DOI: 10.19650/j.cnki.cjsi.J2412665

基于图神经网络特征点匹配的视觉 SLAM 算法

纪泽源,于潇颖,付文兴
(北京机电工程研究所 北京 100074)

摘 要:视觉 SLAM 技术在增强现实和无人驾驶等行业具有重要应用。然而传统视觉 SLAM 在低照度等挑战性场景中存在定位精度较低或定位失败的难题,本文提出一种图神经网络匹配前后帧特征点的视觉 SLAM 算法:VINS-GNN,在视觉 SLAM 前端设计一种把特征点匹配跟踪策略,将图神经网络与视觉 SLAM 结合,有效提升了 SLAM 前端特征点跟踪的性能;在视觉 SLAM 后端设计了一种基于多帧融合的回环重定位算法,近一步提高了全局定位精度。在包含低照度、低纹理的公开数据集对比实验中,VINS-GNN 相比于 VINS-Fusion 定位精度提高了 17.33%;在实际室内低照度实验中,VINS-GNN 相比于 VINS-Fusion 在轨迹终点处精度提升显著,本文还引入了神经网络推理加速技术,以减少算法的资源占用并提升实时性。实验结果表明 VINS-GNN 提出的策略在室内低照度条件下的定位精度提升效果显著,对室内行人与移动机器人定位技术发展具有重要意义。

关键词: 视觉 SLAM;深度学习;低照度;室内定位
中图分类号: TP399 TH701 **文献标识码**: A **国家标准学科分类代码**: 410.55

Visual SLAM algorithm based on graph neural network feature point matching

Ji Zeyuan, Yu Xiaoying, Fu Wenxing
(*Beijing Electro-mechanical Engineering Institute*, *Beijing 100074*, *China*)

Abstract: The vision-based simultaneous localization and mapping (SLAM) technology has significant applications in industries, such as augmented reality and autonomous driving. However, traditional visual SLAM faces challenges such as low positioning accuracy or failure in low-light conditions. This article proposes a visual SLAM algorithm based on graph neural network (GNN) for matching feature points between consecutive frames, e.g., VINS-GNN. In the front end of the visual SLAM, a feature point matching and tracking strategy is designed, integrating GNN with visual SLAM, which could effectively enhance the performance of feature point tracking. In the back end, a loop closure algorithm based on multi-frame fusion is designed to further improve global positioning accuracy. Comparative experiments on public datasets with low light and low texture show that VINS-GNN improves positioning accuracy by 17.33% compared to VINS-Fusion. In real indoor low-light experiments, VINS-GNN significantly improves the accuracy at the end of the trajectory compared to VINS-Fusion. Additionally, the article introduces neural network inference acceleration techniques to reduce resource consumption and enhance real-time performance. Experimental results show that the strategies proposed by VINS-GNN significantly enhance positioning accuracy under indoor low-light conditions, which is of great significance for the development of indoor pedestrian and mobile robot positioning technology.

Keywords: visual SLAM; deep learning; low illumination; indoor positioning

0 引 言

随着机器人技术、虚拟现实和增强现实行业的发展,市场对于行人和机器人在室内定位的需求越来越高,同步定位与建图(simultaneous localization and mapping, SLAM)技术无需环境中额外的定位设施,可以在未知环境中使用自带的传感器完成定位与建图任务,适合在室内卫星定位信号不佳时使用。在应用中最常见的两类 SLAM 为视觉 SLAM 和激光 SLAM,在室内场景中视觉 SLAM 相比于激光 SLAM 有着硬件成本低廉、轻便、高精度等优势,因此受到更多人的关注。

