DOI：10.13973/j.cnki.robot.230195

SUI-SLAM：一种面向室内动态环境的融合语义和不确定度的视觉 SLAM 方法

张玮奇，王 嘉，张 琳，马宗方
（西安建筑科技大学，陕西 西安 710300）

摘 要：为解决动态环境中移动物体对视觉 SLAM（同步定位和地图构建）系统的干扰，提出一种融合深度语义和不确定度的动态视觉 SLAM 算法 SUI-SLAM。首先基于 Mask R-CNN（区域卷积神经网络）的语义分割结果获得场景的动态先验信息。然后利用图像深度信息修正分割区域边缘，进一步区分动态环境的前景及背景特征点。最终利用语义分割结果、移动先验信息以及几何误差，计算特征点与 3D 地图点之间的关联不确定度，同时在相机位姿优化过程中加入正则化项用于提升定位的准确度和鲁棒性。为验证算法的有效性，在 TUM 动态数据集上进行了实验。结果表明，SUI-SLAM 算法相比 ORB-SLAM2 算法，在室内高动态场景中的定位精度最高可提升 98.41%；与其他最先进的动态 SLAM 算法相比，SUI-SLAM 算法的位姿估计精度和鲁棒性均有一定程度的提升。

关键词：动态环境；语义分割；SLAM（同步定位和地图构建）；位姿估计；不确定度估计

SUI-SLAM: A Semantics and Uncertainty Incorporated Visual SLAM Algorithm towards Dynamic Indoor Environments

ZHANG Weiqi，WANG Jia，ZHANG Lin，MA Zongfang
(*Xi'an University of Architecture and Technology*, *Xi'an* 710300, *China*)

Abstract: In order to solve the interference of moving objects on visual SLAM (simultaneous localization and mapping) system in dynamic environment, SUI-SLAM (semantics and uncertainty incorporated SLAM), a dynamic visual SLAM algorithm integrating deep semantics and uncertainty, is proposed. Firstly, the dynamic prior information of the scene is obtained based on semantic segmentation using Mask R-CNN (region-based convolutional neural network). Then, the edge of the segmented area is corrected using the image depth information, to further distinguish the foreground and background feature points in dynamic environment. Finally, the semantic segmentation results, movement priors and geometric errors are used to calculate the uncertainty of the correspondence between the feature points and the 3D map points. Regularization items are added in the process of optimizing the camera pose to improve the accuracy and the robustness. In order to verify the algorithm effectiveness, experiments are carried out on the TUM dynamic datasets. Results show that the pose estimation accuracy of SUI-SLAM algorithm can be increased by up to 98.41% in indoor high dynamic scenes compared with ORB-SLAM2 algorithm. While compared with other SOTA (state-of-the-art) dynamic SLAM algorithms, the pose estimation accuracy and the robustness of SUI-SLAM algorithm are also improved to a certain extent.

Keywords: dynamic environment; semantic segmentation; SLAM (simultaneous localization and mapping); pose estimation; uncertainty estimation

视觉 SLAM 使用相机作为主要传感器获取外界环境信息进行机器人自身定位以及外部环境地图绘制，根据提取图像信息方式的不同，视觉 SLAM 可以分为通过最小化投影误差优化相机运动的特征点法[1] 和根据最小化光度误差获得相机位姿的直接法[2]。但目前多数视觉 SLAM 方案的鲁棒性和高效率是在静态环境的假设下实现的，而实际场景中存在运动的人、车辆等明显的动态物体或大面积遮挡的情况，使得视觉 SLAM 前端图像之间产生错误的帧间匹配，导致定位精度下降及场景泛化性能的降低。因此如何排除动态物体及遮挡对视觉 SLAM 的干扰，提高视觉 SLAM 的定位精度及鲁棒性仍是当下的研究重点。

针对视觉 SLAM 在动态环境下定位精度下降的问题，学者们相继提出了基于几何的传统动态物体剔除方法以及基于深度学习的动态物体剔除

基金项目：陕西省自然科学基础研究基金（2023-JC-QN-0802）；陕西省自然科学基金（2023-JC-YB-294）.
通信作者：马宗方，mazf@xauat.edu.cn 收稿/录用/修回：2023-07-14/2023-12-06/2023-12-22

方法。魏彤等[3] 利用改进的半全局块匹配（semi-global block matching，SGBM）算法结合极线约束对场景进行区域分割，通过稀疏动态特征信息对动态区域进行标记，但当动态物体和背景颜色相近时分割会出现错误，而且该算法只适合在低动态场景下运行。Dai 等[4] 在 ORB-SLAM2 算法[5] 的基础上使用 Delaunay 三角剖分方法[6] 滤除图像的动态区域，该方法虽然能够提高 SLAM 系统在动态环境下的精度，但是图像处理的速度比较慢，而且实时性很差。文 [7] 提出一种运动目标去除算法，首先利用稀疏光流检测动态目标的轮廓，再使用 Grab-Cut 算法[8] 对目标进行进一步的分割，但该方法良好的分割效果是在假设相机处于静止状态的前提下获得的，具有一定的局限性。张慧娟等[9] 提出的基于线特征的 RGB-D 视觉里程计方法根据匹配后的特征点计算两帧图像的初始变换矩阵，然后评估提取后的线特征的静态权重，但是线特征的提取难度相比于点特征更大。艾青林等[10] 利用几何约束将室内环境下提取的特征点分类，然后使用卡尔曼滤波减小动态特征点的误差信息后进行位姿计算。

近年来，随着深度学习在图像处理领域的发展，研究人员将语义分割或目标检测网络添加到 SLAM 系统中，提高了动态环境下求解相机运动模型的可靠性。于超等[11] 提出基于 RGB-D 相机的 DS-SLAM，添加 SegNet 网络[12] 对动态目标进行语义分割，在计算相机位姿时使用 RANSAC（随机抽样一致性）方法，结合运动一致性检测算法移除场景中的动态物体，但运动目标仅局限于应用场景中的人。Bescos 等[13] 提出的 DynaSLAM 通过 Mask R-CNN 网络[14] 和多视图几何相结合的方式分割先验动态物体，并针对前景物体遮挡问题进行背景修复，该方法有效提升了动态物体检测的准确性，但对每一帧图像进行语义分割较为耗时，并且同样需要指定动态物体类别。为了提升动态物体检测精确度，文 [15] 提出的 Detect-SLAM 系统结合语义先验知识并使用目标检测网络 SSD（single shot multibox detector）只对关键帧进行动态物体检测，并且通过运动概率的传播在跟踪线程剔除概率较大的动态物体上的特征点，可在光照条件差等复杂条件下有效地检测和识别物体。MID-Fusion 系统[16] 结合几何、光度和语义信息对每帧 RGB-D 输入图像使用 Mask R-CNN 进行实例分割后再进行几何边缘分割，对场景中的每个物体类别通过运动残差估计来判断其是否处于运动状态，使得动态物体检测准确性得到了有效提升。文 [17] 提出的基于单目相机的 Dynamic-SLAM 系统使用 SSD 网络检测先验动态物体，结合漏检补偿算法提高了物体检测的召回率。文 [18] 提出的 CFP-SLAM 算法利用目标检测网络 YOLOv5 和几何约束区分高动态物体和低动态物体，并且使用关键点由粗到精的两阶段静态概率计算方法，提高匹配像素点对在定位时的利用效率，解决了静态关键点误删除的问题。Fan 等[19] 提出的 Blitz-SLAM 系统在 ORB-SLAM2 算法[5] 的基础上增加 Blitz 网络，通过对区域中的匹配点构造对极约束来定位潜在动态区域中的静态匹配点，使用深度图像的几何信息对原始掩模进行修改，去除局部点云中的噪声，使得系统在动态环境下稳定运行。王梦瑶等[20] 提出一种基于自适应语义分割的 RGB-D 算法，通过获取特征点的运动等级信息，从而自适应地判断当前帧的状态，实现语义信息的跨帧检测；根据先验信息及运动状态进行初步的位姿估计，最终根据加权静态约束的结果对位姿进行二次优化。

