Journal of System Simulation

Volume 36 | Issue 4 Article 20

4-15-2024

Research on Dynamic Scene SLAM Based on Improved Object Detection

Lanxi Shi
School of Internet of Things Engineering , Jiangnan University , Wuxi 214100, China,
2066760176@qq.com

Wenxu Yan
School of Internet of Things Engineering , Jiangnan University , Wuxi 214100, China,
ywx01@jiangnan.edu.cn

Hongyu Ni
State Grid Shaoxing Power Supply Company, Shaoxing 312000, China

Feng Zhao
State Grid Shaoxing Power Supply Company, Shaoxing 312000, China

Research on Dynamic Scene SLAM Based on Improved Object Detection

Abstract

Abstract: Aiming at the epipolar constraint matching problem of monocular SLAM in dynamic scenes a dynamic feature point selection method based on object detection is proposed, in which the dynamic feature points in the front-end image frame of SLAM system is eliminated during feature extraction to improve the localization accuracy of SLAM. An improved target detection network is proposed to construct a loss function to describe the bounding box by using the overlap area, distance similarity and cosine similarity, which can achieve the accurate localization of target objects and obtain the range of object feature points in the current image frame. The object category is judged in SLAM, and the dynamic feature points in the front-end image frame are rejected according to the target detection result for the objects marked as dynamic. Based on the static feature point results, the epipolar geometry is used for the feature matching between two frames to estimate pose the to carry out the tracking, map building and closed-loop detection of monocular camera motion. The speed of the inference process is improved by the structural reparameterization of the backbone of target detection network to ensure the real-time operation of the overall system. Experimental results on KITTI dataset show that the improved system improves the localization accuracy by 23.4% over ORB-SLAM3 system, and the frame rate can reach more than 30fps. The algorithm can effectively improve the localization accuracy of monocular SLAM system in dynamic scenes under the condition of ensuring the real-time operation.

Keywords

visual SLAM, epipolar geometry, feature matching, object detection, IoU loss function, structural reparameterization

Recommended Citation

Shi Lanxi, Yan Wenxu, Ni Hongyu, et al. Research on Dynamic Scene SLAM Based on Improved Object Detection[J]. Journal of System Simulation, 2024, 36(4): 1028-1042.

第 36 卷第 4 期 系统仿真学报© Vol. 36 No. 4
2024 年 4 月 Journal of System Simulation Apr. 2024

基于改进目标检测的动态场景SLAM研究

史蓝兮[1]，颜文旭[1*]，倪宏宇[2]，赵峰[2]
(1. 江南大学 物联网工程学院，江苏 无锡 214100；2. 国网绍兴供电公司，浙江 绍兴 312000)

摘要：针对单目SLAM在动态场景下存在的对极约束误匹配问题，*提出一种基于目标检测的动态特征点选择方法*，通过在特征提取时剔除SLAM系统前端图像帧中动态特征点，提高SLAM的定位精度。*提出了一个改进的目标检测网络，利用重叠面积、距离相似度和余弦相似度构建描述边界框的回归损失函数*，实现目标的准确定位，获得当前图像帧中物体特征点范围。判断物体类别，对于标记为动态的物体根据目标检测结果剔除前端图像帧中的动态特征点。根据静态特征点，采用对极约束进行两帧图像间的特征匹配估计位姿，对单目相机运动进行跟踪、建图与闭环检测。通过对目标检测网络的主干进行结构重参数化改进，提升推理过程的速度，保证整体系统运行的实时性。在公开数据集KITTI的11个序列上的实验结果表明：改进后的系统比ORB-SLAM3系统定位精度提升了23.4%，帧率可以达到30 帧/s以上，在保证实时运行的条件下能有效提高动态场景下单目SLAM系统定位精度。

关键词：视觉SLAM；对极约束；特征匹配；目标检测；IoU损失函数；结构重参数化
中图分类号：TP391.9；TP249　　文献标志码：A　　文章编号：1004-731X(2024)04-1028-15
DOI: 10.16182/j.issn1004731x.joss.22-1332
引用格式: 史蓝兮, 颜文旭, 倪宏宇, 等. 基于改进目标检测的动态场景SLAM研究[J]. 系统仿真学报, 2024, 36(4): 1028-1042.
Reference format: Shi Lanxi, Yan Wenxu, Ni Hongyu, et al. Research on Dynamic Scene SLAM Based on Improved Object Detection[J]. Journal of System Simulation, 2024, 36(4): 1028-1042.

Research on Dynamic Scene SLAM Based on Improved Object Detection

Shi Lanxi[1], *Yan Wenxu*[1*], *Ni Hongyu*[2], *Zhao Feng*[2]
(1. School of Internet of Things Engineering , Jiangnan University , Wuxi 214100, China;
2. State Grid Shaoxing Power Supply Company, Shaoxing 312000, China)

Abstract: Aiming at the epipolar constraint matching problem of monocular SLAM in dynamic scenes *a dynamic feature point selection method based on object detection is proposed,* in which the dynamic feature points in the front-end image frame of SLAM system is eliminated during feature extraction to improve the localization accuracy of SLAM. *An improved target detection network is proposed to construct a loss function to describe the bounding box by using the overlap area*, distance similarity and cosine similarity, which can achieve the accurate localization of target objects and obtain the range of object feature points in the current image frame. The object category is judged in SLAM, and the dynamic feature points in the front-end image frame are rejected according to the target detection result for the objects marked as dynamic. Based on the static feature point results, the epipolar geometry is used

收稿日期：2022-11-09　　修回日期：2023-01-06
基金项目：国网浙江省电力有限公司科技项目(5211SX220003)
第一作者：史蓝兮(1997-)，女，硕士生，研究方向为视觉SLAM。E-mail：2066760176@qq.com
通讯作者：颜文旭(1971-)，男，教授，博士，研究方向为电力电子及智能控制、电力系统及其自动化、电力特种机器人和智能装备。E-mail：ywx01@jiangnan.edu.cn

for the feature matching between two frames to estimate pose the to carry out the tracking, map building and closed-loop detection of monocular camera motion. The speed of the inference process is improved by the structural reparameterization of the backbone of target detection network to ensure the real-time operation of the overall system. Experimental results on KITTI dataset show that the improved system improves the localization accuracy by 23.4% over ORB-SLAM3 system, and the frame rate can reach more than 30fps. The algorithm can effectively improve the localization accuracy of monocular SLAM system in dynamic scenes under the condition of ensuring the real-time operation.

Keywords: visual SLAM; epipolar geometry; feature matching; object detection; IoU loss function; structural reparameterization

0 引言

视觉SLAM是指在没有环境先验信息的情况下，利用主体搭载的相机提取周围环境信息以建立地图模型并估计主体的运动[1-2]。其中，单目视觉SLAM由于单目相机的广泛应用而受到关注[3]。单目视觉测量与双目视觉测量相比，具有系统结构简单、视场范围大等优点[4]。但是，单目SLAM估计得到的轨迹和地图，与真实的轨迹和地图之间相差一个因子，即尺度。因此单目SLAM无法通过图像确定真实的尺度，称为“尺度不确定”。为固定尺度，系统只能根据单目运动的视差通过对极约束求解深度进行定位[5]。但是当单目相机受环境和运动物体影响时，图像中特征点间的匹配会发生误差，影响定位精度。剔除对极约束误匹配的特征点，使用静态特征点进行定位是提高定位精度的前提。

剔除误匹配特征点的方法可分为4类：基于图论的方法、基于块理论的方法、基于特征点几何坐标信息的重采样法以及结合深度学习的方法[6-7]。基于图论的方法模型复杂，仅适用于实时性要求不高的任务。基于块理论的方法仅从局部上剔除不一致的匹配，缺少全局约束。基于特征点几何坐标信息的重采样法通过计算模型筛选最大静态特征点子集。文献[8]先设定最小阈值对特征点粗匹配，再采用随机抽样一致(random sample consensus, RANSAC)的方法筛选静态特征点，但是RANSAC受动态特征点数量影响，特征点数量越多模型的迭代次数越多。随着深度学习的发展，通过深度学习对环境中的目标进行检测，获取目标类别和预测框数据，在SLAM系统中对目标判断并剔除的方法应用比较广泛[9-11]。

为准确剔除影响定位精度的特征点，需要提高目标定位精度。回归损失函数的设定直接影响定位精度和收敛速度[12]。卷积神经网络在评价边界框回归效果时使用了表示重叠面积的交并比(IoU)。但当预测框与真值框不重叠时，损失函数无法准确描述两框的距离。文献[13]提出GIoU损失函数，增加了对最小闭包区的计算。但当预测框与真值框互相包含时，该损失函数失效，退化为IoU损失函数。文献[14]提出CIoU损失函数，增加了长宽比的计算。由于该函数仅表示出长宽比差异而没有明确定义，所以在边界框回归问题中，仍存在收敛速度慢、回归结果不准确的缺点。在SLAM与深度学习结合时，实时性也是重要问题。NAS[15]、NASNet[16]等特征提取网络虽然提高了性能，但是处理速度慢。因此，这些方法不适用于室外动态场景，尤其是多目标检测任务。文献[17]提出的特征提取网络RepVGG使用多分支结构进行训练，然后，基于结构重参数化将其转换为单分支结构进行推理。该方法除了具有推理过程中多分支模型性能的优点外，还具有与单分支模型相似的推理时间较短的优点。为了减少目标检测提取动态特征点对SLAM系统实时性的影响，基于结构重参数化改进YOLOv5s的主干，通过提高目标检测效率，提高SLAM系统的实时性。