收稿日期:2024-03-27 Received Date: 2024-03-27

2015 年 Mur-Artal 等[1] 在 ORB-SLAM（versatile and accurate monocular SLAM system）研究的基础上又提出了一套开源的 ORB-SLAM2 系统，该系统使用了 3 个线程进行 SLAM 计算，并且采用了光束平差法（bundle adjustment，BA）对相机位姿进行优化，该系统支持于单目、双目相机和深度相机等多种传感器。2018 年 Qin 等[2] 提出了 VINS-Mono 算法，算法前端对惯性测量单元（inertial measurement unit，IMU）的外参以及 IMU 和图像之间的时间差进行在线校准，提高了算法的鲁棒性。在算法后端提出了基于图优化的滑动窗口优化，滑动窗口按照时间顺序优化窗口中的 IMU 残差项、视觉残差项等，并对过去的信息边缘化。2019 年香港科技大学开源了基于优化的多传感器状态估计器（visual-inertial state estimator-fusion，VINS-Fusion），支持多种视觉/惯性传感器类型，VINS Fusion 在 VINS Mono 的基础上，添加了 GPS 等可以获取全局观测信息的传感器，使得 VINS 可以利用全局信息消除累积误差，进而减小闭环依赖，实现自主精确定位，是目前主流的视觉 SLAM 框架。

传统的视觉 SLAM 框架，其前端光流估计或特征提取算法均由人工设计，传感器模型是理想状态下人工设计的模型，在极端环境下有局限性[3]。近年来基于深度学习的 SLAM 算法得到更多人的关注。Dosovitskiy 等[4-5] 在 2017 年提出了 DeepVO（deep visual odometry），DeepVO 网络分为 2 部分，第 1 部分网络使用 FlowNet 提取帧与帧的光流并进行帧间运动估计，第 2 部分循环神经网络（recurrent neural network，RNN）用于提取光流的时间特征来估计更精确的相机位姿。DeepVO 的里程计精度可以与没有开启回环校正的 ORB-SLAM 的精度相比，为后续的研究提供了可行的技术方案。Zhan 等[6] 在 2018 年将基于学习的深度估计与 2D 光流预测网络结合，并使用传统的相机模型从神经网络估计的深度与光流预测结果计算相机位姿，深度学习模型与精确的相机运动模型结合，可以兼顾鲁棒性与定位精度。Li 等[7] 提出了 DeepSLAM，使用无监督方法实现单目视觉 SLAM 系统，通过立体图像进行训练，利用深度学习技术从图像序列中提取特征并进行位姿估计，但是其精度依赖于数据集的数量。Cao 等[8] 提出了 Deep Fusion SLAM，通过融合 RGB 图像和激光雷达进行训练，利用深度学习技术从图像序列中提取特征并进行位姿估计，但其对低照度条件下的鲁棒性较差。

1　基于图神经网络特征点匹配的视觉 SLAM 算法

深度学习方法和传统的非线性优化理论各有优势，如何使两种方法的优势互补，已成为重要的课题方向。

针对如图 1 所示的复杂条件下的视觉 SLAM 定位任务，本文提出了一种基于图神经网络匹配前后帧特征点的视觉 SLAM 算法：VINS-GNN（visual-inertial state estimator with graph neural network），并在视觉 SLAM 前端设计了一种特征点匹配跟踪策略图神经网络。网络参考 2020 年提出的 Superglue 算法[9] 的结构，该算法在特征点匹配任务中的精度超过了传统图像匹配算法，具有优秀的匹配性能。为方便描述，将本文基于图神经网络特征点匹配的视觉 SLAM 算法简称为 VINS-GNN。

图 1　视觉 SLAM 典型的复杂场景

Fig. 1　Typical complex scenes of visual SLAM

1.1　算法框架

VINS-GNN 算法框架如图 2 所示，本文主要工作和创新点在 SLAM 前端和后端部分：在视觉 SLAM 前端集成了基于卷积神经网络（convolutional neural network，CNN）特征点提取网络，在特征点跟踪部分，使用基于注意力的图神经网络匹配描述子得到图像前后帧特征点的匹配关系，这一方法避免了光照变化和运动模糊对光流跟踪的影响。设计一种基于基础矩阵的异常点剔除策略剔除误匹配的特征点，进一步提高特征点匹配的质量。在视觉 SLAM 后端设计了基于多帧融合的回环重定位算法，进一步提高视觉 SLAM 回环重定位的精度。

1.2　特征点匹配的图神经网络结构

图神经网络的特征点匹配网络包括特征融合和注意力模块。如图 3 所示，特征点匹配网络的输入为前后帧图像的特征点位置、置信度及描述子：(p_i^0, D_i^0) 和 (p_i^1, D_i^1)，在视觉 *SLAM* 的背景下，(p_i^0, D_i^0) 为上一帧图像的特征点及描述子，(p_i^1, D_i^1) 为当前帧图像的特征点及描述子。设上一帧图像中检测到 N 个特征点，当前帧检测到 m 个特征点，则输出的匹配关系矩阵尺寸为 $N \times M$，匹配关系矩阵中存储了相邻两帧图像之间特征点序列的匹配关系。

