DOI:10.19651/j.cnki.emt.2416024

结合轻量化 YOLOv5s 的动态视觉 SLAM 算法*

黄友锐[1] 王照锋[2] 韩 涛[1] 宋红萍[1]
(1.安徽理工大学电气与信息工程学院 淮南 232001; 2.安徽理工大学计算机科学与工程学院 淮南 232001)

摘 要: 针对视觉 SLAM 在真实环境中易受车辆、行人等运动物体影响而导致位姿估计降低的问题,提出一种结合轻量化 YOLOv5s 的动态视觉 SLAM 算法,将改进的轻量级 YOLOv5s 作为目标检测算法用于判断运动物体;结合提出的动态特征点剔除方法,剔除动态特征点,仅采用静态特征点进行位姿估计和地图跟踪。在 TUM 数据集进行实验,相较于 ORB-SLAM3 算法,改进后的算法在高动态序列上的位姿估计精度分别提升了 89.29%、65.34%、94.42%,结果表明改进后的算法能够有效剔除动态特征点,提高了视觉 SLAM 算法在动态环境下的位姿估计精度和定位精度。

关键词: 动态 SLAM;目标检测;光流法;ORB 特征点
中图分类号: TP391.9;TN98 **文献标识码**: A **国家标准学科分类代码**: 510.99

Dynamic visual SLAM algorithm combined with lightweight YOLOv5s

Huang Yourui[1] Wang Zhaofeng[2] Han Tao[1] Song Hongping[1]
(1. Collage of Electrical and Information Engineering, Anhui University of Science and Technology, Huainan 232001, China;
2. Collage of Computer Science and Engineering, Anhui University of Science and Technology, Huainan 232001, China)

Abstract: Aiming at the problem that visual SLAM is easily affected by moving objects such as vehicles and pedestrians in real environment, resulting in low pose estimation accuracy, a dynamic visual SLAM algorithm combined with lightweight YOLOv5s is proposed. The improved lightweight YOLOv5s is used as an object detection algorithm to judge moving objects. Combined with the proposed method of dynamic feature points elimination, dynamic feature points are eliminated, and only static feature points are used for pose estimation and map tracking. Experiments on TUM data set show that compared with ORB-SLAM3 algorithm, the pose estimation accuracy of the improved algorithm on high dynamic sequence is improved by 89.29%, 65.34% and 94.42% respectively. The results show that the improved algorithm can effectively eliminate dynamic feature points. The pose estimation and positioning accuracy of visual SLAM algorithm in dynamic environment are improved.

Keywords: dynamic SLAM;object detection;LK optical flow;ORB feature point

0 引 言

同时定位与建图(simultaneous localization and mapping, SLAM)是指机器人在未知环境中,确定自身位置和姿态并构建周围环境地图的技术[1]。视觉 SLAM 是借助相机作为外部传感器进行自身定位与地图构建的技术。

现有的视觉 SLAM 系统根据视觉里程计的计算方法不同将视觉 SLAM 分为直接法和特征点法[2]。直接法基于光度一致性假设,利用输入图像中像素的亮度信息作为光度误差并构建稠密地图[3]。特征点法通常使用特征提取器和描述符提取特征点,依靠特征点进行匹配并构建稀疏地图[4]。关于直接法,Engel 等[5]提出的 LSD-SLAM 算法根据图像光度一致性对图像帧进行对齐并估计像素深度,实现定位和半稠密地图构建。关于特征点法,Campos 等[6]发布了 ORB-SLAM3 算法,新增了 IMU、相机融合算法,是第一个能够在算法的所有阶段重新利用来自高视差共视关键帧的所有先前信息的系统。谢波等[7]提出一种单目视觉惯导里程计优化方法,通过 IMU 预积分信息与特征点深度增强方法对系统进行重定位,提高了系统的跟踪稳定性。传统的基于直接法和特征点法的视觉 SLAM 算法都是基于静态假设,如 LSD-SLAM、ORB-SLAM3 等,这些算法在静态环境中或具有少量动态对象的环境中都能取得较高的

收稿日期:2024-05-12
* 基金项目:安徽省高效协同创新项目(GXXT-2023-068)资助

精度和鲁棒性。但传统 SLAM 算法在动态场景中响应比较慢，算法性能会显著下降[8]。

为提高系统在动态环境下的定位精度和鲁棒性，减少运动物体对系统的影响，目前的视觉 SLAM 算法主要是对图像帧中的动态特征点进行处理，动态特征点的处理方法有基于几何的方法和基于深度学习的方法。基于几何的方法，Zeller 等[9]提出一种直接全光学里程计(direct plenoptic odometry，DPO)算法，利用多视图几何跟踪和映射直接作用于微透镜阵列形成的微图像上，根据后续光场帧之间的对应关系生成半稠密深度图，同时能够提供更详细的点云。基于深度学习的方法，Yu 等[10]提出的 DS-SLAM 算法，结合 SegNet[11]语义分割网络与运动一致性检测，剔除对系统影响较大的动态特征点，提高了系统在动态环境下的定位精度，但语义分割比较耗时，无法满足实时性要求。Liu 等[12]提出利用语义信息去除相机运动估计中的异常值，并提出一种数据关联算法，尽可能使用静态对象上的特征，同时将语义信息保存在 Atlas 图集中，利用图集中的语义信息进行相机的位姿优化。邹斌等[13]提出一种点云分割方法，结合 YOLOv3 检测语义信息，并构建了几何信息复杂的语义地图。吴丽凡等[14]通过对动态目标进行识别并剔除不稳定的特征点，实现了对视觉 SLAM 的定位性能的优化，但系统实时性较差。Gökcen 等[15]将 YOLOv3 集成到 ORB-SLAM3 中，利用 YOLOv3 检测动态物体并剔除动态特征点，提高了系统的定位精度。阮晓钢等[16]利用 Mask R-CNN 网络分割动态对象，通过运动一致性检测算法剔除动态物体的特征点。Gao 等[17]提出 RTD-SLAM，将 YOLOv5 与 ORB-SLAM2 相结合，利用改进目标检测网络和多视图几何消除动态特征点，降低了动态特征点对系统的影响。仉新[18]等通过 YOLOv4 识别场景动态语义信息，剔除动态物体对位姿估计的干扰，并利用点线特征实现位姿跟踪和回环检测。李嘉铭等[19]提出一种结合目标检测和多视图几何约束的动态 SLAM 系统，并根据运动概率判别与剔除动态特征点，但平均每帧图像处理时间太长，且在大规模视野情况下系统稳定性较低。

针对环境中动态物体干扰而导致视觉 SLAM 系统位姿估计精度降低的问题，本文提出一种结合轻量化 YOLOv5s 的动态视觉 SLAM 算法，融合动态特征点剔除方法，利用改进后的 YOLOv5s 目标检测算法和 LK 光流法来剔除环境中的动态特征点，最后只利用保留的静态特征点进行定位和建图。