DOI: 10.16182/j.issn1004731x.joss.22-1332

第 36 卷第 4 期 系统仿真学报 Vol. 36 No. 4
2024 年 4 月 Journal of System Simulation Apr. 2024

本文提出一种基于目标检测的动态特征点选择方法，通过在特征提取时剔除SLAM系统前端图像帧中动态特征点，提高单目SLAM在动态场景的定位精度。为了提高定位精度，本文考虑了重叠面积，距离相似度和余弦相似度三个要素。

1 系统结构

ORB-SLAM3是目前优秀的特征点SLAM算法之一，包含跟踪线程，局部建图线程和闭环检测线程等[18]，系统框架如图1所示。跟踪线程包括ORB(oriented FAST and rotated BRIEF)特征点的提取和匹配，对极约束等几何计算方式估计帧间相对位姿；局部建图线程通过非线性优化来优化局部地图帧的位姿；闭环检测线程通过关键帧检测全局地图的闭环并校正误差。

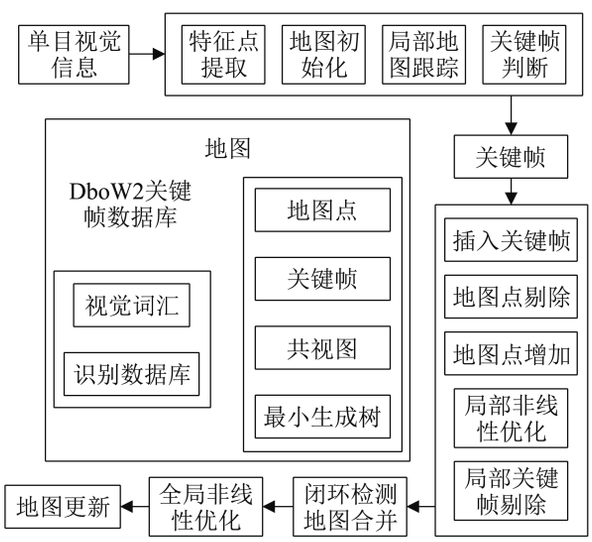


图1 ORB-SLAM3结构
Fig. 1 Structure of ORB-SLAM3 system

为解决单目SLAM系统在动态环境下产生的对极约束误匹配的问题，本文在处理前端图像帧的跟踪线程中加入了YOLOv5s识别动态目标的特征点。为准确识别到特征点，构建了包含重叠面积、距离相似度和余弦相似度的回归损失函数。为保证系统的图像处理实时性，改进了主干网络，提高了目标检测的推理计算速度，即结构重参数化。改进后的系统中，首先由单目相机采集图像信息，完成特征点的提取。同时，通过YOLOv5s目标检测网络识别目标并实时地传输目标分类和目标边界框信息给SLAM系统。在SLAM系统中对目标分类进行判断。如果该目标分类是动态目标，如人和车辆，则剔除该目标边界框中的特征点。为防止剩余特征点过少，对当前帧特征点数目进行判断，大于设定的数目则进行初始化。然后，进一步用对极约束的阈值剔除掉可疑特征点，得到静态特征点的特征匹配，用于估计初始位姿进行下一步跟踪。在后端，对系统进行非线性优化。最后，进行闭环检测和建图。本文算法对跟踪线程的改进结构如图2所示。

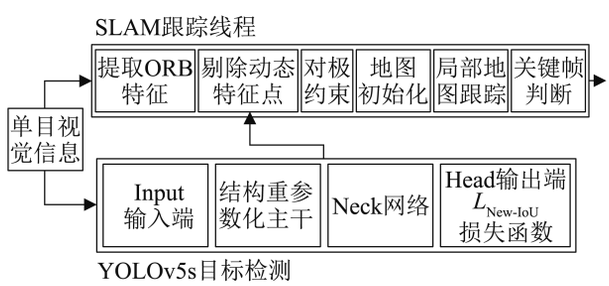


图2 改进的跟踪线程
Fig. 2 Improved trace threads

2 问题分析

单目SLAM中为固定尺度，通过对极约束定义的本质矩阵求解帧间相对位姿，原理如图3所示。图中，I_1和I_2表示连续两帧图像的像平面，p_1和p_2表示特征匹配后观测目标点P在前后两帧的成像点，O_1和O_2表示两帧每一帧对应的相机中心点。O_1，O_2和P确定一个极平面。O_1和O_2的连接线为基线。基线与像平面的交点为极点，即e_1和e_2。极平面与像平面的交线为极线，即l_1和l_2。

对极约束要求帧间匹配的特征点位于当前帧的极线上[19]，那么需要设定一个阈值判断是否在极线范围内。设I_1到I_2的基础矩阵为$\boldsymbol{F}$，设(u, v)为p_1投影点的像素坐标，极线l_2表示为

$$ax+by+c=0 \tag{1}$$

$$[a \quad b \quad c]^{\mathrm{T}} = \boldsymbol{FK}[u_1 \quad v_1 \quad 1] \tag{2}$$

式中：$\boldsymbol{K}$为相机内参矩阵。

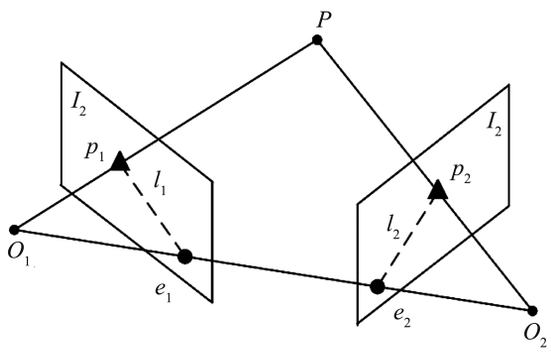


图3 对极约束的原理图
Fig. 3 Schematic of epipolar geometry

特征点到极线l_2距离的平方为

$$d^2 = \frac{(au_2 + bv_2 + c)^2}{a^2 + b^2} \tag{3}$$

由式(3)能得到d^2服从卡方分布。在视觉SLAM中，为获得不同分辨率的图像，用金字塔对图像进行不同层次的降采样。在金字塔第n层，取95%置信度时，d^2的约束条件为

$$d^2 < 3.84 \times 1.2^{2n} \tag{4}$$

如果当前特征点的d^2在阈值内，则进行下一步求解位姿，不在阈值内则舍弃该点。根据针孔相机模型，可以得到p_1和p_2的位置公式：

$$\begin{cases} p_1 = \boldsymbol{K}P \\ p_2 = \boldsymbol{K}(\boldsymbol{R}P + \boldsymbol{t}) \end{cases} \tag{5}$$

式中：$\boldsymbol{R}$为旋转矩阵；$\boldsymbol{t}$为平移向量，设$\boldsymbol{t}$和$\boldsymbol{t}^\wedge$互为反对称矩阵。

设x_1和x_2为p_1和p_2在归一化平面上的坐标：

$$x_2 = \boldsymbol{R}x_1 + \boldsymbol{t} \tag{6}$$

对式(6)两边左乘反对称矩阵$\boldsymbol{t}^\wedge$：

$$\boldsymbol{t}^\wedge x_2 = \boldsymbol{t}^\wedge \boldsymbol{R}x_1 \tag{7}$$

式(7)左侧与x_2和t方向垂直，为消去x_2，左乘x_2^{T}：

$$x_2^{\mathrm{T}} \boldsymbol{t}^\wedge x_2 = x_2^{\mathrm{T}} \boldsymbol{t}^\wedge \boldsymbol{R}x_1 = 0 \tag{8}$$

代入(5)，得到对极约束表达式：

$$p_2^{\mathrm{T}} \boldsymbol{K}^{-\mathrm{T}} t^\wedge \boldsymbol{R}\boldsymbol{K}^{-1} p_1 = 0 \tag{9}$$

$\boldsymbol{R}$和$\boldsymbol{t}$表示世界坐标系和相机坐标系之间的转换。由式(9)可知，通过对极约束可以求解$\boldsymbol{R}$和$\boldsymbol{t}$，即当前位姿。

在实际应用中，相机和动态物体都是运动的，对极约束无法区分这类动态特征点，造成误匹配[20]。设视野中存在一个动态目标，P点为该动态目标的位置，则动态目标干扰下的对极约束如图4所示。设动态目标从P移动到P_{dyna}，则极平面由P_0、O_1和P_{dyna}组成。最后得到结果是O_1到O_{dyna}的位姿，不是O_1到O_2的位姿，从而产生错误的相机位姿解算结果。针对动态场景，本文通过目标检测对动态目标进行识别，剔除干扰特征点，进而降低误匹配对定位精度的影响。