特征融合部分使用多层感知器[10]（multi layer perceptron，MLP）对特征点的位置及置信度编码，得到与描述子维度相同的向量并与描述子的向量相加实现特征融合，注意力模块对输入的向量进行注意力增强，获得特征点向量之间的匹配置信度。

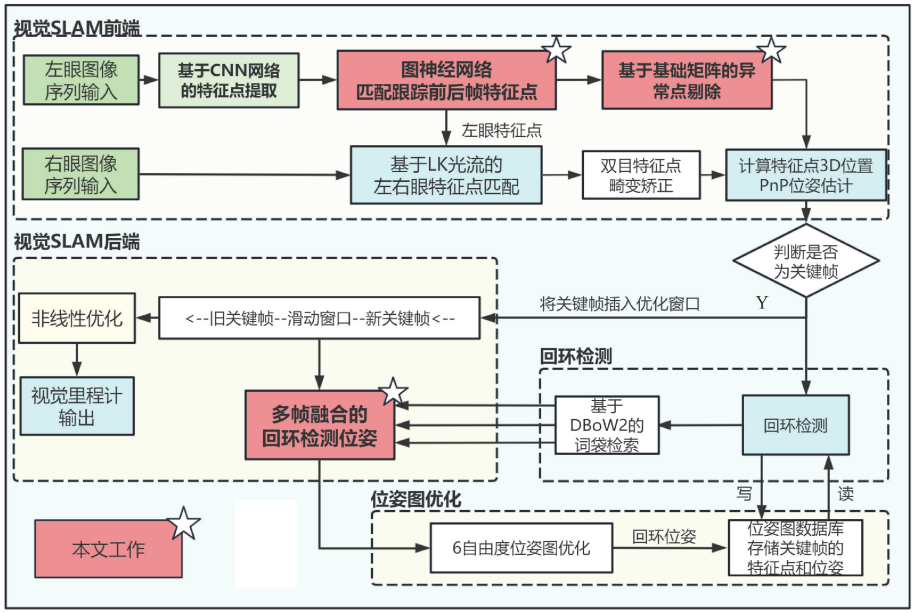


图 2 VINS-GNN 算法框架

Fig. 2 VINS-GNN algorithm framework

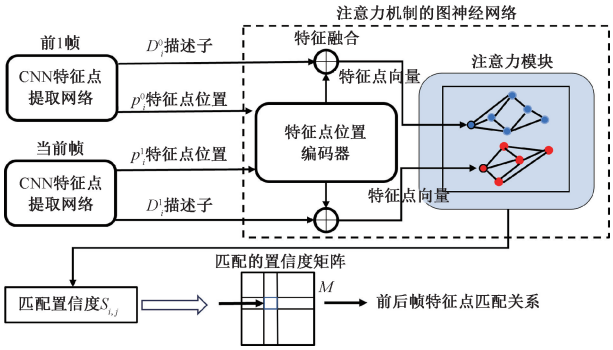


图 3 基于图神经网络的特征点匹配网络的组成

Fig. 3 Composition of feature point matching network based on graph neural network

如图 4 所示,注意力模块中包含自注意力层和交叉注意力层[11],在视觉 SLAM 的背景下自注意力层的作用是削弱单张图像中特征点向量的相关性,减少重复纹理环境中的误匹配;交叉注意力层的作用是强化前后帧图像间特征点向量的相关性,加强前后帧特征点自注意力之后的相关性。

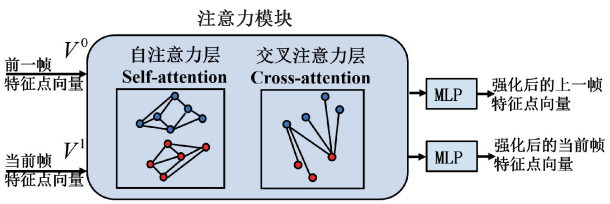


图 4 图神经网络中注意力机制的结构

Fig. 4 The structure of attention mechanism in neural networks

1.3 基于基础矩阵约束的匹配外点剔除策略

为了进一步提高特征点匹配的质量,需要检测图像特征点在前后帧之间匹配后的特征点是否正确,VINS-GNN 在后端集成一种基于基础矩阵约束和随机采样一致性的匹配外点剔除策略。