1　关键技术研究

ORB-SLAM3 包括跟踪线程、局部建图和闭环检测 3 个主要线程，其主要组成如图 1 所示。跟踪线程处理传感器信息，实时计算当前帧相对于活动地图的位姿，使匹配地图特征的重投影误差最小化，同时判断当前帧是否为关键帧；局部建图线程根据关键帧构建局部地图并删除冗余关键帧；回环检测线程利用所有关键帧的信息进行关联检测，寻找关联错误，并依据全局关系进行位姿校正和全局位姿优化。本文提出结合轻量化 YOLOv5s 的动态视觉 SLAM 算法，以 ORB-SLAM3 为主体框架，改进部分主要集中在跟踪线程。

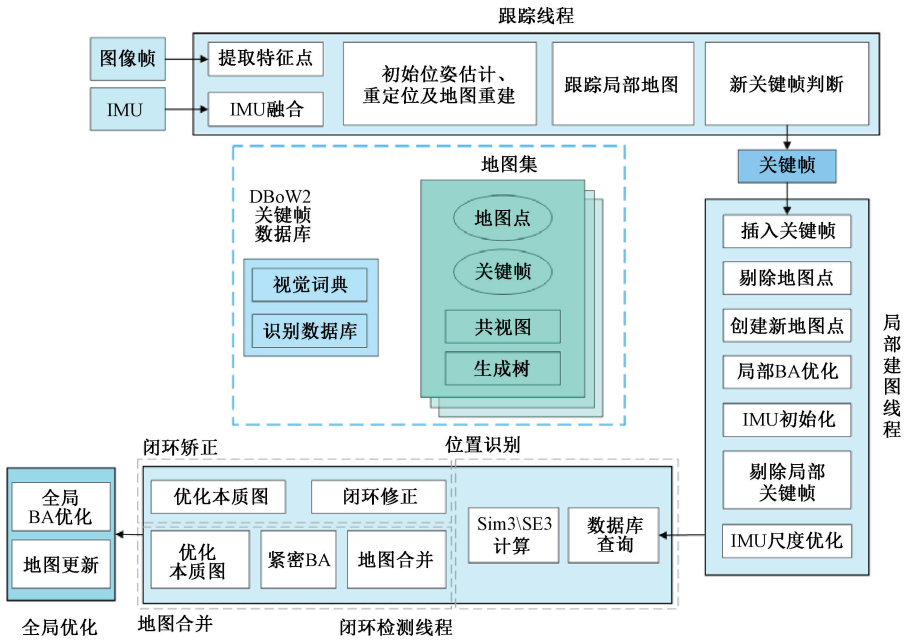


图 1　ORB-SLAM3 主要组成

Fig. 1　Main components of ORB-SLAM3

改进后的跟踪线程如图 2 所示。在 ORB-SLAM3 的基础上,增加了轻量化 YOLOv5s 目标检测线程,利用先验信息检测动态目标和潜在动态目标,并输出相应锚框坐标;在跟踪线程新增动态特征点剔除模块,通过 LK 光流法对动态目标及潜在动态目标锚框内的特征点进行判断,将判断后的静态特征点用于后续的位姿优化,以减少动态特征点对定位精度的影响。

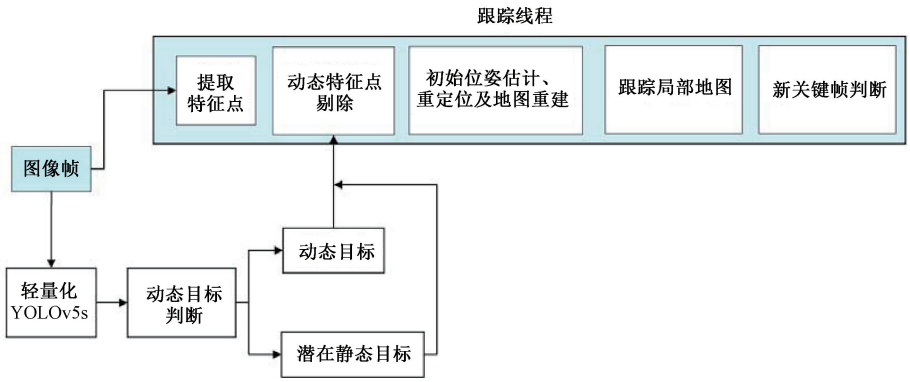


图 2　改进后的跟踪线程

Fig. 2　Improved trace threads

1.1　改进的轻量化 YOLOv5s 目标检测算法

本文采用 YOLO 系列中稳定性较高的 YOLOv5 作为动态目标检测算法,选择深度和特征图宽度均较小、检测速度最快、模型体积最小的 YOLOv5s 网络作为动态目标检测的主体网络。为满足动态环境下的目标检测和系统实时运行的需求,本节在 YOLOv5s 网络的基础上进行改进,结合 Ghost 模块[20]将 YOLOv5s 网络中的普通卷积和 C3 模块分别替换为轻量化的 GhostConv 和 C3Ghost 模块;在主干网络中添加 SimAM[21]注意力机制,以增强网络对于重要特征的捕捉能力。改进后的 YOLOv5s 网络结构如图 3 所示。

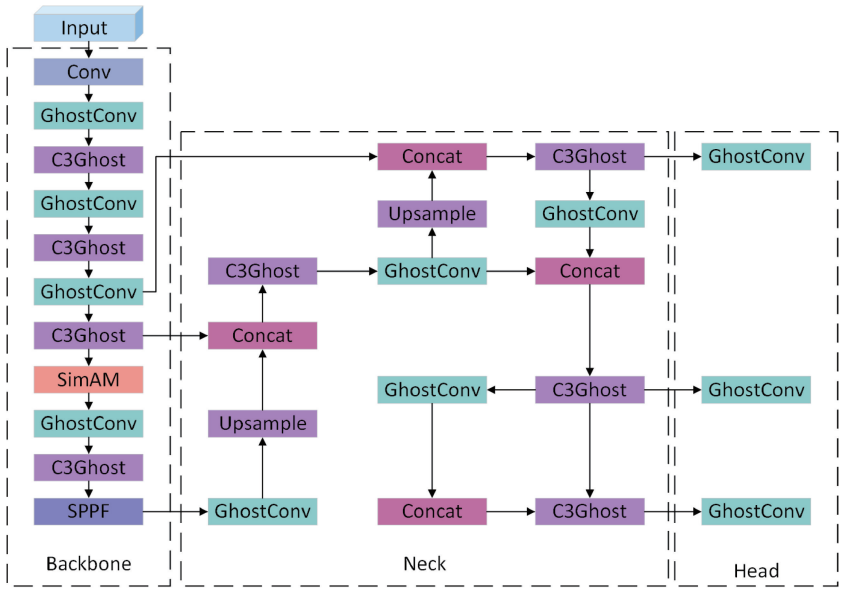


图 3　改进后的 YOLOv5s 网络结构

Fig. 3　The improved YOLOv5s networks structure

1)Ghost 模块

为了使 YOLOv5s 网络更加轻量化,本文在 YOLOv5s 的主干网络中引入轻量化的 Ghost 模块,Ghost 模块如图 4 所示。相比于使用常规卷积直接生成特征图,Ghost 模块没有使用大量的参数逐个生成这些冗余特征图,而是利用廉价操作生成冗余特征图。Ghost 模块将原始卷积层分为两部分,第一部分使用较少的卷积核来生成原始特征图;第二部分使用 DWConv 深度可分离卷积进行廉价操作 ϕ 生成冗余特征图,并利用恒等映射(Identity)将原始特征图直接输出;最后将上述两部分生成的原始特征图和冗余特征图进行拼接得到输出特征图。Ghost 模块的计算量大幅度降低。