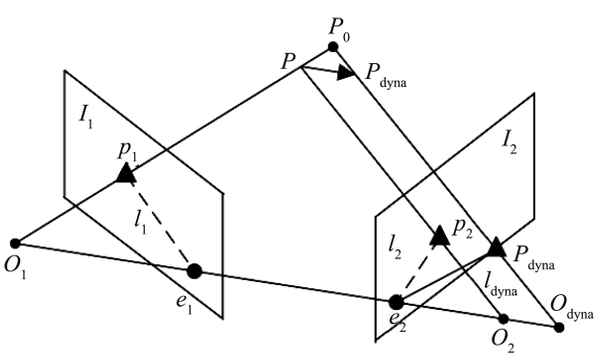


图4 动态目标对对极约束的干扰
Fig. 4 Interference of dynamic target to epipolar geometry

3 动态特征点的剔除

3.1 算法设计

本文系统通过Socket接口实现双线程并行模式。目标检测在YOLOv5s中实现，目标分类在SLAM跟踪线程中实现。Socket是用户层应用程序访问TCP/IP协议层的编程接口，可以在用户层实现同一主机中2个应用程序间的数据交换，该网络结构如图5所示。

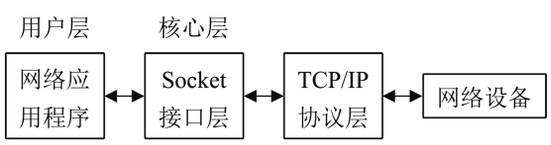


图5 网络结构
Fig. 5 Network structure

剔除动态特征点算法流程如算法1所示。

算法1：剔除动态特征点算法

输入：当前帧F_{cur}，历史帧F_{last}，当前帧特征点数量N_{cur}，阈值T_1，阈值T_2

输出：保留静态特征的图像帧F_{sta}

```
begin
for  F_cur_i in F_last
提取F_cur中特征点//SLAM线程
对F_cur进行目标识别//YOLOv5s线程
传输B_coo、B_cla、B_con到SLAM线程
if  B_con>T_1
    continue
if  B_cla= “person”“car”“bicycle” etc then
    删除B_coo内特征点
    if N_cur>T_2
    continue
  return F_sta
end
```

3.2 边界框回归

在目标检测中，回归损失函数的设定直接影响目标的定位精度。目标检测通常用交并比(IoU)衡量2个边界框的相似性：

$$q_{\mathrm{IoU}}=\frac{|\mathrm{A}\cap\mathrm{B}|}{|\mathrm{A}\cup\mathrm{B}|} \tag{10}$$

YOLOv5s采用CIoU损失函数计算边界框坐标的回归损失。该损失函数包含了重叠面积、中心点距离和长宽比：

$$L_{\mathrm{CIoU}}=1-q_{\mathrm{IoU}}+\frac{\rho^2(b,b_{\mathrm{gt}})}{d_{\mathrm{E}}^2}+\alpha v \tag{11}$$

式中：b为预测框的中心；b_{gt}为真值框的中心；d_{E}为最小闭包区的对角线距离；ρ为预测框和真值框的中心点距离；α为正平衡参数；v用于测量预测框和真值框的长宽比一致性。

$$\begin{cases} v=\dfrac{4}{\pi^2}\left(\arctan\dfrac{w_{\mathrm{gt}}}{h_{\mathrm{gt}}}-\arctan\dfrac{w}{h}\right)^2 \\ \alpha=\dfrac{v}{(1-q_{\mathrm{IoU}})+v} \end{cases} \tag{12}$$

与其他IoU损失函数相比，CIoU损失函数的收敛速度和检测精度都有提高[13-14]。但是，仍存在2个问题：

(1) v没有明确的定义，只是表示长宽比的差异，而不是w与w_{gt}或h与h_{gt}之间的真实关系。

(2) 当v对w和h求偏导数，可得

$$\begin{cases} \dfrac{\partial v}{\partial w}=\dfrac{8}{\pi^2}\left(\arctan\dfrac{w_{\mathrm{gt}}}{h_{\mathrm{gt}}}-\arctan\dfrac{w}{h}\right)\dfrac{h}{w^2+h^2} \\ \dfrac{\partial v}{\partial h}=-\dfrac{8}{\pi^2}\left(\arctan\dfrac{w_{\mathrm{gt}}}{h_{\mathrm{gt}}}-\arctan\dfrac{w}{h}\right)\dfrac{w}{w^2+h^2} \end{cases} \tag{13}$$

即有$\dfrac{\partial v}{\partial w}=-\dfrac{h}{w}\cdot\dfrac{\partial v}{\partial h}$，式中两个偏导数符号相反，那么变量之一增长，另一个下降，降低了CIoU的收敛速度。

基于以上问题的分析，本文提出能准确定义距离的计算部分，即距离相似度，并且对角度方面进行计算用于描述余弦相似度。假设预测框和真值框以及它们的最小闭包区在像素坐标系中的位置如图6所示。预测框P的顶点坐标由x_1、x_2、y_1、y_2表示，真值框P_{gt}由x_1^{gt}、x_2^{gt}、y_1^{gt}、y_2^{gt}表示，虚线框为两框最小闭包区。

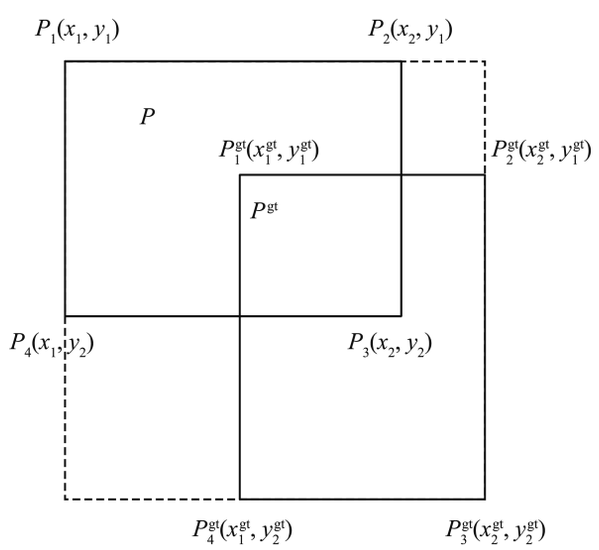


图6 距离对损失函数的影响

Fig. 6 Impact of distance on loss function

根据图6的坐标计算边界框点之一的P_1和P_1^{gt}

的距离，并对预测框坐标之一$P_1(x_1, y_1)$求偏导数，得到距离和求偏导后公式：

$$L_{\mathrm{d}}=\sum_{i=1}^{4}\frac{\rho^2(P_i,P_i^{\mathrm{gt}})}{d_{\mathrm{E}}^2}=\frac{1}{2}\times\frac{(x_1-x_1^{\mathrm{gt}})^2+(x_2-x_2^{\mathrm{gt}})^2+(y_1-y_1^{\mathrm{gt}})^2+(y_2-y_2^{\mathrm{gt}})^2}{d_{\mathrm{E}}^2} \tag{14}$$

$$\begin{cases}\dfrac{\partial L_{\mathrm{d}}}{\partial x_1}=\dfrac{1}{d_{\mathrm{E}}^2}(x_1-x_1^{\mathrm{gt}})\\ \dfrac{\partial L_{\mathrm{d}}}{\partial y_1}=\dfrac{1}{d_{\mathrm{E}}^2}(y_1-y_1^{\mathrm{gt}})\end{cases} \tag{15}$$

由式(15)可知，当偏导数为0，式(14)可以得到极值，即得到准确的收敛方向，同理可得到其他坐标点的收敛方向。

仅依靠距离相似度不能完全描述两框的位置。余弦相似度用于向量空间中两个向量夹角的余弦值，比较两个个体间的差异大小。在图7中，由坐标原点O到预测边界框的中心点O_1和真值边界框的中心点O_2，表示向量$\boldsymbol{OO}_1$和向量$\boldsymbol{OO}_2$。通过求向量$\boldsymbol{OO}_1$和向量$\boldsymbol{OO}_2$之间的夹角θ余弦值表示边界框的相似度，夹角θ取值范围在0~90°，如图8所示。通过余弦相似度的计算，模型会沿着角度减小的方向进行优化。