综上所述，基于传统几何方法的动态 SLAM 算法在对象移动方式多样或存在遮挡等情况下无法有效识别移动特征点，在零先验信息的情况下与一般视觉 SLAM 方法相比定位精度提升不高；深度学习方法获取的动态物体需要预先作出类别定义，但实际场景中动态物体类别多样，且动态物体并非一直处于运动状态，前景中某些可移动的物体也可能处于运动状态，因此该筛选方式只能针对某些场景，并有一定局限性。同时，剔除大面积的动态物体中的特征点后，可跟踪的特征点过少会导致定位失效，这种方法只能应用于低动态且动态特征点较少的环境；判断特征点动态概率能够有效区分其属于高、低动态及静态物体类别，避免有效特征点过少以及过度筛选导致的低动态场景精度下降，但该方法仍旧依赖于动态物体分割结果。为了解决上述问题，本文基于语义分割的先验信息和图像深度信息计算特征点移动概率，估计特征匹配的不确定度，并将其作为权重优化位姿估计。本文提出：

1) 基于语义分割获得场景物体移动先验信息，判断关键帧的特征点属于背景区域或前景物体中的哪种类别，根据深度信息修正分割出边缘区域或交界处，计算不同区域的移动概率；

2) 根据场景语义先验信息及移动概率，计算特征点几何投影误差，估计图像像素点、场景 3 维点对应的不确定度；

3) 在位姿求解过程中加入正则化项，提升定位的精度和鲁棒性。

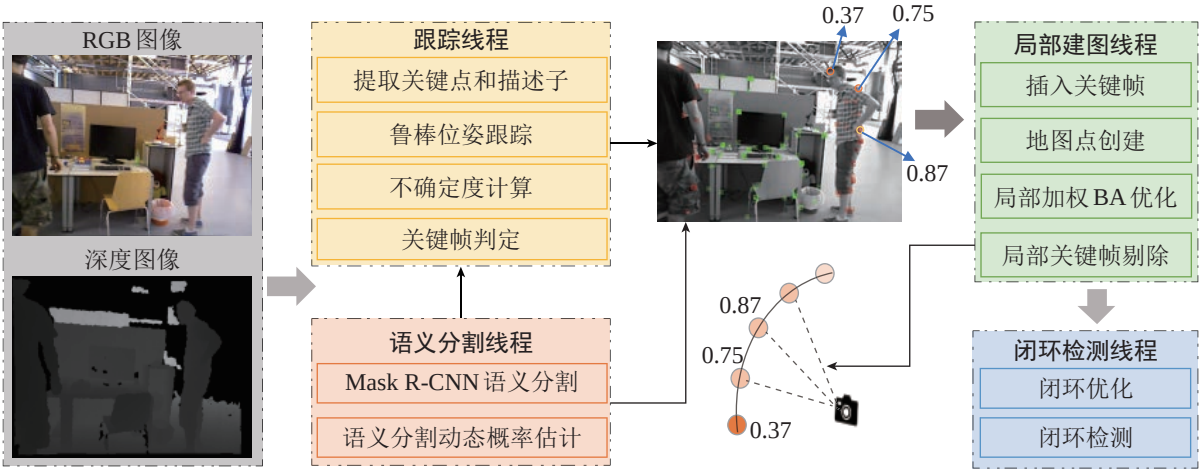


图 1 本文系统框架

Fig.1 Framework of the proposed system

1 系统框架（System framework）

本文的 SUI-SLAM 系统框架主要基于 ORB-SLAM2 算法[5] 构建，如图 1 所示，系统主要包括语义分割、跟踪、局部建图和闭环检测 4 个线程。当 RGB-D 图像传入跟踪线程和语义分割线程后，对其进行并行处理，其中语义分割线程利用 Mask R-CNN 网络[14] 区分动态环境中的自主移动物体、伴随移动物体以及静态背景，通过获取各类别对象的像素级语义标签，即特征的移动先验信息，针对所有非静态背景特征点进行不确定度计算，并利用估计值对位姿优化目标方程进行正则化处理。跟踪线程将筛选后的特征点、关联不确定度等信息传入局部建图以及闭环检测线程，利用 BA（光束平差）优化方法对位姿、地图点进行二次优化后，最终获得相机位姿的估计结果。

2 融合深度语义和不确定度的视觉 SLAM 方法（A deep semantics and uncertainty incorporated visual SLAM method）

2.1 基于语义分割的动态概率估计

本文使用经 MS COCO 数据集预训练的神经网络 YOLOv5 实现室内场景下的可移动物体分割。该数据集包含人、汽车、椅子等 80 个不同类别，本文选择室内场景下出现频率最高的几个类别，对每帧输入图像进行分割。根据 Mask R-CNN 语义分割的结果计算动态概率，将特征点 p_i 的动态概率表示为 $M(p_i)\in[0,1]$，如果特征点 p_i 属于先验动态物体，比如人，则 $M(p_i)=1$；反之，如果特征点属于天花板、墙面等背景，则 $M(p_i)=0$。由于分割后的动态区域边界附近包含一定数量的错误分类的特征点，为了有效筛选特征点，本文首先通过深度值范围区分边缘位置属于前景和背景的特征点。对于当前动态区域特征点，判断其邻近像素 p_j 的深度，当 $\delta_{\mathrm{b}}d_{\mathrm{b}}\leqslant d_j\leqslant\delta_{\mathrm{f}}d_{\mathrm{f}}$ 时，将动态区域进行扩充，其中，d_{b} 为邻近上下左右 4 个像素点的深度值和该点深度值之差的最小值，d_{f} 为 4 个像素点与该点的深度值之差的最大值，实验中参数 δ_{b} 和 δ_{f} 的取值分别为 0.015 和 0.04[22]，同时根据边缘距离定义特征点的移动概率。

定义特征点的集合为 $P=\{p_1,p_2,\cdots,p_n\}$，位于分割区域边界的像素点集合为 $B=\{b_1,b_2,\cdots,b_l\}$。则位于内部的特征点 p_i 与图像前景或背景分割边界的距离 d_i 可以由式 (1) 表示：

$$d_i=\min_{b_s\in B}\|\boldsymbol{p}_i-\boldsymbol{b}_s\|_2 \tag{1}$$

式中，$\|\cdot\|_2$ 表示位于分割区域内部的特征点 p_i 与边界的像素点 b_s 之间的欧氏距离。

如图 2 所示，动态特征点与动态分割边界之间的距离越小，特征点为动态点的概率越低，则特征点 p_i 基于语义分割的动态概率 $M(p_i)$ 如下：

$$M(p_i)=\frac{1}{\exp(-\eta\cdot d_i)+1} \tag{2}$$

式 (2) 中 η 取值为 0.6。

为了提升系统在动态环境下检测的稳定性，确保每一帧图像帧的动态特征点能被有效区分，本文在进行语义分割的同时，采用传统的几何预测方法，对提取的特征点是否产生移动进行判断。这相比于利用 EKF（扩展卡尔曼滤波）等方法，在减少跟踪动态物体耗时的同时，还具有筛选错误的静态区域特征点的作用。不同于 CFP-SLAM 算法[19] 通过计算物体的静态概率来区分物体的高、低动态属性，本文方法将物体类别的初步筛选和移动概

率的计算结果作为判断不确定性的依据，减小对 ORB-SLAM2 系统[5] 位姿估计实时性的影响，提升算法的运行效率。获得特征点移动概率后，对于静态区域或移动概率极小的特征点，可认为特征点间匹配的准确性较高，适用于后续优化；对于其他特征点，通过几何关系进一步估计匹配不确定度。