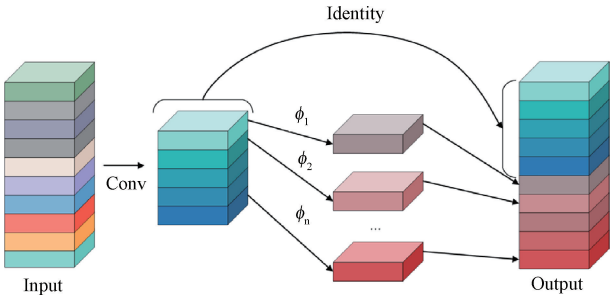


图 4　Ghost 模块

Fig. 4　Ghost module

2)SimAM 注意力机制

引入 Ghost 模块有效降低了模型的参数量和计算量，但在一定程度上降低了网络对图像特征的提取能力。因此，为增强网络特征的表达能力，本文在 Backbone 中添加 SimAM 注意力机制。SimAM 注意力模块如图 5 所示，特征 X 直接生成三维权重，在每个子图中，相同的颜色表示空间位置或特征上的某个点，不同的点使用不同的权重。

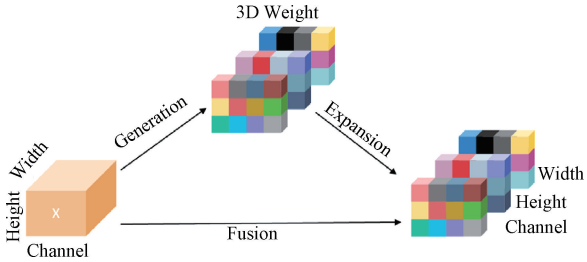


图 5　SimAM 注意力模块

Fig. 5　SimAM attention module

SimAM 注意力模块是一种无参数的轻量化注意力，提出一种基于神经科学理论的能量函数，根据能量函数计算三维注意力权重，能量函数定义如下：

$$e_t(w_t,b_t,\boldsymbol{y},x_i)=(y_t-\hat{t})^2+\frac{1}{M-1}\sum_{i=1}^{M-1}(y_o-\hat{x}_i)^2 \tag{1}$$

式中：$\hat{t}=w_t t+b_t$ 和 $\hat{x}_i=w_t x_i+b_t$ 分别是 t 和 x_i 的线性变化，其中 t 是目标神经元，x_i 是输入特征的单个通道的其他神经元，y_t 与 y_o 分别是神经元 t 和 x_i 的标签。i 是空间维度上的索引，$M=H\times W$ 是该通道上的神经元，w_t 和 b_t 线性变换中的权重和偏置。

当 $\hat{t}$ 等于 y_t，$\hat{x}_i$ 等于 y_o 时，式(1)达到最小值，通过最小化该方程，找到目标神经元 t 与同一通道中的所有其他神经元之间的线性可分性。对式(1)中的 y_t 和 y_o 采用二值标签标记(即分别为 1 和−1)，并在式(1)中加入正则化项，能量函数如式(2)表示为：

$$e_t(w_t,b_t,\boldsymbol{y},x_i)=\frac{1}{M-1}\sum_{i=1}^{M-1}(-1-(w_t x_i+b_t))^2+(1-(w_t t+b_t))^2+\lambda w_t^2 \tag{2}$$

式中：λ 是正则化系数，λw_t^2 是正则化项。

对于 w_t 和 b_t 有如下解析解：

$$w_t=-\frac{2(t-\mu_t)}{(t-\mu_t)^2+2\sigma_t^2+2\lambda} \tag{3}$$

$$b_t=-\frac{1}{2}(t+\mu_t)w_t \tag{4}$$

式中：$\mu_t=\frac{1}{M-1}\sum_i^{M-1}=1x_i$ 和 $\sigma_t^2=\frac{1}{M-1}\sum_i^{M-1}(x_i-\mu_t)^2$ 是该通道中除 t 外所有神经元的平均值和方差。

能量函数最终可表示为：

$$e_t^*=\frac{4(\hat{\sigma}^2+\lambda)}{(t-\hat{\mu})^2+2\hat{\sigma}2+2\lambda} \tag{5}$$

为方便计算，式(5)中 $\hat{\mu}=\frac{1}{M}\sum_{i=1}^{M}x_i$ 和 $\hat{\sigma}^2=\frac{1}{M}\sum_{i=1}^{M}(x_i-\hat{\mu})^2$ 分别为该通道中所有神经元的平均值和方差。

能量 e_t^* 越低，表明神经元 t 与周围神经元的区别越大，在视觉处理中则应该被赋予更高的权重。因此，每个神经元的权重可由 $1/e_t^*$ 得到。

SimAM 注意力机制是一种轻量级的三维注意力机制，与通道注意力、空间注意力机制不同，其同时考虑通道和空间维度，充分利用能量函数，加快了权重的计算速度，并有效提升了 YOLOv5s 网络的准确性。

1.2　动态特征点剔除方法

1)LK 光流法

在特征跟踪和匹配中，没有必要计算所有像素的光流矢量，仅计算特征点的光流矢量即可，因此本文采用稀疏的 LK 光流法计算视觉里程计提取到的所有特征点的光流场。LK 光流法是一种用于估计图像序列中像素运动的技术，基于 3 个假设条件：(1)像素灰度不变性假设；(2)相邻图像帧时间间隔极短，认为图像帧间运动为“小运动”；(3)图像帧中的相邻像素具有相似的运动。

