一种高效可靠的解决方案。

参考文献：

[1] DAVISON A J, REID I D, MOLTON N D, et al. MonoSLAM: Real-time single camera SLAM[J]. IEEE transactions on pattern analysis and machine intelligence, 2007, 29(6): 1052-67.

[2] MUR-ARTAL R, MONTIEL J M M, TARDOS J D. ORB-SLAM: a versatile and accurate monocular SLAM system[J]. IEEE transactions on robotics, 2015, 31(5): 1147-63.

[3] 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): 1255-62.

[4] CAMPOS C, ELVIRA R, RODRíGUEZ J J G, et al. Orb-slam3: An accurate open-source library for visual, visual – inertial, and multimap slam[J]. IEEE Transactions on Robotics, 2021, 37(6): 1874-90.

[5] 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): 4076-83.

[6] 王红星，贺文龙，王璟源，等. 动态场景下移动机器人实时语义视觉SLAM研究[J]. 兵器装备工程学报，2025，46(4)：252-262.

[7] 刘砚菊，晏佳华，冯迎宾. 动态场景下基于视觉的SLAM技术研究[J]. 半导体光电，2024，45(2)：327-335.

[8] 罗元，沈吉祥，李方宇. 动态环境下基于深度学习的视觉SLAM研究综述[J]. 半导体光电，2024，45(1)：1-10.

[9] 仉新，郑飚默，谭振华，等. 动态场景下移动机器人视觉SLAM[J]. 机床与液压，2023，51(3)：57-63.

[10] 阮晓钢，郭佩远，黄静. 动态场景下基于深度学习的语义视觉SLAM[J]. 北京工业大学学报，2022，48(1)：16-23.

[11] 房立金，王科棋. 一种结合深度学习的运动重检测视觉SLAM算法[J]. 计算机工程，2022，48(5)：18-26.

[12] 梁鸿，陈俊熹，李丽华，等. 动态场景下结合语义的半直接法视觉里程计[J]. 计算机应用研究，2021，38(3)：941-945.

（责任编辑：夏侯国论）

Dynamic Scene Visual Algorithm SLAM Based on Deep Learning

WANG Xiaodong[1], *CHEN Yuanfeng*[1], *YANG Weigao*[2]

(1. School of Intelligent Manufacturing, Guangzhou City Polytechnic;2. Equipment Management Office, Guangzhou City Polytechnic, Guangzhou 511300,China)

Abstract: Simultaneous Localization and Mapping (SLAM) technology is a key research direction for autonomous exploration by drones or robots in unknown and complex environments. Using this technology, the robot can calculate its own pose in real time through the data acquired by its sensors while simultaneously constructing a high-precision map of the external environment. With this information, the robot can not only formulate path planning, but also dynamically correct the position error, thus significantly improving the accuracy and stability of navigation in unknown environments. In visual SLAM systems, pose calculation typically relies on geometric algorithms and feature matching techniques. These methods generally assume that the external environment consists of static objects, i.e., they are based on the premise of a static scene. However, in practical applications, dynamic objects such as pedestrians and vehicles frequently appear, posing significant challenges to system performance and robustness. Therefore, this study integrates deep learning with visual SLAM algorithms by adding a dynamic object detection thread to the existing ORB-SLAM2 framework. This thread identifies dynamic objects and removes dynamic points in the odometer calculation to reduce the impact of dynamic targets on the system positioning accuracy. Experimental results show that proposed method significantly reduces the absolute trajectory error and improve the applicability and stability of the SLAM algorithm in dynamic scenarios.

Key words: visual SLAM; dynamic scenes; deep learning; object detection