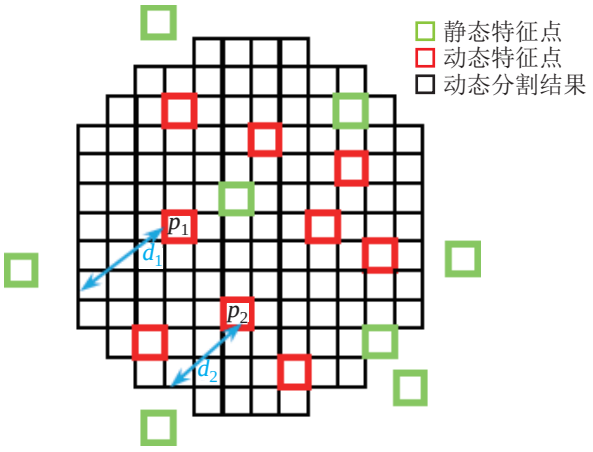


图 2　动态特征点与动态分割边界之间的距离

Fig.2　The distance between the dynamic feature points and the dynamic segmentation boundary

2.2　基于几何信息的不确定度估计

在室内场景下先验物体被划分为不同类别，根据相应类别的属性标注物体的运动状态。为了按属性有效筛选各动态特征点，首先利用移动概率对数据集中的物体类别进行划分：第 t 帧特征点 p_i 的移动概率 $M_t(p_i)$ 越接近 1，则 p_i 处于运动状态的可能性越大；反之，$M_t(p_i)$ 越接近 0，则 p_i 处于静止状态的可能性越大。图 3 中，人是移动物体，椅子、书本为伴随移动物体，对应的移动概率 $M_t(p_i) \geqslant 0.5$；而建筑物等静止物体的移动概率 $M_t(p_i) < 0.5$。根据 Mask R-CNN 网络的分割结果以及移动概率计算获得的第 t 帧图像的掩模 M_t，计算超过阈值（$M_t(p_i) > T_{\mathrm{h}}$）的特征点的几何投影误差。

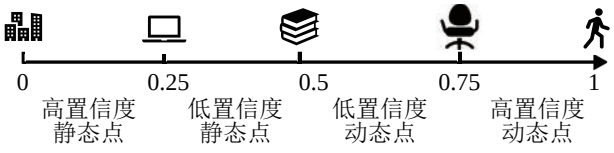


图 3　室内先验物体的移动概率划分

Fig.3　Moving probability division of indoor prior objects

在图像 I_i 中观测到的第 k 个特征点坐标 $\boldsymbol{p}_k^{c_i} = (u,v)$ 与其对应的 3 维坐标 $\boldsymbol{P}_k^{c_i} = (X,Y,Z)$ 之间的变换关系可以表示为

$$\boldsymbol{p}_k^{c_i} = \boldsymbol{\pi}(\boldsymbol{P}_k^{c_i}) = \left(\frac{Xf_x}{Z} + c_x, \frac{Xf_y}{Z} + c_y\right) \tag{3}$$

其中，$\boldsymbol{\pi}(\cdot)$ 表示相机的投影变换函数，f_x 和 f_y 表示相机焦距，(c_x,c_y) 为针孔相机模型的光学中心。对应的，$\boldsymbol{\pi}^{-1}(\cdot)$ 表示反投影变换函数。

第 j 帧图像中第 k 个特征点 $p_k^{c_j}$ 在当前第 i 帧图像中投影点的坐标可以表示为

$$(u',v') = \boldsymbol{\pi}(\boldsymbol{T}_{ij}\boldsymbol{\pi}^{-1}(\boldsymbol{p}_k^{c_j})) \tag{4}$$

其中 $\boldsymbol{T}_{ij} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{t} \\ 0 & 1 \end{bmatrix} \in SE(3)$ 表示两帧图像 I_i 和 I_j 之间的相对变换关系，$\boldsymbol{R}$、$\boldsymbol{t}$ 分别代表相机位姿的旋转量和偏移量。定义深度投影误差公式：

$$E_{\mathrm{prj}}(\boldsymbol{p}_k^{c_i}) = D_i(\boldsymbol{p}_k^{c_i}) - [\boldsymbol{T}_{ij}\boldsymbol{\pi}^{-1}(\boldsymbol{p}_k^{c_j})]_2 \tag{5}$$

式中，D_i 表示第 i 帧对应的深度值，$[\cdot]_2$ 表示取矩阵的第 2 行。对于匹配后关键帧对应的特征点和地图点，重投影误差可以表示为

$$E_{\mathrm{rep}}(\boldsymbol{p}_k^{c_i}) = \left\| \boldsymbol{p}_k^{c_i} - \boldsymbol{\pi}(\boldsymbol{T}_{ij}\boldsymbol{\pi}^{-1}(\boldsymbol{p}_k^{c_j})) \right\|_2 \tag{6}$$

为了衡量场景中移动物体对特征匹配的影响，本文利用关键帧和当前帧投影的几何误差，在特征点移动概率的基础上计算特征匹配的不确定度。将特征点 $p_k^{c_i}$ 的不确定度表示为 $U(\boldsymbol{p}_k^{c_i})$，则：

$$U(\boldsymbol{p}_k^{c_i}) = \frac{1}{\alpha \cdot G(E_{\mathrm{prj}}(\boldsymbol{p}_k^{c_i})) + \beta \cdot G(E_{\mathrm{rep}}(\boldsymbol{p}_k^{c_i}))} \tag{7}$$

其中，α 和 β 分别表示当前帧的 E_{prj} 和 E_{rep} 对应的比例系数，$G(\cdot) = \exp(1+\cdot)^{-1}$。

2.3　位姿估计正则化

在检测出大面积动态物体区域，尤其是物体遮挡场景的情况下，剔除区域内的特征点会导致无法产生有效的匹配。一些方法提出用深度信息聚类方法将区域内的稳定的匹配筛选出来，用于后续位姿估计。本文提出利用特征点匹配不确定度对目标位姿估计方程正则化：不确定度较高的特征点，对应的优化权重趋于 0；而不确定度较低的特征点，对应的权重更高，他们将在位姿优化过程中起到更重要的作用。

设第 i 帧关键帧的特征点坐标和相对应的 3D 地图点坐标分别为 $\boldsymbol{p}_k^{c_i} \in \mathrm{R}^2$，$\boldsymbol{P}_k^{\mathrm{w}} \in \mathrm{R}^3$。对关键帧位姿和地图点构建最小二乘问题：

$$\arg\min_{\boldsymbol{R},\boldsymbol{t}} \sum_{i=1}^{n} \rho(\| \boldsymbol{p}_k^{c_i} - \boldsymbol{\pi}(\boldsymbol{T}_i\boldsymbol{P}_k^{\mathrm{w}}) \|_2^2) \tag{8}$$

其中，ρ 表示鲁棒的 Huber 核函数。

为提升方法在不同动态场景下位姿估计的准确性和鲁棒性，获取特征点不确定度后，进一步改进位姿优化目标方程，即在式 (8) 中添加正则项 $\omega_{p_k^{c_i}}$：

$$\arg\min_{\boldsymbol{R},\boldsymbol{t}} \sum_{i=1}^{n} \rho\left(\left\|\omega_{p_k^{c_i}}\left(\boldsymbol{p}_k^{c_i}-\boldsymbol{\pi}\left(\boldsymbol{T}_i \boldsymbol{P}_k^{\mathrm{w}}\right)\right)\right\|_2^2\right) \tag{9}$$

对于静态物体或位于背景的特征点，令 $\omega_{p_k^{c_i}}=1$；相反，对于动态概率较大的点，对应的 $\omega_{p_k^{c_i}}$ 可表示为

$$\omega_{p_k^{c_i}}=\frac{1}{U\left(\boldsymbol{p}_k^{c_i}\right)\cdot M\left(\boldsymbol{p}_k^{c_i}\right)} \tag{10}$$