1)基础矩阵约束

基础矩阵建立了点与极线之间的约束关系,其原理来自对极几何约束[12],具有共同视角、相邻两帧的基础矩阵约束如图 5 所示。

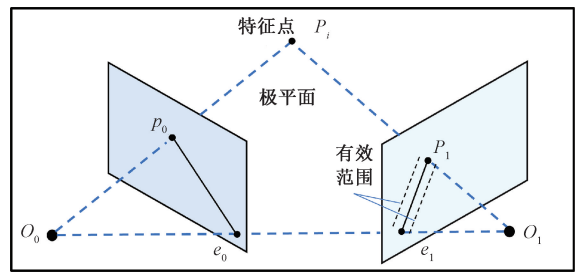


图 5 对极几何约束示意图

Fig. 5 Diagram of polar geometric constraints

当确定特征点匹配点对 $(\boldsymbol{p}_0,\boldsymbol{p}_1)$ 在 O_0 视角图像上的投影 $\boldsymbol{p}_0$ 时,如果匹配正确,则另一个匹配特征点 $\boldsymbol{p}_1$ 会在极线 $\overline{e_1p_1}$ 的有效范围附近,如果 p_1 的位置超出了有效范围,则可以认为特征点匹配错误,有效范围与特征点匹配的质量和图像分辨率有关。对极约束从数学上的描述为:

$$\begin{aligned}&\boldsymbol{p}_0^{\mathrm{T}}\boldsymbol{F}\boldsymbol{p}_1=0\\&\boldsymbol{F}=\boldsymbol{K}^{-\mathrm{T}}t^{\wedge}\boldsymbol{R}\boldsymbol{K}^{-1}\end{aligned}\tag{1}$$

在式（1）中，$\boldsymbol{F}$ 为 O_0 和 O_1 之间的基础矩阵，基础矩阵中包含了相机内参、O_0 和 O_1 之间的旋转与平移信息，相机内参在标定相机后是已知的，在工程中，计算基础矩阵常采用 8 点法计算。本文的异常点剔除策略结合了 8 点法计算基础矩阵方法和随机采样一致性算法。

8 点法计算基础矩阵的方法也称直接线性变换方法，8 点法输入 8 对来自前后帧的匹配点，通过解方程的方式计算基础矩阵。设 $\boldsymbol{p}_1=[u_1,v_1,1]^{\mathrm{T}}$，$\boldsymbol{p}_2=[u_2,v_2,1]^{\mathrm{T}}$，可得到具体的对极约束的表达式（2）：

$$\begin{pmatrix}u_1 & v_1 & 1\end{pmatrix}\begin{bmatrix}F_{11} & F_{12} & F_{13}\\ F_{21} & F_{22} & F_{23}\\ F_{31} & F_{32} & F_{33}\end{bmatrix}\begin{pmatrix}u_2\\ v_2\\ 1\end{pmatrix}=0 \tag{2}$$

如图将基础矩阵 $\boldsymbol{F}$ 中的元素序列化，$\boldsymbol{f}$ 为待求解的向量，其中 $F_{33}=0$：

$$\boldsymbol{f}=[F_{11},F_{12},F_{13},F_{21},F_{22},F_{23},F_{31},F_{32},F_{33}]^{\mathrm{T}} \tag{3}$$