根据灰度不变性假设可得：

$$I(x+\mathrm{d}x,y+\mathrm{d}y,t+\mathrm{d}t)=I(x,y,t) \tag{6}$$

式中：$I(x,y,t)$ 是 t 时刻位于 (x,y) 处的特征点的灰度，$I(x+\mathrm{d}x,y+\mathrm{d}y,t+\mathrm{d}t)$ 是 $t+\mathrm{d}t$ 时刻该特征点的灰度，根据假设 2，对式(6)左侧泰勒展开保留一阶项如下：

$$I(x+\mathrm{d}x,y+\mathrm{d}y,t+\mathrm{d}t)\approx I(x,y,t)+\frac{\partial I}{\partial x}\mathrm{d}x+\frac{\partial I}{\partial y}\mathrm{d}y+\frac{\partial I}{\partial t}\mathrm{d}t \tag{7}$$

由灰度不变性假设，特征点在 t 时刻和 $t+\mathrm{d}t$ 时刻灰度值相等，则有：

$$\frac{\partial I}{\partial x}dx+\frac{\partial I}{\partial y}dy+\frac{\partial I}{\partial t}\mathrm{d}t=0 \tag{8}$$

记 $\frac{\partial I}{\partial x}$、$\frac{\partial I}{\partial y}$ 为 I_x、I_y，分别是特征点在 x 轴、y 轴方向

上灰度的梯度。在式(8)两边同除 dt 有:

$$I_x \frac{\mathrm{d}x}{\mathrm{d}t} + I_y \frac{\mathrm{d}y}{\mathrm{d}t} = -\frac{\partial I}{\partial t} \tag{9}$$

记 $\frac{\mathrm{d}x}{\mathrm{d}t}$、$\frac{\mathrm{d}y}{\mathrm{d}t}$ 为 u、v,分别是特征点在 x 轴、y 轴方向上的速度。记 $\frac{\partial I}{\partial t}$ 为 I_t,即特征点对时间的变化量,式(9)写成矩阵形式有:

$$\begin{bmatrix} I_x & I_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = -I_t \tag{10}$$

为计算特征点 (x,y) 光流矢量,以特征点为中心选取一个 5×5 的像素窗口,根据假设 3,内部 25 个像素具有相同的运动,该方程转化为关于 u、v 的超定方程,如式(11)所示。

$$\begin{bmatrix} I_x & I_y \end{bmatrix}_k \begin{bmatrix} u \\ v \end{bmatrix} = -I_{tk}, k = 1, \cdots, 5^2 \tag{11}$$

记:

$$\boldsymbol{A} = \begin{bmatrix} \begin{bmatrix} I_x, I_y \end{bmatrix}_1 \\ \vdots \\ \begin{bmatrix} I_x, I_y \end{bmatrix}_k \end{bmatrix}, \boldsymbol{b} = \begin{bmatrix} I_{t1} \\ \vdots \\ I_{tk} \end{bmatrix} \tag{12}$$

则式(11)可以写成式(13):

$$\boldsymbol{A} \begin{bmatrix} u \\ v \end{bmatrix} = -\boldsymbol{b} \tag{13}$$

当系数矩阵 $\boldsymbol{A}$ 超定时,利用最小二乘法计算 u、v 结果如下:

$$\begin{bmatrix} u, v \end{bmatrix}^{\mathrm{T}} = -(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A})^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{b} \tag{14}$$

因此,由 LK 光流法求得 t 时刻特征点 (x,y) 的光流矢量,用于跟踪估计某个特征点在图像中的位置。

2)动态特征点剔除算法

本文提出一种动态特征点剔除算法,该算法流程如算法 1 所示。首先,对日常场景中常见的物体进行分类,例如人、动物、汽车等会经常移动,归类为动态目标;桌子、柜子等不经常移动,归类为潜在动态目标。动态目标锚框表示为 B_d,潜在动态目标锚框表示为 B_p,动态特征点集合为 D,静态特征点集合为 S。首先,计算动态目标和潜在动态目标锚框内特征点的光流矢量 (u,v)。其次,计算集合 S 中特征点的光流矢量,代入式(15),计算出集合 S 中特征点的平均光流矢量 (U,V)。接着将锚框内的特征点经过式(16)进行判断,若大于光流阈值 L,则说明该特征点是动态特征点,放入集合 D 中;若小于光流阈值 L,则说明该特征点是静态特征点,放入静态特征点集合 S 中。最后,保留集合 S 中的静态特征点用于位姿估计。

$$\begin{bmatrix} U \\ V \end{bmatrix} = \frac{1}{M}\sum_{k=1}^{M} \begin{bmatrix} u_i \\ v_i \end{bmatrix} \tag{15}$$

$$\sqrt{(u-U)^2 + (v-V)^2} > L \tag{16}$$

本文的动态特征点剔除策略没有将动态目标框内的特征点和潜在动态目标框内的特征点全部剔除,而是利用 LK 光流法进行判断,将动态目标框和潜在动态目标框内的静态背景点进行保留,减小了动态特征点对系统的影响,进一步提高了系统的定位精度。

算法 1:动态特征点剔除算法

1. 输入:当前帧 F_t 中特征点,动态目标锚框 B_d,潜在动态目标锚框 B_p,光流阈值 L
2. 输出:当前帧静态特征点
3. **begin**
4. for(f_i in F_t)//判断当前帧中的特征点是否在目标锚框内
5. **if** (f_i in B_p ‖ f_i in B_d)
6. f_i 放入动态特征点集合 D
7. **else**
8. f_i 放入静态集合 S
9. 计算集合 S 中各特征点光流,并代入式(15)计算平均光流 (U,V)
10. 计算集合 D 中各特征点光流(u,v)
11. for(f_i in D)//判断集合 D 中特征点是否为动态特征点
12. **if** $\sqrt{(u-U)^2+(v-V)^2} > L$
13. 将该特征点放入集合 D 中
14. **else**
15. 将该特征点放入集合 S 中
16. **return** 集合 S
17. **end**

2 实验与结果分析

为验证本文算法的有效性,分别对改进后的轻量化 YOLOv5s 算法和改进后的 ORB-SLAM3 算法进行实验,实验在 Ubuntu18.04 操作系统上进行,硬件设备 CPU 型号为 R7-6800H,GPU 为 8 G 显存的 NVIDIA Geforce RTX 3050 Ti。

2.1 轻量化目标检测算法实验

为验证轻量化 YOLOv5s 算法的有效性,本实验选择 VOC2007 数据集进行训练和测试,该数据集包含人、车、马、羊等 20 种类别,将数据集按 7∶2∶1 划分为训练集、验证集和测试集,共 8 713 张,部分类别如图 6 所示。