对于在前景中并未划分具体类别且被几何方法判定为移动点的特征点，同样计算其不确定度，根据式 (10) 计算优化权重。同时，在后端优化过程中通过加权局部 BA 算法优化相机位姿。

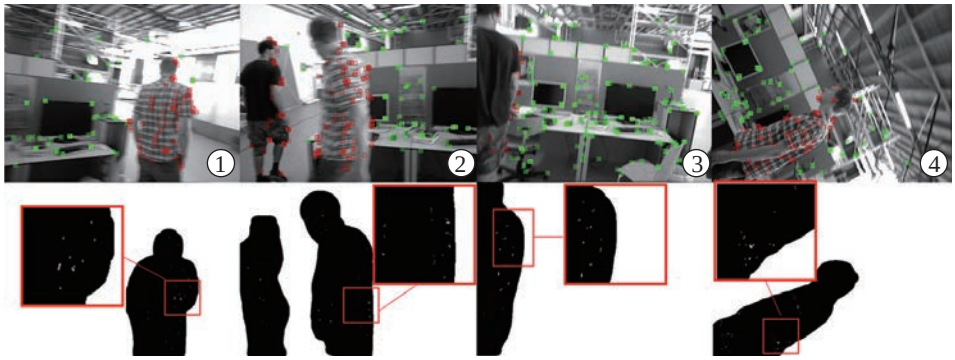
3 实验与分析（Experiment and analysis）

将本文提出的 SUI-SLAM 算法分别与 ORB-SLAM2 算法及其他几种目前在动态场景下表现良好的动态 SLAM 算法进行对比，包括 DS-SLAM[11]、DynaSLAM[13]、CFP-SLAM[18]、Blitz-SLAM[19]，并采用慕尼黑工业大学发布 TUM RGB-D 数据集和 RealSense D435i 深度相机采集的数据集评估系统在动态环境下的性能。TUM 数据集不仅提供了动态环境下的多个序列，而且具有由外部运动捕捉系统获得的准确运动轨迹。其序列可分为高动态场景和低动态场景两类，实验选取了 freiburg3_walking 序列，其中人处于大幅度的运动状态；freiburg3_sitting 序列，其中人处于坐立状态，只有细微的肢体动作。本文算法的运行环境为 64 位 Ubuntu18.04 系统，Intel Core i7-9700 CPU，内存大小为 16 GB，显卡型号为 NVIDIA GTX1650。取 10 次运行的定位结果的中位数，与 TUM 数据集提供的真实值进行对比，得到算法的定位精度和运行时间数据。采用绝对轨迹误差（ATE）、相对位姿误差（RPE）、每帧图像的平均跟踪时间以及 SUI-SLAM 系统各模块的平均运行时间作为评价系统定位精度和运行效率的性能指标。

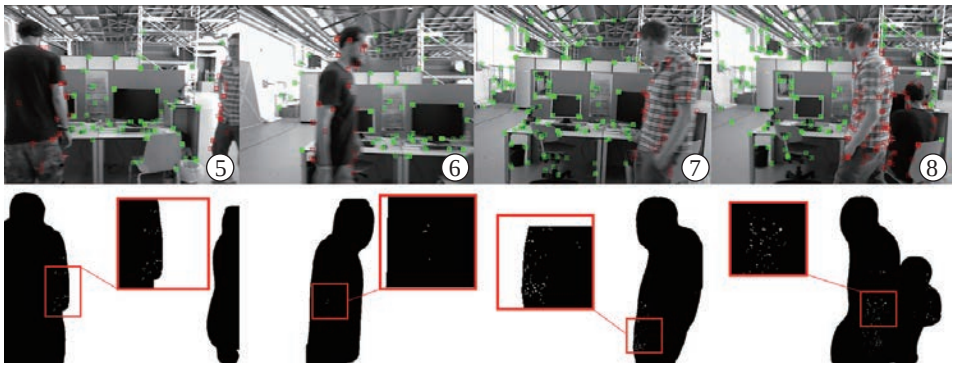
3.1 系统定位精度评估

3.1.1 移动先验概率与不确定度估计

在不确定度估计实验中，令特征点与边界之间的距离阈值 T_{d} 为 0.75，令式 (7) 中深度投影误差 E_{prj} 和重投影误差 E_{rep} 对应的比例系数 α、β 分别取值为 0.4 和 0.6，移动概率的阈值 T_{h} 为 0.5。图 4 中 (a)(b) 分别为 SUI-SLAM 算法在 fr3_walking_ rpy 序列和 fr3_walking_xyz 序列下连续帧的彩色图像及对应位姿化权重估计结果，其中先验动态物体对应的特征点在图像中标记为红色，背景及静态区域的特征点标记为绿色。移动概率计算结果超过阈值 $T_{\mathrm{h}}=0.5$ 的像素点的值标记为 0，计算特征点优化权



(a) fr3_walking_rpy 序列



(b) fr3_walking_xyz 序列

图 4 SUI-SLAM 算法获得的动静态特征点及其对应权重估计结果

Fig.4 Dynamic and static feature points and their corresponding weight estimation results by SUI-SLAM algorithm

重后重新赋值，结果小于 0.5 的像素点的值标记为 1 后，再乘以 255，如图 4 中彩色图像对应黑白图像所示。其中黑色部分表示动态区域，白色部分表示静态区域。在分割的基础上，黑色动态区域特征点权重不为 0 的像素点被标记了出来，如黑白图中红色框线放大部分所示。在某些情况下，如动态物体运动速率较慢，筛选出的权重较大的特征点可以作用于后续的位姿优化过程，提升位姿估计准确度。

图 4 结果表明，在图像序列出现运动模糊、相机旋转及运动物体超出相机视野范围的情况下，基于 Mask R-CNN 的分割结果以及计算获得的移动概率，能够较好地判断图像中物体移动先验信息及边界区域特征点性质，图 4 子图 ⑦中 fr3_walking_xyz 序列存在运动物体的边界相互重合的图像帧时，仍能够准确地分割先验动态物体。fr3_walking_rpy 序列中物体和相机移动得较快，动态概率较大像素区域其特征点优化权重几乎为 0，仅在运动物体和背景临界区域有少量特征点优化权重较高。由于 fr3_walking_xyz 序列中物体和相机移动得较慢，因此特征点的权重在肢体、躯干和背景临界处，以及运动物体的速率减缓时较高，在位姿求解过程中能够起到一定的作用。