要求解基础矩阵中的 8 个未知数，至少需要 8 个匹配点对，当有 n 对匹配的特征点时，存在以下的方程：

$$\boldsymbol{A}\boldsymbol{f}=0$$

$$\boldsymbol{A}=\begin{pmatrix}u_0^{(1)}u_1^{(1)}, & u_0^{(1)}v_1^{(1)}, & u_0^{(1)}, & v_0^{(1)}u_1^{(1)}, & v_0^{(1)}v_1^{(1)}, & v_0^{(1)}, & u_1^{(1)}, & v_1^{(1)}, & 1\\ u_0^{(2)}u_1^{(2)}, & u_0^{(2)}v_1^{(2)}, & u_0^{(2)}, & v_0^{(2)}u_1^{(2)}, & v_0^{(2)}v_1^{(2)}, & v_0^{(2)}, & u_1^{(2)}, & v_1^{(2)}, & 1\\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots\\ u_0^{(n)}u_1^{(n)}, & u_0^{(n)}v_1^{(n)}, & u_0^{(n)}, & v_0^{(n)}u_1^{(n)}, & v_0^{(n)}v_1^{(n)}, & v_0^{(n)}, & u_1^{(n)}, & v_1^{(n)}, & 1\end{pmatrix} \tag{4}$$

在式（4）中 n 为匹配点对的数量，当 $n=8$ 时，方程可以直接解算 $\boldsymbol{f}$；当 $n>8$ 时，使用最小二乘法解算 $\boldsymbol{f}$。在本文剔除匹配外点的应用背景下，算法的重点在于使用基础矩阵 $\boldsymbol{F}$ 来评估外点剔除的效果。

如图 6 所示，外点剔除的评估模型为基础矩阵模型，求解模型最少的点个数 $K=8$，算法样本点的数量 N 根据场景而不同，分布在 100~500 之间。算法输入 N 个特征点匹配点对，接着在样本中随机采样 8 个匹配点对，按照 8 点法计算基础矩阵；为了评估随机采样的准确性，按顺序计算其余（$N-K$）个匹配点对按照当前计算的基础矩阵投射的距离误差，如果某一匹配点对的距离误差小于阈值，则认为该点为内点；循环 m 次之后，统计内点数最多的内点分布，以此作为异常点剔除的依据。

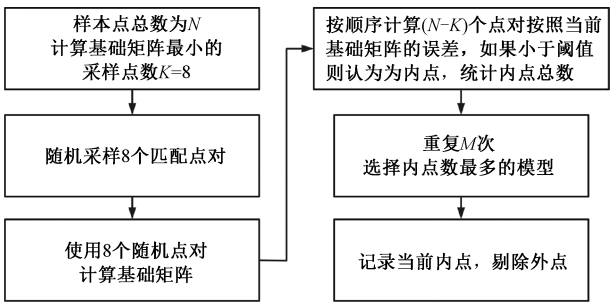


图 6　基于基础矩阵的匹配外点剔除算法流程

Fig. 6　Process of matching outlier removal algorithm based on basic matrix

重复采样次数 M 由式（5）所示，z 为期望的采样成功率，p 代表当前迭代中内点占样本总数的比例，工程中一般设置 $z=0.99$。

$$M=\frac{\log(1-z)}{\log(1-p^K)},K=8 \tag{5}$$

图 6 中右上角计算当前基础矩阵的距离误差表达式如式（6）所示：

$$d(p_0,p_1)=\frac{(\boldsymbol{p}_1^{\mathrm{T}}\boldsymbol{F}\boldsymbol{p}_0)^2}{(\boldsymbol{F}\boldsymbol{p}_0)_x^2+(\boldsymbol{F}\boldsymbol{p}_0)_y^2+(\boldsymbol{p}_1^{\mathrm{T}}\boldsymbol{F})_x^2+(\boldsymbol{p}_1^{\mathrm{T}}\boldsymbol{F})_y^2} \tag{6}$$

每次迭代都将计算基础矩阵的距离误差，当 $d(\boldsymbol{p}_0,\boldsymbol{p}_1)<\tau$ 时，认为当前匹配点为内点，否则为外点。经过 M 次迭代后，使用迭代中内点最多的随机采样结果作为异常点剔除的依据。

1.4　融合多帧回环重定位的数学模型

为方便描述回环重定位的算法模型，如图 7 所示，相机曾经经过此处建筑物，再次经过此处时，词袋检索到与当前关键帧 O_c^i 相似的 k 个回环关键帧：$\{O_l^1,O_l^2,\cdots,O_l^k\}$。每个关键帧视角的图像如图 8 所示，图像之间存在共视关系，有一定的相似性。如图 9 所示，传统视觉 SLAM 在后端使用单帧的特征点 2D 匹配与透视 N 点投影（perspective-n-point，PnP）进行位姿估计的方法校正视觉 SLAM 前端漂移误差，在此基础上本文提出一种融合多帧回环重定位的重定位算法。