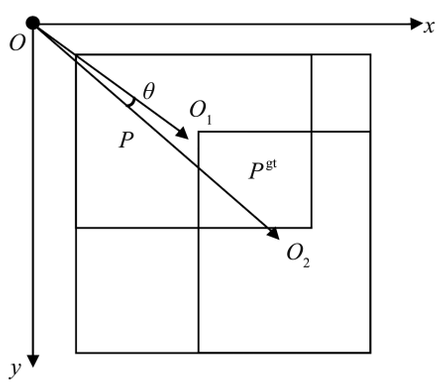


图7 角度对损失函数的影响
Fig. 7 Impact of angle on loss function

对余弦相似度进行详细推导，设$\boldsymbol{a}=(x_1,y_1)$，$\boldsymbol{b}=(x_2,y_2)$，余弦相似度的几何定义为

$$\cos\theta=\frac{\boldsymbol{a}\cdot\boldsymbol{b}}{|\boldsymbol{a}||\boldsymbol{b}|}=\frac{x_1x_2+y_1y_2}{\sqrt{x_1^2+y_1^2}\times\sqrt{x_2^2+y_2^2}} \tag{16}$$

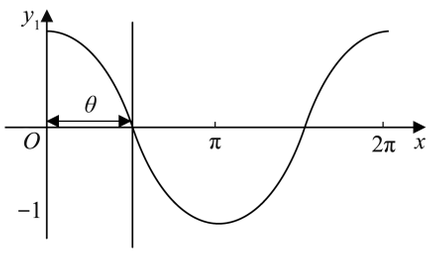


图8 余弦相似度取值范围
Fig. 8 Range of cosine similarity values

根据图7和图8可得

$$\begin{cases}\boldsymbol{OO}_1=((x_1+x_2)/2,\ (y_1+y_2)/2)\\ \boldsymbol{OO}_2=((x_1^{\mathrm{gt}}+x_2^{\mathrm{gt}})/2,\ (y_1^{\mathrm{gt}}+y_2^{\mathrm{gt}})/2)\end{cases} \tag{17}$$

由此得到

$$L_{\cos\theta}=\cos(b,b^{\mathrm{gt}})=\frac{\dfrac{x_1+x_2}{2}\times\dfrac{x_1^{\mathrm{gt}}+x_2^{\mathrm{gt}}}{2}+\dfrac{y_1+y_2}{2}\times\dfrac{y_1^{\mathrm{gt}}+y_2^{\mathrm{gt}}}{2}}{\sqrt{\left(\dfrac{x_1+x_2}{2}\right)^2+\left(\dfrac{y_1+y_2}{2}\right)^2}\times\sqrt{\left(\dfrac{x_1^{\mathrm{gt}}+x_2^{\mathrm{gt}}}{2}\right)^2+\left(\dfrac{y_1^{\mathrm{gt}}+y_2^{\mathrm{gt}}}{2}\right)^2}}=\frac{(x_1+x_2)(x_1^{\mathrm{gt}}+x_2^{\mathrm{gt}})+(y_1+y_2)(y_1^{\mathrm{gt}}+y_2^{\mathrm{gt}})}{\sqrt{(x_1+x_2)^2+(y_1+y_2)^2}\sqrt{(x_1^{\mathrm{gt}}+x_2^{\mathrm{gt}})^2+(y_1^{\mathrm{gt}}+y_2^{\mathrm{gt}})^2}} \tag{18}$$

最终得到损失函数：

$$L_{\mathrm{New-IoU}}=1-q_{\mathrm{IoU}}+\sum_{i=1}^{4}\frac{\rho^2(P_i,P^{\mathrm{gt}})}{d_{\mathrm{E}}^2}+\lambda\cos(b,b^{\mathrm{gt}}) \tag{19}$$

式中：λ取经验值0.01。

在式(19)中，将IoU作为边界框回归的损失项。在回归过程中，由于IoU具有尺度不变性，这能让不同尺度的边界框获得平均的优化权重。在IoU损失项基础上添加了距离相似度损失项。通过此项描述长宽，而不是CIoU中长宽比的差异。最后在此基础上添加惩罚项，通过最小化预测框与真值框中心点向量间的余弦值，加快预测框中心点与真值框中心点的重叠速度。

图9表示了预测框P和真值框P^{gt}相交的情况和两框相互包含的情况。针对相交的情况，距离相似度起主要作用，对预测框进行优化。针对包含的情况，余弦相似度起到更大的作用，避免损失函数退化为IoU。

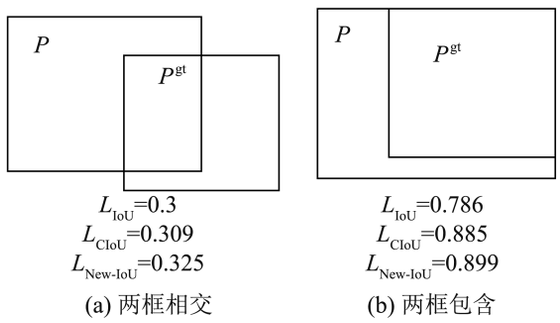


图9 预测框与真值框的位置
Fig. 9 Location of anchor box and target box

3.3 结构重参数化主干

结合深度学习的方法分为以YOLO为代表的目标检测法和以Mask R-CNN为代表的语义分割法。Mask R-CNN在CPU中每帧运行速度超过2 s，难以实时运行[21-22]。通过改进目标检测网络YOLOv5s能够满足实时性的要求。在目标检测框架中，YOLOv5s模型参数量最少，浮点运算量最低，运行速度最快。推理时间和非极大值抑制处理时间是目标检测耗时的主要部分，直接影响SLAM系统对每一帧的跟踪。由于非极大值抑制处理时间与边界框回归相关，所以，本文从推理时间进行改进。YOLOv5s的结构如图10所示。

与RepVGG相比，BottleneckCSP模块有一个更复杂的架构和更多的分支。它通过融合不同域的特征图获得更全面的特征信息。虽然BottleneckCSP模块的结构比较复杂，但通过结构重参数化，可将其转换为一个3×3的计算。

合并相同分支的卷积，即在同一分支上的1×1和3×3卷积：

$$F_{\text{out}}=(F_{\text{in}}\otimes F_{1\times1}+b_{1\times1})\otimes F_{3\times3}+b_{3\times3} \tag{20}$$

式中：$F_{1\times1}$为1×1的卷积核；$b_{1\times1}$为1×1的卷积对应的偏差；$F_{3\times3}$为3×3的卷积核；$b_{3\times3}$为3×3卷积对应的偏差；$\otimes$为卷积运算；F_{in}为输入；F_{out}为卷积操作后的输出。

合并不同分支的卷积：

$$\begin{cases}F_{\text{out},i}=F_{\text{in}}\otimes F_i+b_i, & i=1,2,\cdots,n\\ F_{\text{out}}=F_{\text{out},1}+F_{\text{out},2}+\cdots+F_{\text{out},n}\end{cases} \tag{21}$$

卷积运算的可加性使具有相同形式的卷积运算可以合并成一个：

$$\begin{cases}F'=F_1+F_2+\cdots+F_n\\ b'=b_1+b_2+\cdots+b_n\end{cases} \tag{22}$$