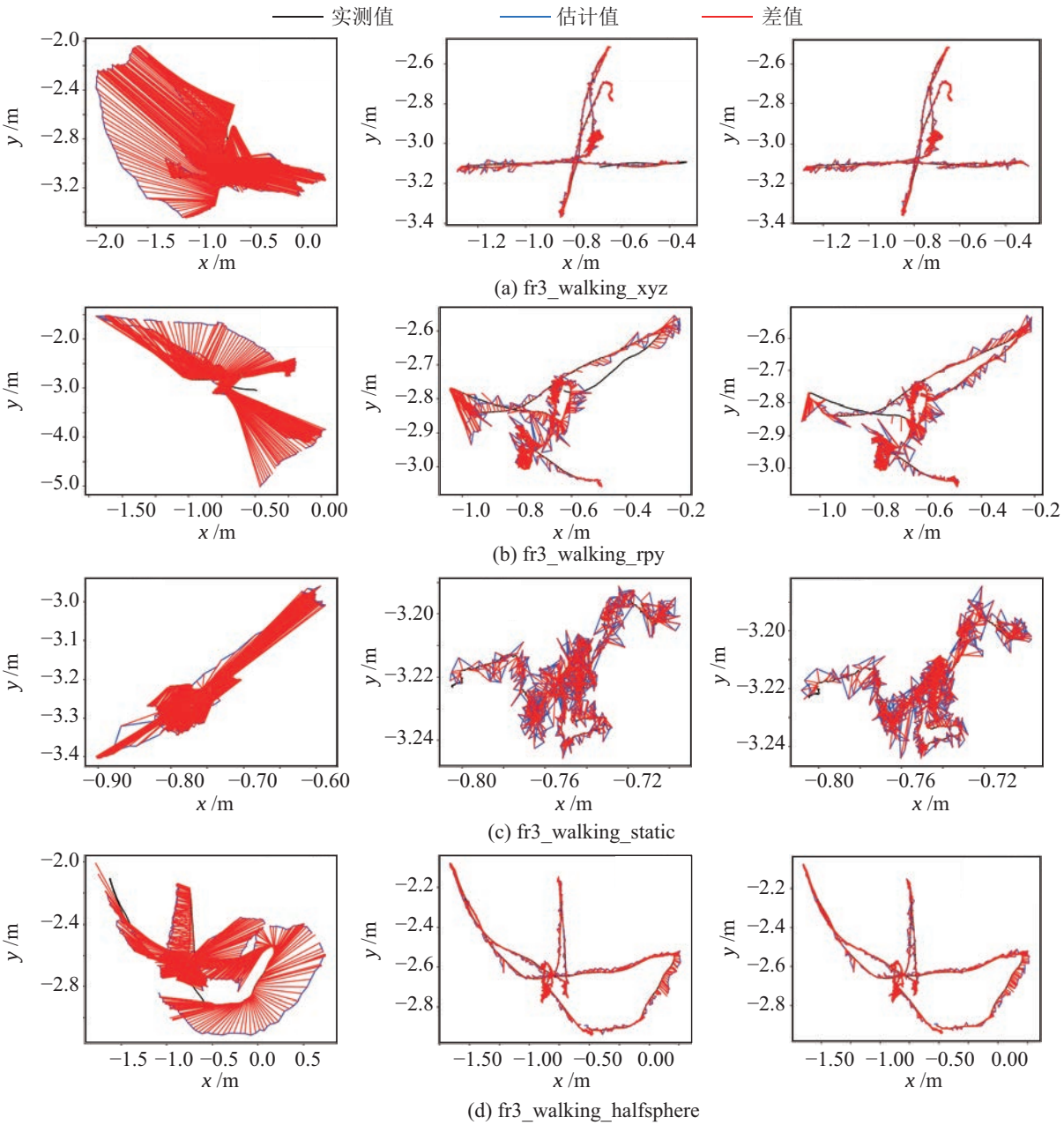


图 5　ORB-SLAM2（第 1 列）与 DynaSLAM（第 2 列）及 SUI-SLAM（第 3 列）算法在高动态场景下的绝对轨迹误差

Fig.5　Absolute trajectory error of ORB-SLAM2 (the 1st column) and DynaSLAM (the 2nd column) and SUI-SLAM (the 3rd column) algorithms in high dynamic scenes

3.1.2 算法评估

为了验证本文算法（SUI-SLAM）在动态环境下的性能，分别将其与 ORB-SLAM2、DynaSLAM 算法在高动态序列中的运行结果进行对比，绝对轨迹误差（ATE）结果如图 5 所示，其中黑色曲线表示相机的真实运动轨迹，蓝色曲线表示相机轨迹的估计值，红色曲线表示每一时刻真实值与估计值之间的偏差值。由图 5 可知，ORB-SLAM2 算法在场景中存在动态目标的情况下，相机位姿估计会出现较大的偏差，其中 ATE 最大能达到 1 m。而本文 SUI-SLAM 算法相比于 ORB-SLAM2 和 DynaSLAM 算法在 fr3_walking 高动态序列下能够较为准确地获得相机位姿，同时能够保证系统稳定运行。

图 6 为 3 种算法在动态序列中平移部分的相对位姿误差（RPE）。经对比，SUI-SLAM 算法通过计算图像语义信息和特征点几何不确定度，并在位姿优化过程中改变不同时刻特征点的权重，提升了定位的鲁棒性。相比于直接剔除动态特征点的方法，本文算法在 fr3_walking 序列上能够更有效地减小定位误差。

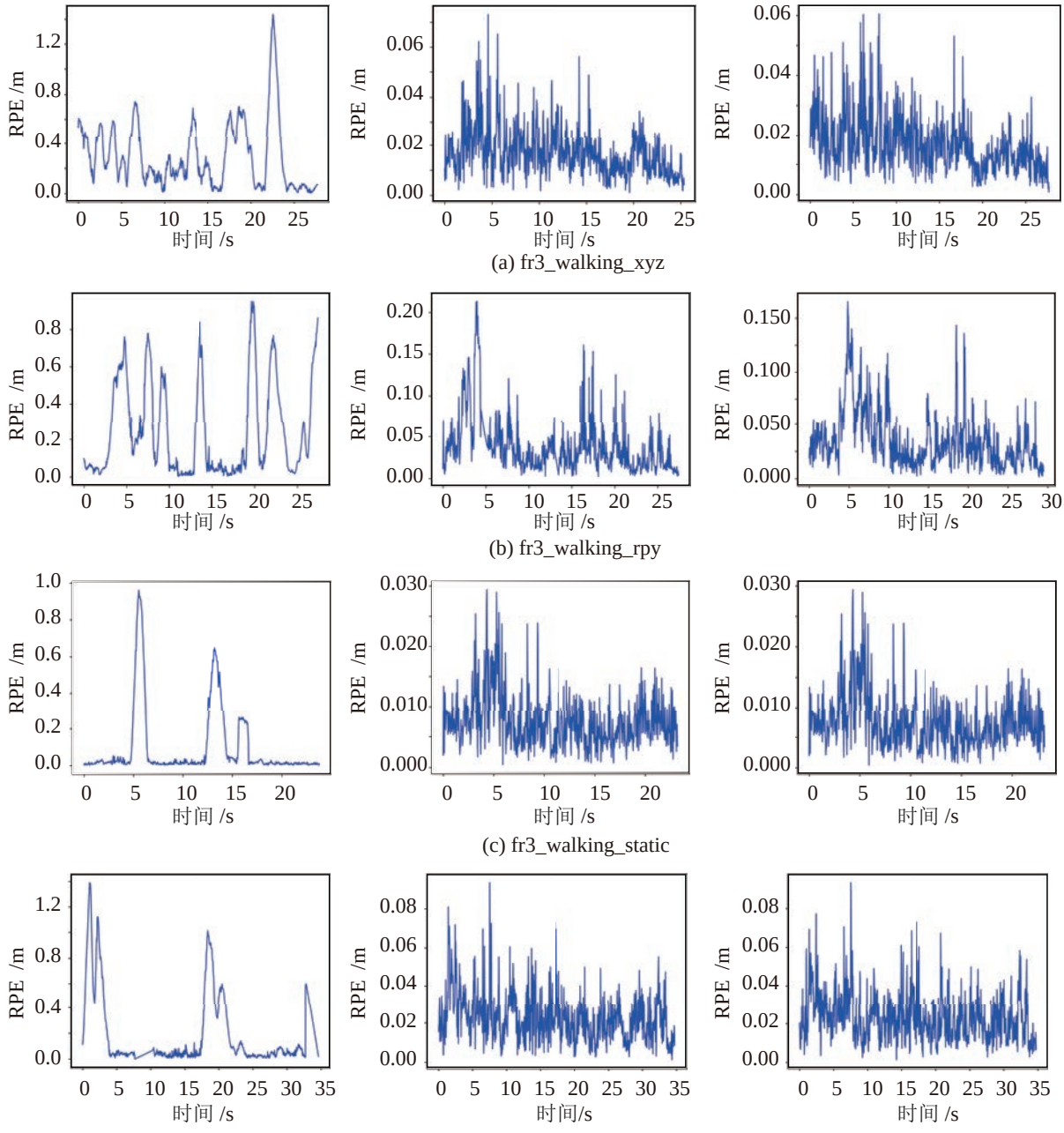


图 6 ORB-SLAM2（第 1 列）与 DynaSLAM（第 2 列）及 SUI-SLAM（第 3 列）算法在高动态场景下的相对位姿误差

Fig.6 Relative pose error of ORB-SLAM2 (the 1st column) and DynaSLAM (the 2nd column) and SUI-SLAM (the 3rd column) algorithms in high dynamic scenes