目标检测算法通常采用均值平均精确度(mean average precision,mAP)评价模型的性能,mAP 值越大,说明模型的性能越好。mAP 计算方式如下:

$$mAP = \frac{1}{C}\sum_{j=1}^{C} AP_j \tag{17}$$

式(17)中 AP_j 表示某类样本的平均精度,C 表示样本类别总数,j 表示当前类样本。

利用改进后的 YOLOv5s 检测效果如图 7 所示。可以

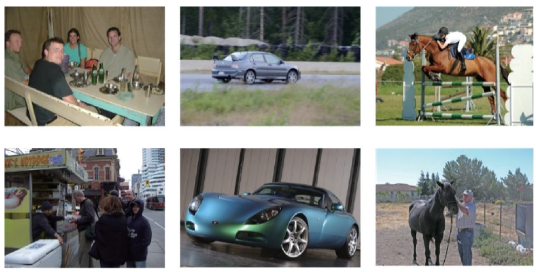


图 6　VOC2007 数据集部分类别

Fig. 6　Part of the categories in the VOC2007 dataset

观察到图 7(a)和(b)中仅露出一小部分特征及远距离的小目标，均能够被准确识别。

(a) 部分特征
(a) Partial characteristics

(b) 远距离小目标
(b) Small targets at long range

图 7　检测效果

Fig. 7　Detection performance

为验证本文所提模块的有效性，现分别添加 Ghost 模块、SimAM 注意力进行实验，实验结果如表 1，其中 Baseline 是 YOLOv5s 算法。从表 1 中可以看出，添加 Ghost 模块后模型参数量降低了 49.8%，模型大小减小了 45.7%，mAP 则降低了 3.5%。仅添加 SimAM 注意力后，模型参数量不变，mAP 提升了 1.3%。相较于 YOLOv5s 模型，本文算法添加 Ghost 模块、SimAM 注意力，模型参数量和模型大小分别降低 49.8%、45.7%，同时检测精度均略高于 YOLOv5s 模型。本文改进实现了对 YOLOv5s 的轻量化，在不降低检测准确度的同时减小了模型参数量。

表 1　消融实验

Table 1　Ablation study

实验序号	Ghost	SimAM	mAP/%	模型参数量/M	模型大小/MB
Baseline	—	—	63.2	7.41	14.1
1	√		59.7	3.72	7.51
2	—	√	64.5	7.41	14.4
3	√	√	63.5	3.72	7.57

2.2　位姿估计误差分析实验

1)TUM 数据集

为验证改进后的 ORB-SLAM3 算法有效性，本文采用德国慕尼黑工业大学提供的 TUM 数据集进行实验。本文选取 TUM 数据集中的 sitting_static、sitting_xyz、walking_half、walking_static、walking_xyz 5 个图像序列，其中 sitting_static、sitting_xyz 2 个序列是低动态序列数据集，walking_half、walking_static、walking_xyz 3 个序列是高动态序列数据集。

2)实验结果分析

实验结果采用计算估计位姿与真实位姿之间差异性的绝对轨迹误差(absolute trajectory error，ATE)和能够直接测量里程计漂移的相对轨迹误差(relative pose error，RPE)作为评价指标。使用 ATE 的均方根差(RMSE)、标准差(STD)、误差均值(Mean)和误差中值(Median)评估 SLAM 系统的性能，其中加粗部分代表最佳结果。表 2、表 3 分别为 ATE、RPE 实验结果。

表 2　绝对轨迹误差(ATE)对比

Table 2　Comparison of absolute trajectory error (ATE)　　m

数据集	ORB-SLAM3				改进算法			
	RMSE	STD	Mean	Median	RMSE	STD	Mean	Median
sitting_static	**0.006 5**	0.003 3	0.005 6	0.004 9	0.006 8	0.003 3	0.005 9	0.005 2
sitting_xyz	**0.008 9**	0.004 6	0.007 6	0.006 8	0.011 8	0.006 7	0.009 8	0.008 3
walking_half	0.184 9	0.074 4	0.169 3	0.148 7	**0.019 8**	0.009 7	0.017 3	0.015 4
walking_static	0.025 1	0.013 5	0.021 2	0.018 4	**0.008 7**	0.004 6	0.007 4	0.006 5
walking_xyz	0.260 1	0.100 7	0.239 8	0.238 2	**0.014 5**	0.007 1	0.012 6	0.011 2

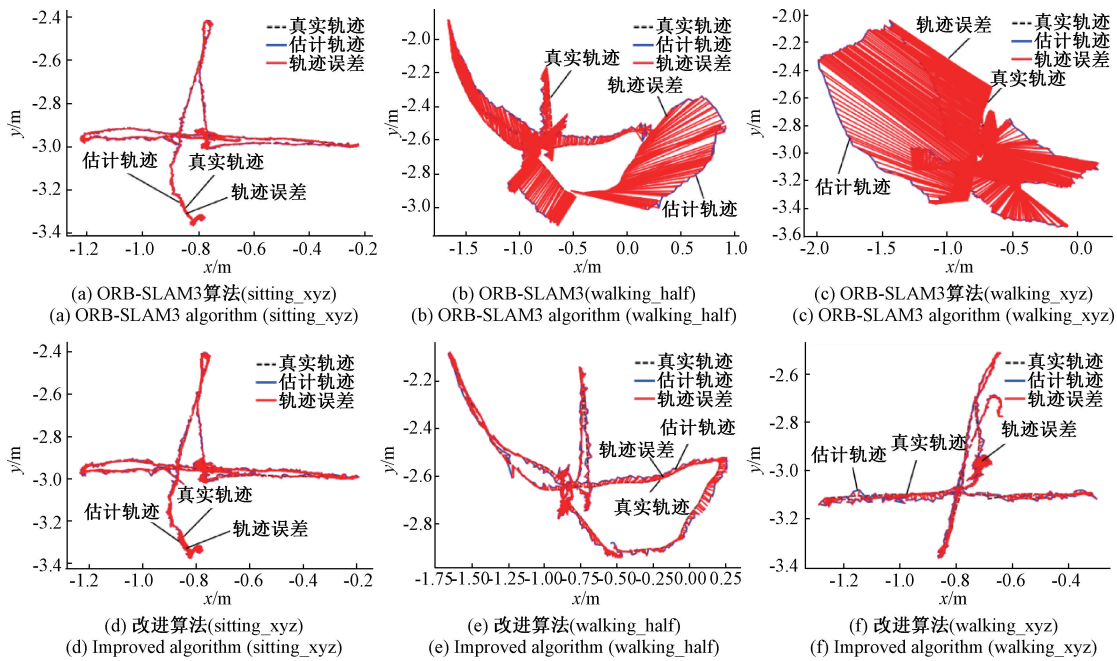
根据表 2，改进算法在 walking_half、walking_static 和 walking_xyz 高动态序列中相比于 ORB-SLAM3 算法，绝对轨迹误差分别提升 89.29%、65.34%、94.43%。根据表 3，改进算法在 walking_half、walking_static 和 walking_xyz 高动态序列中相比于 ORB-SLAM3 算法分别提升 39.58%、51.41%、52.36%。在低动态场景下，动态目标几乎静止，原 ORB-SLAM3 算法可以将图像帧中的所有特征点用于特征匹配和跟踪，改进算法则将部分静态特征点检测为动态特征点而剔除，导致每帧图像中可用的特征点数目减少，故改进算法在 sitting_static、sitting_xyz 低动态序列中，绝对轨迹误差和相对轨迹误差均没有提升。