可以将不同分支的3×3卷积合并为一个分支：

$$F_{\text{out}}=F'+b' \tag{23}$$

结构化参数重构过程的原理如图11所示。

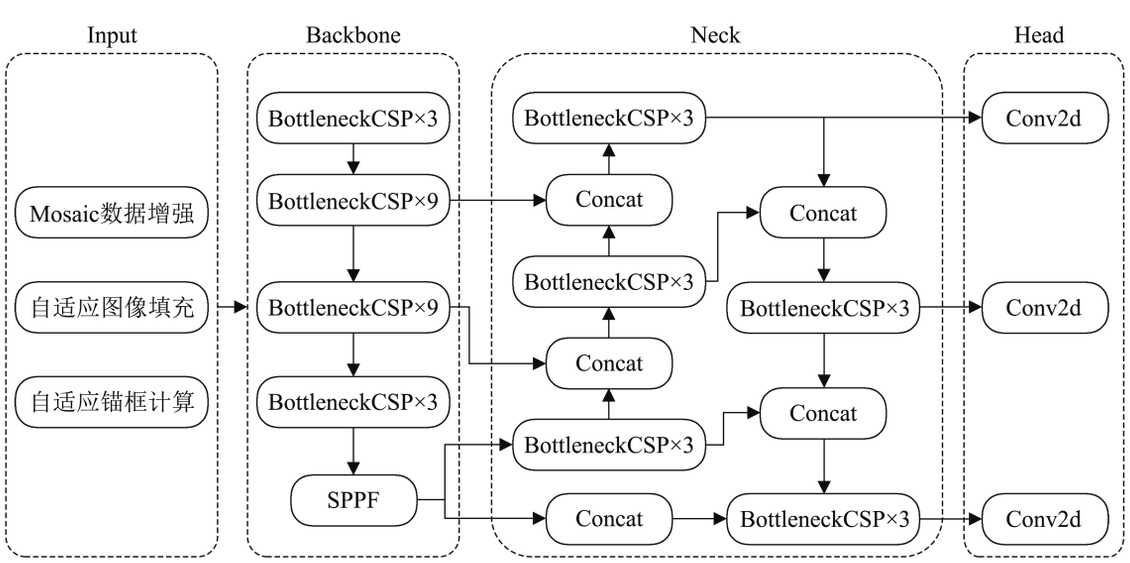


图10 YOLOv5s网络结构
Fig. 10 Structure of YOLOv5s

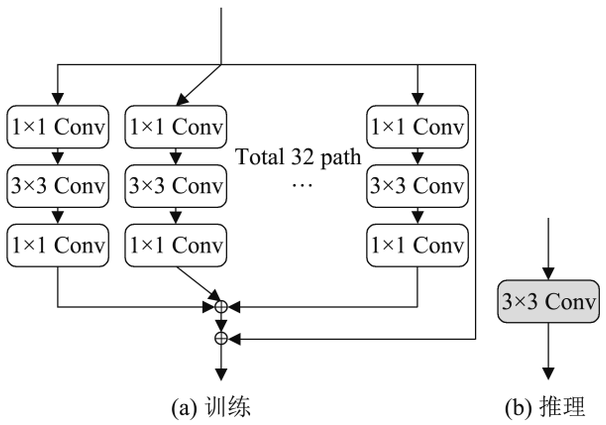


图11 改进后的BottleneckCSP模块
Fig. 11 Improved BottleneckCSP module

在本文研究的环境中，存在多目标的情况，如果使用复杂的多分支主干提取特征，检测速度会受到很大影响，SLAM系统中不能实时跟踪。通过对主干进行结构重参数化，在训练过程中，使用多分支结构来充分提取目标的特征并在推理过程中将其转换为单分支结构，能够在不影响检测精度的情况下提高检测速度。

4 实验结果与分析

本文算法在公开数据集KITTI中的11个视频序列展开实验，针对单目模式下的视觉SLAM系统进行评估[23]。实验环境配置为6核i7处理器，16 GB内存，RTX 2060，操作系统Ubuntu18.04。

4.1 目标检测效果对比

表1数据基于目标检测网络YOLOv5s，损失函数部分为L_{CIoU}和本文的$L_{\text{New-IoU}}$，分别在MS COCO 2017的训练集中进行性能评估。实验采用平均精度反映每类目标的检测效果。平均精度是用准确率和召回率来衡量检测算法的准确性，直观表现了模型对单个类别的检测效果。采用平均精度均值来衡量多类目标的平均检测精度。mAP值越高表示模型在全部类别中综合性能越高。本文的方法比CIoU损失函数在平均精确率mAP中上涨1.7个百分点。

表1 训练结果
Table 1 Training results

损失函数	准确率	召回率	mAP@.5	mAP@.5:.95
L_{CIoU}	0.952	0.893	0.957	0.837
$L_{\text{New-IoU}}$	0.956	0.980	0.974	0.856

YOLOv5训练可视化结果如图12~13所示。该曲线表示边界框损失函数均值，值越小证明边界框越准。

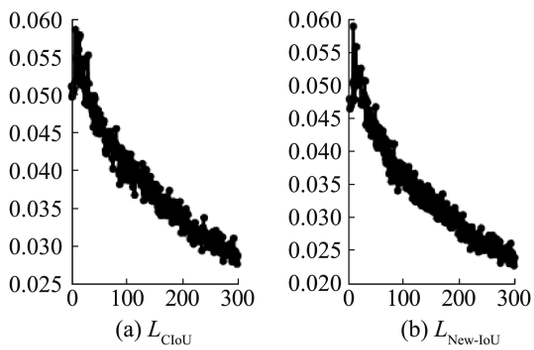


图12 训练集损失函数曲线
Fig. 12 Loss function plot for training set

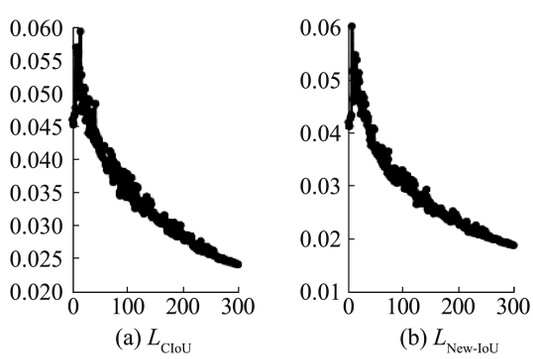


图13 验证集损失函数曲线
Fig. 13 Loss function plot for validation set

在训练集中选择以下几个动态目标的识别平均精度进行对比，如图14所示。

4.2 特征点剔除效果

本文研究的KITTI数据集中城市道路和高速公路场景序列属于复杂环境。在11个序列中，01、03、04和10是开环序列。09中，虽然车辆以相同的方向重新访问起点，但重叠轨迹很短导致视觉闭环检测无法找到起点。在该数据集中，动态目标数量统计如图15所示。

第 36 卷第 4 期 系统仿真学报 Vol. 36 No. 4
2024 年 4 月 Journal of System Simulation Apr. 2024

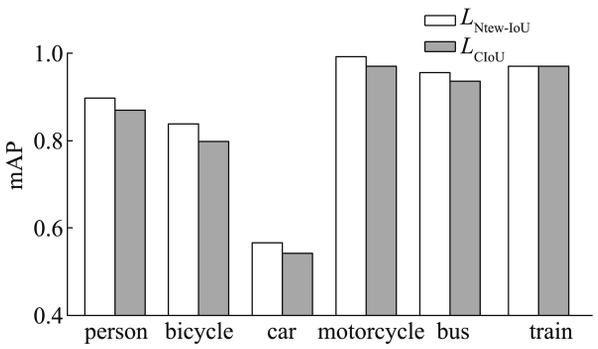


图 14 不同算法的 mAP 对比
Fig. 14 Comparison of mAP in different loss functions

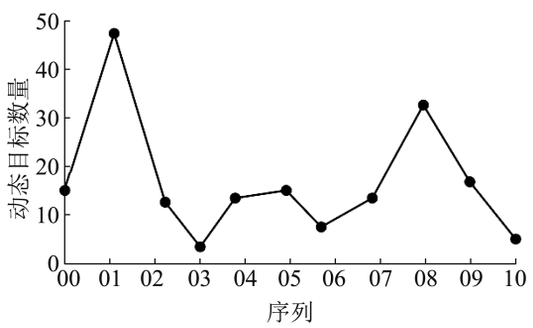
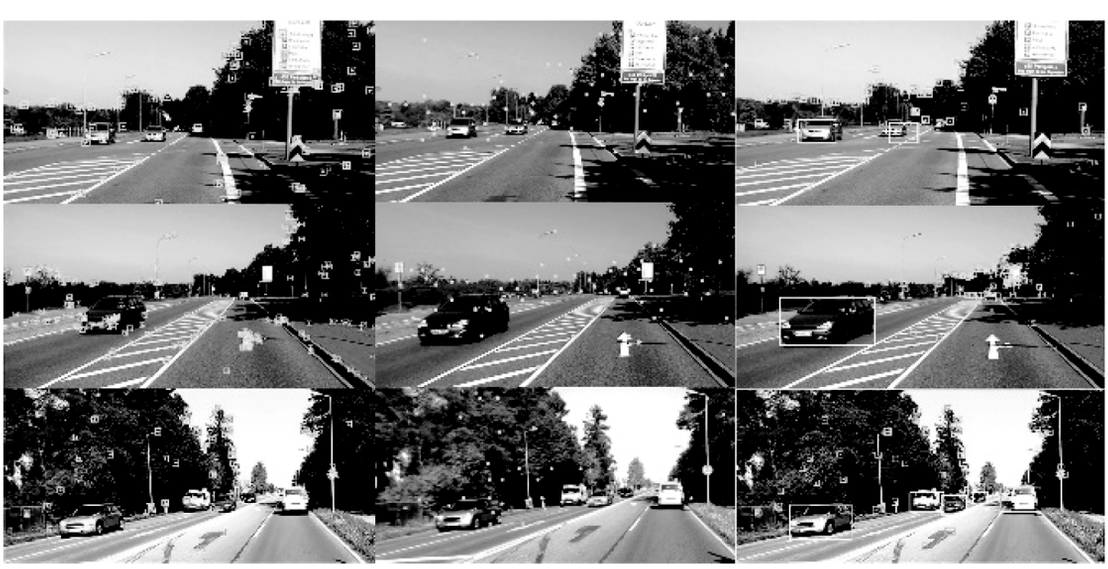


图 15 KITTI 数据集中动态目标数量统计
Fig. 15 Number of dynamic targets in KITTI dataset

对比实验选择文献[24]中 VINS-Fusion 的 VSLAM 模式。VINS-Fusion 前端通过 LK 光流法提取动态目标附近的光流变化方向表示动态目标信息，并剔除干扰点。

本文改进的目标检测系统通过提出新的损失函数提高对目标的识别准确性，例如，在改进之前的目标检测系统中，运动的车辆不能在每一帧都被识别到，而改进后的系统在动态目标出现的连续几帧都被识别到了。进而能解决动态特征点干扰的问题，尤其是多个动态目标重叠的情况。通过剔除目标检测的信息相关动态特征点后，在原特征点提取数量不变的情况下，能有效提高定位精度。当前帧存在动态车辆和行人时，ORB-SLAM3 对动态目标不作处理，导致动态车辆和行人上的特征点一并进行位姿估计。本文算法剔除了动态车辆和行人边界框中的特征点，因为当前帧的前后几帧存在少量密集动态目标，所以边界框内除动态目标主体外的背景特征点不产生跟踪结果。