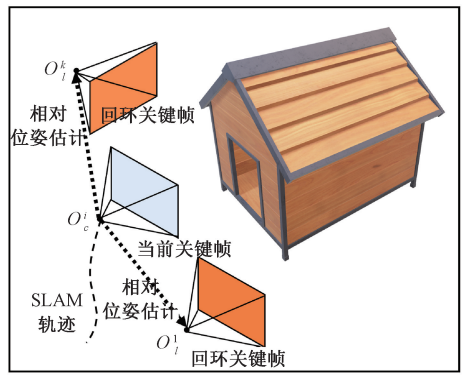


图 7　回环检测中关键帧的视角

Fig. 7　Diagram of keyframe perspective in loop detection

图 8 回环检测中关键帧的视角图像

Fig. 8 View image of keyframes in loop detection

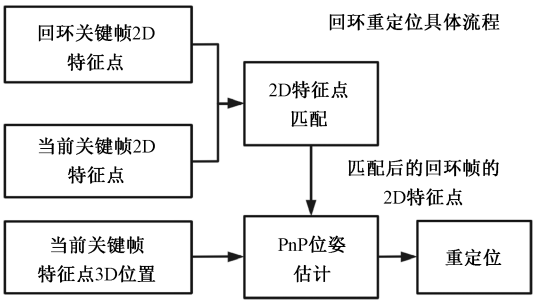


图 9 回环帧匹配重定位流程

Fig. 9 Loop frame matching and relocation process

为了提高回环检测重定位的精度,本文提出融合多个回环关键帧计算相对位姿的方法。为了消除变换矩阵内部的正交约束,使用李代数工具融合多个重定位的结果。

多个回环帧融合的算法流程如图 10 所示,依次匹配当前关键帧和回环关键帧,计算得到多个单次回环重定位的结果,接着使用李代数融合多个重定位的结果。

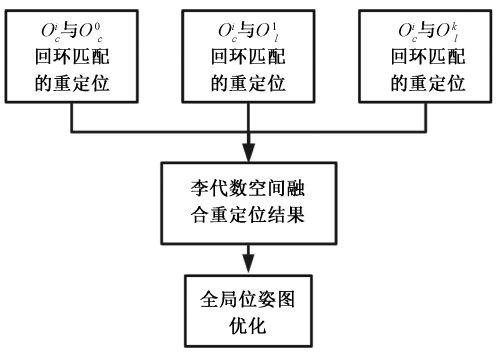


图 10 多个回环帧融合的算法流程

Fig. 10 The algorithm process for fusing multiple loop frames

每个回环帧匹配重定位的流程如图 9 所示,重定位的主要计算量在 PnP 位姿估计,在位姿估计之前,首先进行 2D 特征点匹配,找到回环帧与当前关键帧特征点的匹配关系,目的是找到回环帧 2D 特征点与当前关键帧特征点 3D 位置的一一对应关系,按照 2D~3D 点对的投影关系,使用 PnP 算法估计回环帧相对当前关键帧的相对位姿,由于回环帧在世界坐标系下的位姿在第 1 次经过建筑物后已经存储到词袋数据库中,因此根据回环帧位姿和当前帧的相对位姿,可以计算回环校正后的、更准确的当前帧位姿。

设当前帧为 O_c,视觉里程计计算的当前帧位姿为 T_c;词袋模型检索到的 k 个回环帧为 $\{O_l^1, O_l^2, \cdots, O_l^k\}$,它们在世界坐标系下的位姿为 $\{T_1, T_2, \cdots, T_k\}$。