图 8 为 ORB-SLAM3 算法和改进算法的估计轨迹与

表 3　相对轨迹误差(RPE)对比

Table 3　Comparison of relative pose error (RPE)　　m

数据集	ORB-SLAM3				改进算法			
	RMSE	STD	Mean	Median	RMSE	STD	Mean	Median
sitting_static	**0.005 3**	0.002 7	0.004 5	0.003 9	0.006 1	0.003 2	0.005 1	0.004 4
sitting_xyz	**0.008 4**	0.004 2	0.007 2	0.006 5	0.011 3	0.006 8	0.008 9	0.007 4
walking_half	0.023 5	0.015 9	0.017 4	0.012 7	**0.014 2**	0.008 2	0.011 6	0.009 7
walking_static	0.014 2	0.010 9	0.009 1	0.005 7	**0.006 9**	0.003 8	0.005 7	0.004 6
walking_xyz	0.025 4	0.014 8	0.020 6	0.016 6	**0.012 1**	0.006 7	0.010 1	0.008 1



(a) ORB-SLAM3算法(sitting_xyz)
(a) ORB-SLAM3 algorithm (sitting_xyz)

(b) ORB-SLAM3(walking_half)
(b) ORB-SLAM3 algorithm (walking_half)

(c) ORB-SLAM3算法(walking_xyz)
(c) ORB-SLAM3 algorithm (walking_xyz)

(d) 改进算法(sitting_xyz)
(d) Improved algorithm (sitting_xyz)

(e) 改进算法(walking_half)
(e) Improved algorithm (walking_half)

(f) 改进算法(walking_xyz)
(f) Improved algorithm (walking_xyz)

图 8　估计轨迹和真实轨迹对比图

Fig. 8　Comparison chart of estimated trajectory and true trajectory

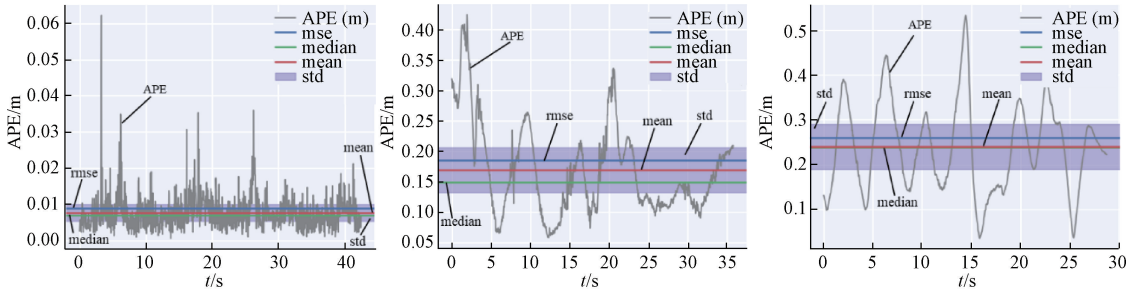
真实轨迹对比图。图 8 中虚线表示真实轨迹,实线表示估计轨迹,实线阴影区域表示估计位姿和真实位姿之间的误差,阴影区域越小,表明系统的定位精度越高。根据图 8 可以看出,在动态序列 walking_half、walking_xyz 中,改进算法的轨迹误差明显小于 ORB-SLAM3 算法的轨迹误差。

图 9 为 ORB-SLAM3 算法与改进算法的绝对轨迹误差分布图。从图 9 中可以看出 ORB-SLAM3 算法在低动态序列上,其平均误差 mean 很小;改进算法由于丢失部分特征点,其平均误差 mean 略低于原算法;在高动态场景下,改进算法较 ORB-SLAM3 算法有明显的提升,系统的定位精度更高。

图 10 为 ORB-SLAM3 算法与改进算法的相对轨迹误差分布图,在低动态序列 sitting_xyz 上,改进算法的平均误差 mean 比 ORB-SLAM3 算法略高;在高动态序列 walking_half、walking_xyz 上,改进算法的平均误差 mean 远小于 ORB-SLAM3 算法,说明本文算法在高动态场景下的定位精度更高。

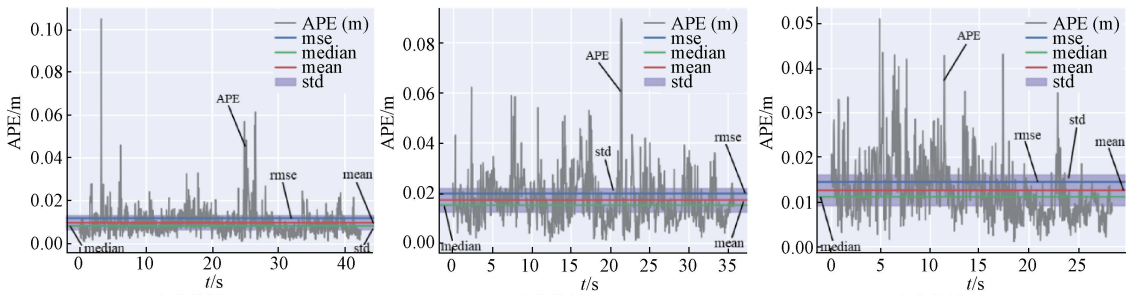
为验证本文所提算法在实际动态场景中的效果,设计了两组基于 Turtlebot2 移动机器人的真实场景实验,如图 11 所示。实验一,移动机器人保持不动,人在相机前移动实验,动态特征点去除效果如图 11(a)所示,位姿估计结果如图 11(b)所示;实验二,移动机器人匀速直线运动,同时人在移动机器人前移动模拟动态环境,动态特征点去除效果如图 11(c)所示,位姿估计结果如图 11(d)所示。真实场景实验结果表明,本文所提动态特征点剔除方法能够检测动态物体,并剔除动态特征点,保留了静态背景点,提升了算法在动态环境中的定位精度。

本文算法还与近年来比较优秀的动态 SLAM 算法进行了比较。其中 DS-SLAM、RDS-SLAM 结合语义分割算法剔除动态特征点;Object-SLAM、RTD-SLAM 采用目标检测算法检测动态物体并剔除动态锚框内所有特征点。本文算法与上述算法的 ATE 对比如表 4 所示,其中加粗部分是最优结果,"—"表示原论文未对该数据集进行实验。从表 4 可以看出最佳结果来自 DS-SLAM 与改进算



(a) ORB-SLAM3算法(sitting_xyz)　(b) ORB-SLAM3算法(walking_half)　(c) ORB-SLAM3算法(walking_xyz)

(a) ORB-SLAM3 algorithm (sitting_xyz)　(b) ORB-SLAM3 algorithm (walking_half)　(c) ORB-SLAM3 algorithm (walking_xyz)