图 16~17 是 00、02、04、07 和 08 序列的部分仿真图片。图 16 中 ORB-SLAM3 和 VINS-Fusion 均选取了动态车辆上的部分特征点作为静态特征点进行计算。尤其在 ORB-SLAM3 中，近景车辆占有较多特征点。本文算法剔除了动态车辆上的特征点。当存在连续动态车辆影响时，本文算法也能剔除这一部分的特征点。



(a) ORB-SLAM3 (b) VINS-Fusion (c) 本文方法

图 16 对动态车辆的识别效果
Fig. 16 Recognition effect on dynamic vehicles

4.3 定位精度分析

本实验采用SLAM轨迹评估工具evo评价绝对轨迹误差和相对轨迹误差。绝对轨迹误差是指相机的位姿真值和SLAM算法的位姿估计值之间的差值，用于评估视觉SLAM系统的性能。相对轨迹误差是指当前帧的漂移。设待对比算法的精度误差为p，本文算法误差为q。精度提升率计算式

$$\text{提升率}=(p-q)/p\times 100\% \tag{24}$$

4.3.1 绝对轨迹误差和轨迹

表2对比了平均值、标准差和均方根误差。标准差表示数据偏离平均值的离散程度。均方根误差表示数据偏离真值的离散程度。同类中最优值在表中加粗表示。01高速公路场景下存在大量动态目标干扰和公路相似环境，导致ORB-SLAM3跟踪较差，仅跟踪部分轨迹。因此01的ORB-SLAM3部分标记“—”。

(a) ORB-SLAM3 (b) VINS-Fusion (c) 本文方法

图17 对行人的识别效果
Fig. 17 Recognition effect on people

表2 绝对轨迹误差对比
Table 2 Comparison of absolute trajectory error

序列	平均值			标准差			均方根误差		
	ORB-SLAM3	文献[24]	本文	ORB-SLAM3	文献[24]	本文	ORB-SLAM3	文献[24]	本文
00	0.513	13.238	**0.396**	0.286	5.222	**0.245**	0.587	15.165	**0.466**
01	—	6.505	2.021	—	2.631	1.407	—	7.017	2.462
02	0.735	9.155	**0.727**	**0.535**	6.015	0.618	**0.907**	10.594	0.954
03	0.042	0.865	**0.025**	0.042	0.267	**0.019**	0.059	0.906	**0.032**
04	0.043	0.247	**0.006**	0.022	0.126	**0.003**	0.048	0.277	**0.007**
05	**0.164**	5.997	0.185	**0.056**	3.093	0.075	**0.173**	6.748	0.200
06	0.513	3.157	**0.434**	0.286	1.786	**0.259**	0.587	3.627	**0.505**
07	0.229	1.353	**0.175**	0.114	1.515	**0.101**	0.256	2.484	**0.212**
08	4.770	4.913	**3.758**	3.743	3.432	**2.723**	6.063	5.993	**4.641**
09	0.608	4.866	**0.287**	0.570	3.953	**0.126**	0.834	6.269	**0.314**
10	**0.293**	3.158	0.379	**0.295**	1.778	0.331	**0.474**	3.753	0.503

图 18 是本文算法对比 ORB-SLAM3 和 VINS-Fusion 的绝对轨迹误差 RMSE 在不同序列中的提升率折线图。ORB-SLAM3 平均提升率为 23.4%，VINS-Fusion 平均提升率为 85.9%。

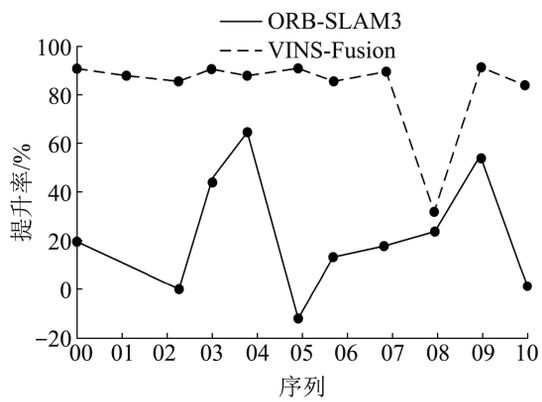
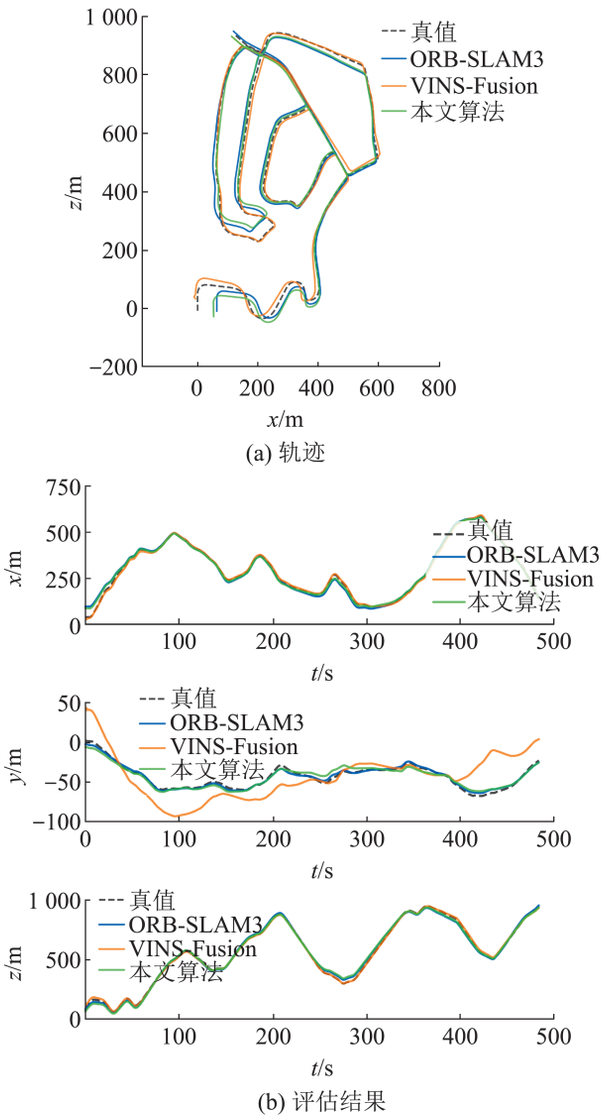


图 18　绝对轨迹误差的提升率

Fig. 18　Rate of improvement of absolute trajectory error

图 19~21 分别是序列 02、06 和 09 的轨迹。在 02 序列中，ORB-SLAM3 和本文算法在旋转时精度较差。02 序列中静态车辆较多，动态目标与静态目标比例约为 6.7%，而本文算法中由于在剔除动态特征点后，进行阈值判断导致删除了部分特征点数量不够的图像帧。因此，本文算法比 ORB-SLAM3 的绝对轨迹误差下降 5.2%。在 06 序列中，VINS-Fusion 的闭环效果较差。ORB-SLAM3 系统的轨迹在 z 方向最大和最小时，即在同一地点旋转角度过大时的漂移较大，而在本文算法中可以看到跟踪误差更小，比 ORB-SLAM3 的绝对轨迹误差提高 14%。在 09 序列中，动态目标与静态目标比例约为 20%。VINS-Fusion 未能识别到同一地点，未能闭环。由于动态干扰较大，本文算法比 ORB-SLAM3 效果更好，绝对轨迹误差提高 37.6%。

从以上 3 个序列分析得到，VINS-Fusion 在 VSLAM 模式下精度较差，主要反映在垂直方向 y 上，xz 平面也受到影响。ORB-SLAM3 算法在动态物体干扰时有明显轨迹漂移，进而精度下降，而本文算法的跟踪轨迹精度更好。