以 O_c 与 O_l^1 的 PnP 位姿估计为例,PnP 算法估计回环帧 O_l^1 相对当前关键帧 O_c 的相对位姿为 ${}^1T_c$,则经过与 O_l^1 回环校正后的当前帧位姿为 $\overline{T_c^1}$:

$$\overline{T_c^1} = T_1 \cdot {}^1T_c^{-1} \tag{7}$$

根据李群李代数的映射关系,计算特殊欧式群 $\overline{T_c^1}$ 到李代数 ξ_c^1 之间的映射关系:

$$\xi_c^1 = \log(\overline{T_c^1})^{\vee} \tag{8}$$

基于多帧融合的回环重定位算法在李代数空间融合多个回环帧重定位的结果,在李代数空间求重定位结果的均值:

$$\xi_c = \frac{1}{k}\sum_{j=1}^{k}\xi_c^j \tag{9}$$

接着将 ξ_c 映射到特殊欧氏群空间:

$$T_c = \exp(\xi_c^{\wedge}) \tag{10}$$

T_c 作为融合多帧回环帧重定位的结果,相比于单帧回环提高了回环重定位的精度。回环检测重定位后,当前关键帧插入到全局位姿图优化序列中,在视觉 SLAM 背景下,全局位姿图优化的自由度为 6,同时优化历史轨迹的位置与姿态。

在图优化中,存在边和节点,每次回环检测重定位的相对位姿约束为图的边,每个关键帧的位姿为图中的节点,在边的约束下优化节点中的位姿。图优化理论和技术在目前逐渐成熟,在工程中常用 Ceres 和 g2o 优化库构建并计算图优化问题[13]。

2 实验验证

实验部分开展视觉 SLAM 领域公开数据集对比实验和实际场景对比实验,验证 VINS-GNN 算法在包含低照度等挑战性场景下的定位精度。

2.1 公开数据集定位精度对比实验

在视觉 SLAM 领域中 EuRoC 数据集[14]是公认的测评视觉 SLAM 轨迹精度的数据集,数据集中包括精准同步过的双目相机图像与 IMU 惯性传感器数据,参考轨迹通过高精度激光跟踪仪和红外动捕设备获取,参考轨迹具有较高的精度。

1) EuRoC 实验场景

EuRoc 数据集场景如图 11 所示。

(a) 锅炉房场景　(b) 室内场景
(a) Machine hall　(b) Room

图 11　EuRoC 数据集场景
Fig. 11　Scenarios of the EuRoC dataset

2)对比实验结果

(1)定量对比分析

选用包括 Machine Hall 场景 MH01 到 MH05、Room 场景 V1_01 到 V1_03,共 8 个序列的场景测评,通过对比实验验证 VINS-GNN 的精度,其中中等和困难序列的数据集中包含低光照度等挑战性场景。

目前视觉 SLAM 领域有多种开源技术方案,比如基于滤波方法的 MSCKF[15](2007),基于优化方法的 OKVIS(2015)、VINS-Fusion(2019)、基于深度学习的 TartanVO[16](2021)和 TRVO[17]。

如表 1 所示,VINS-GNN 在多个序列中的均方根定位误差小于传统的视觉 SLAM 算法和基于深度学习的 TartanVO 算法,在与最新的基于深度学习的 TRVO 算法比较中,在 MH03 到 MH05 和 V1_01 到 V1_03 序列中,VINS-GNN 算法精度均优于 TRVO。

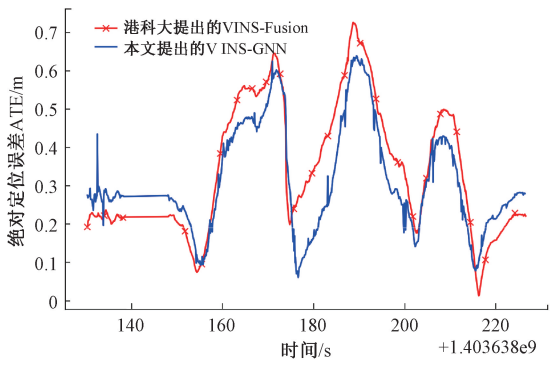
表 1　EuRoC 数据集序列上的定位的均方根误差
Table 1　RSME of Localization on the EuRoC Dataset Sequence　(m)

算法类型	数据集序列							
	MH01 简单	MH02 简单	MH03 中等	MH04 困难	MH05 困难	V1_01 简单	V1_02 中等	V1_03 困难
MSCKF(2007)	0.42	0.45	0.23	0.37	0.48	0.34	0.20	0.37
OKVIS(2015)	0.161	0.22	0.24	0.36	0.47	0.09	0.20	0.24
VINS-Fusion(2019)	0.168	0.186	0.17	0.402	0.35	0.198	0.100	0.114
TartanVO (2021)	0.639	0.325	0.550	1.153	1.021	0.447	0.389	0.622
TRVO (2023)	**0.124**	**0.152**	0.172	0.341	0.465	0.092	0.099	0.124
VINS-GNN(本文)	0.159	0.156	**0.147**	**0.335**	**0.323**	**0.073**	**0.093**	**0.101**