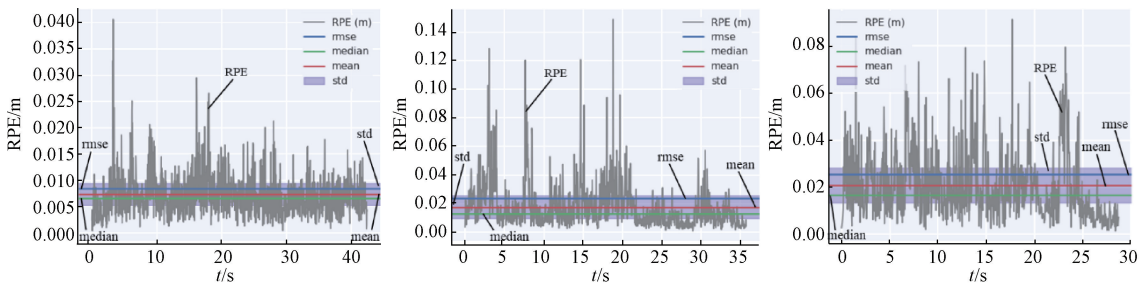


(d) 改进算法(sitting_xyz)　(e) 改进算法(walking_half)　(f) 改进算法(walking_xyz)

(d) Improved algorithm (sitting_xyz)　(e) Improved algorithm (walking_half)　(f) Improved algorithm (walking_xyz)

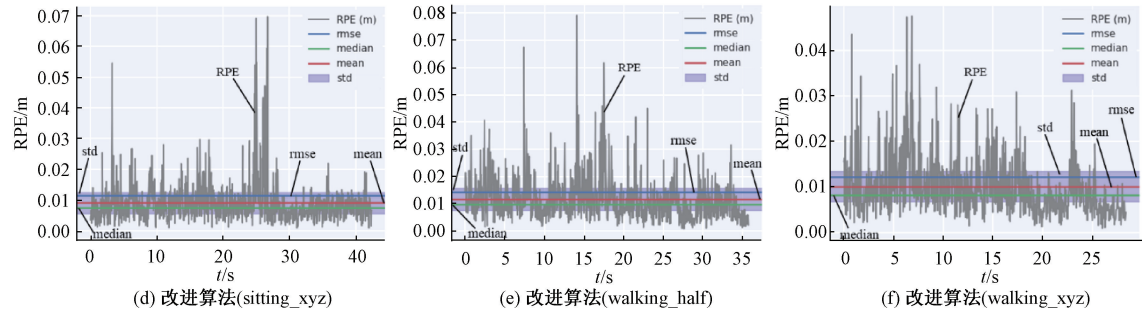
图 9　绝对轨迹误差分布对比

Fig. 9　Comparison of absolute trajectory error distribution



(a) ORB-SLAM3算法(sitting_xyz)　(b) ORB-SLAM3(walking_half)　(c) ORB-SLAM3算法(walking_xyz)

(a) ORB-SLAM3 algorithm (sitting_xyz)　(b) ORB-SLAM3 algorithm (walking_half)　(c) ORB-SLAM3 algorithm (walking_xyz)



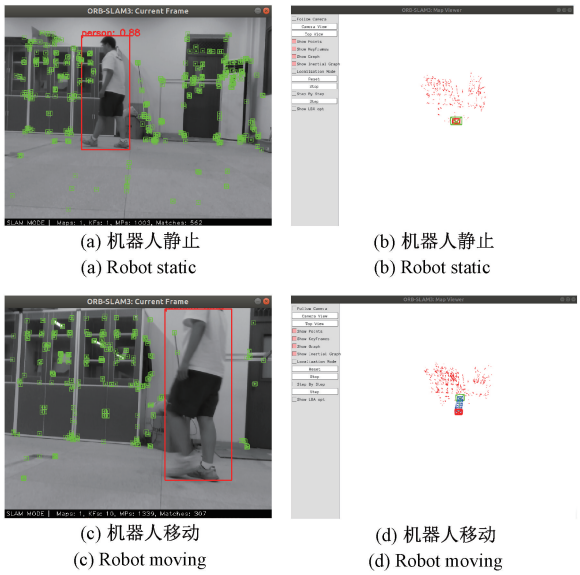
(d) 改进算法(sitting_xyz)　(e) 改进算法(walking_half)　(f) 改进算法(walking_xyz)

(d) Improved algorithm (sitting_xyz)　(e) Improved algorithm (walking_half)　(f) Improved algorithm (walking_xyz)

图 10　相对轨迹误差分布对比

Fig. 10　Comparison of relative trajectory error distribution

法，在静态序列 sitting-static 和动态序列 walking-static 上，DS-SLAM 与改进算法分别相差 0.000 3、0.000 6，改进算法的误差略高于 DS-SLAM；在动态序列 walking-half、walking-xyz 上，DS-SLAM 与改进算法分别相差 0.010 5、0.010 2，改进算法的误差远低于 DS-SLAM，改进算法具有更高的定位精度。根据表 4、5，相较于同样使用目标检测网络的 RTD-SLAM，本文所提算法在位姿估计的精度与算法运行实时性两方面都有较大提升；与基于分割网络的



(a) 机器人静止 (b) 机器人静止
(a) Robot static (b) Robot static

(c) 机器人移动 (d) 机器人移动
(c) Robot moving (d) Robot moving

图 11 真实环境下动态特征点去除效果

Fig. 11 The effect of removing dynamic feature points in a real-world environment

DS-SLAM 相比,本文算法在动态序列中精度更高,且实时性远胜于 DS-SLAM。

表 4 本文算法与其他算法绝对轨迹误差对比

Table 4 This paper compares the absolute trajectory error of our algorithm with that of other algorithms

数据集	DS-SLAM	RDS-SLAM	Object-SLAM	RTD-SLAM	改进算法
sitting-static	**0.006 5**	0.008 8	0.018 4	—	0.006 8
walking-half	0.030 3	0.025 9	0.097 1	0.027 8	**0.019 8**
walking-static	**0.008 1**	0.085 1	0.013 4	0.120 7	0.008 7
walking-xyz	0.024 7	0.021 3	0.042 9	0.019 9	**0.014 5**

表 5 本文算法与其他算法每帧追踪时间对比

Table 5 The tracking time per frame of the proposed algorithm is compared with other algorithms

算法	GPU	Semantic/Detect	每帧追踪时间/ms
ORB-SLAM3	NVIDIA RTX 3050 Ti	—	25.36
DS-SLAM	P4000 GPU	Semantic	59.79
RTD-SLAM	NVIDIA RTX2070	Detect	43.47
改进算法	NVIDIA RTX 3050 Ti	Detect	29.86