表 1　TUM 数据集上的绝对轨迹误差结果

Tab.1　Results of absolute trajectory error on TUM dataset

fr3 序列	ORB-SLAM2 [5] /m			SUI-SLAM /m			提升幅度 /%		
	RMSE	Mean	S.D.	RMSE	Mean	S.D.	RMSE	Mean	S.D.
fr3_walking_xyz	0.648 5	0.548 9	0.345 2	**0.014 4**	**0.012 6**	**0.007 0**	97.78	97.71	97.97
fr3_walking_rpy	0.823 8	0.693 2	0.444 5	**0.033 6**	**0.027 6**	**0.023 3**	95.92	96.02	94.76
fr3_walking_static	0.403 6	0.372 3	0.156 0	**0.006 4**	**0.005 6**	**0.003 0**	98.41	98.50	98.08
fr3_walking_half	0.386 3	0.339 4	0.184 5	**0.024 5**	**0.021 3**	**0.012 0**	93.66	93.73	93.50
fr3_sitting_xyz	0.009 2	0.007 7	0.004 7	**0.008 8**	0.007 8	**0.004 1**	4.35	−1.29	12.76
fr3_sitting_static	0.008 4	0.007 5	0.003 9	**0.007 1**	**0.006 4**	**0.003 2**	15.48	14.67	17.95
fr3_sitting_half	0.022 5	0.018 6	0.012 3	**0.019 3**	**0.016 1**	**0.010 7**	14.22	13.44	13.00
fr3_sitting_rpy	0.021 5	0.016 3	0.012 9	**0.020 0**	**0.016 0**	**0.011 6**	6.98	1.85	10.08

表 2　平移相对位姿误差结果

Tab.2　Results of translational relative pose error

fr3 序列	ORB-SLAM2 [5] /m			SUI-SLAM /m			提升幅度 /%		
	RMSE	Mean	S.D.	RMSE	Mean	S.D.	RMSE	Mean	S.D.
fr3_walking_xyz	0.398 1	0.296 9	0.265 1	**0.019 4**	**0.012 6**	**0.007 0**	92.12	95.75	97.35
fr3_walking_rpy	0.364 6	0.256 2	0.259 5	**0.051 3**	**0.042 5**	**0.028 2**	85.93	83.41	89.13
fr3_walking_static	0.217 2	0.096 4	0.194 6	**0.008 6**	**0.007 5**	**0.004 1**	96.04	92.21	97.89
fr3_walking_half	0.357 2	0.195 7	0.298 8	**0.034 8**	**0.031 0**	**0.015 6**	90.25	84.16	94.78
fr3_sitting_xyz	0.014 4	0.009 8	0.005 7	**0.011 5**	**0.010 1**	**0.005 4**	20.13	−3.06	5.26
fr3_sitting_static	0.009 1	0.007 9	0.004 5	0.010 1	0.009 0	**0.004 4**	−10.98	−13.92	2.22
fr3_sitting_half	0.023 7	0.017 0	0.016 5	**0.020 3**	**0.016 3**	**0.012 0**	14.35	4.12	27.27
fr3_sitting_rpy	0.026 6	0.021 8	0.015 6	**0.026 0**	**0.021 3**	**0.014 0**	2.26	2.29	10.26

表 3　旋转相对位姿误差结果

Tab.3　Results of rotational relative pose error

fr3 序列	ORB-SLAM2 [5] /(°)			SUI-SLAM /(°)			提升幅度 /%		
	RMSE	Mean	S.D.	RMSE	Mean	S.D.	RMSE	Mean	S.D.
fr3_walking_xyz	7.453 8	5.622 3	4.893 3	**0.611 5**	**0.484 1**	**0.373 5**	91.80	91.38	92.37
fr3_walking_rpy	7.098 2	5.017 6	5.020 8	**0.987 0**	**0.830 4**	**0.533 4**	86.10	83.45	89.38
fr3_walking_static	3.858 1	1.786 7	3.419 4	**0.246 0**	**0.221 3**	**0.107 3**	93.62	87.61	96.86
fr3_walking_half	7.335 5	4.120 2	6.069 1	**0.852 6**	**0.763 0**	**0.380 5**	88.38	88.76	93.73
fr3_sitting_xyz	0.477 0	0.403 9	0.253 8	**0.479 0**	**0.410 4**	**0.247 0**	0.42	−1.61	2.68
fr3_sitting_static	0.277 5	0.250 2	0.127 4	**0.300 1**	0.272 2	**0.120 1**	−8.14	−8.79	5.72
fr3_sitting_half	0.602 5	0.533 7	0.279 4	**0.601 7**	**0.531 3**	**0.274 5**	0.13	0.45	1.75
fr3_sitting_rpy	0.866 4	0.707 7	0.499 8	**0.762 6**	**0.671 8**	**0.360 9**	11.98	5.07	27.79

表 1～3 分别将 SUI-SLAM 和 ORB-SLAM2 算法在动态环境下的绝对轨迹误差（ATE）、平移相对位姿误差和旋转相对位姿误差进行对比。表中分别记录了 ATE和 RPE 的均方根误差 RMSE、平均值 Mean 以及标准差 S.D.。

结果表明，与 ORB-SLAM2 算法相比，SUI-SLAM 算法的系统位姿估计精度有一定提升。在 fr3_walking 高动态序列下，相机位姿的 ATE 的 RMSE 最大可提升 92.67%～98.41%，RPE 平移部分的 RMSE 可提升 85.33%～96.04%，旋转部分的 RMSE 可提升 85.92%～91.80%。但在 fr3_sitting_xyz 低动态序列中，人多数时刻处于静止状态，并且由于 ORB-SLAM2 算法在静态假设的前提下利用 RANSAC 算法处理图像中的特征点，因此能较准

表 4 SUI-SLAM 与其他动态 SLAM 算法的绝对轨迹误差对比

Tab.4 Comparison of absolute trajectory errors (ATE) of SUI-SLAM and other dynamic SLAM algorithms

单位：m

fr3 序列	DS-SLAM[11]		DynaSLAM[13]		CFP-SLAM[18]		Blitz-SLAM[19]		SUI-SLAM	
	RMSE	S.D.	RMSE	S.D.	RMSE	S.D.	RMSE	S.D.	RMSE	S.D.
fr3_sitting_static	0.007 8	0.003 8	0.006 2	0.002 9	**0.005 3**	**0.002 7**	–	–	0.007 1	0.003 2
fr3_sitting_half	0.016 6	0.007 7	0.019 6	0.009 1	**0.014 7**	**0.006 9**	0.016 0	0.007 6	0.019 3	0.010 7
fr3_sitting_rpy	–	–	0.072 2	0.057 8	0.025 3	0.015 4	–	–	**0.020 0**	**0.011 6**
fr3_sitting_xyz	0.011 5	0.005 6	0.016 2	0.007 1	0.009 0	0.004 2	0.014 8	0.006 9	**0.008 8**	**0.004 1**
fr3_walking_xyz	0.024 7	0.016 1	0.015 4	0.008 0	**0.014 1**	**0.007 2**	0.015 3	0.007 8	0.014 4	0.007 0
fr3_walking_rpy	0.444 2	0.235 0	0.040 0	0.025 9	0.036 8	0.023 0	0.035 6	0.022 0	**0.033 6**	**0.023 3**
fr3_walking_half	0.030 3	0.015 9	0.027 5	0.014 4	**0.023 7**	**0.011 4**	0.025 6	0.012 6	**0.024 5**	**0.012 0**
fr3_walking_static	0.008 1	0.003 6	0.006 5	0.003 2	0.0066	0.003 0	0.010 2	0.005 2	0.006 4	0.003 0