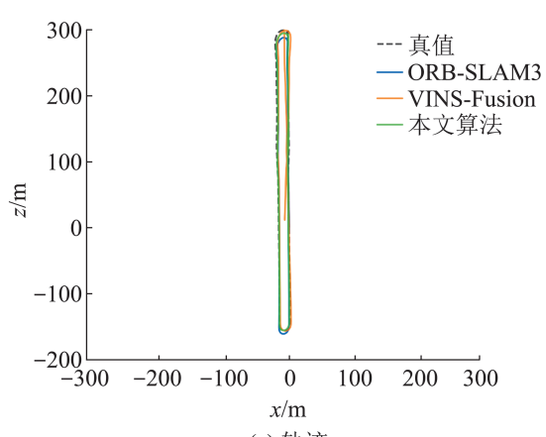


(a) 轨迹

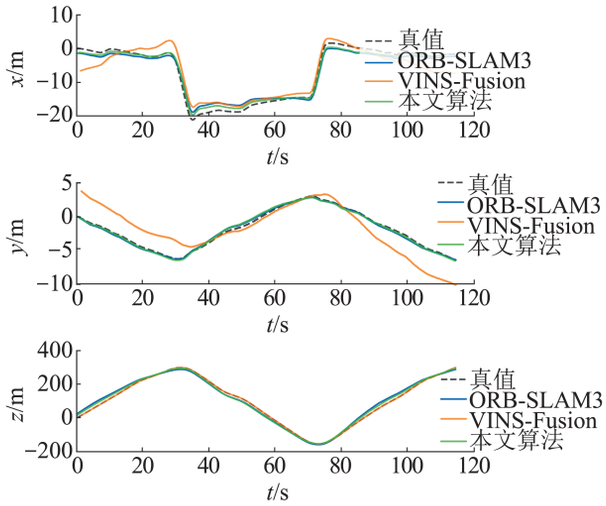
(b) 评估结果

图 19　02 序列

Fig. 19　Sequence 02



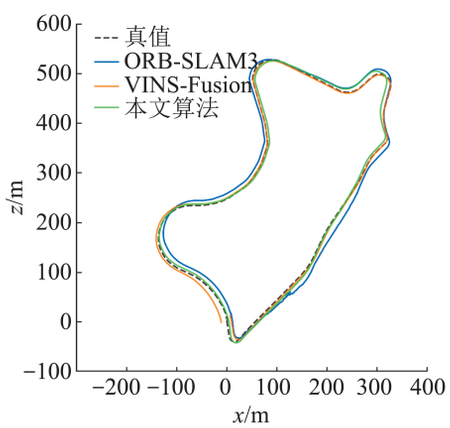
(a) 轨迹



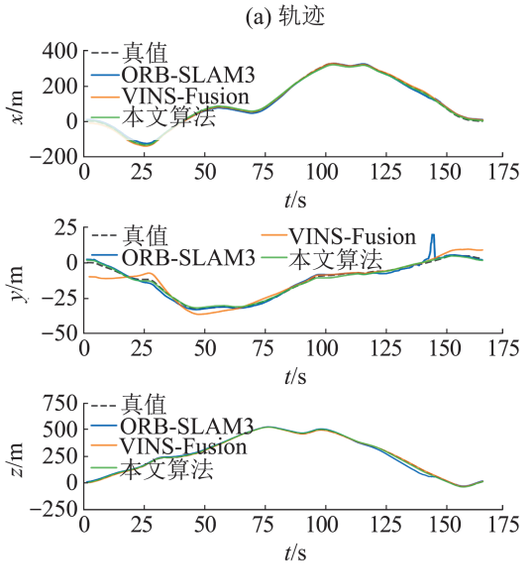
(b) 评估结果

图20　06序列

Fig. 20　Sequence 06



(a) 轨迹



(b) 评估结果

图21　09序列

Fig. 21　Sequence 09

4.3.2 相对轨迹误差

表3是相对轨迹误差的平均平移误差和旋转误差的对比。实验数据证明了VINS-Fusion可通过视觉信息提高旋转精度这一优点。ORB-SLAM3和本文算法的旋转误差略差。本文算法比ORB-SLAM3在平均平移误差精度提升率提高26.3%，旋转误差精度提升率下降4.9%。

表3　相对轨迹误差对比

Table 3　Comparison of relative trajectory error

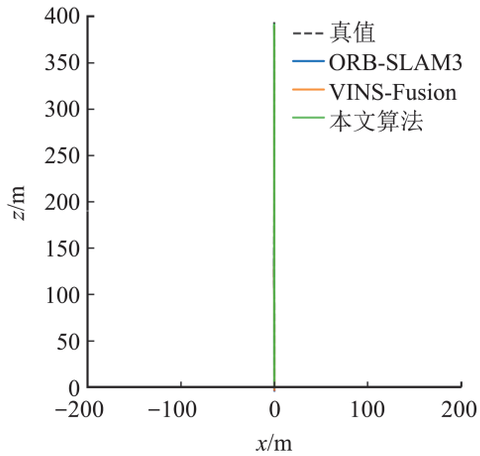
序列	均方根误差/%			旋转误差/((°)/m)		
	ORB-SLAM3	文献[24]	本文	ORB-SLAM3	文献[24]	本文
00	**1.334**	1.791	1.468	0.767	**0.216**	0.474
01	—	3.632	**0.958**	—	**0.064**	2.352
02	0.692	1.207	**0.166**	0.272	**0.158**	0.302
03	0.591	1.174	**0.403**	0.107	**0.058**	0.169
04	0.669	0.617	**0.552**	0.147	**0.046**	0.14
05	1.022	**0.988**	1.347	0.328	**0.077**	0.497
06	2.601	**0.797**	1.607	0.229	**0.057**	0.193
07	1.122	**0.812**	1.126	0.191	**0.075**	0.182
08	10.855	**1.479**	6.392	0.184	**0.072**	0.19
09	4.113	1.612	**1.041**	0.282	**0.067**	0.246
10	2.022	**1.318**	1.504	0.183	**0.083**	0.186

4.3.3 目标检测优化对比

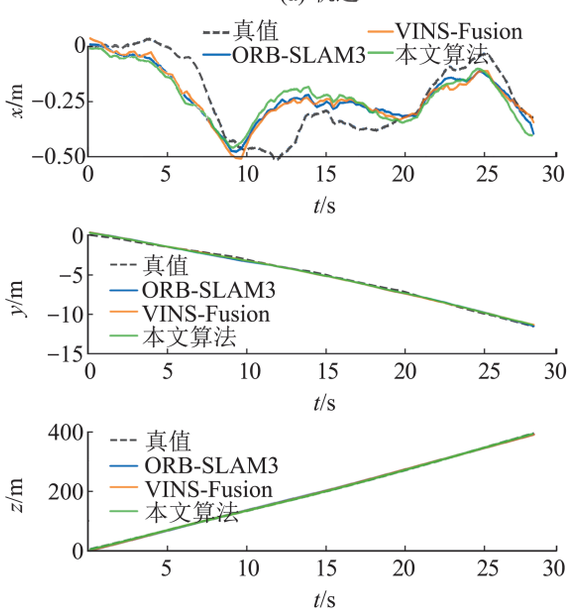
实验在同一置信度下选择了04和07序列进行分析。图22~23中，YOLO表示增加了YOLO的ORB-SLAM3系统。表4中，O+Y表示增加了YOLOv5s的ORB-SLAM3系统数据。04序列包含大部分动态车辆，路程较短。

由图22和表4可以看出，由于对目标检测进行了优化，本文方法比ORB-SLAM3精度提高18.8%。07序列中，动态目标和静态目标比值约为11.8%。如果仅对动态目标进行简单分类，会在SLAM中损失大量特征点，使得二者融合后的精度降低。本文算法对特征点数量进行了筛选，因此会提高系统精度。

https://dc-china-simulation.researchcommons.org/journal/vol36/iss4/20
DOI: 10.16182/j.issn1004731x.joss.22-1332



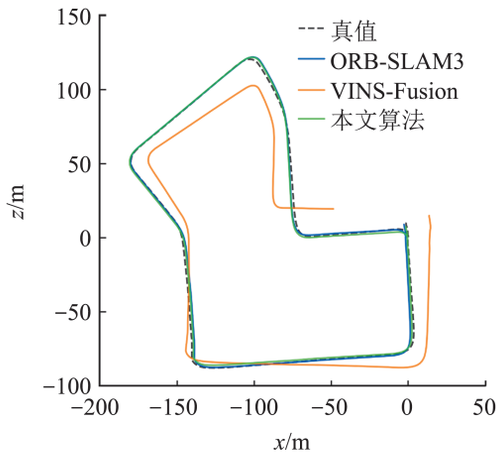
(a) 轨迹



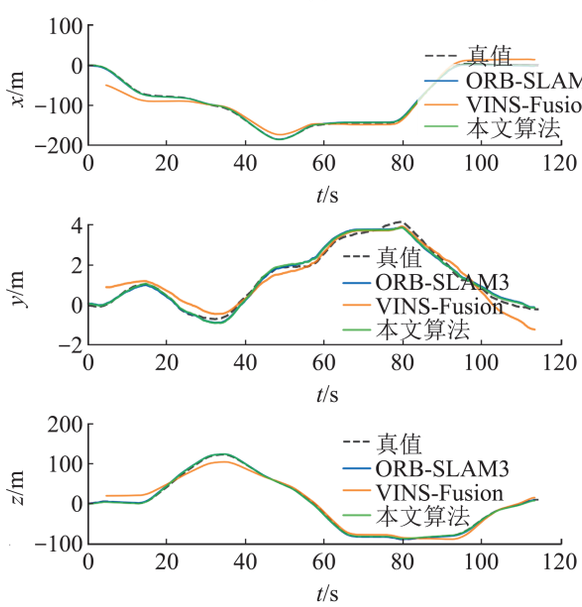
(b) 评估结果

图22　04序列

Fig. 22　Sequence 04



(a) 轨迹



(b) 评估结果

图23　07序列

Fig. 23　Sequence 07

表4　Table 4 目标检测优化对比

Comparison of object detection optimization　　m

序列	平均值			标准差			均方根误差		
	ORB-SLAM3	O+Y	本文方法	ORB-SLAM3	O+Y	本文方法	ORB-SLAM3	O+Y	本文方法
04	0.043	0.034	0.006	0.022	0.02	0.003	0.048	0.039	0.007
07	0.229	0.377	0.175	0.114	0.237	0.101	0.256	0.446	0.212