3 结 论

本文提出一种结合轻量化 YOLOv5s 的动态视觉 SLAM 算法,以解决环境中的动态物体干扰问题。基于 ORB-SLAM3 主体框架,将改进的 YOLOv5s 网络与 LK 光流法相结合,利用动态特征点剔除方法剔除动态特征点。首先,使用 Ghost 模块减少 YOLOv5s 网络的模型参数量和计算量,并添加 SimAM 注意力机制提升轻量化后的模型检测精度;其次,利用 LK 光流法计算特征点光流矢量;最后,根据动态特征点剔除方法剔除图像帧中的动态特征点,减少动态物体对系统定位精度的影响。在 TUM 数据集的高动态序列实验表明,与 ORB-SLAM3 系统相比,本文算法在高动态序列中位姿估计精度分别提升 89.29%、65.34%、94.43%,提高了系统定位精度。相比 DS-SLAM、RDS-SLAM、Object-SLAM 等动态 SLAM 系统,本文算法在定位精度上均有一定提升。针对改进算法的实时性下降问题,下一步将优化动态特征点剔除方法,缩短系统的运行时间,保证算法精度的同时提升算法实时性。

参考文献

[1] 张耀,吴一全,陈慧娴. 基于深度学习的视觉同时定位与建图研究进展[J]. 仪器仪表学报,2023,44(7):214-241.
ZHANG Y, WU Y Q, CHEN H X. Research progress of visual simultaneous localization and mapping based on deep learning[J]. Chinese Journal of Scientific Instrument, 2023,44(7): 214-241.

[2] 谷晓琳,杨敏,张燚,等. 一种基于半直接视觉里程计的 RGB-DSLAM 算法[J]. 机器人,2020,42(1): 39-48.
GU X L, YANG M, ZHANG Y, et al. An RGB-D SLAM algorithm based on semi-direct visual odometry[J]. Robot, 2020,42(1): 39-48.

[3] 马科伟,张锲石,康宇航,等. 移动机器人中视觉里程计技术综述[J]. 计算机工程,2021,47(11): 1-10.
MA K W, ZHANG Q SH, KANG Y H, et al. Overview of visual odometry technology in mobile robots[J]. Computer Engineering, 2021, 47(11): 1-10.

[4] YUAN J, ZHU S H, TANG K T, et al. ORB-TEDM: An RGB-D SLAM Approach Fusing ORB Triangulation Estimates and Depth Measurements[J]. IEEE Transactions on Instrumentation and Measurement, 2022,71: 1-15.

[5] ENGEL J, STÜCKLER J, CREMERS D. Large-scale direct SLAM with stereo cameras[C]. IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS). 2015: 834-849.

[6] 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(4): 1051-1062.

[7] 谢波,张国良,李歆,等. 一种单目 VIO 定位精度与跟

踪稳定性优化方法[J]. 国外电子测量技术,2023,42(4):23-30.
XIE B, ZHANG G L, LI X, et al. Optimization method for positioning accuracy and tracking stability of monocular VIO [J]. Foreign Electronic Measurement Technology,2023,42(4): 23-30.

[8] 杨雪梅,李帅永. 移动机器人视觉 SLAM 回环检测原理、现状及趋势[J]. 电子测量与仪器学报,2022,36(8): 1-12.
YANG X M, LI SH Y. Prinple, current situation and trend of visual SLAM loop closure detection for mobile robot[J]. Journal of Electronic Measurement and Instrumentation,2022,36(8): 1-12.

[9] ZELLER N, QUINT F, STILLA U. From the calibration of a light-field camera to direct plenoptic odometry[J]. IEEE Journal of Selected Topics in Signal Processing,2017,11(7): 1004-1019.

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

[11] BADRINARAYANAN V, KENDALL A, CIPOLLA R, et al. SegNet: A deep convolutional encoder-decoder architecture for image segmentation [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence,2017,39(12):2481-2495.

[12] LIU Y B, MIURA J. RDS-SLAM: Real-time dynamic SLAM using semantic segmentation methods[J]. IEEE Access,2021, 9:23772-23785.

[13] 邹斌,林思阳,尹智帅. 基于 YOLOv3 和视觉 SLAM 的语义地图构建[J]. 激光与光电子学进展,2020,57(20): 124-130.
ZOU B, LIN S Y, YIN ZH SH. Semantic mapping based on YOLOv3 and visual SLAM[J]. Laser & Optoelectronics Progress,2020,57(20): 124-130.

[14] 吴丽凡,魏东岩,袁洪. 基于 YOLO 的复杂环境视觉 SLAM 优化方法[J]. 计算机应用,2021,41(S2): 208-213.
WU L F, WEI D Y, YUAN H. YOLO-based SLAM optimization method for complex environment vision[J]. Journal of Computer Applications, 2021, 41(S2): 208-213.

[15] GÖKCEN B, USLU E. Object aware RGBD SLAM in dynamic environments[C]. International Conference on INnovations in Intelligent SysTems and Applications(INISTA),2022: 1-6.

[16] 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.
RUAN X G, GUO P Y, HUANG J. Semantic visual SLAM based on deep learning in dynamic scenes[J]. Journal of Beijing University of Technology, 2022, 48(1): 16-23.

[17] GAO R Z, LI Z H, LI J F, et al. Real-time SLAM based on dynamic feature point elimination in dynamic environment [J]. IEEE Access, 2023, 11: 113952-113964.

[18] 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.
ZHANG X, ZHENG L M, TAN ZH H, et al. Visual SLAM for mobile robots in dynamic scenes [J]. Machine Tool & Hydraulics,2023,51(3): 57-63.

[19] 李嘉铭,解明扬,张民,等. 动态环境下基于语义信息与几何约束的视觉 SLAM 系统[J]. 智能科学与技术学报,2023,5(4): 477-485.
LI J M, XIE M Y, ZHANG M, et al. Visual SLAM based on semantic information and geometric constraints in dynamic environment [J]. Chinese Journal of Intelligent Science and Technology, 2023, 5(4): 477-485.

[20] HAN K, WANG Y H, TIAN Q, et al. Ghostnet: More features from cheap operations[C]. IEEE/CVF Conference on Computer Vision and Pattern Recognition(CVPR), 2020: 1577-1586.

[21] YANG L X, ZHANG R Y, LI L D, et al. SimAM: A simple, parameter-free attention module for convolutional neural networks [C]. International Conference on Machine Learning, 2021, 139: 11863-11874.

作者简介

黄友锐,博士研究生,教授,博士生导师,主要研究方向为智能控制、智能制造。
E-mail:hyr628@163.com

王照锋(通信作者),硕士研究生,主要研究方向为视觉 SLAM、自主导航。
E-mail:1526335437@qq.com

韩涛,硕士研究生,高级工程师,主要研究方向为计算机视觉、机器人自主定位与导航。
E-mail:than@aust.edu.cn

宋红萍,博士研究生,主要研究方向为图像识别和机器人控制。
E-mail:shp7420@163.com