确地估计相机的位姿。SUI-SLAM 算法在 fr3_sitting 低动态场景下的定位精度相比于 ORB-SLAM2 算法的提升效果不明显，在 fr3_sitting 场景中相机位姿的 ATE 的 RMSE 提升范围为 6.98%～15.48%，RPE 平移部分的 RMSE 最大可提升 14.22%，旋转部分的 RMSE 最大可提升 11.98%。

3.1.3 与 SOTA 动态 SLAM 算法对比

为了进一步验证 SUI-SLAM 算法的有效性，如表 4 所示，将本文提出的 SUI-SLAM 算法与目前 SOTA 动态 SLAM 算法 DS-SLAM[11]、DynaSLAM[13]、CFP-SLAM[18] 及 Blitz-SLAM[19] 在 fr3 动态序列下的运行结果进行对比，表中“–”表示对应的文献中没有该数据集的位姿精度结果。由表 4 可得，SUI-SLAM 和 DS-SLAM、DynaSLAM 在 fr3_sitting_static 和 fr3_sitting_half 低动态序列下的运行结果相比，差别不大，本文算法对特征点的正则化处理和其他方法中的特征点筛选，都能够有效消除动态区域内特征点对于位姿估计的影响。而在 fr3_sitting_rpy 和 fr3_sitting_xyz 序列上，本文算法精度较高，说明在图像模糊和相机旋转情况下，SUI-SLAM 算法的鲁棒性更好。一方面，由于 DynaSLAM 算法和 DS-SLAM 算法直接剔除了运动物体上的关键点，当人在图像中占大部分面积时，相比于其他动态 SLAM 算法，误差偏大；另一方面，对于处于复杂运动状态的物体采用帧间光流法判断运动一致性的方式，存在一定误差。因此在 fr3_walking 大部分序列中 DynaSLAM 算法的精度低于 SUI-SLAM 和 CFP-SLAM 算法。在 fr3_sitting_rpy、fr3_sitting_xyz 数据集中，SUI-SLAM 算法和 CFP-SLAM 算法的表现相当。在 fr3_walking_rpy 序列下，本文算法 ATE 的 RMSE 相比于 CFP-SLAM 算法有一定提升。CFP-SLAM 算法采用 YOLO 模型检测动态物体，通过 EKF 算法和 DBSCAN（density-based spatial clustering of applications with noise）算法计算物体静态概率，保证动态物体识别的准确性和完整度。其中物体静态概率估算用时较多，为 17.93 ms，而本文算法不完全依赖于分割结果的精确度。CFP-SLAM 算法基于物体及特征点的静态概率对位姿进行两阶段优化，其中静态概率的计算基于投影约束以及极线约束。表 4 中，和 CFP-SLAM 算法对比，本文算法确实没有明显的精度提升，但在 sitting_rpy 和 walking_rpy 序列上相对具有一定的优势。在 sitting_rpy 序列上，本文算法和 ORB-SLAM2 算法的精度基本一致。以上实验验证了在相机运动角度变化比较大、极限约束失去作用的情况下 SUI-SLAM 算法的有效性。在未来工作中，将考虑优化基于深度信息估计移动概率的方法，提升在没有移动先验信息的情况下筛选动态特征点的准确性。

3.1.4 实际场景测试

为了测试 SUI-SLAM 算法在现实场景中的有效性，用 RealSense D435i 深度相机进行相关实验。实验中，标定 D435i 相机后用相机捕获实验室环境中行走状态下的 2 名学生（白色衣服速度较慢，黑色衣服速度较快）对应的深度及 RGB 图像，采用 SUI-SLAM 算法估计动态及静态特征点对应的权重。图 7 中第 1 行的动态特征点标记为红色，静态特征点标记为绿色，第 2 行表示动态特征点和静态特征点对应的权重估计结果。结果表明，当室内环境中运动物体占据面积较大时，SUI-SLAM 算法能有效避免大部分特征点被剔除，保证位姿的稳定跟踪。当动态物体分割边缘包含较多静态背景中的特征点时，该算法能减小静态点被误删除的概率。

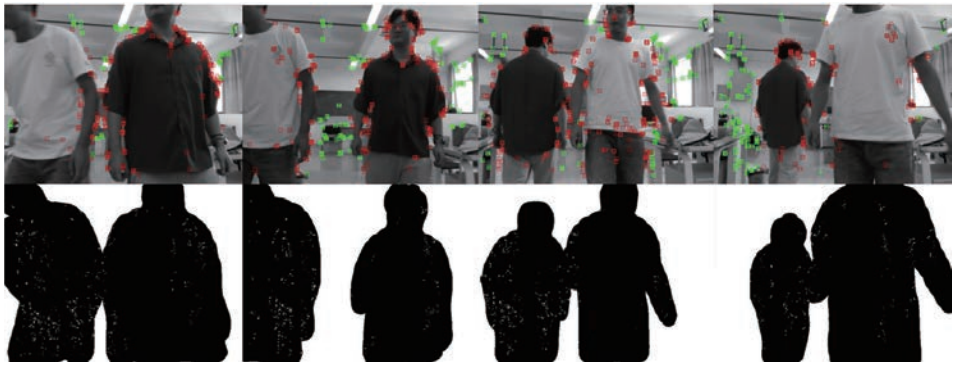


图 7　SUI-SLAM 算法在真实数据集的动静态特征点及其对应权重估计结果

Fig.7　SUI-SLAM dynamic and static feature points and their corresponding weight estimation results on real world sequences

图 8 为 SUI-SLAM 及 DynaSLAM 算法在实际室内场景下的关键帧运行轨迹。经对比，在相机运动过程中，SUI-SLAM 相比于 DynaSLAM 能够更准确地估计关键帧位姿，避免剔除过多动态特征点使得跟踪目标丢失。

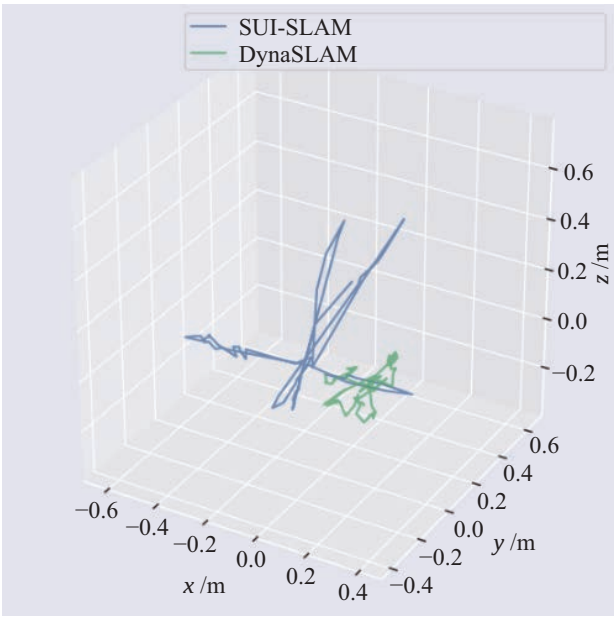


图 8　SUI-SLAM 及 DynaSLAM 算法在真实数据集上的关键帧轨迹

Fig.8　Keyframe trajectories of SUI-SLAM and DynaSLAM on real world sequences

3.2　系统运算速度评估

为了验证本文算法的实时性，采用每帧图像的平均跟踪时间作为评估算法实时性的指标。表 5 给出了 SUI-SLAM 和 DynaSLAM 算法在同一实验平台下的运行结果。同时，表 6 分别统计了 SUI-SLAM 算法在 fr3 序列下语义分割、追踪和局部建图模块的平均消耗时间。在 DynaSLAM 算法中，图像分割以及背景修复过程中会占用一部分运行时间，而 SUI-SLAM 算法中将 Mask R-CNN 网络的图像分割过程和移动概率计算过程放在单独的线程，因此 SUI-SLAM 算法相比于 DynaSLAM 算法的图像处理时间更少，更能够提高系统的运行效率。在 fr3_walking_static 序列中，由于人的运动幅度及活动范围相比于其他 walking 序列更小，因此相比于DynaSLAM 算法，SUI-SLAM 算法在该序列下的平均跟踪时间提升效果不明显。但在其他 walking 序列中，SUI-SLAM 算法相比于 DynaSLAM 算法的平均跟踪时间最高可提升 47.90%。从表 6 中可以看出，SUI-SLAM 算法的语义分割模块的平均用时为 0.06 s，局部建图模块用时相对较长。总体来说，本文算法在保证准确性的前提下，基本能满足实时性的需求。