4.4 帧率分析

由于本文算法比ORB-SLAM3系统增加了计算复杂度，因此运行时间会有增加。每帧跟踪时间统计结果如表5所示。

本文算法跟踪帧率比ORB-SLAM3减少15%。改进后的系统帧率在30帧/s以上，证明能够实时运行，帧率对比如表6所示。

表5 跟踪时间对比
Table 5 Tracking time comparison ms

序列	ORB-SLAM3		本文算法	
	中位数	平均数	中位数	平均数
03	24.7	26.6	28.9	30.2
06	24.4	26.5	29.9	31.4
07	24.5	26.7	30.5	31.1
平均值	24.5	26.6	29.8	30.9

表6 帧率对比
Table 6 Frame rate comparison 帧/s

序列	ORB-SLAM3	本文算法
03	37.6	33.0
06	37.7	31.7
07	37.5	32.0
平均值	37.6	32.2

5 结论

为解决单目SLAM在动态场景下跟踪线程中的对极约束误匹配问题，本文提出一种基于目标检测的动态特征点选择方法。该方法构建由重叠面积、距离相似度和余弦相似度描述的边界框回归损失函数定位动态目标物体，在特征提取时剔除SLAM系统前端图像帧中动态特征点，提高定位精度。在实时性方面，对YOLOv5s的主干网络进行结构重参数化改进，通过减少推理时间减少SLAM跟踪线程耗时，提高系统整体处理速度。在KITTI数据集上的数据表明：本文系统比ORB-SLAM3的定位精度提升了23.4%，速度可以达到30帧/s以上，可以实时运行。后续研究中，对于单目SLAM在动态场景下的应用可以考虑多传感器融合，在精度方面仍有提升空间。

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): 1052-1067.

[2] 张良桥, 陈国良, 许晓东, 等. 一种用于图像特征提取的改进ORB-SLAM算法[J]. 测绘通报, 2019(3): 16-20.
Zhang Liangqiao, Chen Guoliang, Xu Xiaodong, et al. An Improved ORB-SLAM Algorithm for Feature Extraction[J]. Bulletin of Surveying and Mapping, 2019 (3): 16-20.

[3] Zhou Lipu, Wang Shengze, Kaess M. DPLVO: Direct Point-line Monocular Visual Odometry[J]. IEEE Robotics and Automation Letters, 2021, 6(4): 7113-7120.

[4] Ban Xicheng, Wang Hongjian, Chen Tao, et al. Monocular Visual Odometry Based on Depth and Optical Flow Using Deep Learning[J]. IEEE Transactions on Instrumentation and Measurement, 2021, 70: 1-19.

[5] Li Jinquan, Pei Ling, Zou Danping, et al. Attention-SLAM: A Visual Monocular SLAM Learning from Human Gaze[J]. IEEE Sensors Journal, 2021, 21(5): 6408-6420.

[6] Sehyung Lee, Jongwoo Lim, Il Hong Suh. Progressive Feature Matching: Incremental Graph Construction and Optimization[J]. IEEE Transactions on Image Processing, 2020, 29: 6992-7005.

[7] 胡立华, 左威健, 张继福. 采用逆近邻与影响空间的图像特征误匹配剔除方法[J]. 计算机辅助设计与图形学学报, 2022, 34(3): 449-458.
Hu Lihua, Zuo Weijian, Zhang Jifu. A Mismatch Elimination Method Based on Reverse Nearest Neighborhood and Influence Space[J]. Journal of Computer-Aided Design & Computer Graphics, 2022, 34 (3): 449-458.

[8] 任彬, 宋海丽, 赵增旭, 等. 基于RANSAC的视觉里程计优化方法研究[J]. 仪器仪表学报, 2022, 43(6): 205-212.
Ren Bin, Song Haili, Zhao Zengxu, et al. Study on Optimization Method of Visual Odometry Based on RANSAC[J]. Chinese Journal of Scientific Instrument, 2022, 43(6): 205-212.

[9] 盛超, 潘树国, 赵涛, 等. 基于图像语义分割的动态场景下的单目SLAM算法[J]. 测绘通报, 2020(1): 40-44.
Sheng Chao, Pan Shuguo, Zhao Tao, et al. Monocular SLAM System in Dynamic Scenes Based on Image Semantic Segmentation[J]. Bulletin of Surveying and Mapping, 2020(1): 40-44.

[10] Chang Jianfang, Dong Na, Li Donghui. A Real-time Dynamic Object Segmentation Framework for SLAM System in Dynamic Scenes[J]. IEEE Transactions on Instrumentation and Measurement, 2021, 70: 1-9.

[11] 刘瑞军, 王向上, 张晨, 等. 基于深度学习的视觉SLAM综述[J]. 系统仿真学报, 2020, 32(7): 1244-1256.
Liu Ruijun, Wang Xiangshang, Zhang Chen, et al. A Survey on Visual SLAM Based on Deep Learning[J]. Journal of System Simulation, 2020, 32(7): 1244-1256.

[12] 张翠文, 张长伦, 何强, 等. 目标检测中框回归损失函数的研究[J]. 计算机工程与应用, 2021, 57(20): 97-103.
Zhang Cuiwen, Zhang Changlun, He Qiang, et al.

DOI: 10.16182/j.issn1004731x.joss.22-1332

第 36 卷第 4 期 系统仿真学报 Vol. 36 No. 4
2024 年 4 月 Journal of System Simulation Apr. 2024

Research on Loss Function of Box Regression in Object Detection[J]. Computer Engineering and Applications, 2021, 57(20): 97-103.

[13] Rezatofighi H, Tsoi N, Gwak J Y, et al. Generalized Intersection over Union: A Metric and a Loss for Bounding Box Regression[C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE, 2019: 658-666.

[14] Zheng Zhaohui, Wang Ping, Ren Dongwei, et al. Enhancing Geometric Factors in Model Learning and Inference for Object Detection and Instance Segmentation[J]. IEEE Transactions on Cybernetics, 2022, 52(8): 8574-8586.

[15] Xue Hongtao, Wu Meng, Zhang Ziming, et al. Intelligent Diagnosis of Mechanical Faults of In-wheel Motor Based on Improved Artificial Hydrocarbon Networks[J]. ISA Transactions, 2022, 120: 360-371.

[16] Zoph B, Vasudevan V, Shlens J, et al. Learning Transferable Architectures for Scalable Image Recognition[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway, NJ, USA: IEEE, 2018: 8697-8710.

[17] Ding Xiaohan, Zhang Xiangyu, Ma Ningning, et al. RepVGG: Making VGG-style ConvNets Great Again[C]// 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE, 2021: 13728-13737.

[18] Carlos Campos, Richard Elvira, Juan J Gómez Rodríguez, et al. ORB-SLAM3: An Accurate Open-source Library for Visual, Visual-inertial, and Multimap SLAM[J]. IEEE Transactions on Robotics, 2021, 37(6): 1874-1890.

[19] 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 Qinglin, Liu Gangjiang, Xu Qiaoning. 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.

[20] Shao Chunyan, Zhang Chi, Fang Zaojun, et al. A Deep Learning-based Semantic Filter for RANSAC-based Fundamental Matrix Calculation and the ORB-SLAM System[J]. IEEE Access, 2020, 8: 3212-3223.

[21] João Carlos Virgolino Soares, Marcelo Gattass, Marco Antonio Meggiolaro. Visual SLAM in Human Populated Environments: Exploring the Trade-off between Accuracy and Speed of YOLO and Mask R-CNN[C]// 2019 19th International Conference on Advanced Robotics (ICAR). Piscataway, NJ, USA: IEEE, 2019: 135-140.

[22] Berta Bescos, José M Fácil, Javier Civera, et al. DynaSLAM: Tracking, Mapping, and Inpainting in Dynamic Scenes[J]. IEEE Robotics and Automation Letters, 2018, 3(4): 4076-4083.

[23] Geiger A, Lenz P, Stiller C, et al. Vision Meets Robotics: The KITTI Dataset[J]. The International Journal of Robotics Research, 2013, 32(11): 1231-1237.

[24] Wang Ke, Cao Chuan, Ma Sai, et al. An Optimization-Based Multi-sensor Fusion Approach Towards Global Drift-free Motion Estimation[J]. IEEE Sensors Journal, 2021, 21(10): 12228-12235.