表 5　DynaSLAM 和 SUI-SLAM 算法运行时间对比

Tab.5　Uptime comparison between DynaSLAM and SUI-SLAM algorithms

fr3 序列	DynaSLAM /s	SUI-SLAM /s	提升幅度 /%
fr3_walking_rpy	0.3409	**0.2387**	29.98
fr3_walking_half	0.4605	**0.2399**	47.90
fr3_walking_static	0.4921	**0.4278**	13.07
fr3_walking_xyz	0.4518	**0.2523**	44.16

表 6　在 fr3 序列下 SUI-SLAM 算法各模块平均运行时间

Tab.6　Average running time of each module of the SUI-SLAM algorithm on the fr3 sequence

主要模块	用时 /s
语义分割模块	0.060
跟踪模块	0.041
局部建图模块	0.312

4　结论（Conclusion）

提出了融合语义分割及不确定度估计的动态 SUI-SLAM 算法，在 Mask R-CNN 语义分割网络获取动态物体的先验信息基础上，根据深度信息修

正分割区域边缘，结合移动先验信息以及几何误差计算像素点、3D 地图点对应的不确定度，在相机位姿求解过程中加入正则化项用于提升定位的准确度和鲁棒性。实验部分将 SUI-SLAM 算法分别与 ORB-SLAM2 算法和 DS-SLAM、DynaSLAM、CFP-SLAM、Blitz-SLAM 几种动态 SLAM 算法在 TUM 数据集上的运行结果进行对比。实验结果表明，本文提出的 SUI-SLAM 算法在动态环境下相机位姿估计精度相比于 ORB-SLAM2 算法有明显的提升，且相比于 DS-SLAM、DynaSLAM、CFP-SLAM 和 Blitz-SLAM 算法，SUI-SLAM 算法的精确度和鲁棒性也有一定程度的提高，实时性相比于 DynaSLAM 有所改善。在下一个阶段的工作中，将尝试采用物体检测网络和几何信息获取动态物体先验信息，进一步提升运行效率，同时加入其他传感器，以应对视觉退化场景下的位姿漂移等问题。

参考文献（References）

[1] QIN T, LI P L, SHEN S J. VINS-Mono: A robust and versatile monocular visual-inertial state estimator[J]. IEEE Transactions on Robotics, 2018, 34(4): 1004-1020.

[2] ENGEL J, KOLTUN V, CREMERS D. Direct sparse odometry [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(3): 611-625.

[3] 魏彤，李绪．动态环境下基于动态区域剔除的双目视觉 SLAM 算法[J]．机器人，2020，42(3)：336-345.
WEI T, LI X. Binocular vision SLAM algorithm based on dynamic region elimination in dynamic environment[J]. Robot, 2020, 42(3): 336-345.

[4] DAI W C, ZHANG Y, LI P, et al. RGB-D SLAM in dynamic environments using point correlations[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44(1): 373-389.

[5] 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.

[6] BARBER C B, DOBKIN D P, HUHDANPAA H. The quickhull algorithm for convex hulls[J]. ACM Transactions on Mathematical Software, 1996, 22(4): 469-483.

[7] SUN Y X, LIU M, MENG M Q H, et al. Invisibility: A moving-object removal approach for dynamic scene modelling using RGB-D camera[C]//IEEE International Conference on Robotics and Biomimetics. Piscataway, USA: IEEE, 2017: 50-55.

[8] ROTHER C, KOLMOGOROV V, BLAKE A. “GrabCut”: Interactive foreground extraction using iterated graph cuts[J]. ACM Transactions on Graphics, 2004, 23(3): 309-314.

[9] 张慧娟，方灶军，杨桂林．动态环境下基于线特征的 RGB-D 视觉里程计[J]．机器人，2019，41(1)：75-82.
ZHANG H J, FANG Z J, YANG G L. RGB-D visual odometry in dynamic environments using line features[J]. Robot, 2019, 41(1): 75-82.

[10] 艾青林，刘刚江，徐巧宁．动态环境下基于改进几何与运动约束的机器人 RGB-D SLAM 算法[J]．机器人，2021，43(2)：167-176.
AI Q L, LIU G J, XU Q N. An RGB-D SLAM algorithm for robot based on the improved geometric and motion constraints in dynamic environment[J]. Robot, 2021, 43(2): 167-176.

[11] 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. Piscataway, USA: IEEE, 2018: 1168-1174.

[12] 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.

[13] 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.

[14] HE K M, GKIOXARI G, DOLLÁR P, et al. Mask R-CNN [C]//IEEE International Conference on Computer Vision. Piscataway, USA: IEEE, 2017: 2980-2988.

[15] ZHONG F W, WANG S, ZHANG Z Q, et al. Detect-SLAM: Making object detection and SLAM mutually beneficial[C]//IEEE Winter Conference on Applications of Computer Vision. Piscataway, USA: IEEE, 2018: 1001-1010.

[16] XU B B, LI W B, TZOUMANIKAS D, et al. MID-Fusion: Octree-based object-level multi-instance dynamic SLAM[C]// International Conference on Robotics and Automation. Piscataway, USA: IEEE, 2019: 5231-5237.

[17] XIAO L H, WANG J G, QIU X S, 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.

[18] HU X G, ZHANG Y Z, CAO Z Z, et al. CFP-SLAM: A real-time visual SLAM based on coarse-to-fine probability in dynamic environments[C]//IEEE/RSJ International Conference on Intelligent Robots and Systems. Piscataway, USA: IEEE, 2022: 4399-4406.

[19] FAN Y C, ZHANG Q C, TANG Y L, et al. Blitz-SLAM: A semantic SLAM in dynamic environments[J]. Pattern Recognition, 2022, 121. DOI: 10.1016/j.patcog.2021.108225.

[20] 王梦瑶，宋薇．动态场景下基于自适应语义分割的 RGB-D SLAM 算法[J]．机器人，2023，45(1)：16-27.
WANG M Y, SONG W. An RGB-D SLAM algorithm based on adaptive semantic segmentation in dynamic environment[J]. Robot, 2023, 45(1): 16-27.

[21] STURM J, ENGELHARD N, ENDRES P, et al. A benchmark for the evaluation of RGB-D SLAM systems[C]//IEEE/RSJ International Conference on Intelligent Robots and Systems. Piscataway, USA: IEEE, 2012: 573-580.

[22] LI S L, LEE D. RGB-D SLAM in dynamic environments using static point weighting[J]. IEEE Robotics and Automation Letters, 2017, 2(4): 2263-2270.

作者简介：

张玮奇（1992–），女，博士，副教授。研究领域：计算机视觉，SLAM，机器人自主导航。

马宗方（1980–），男，博士，教授。研究领域：智能信息处理，智慧城市，机器视觉工业应用。