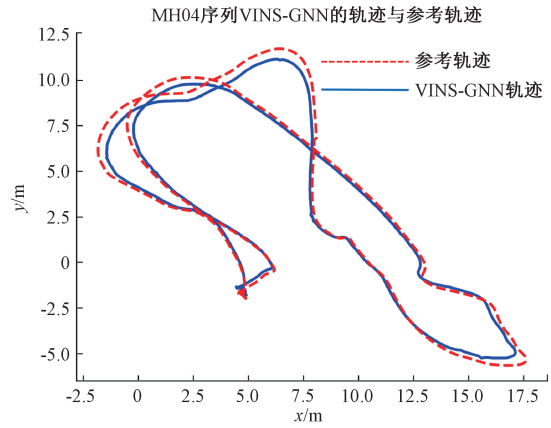
(2)定性分析

如图 12(a)所示,在本次实验中,VINS-GNN 的定位误差在大多数时间低于 VINS-Fusion,且图 12(b)中 VINS-GNN 的定位轨迹与参考轨迹全程一致,验证了 VINS-GNN 对提高视觉 SLAM 定位精度的有效性。

结合表 1 和图 12,在 MH01 到 MH05 序列的实验中,VINS-GNN 相比于 VINS-Fusion 定位精度提高 19.06%,在 MH01 到 MH05 和 V1_01 到 V1_03 序列中相比于 VINS-Fusion 定位精度提高 17.33%。实验结果表明,VINS-GNN 对普通场景和低照度场景均有显著的定位精度提升。



(a) MH04(困难)序列定位误差比较
(a) Comparison of MH04 (difficult) sequence positioning errors



(b) MH04(困难)序列的定位轨迹
(b) Localization trajectory map of MH04 (difficult) sequence

图 12　VINS-GNN 定位精度的定性分析
Fig. 12　Qualitative analysis of VINS-GNN positioning accuracy

2.2　实际低照度条件下运动场景的对比实验

1)实验参数

本节在实际低照度场景中评估 VINS-GNN 算法,实际实验条件如图 13 所示。图像采集设备为自行搭建的双目惯性相机和移动端计算机如图 13(a)所示,双目惯

性相机参数如表 2 所示,移动端计算机用于记录和传输双目图像数据,移动载具为四轮移动机器人底盘。

(a) 图像采集设备
(a) Image acqu isition equipment

(b) 数字光照度计
(b) Digital illuminometer

(c) 实验场地及移动轨迹(亮灯拍摄)
(c) Experimental site and movement trajectory (shooting with lights on)

图 13 实际实验条件

Fig. 13 Actual experimental conditions

表 2 双目惯性相机参数

Table 2 Stereo inertial camera parameters

参数类型	参数值
双目基线长度	120 mm
摄像头模组型号	KS2A543-3.0(Global Shutter)
摄像头模组快门类型	全局快门
镜头视场角	110°
支持的分辨率	1 920×1 080,1 280×720,640×480
摄像头通讯接口	USB3.0
惯性传感器型号及类型	CH100,MEMS

关闭灯光后,实验场地的环境光的照度值如表 3 所示,使用图 13(b)所示的照度计测量场景四周和中心处共 5 个地点的照度值,平均照度值为 4.58 Lux。在低照度条件下,图像的纹理也会随着光照降低而减少。

在关闭灯光的条件下,机器人底盘在场景中环绕一圈,并通过参考地面瓷砖的边缘控制机器人的起点与终点重合,机器人行进路线的示意图如图 13(b)所示。

表 3 关闭灯光后场地光照度测量值

Table 3 Measurement values of on-site illumination after turning off the lights

测量地点	#1	#2	#3	#4	#5	平均
照度值/Lux	4.8	1.9	7.8	5	3.4	4.58

在实验中移动机器人搭载双目相机以最高 2 m/s 的线速度平稳运动,从起点出发在室内环绕行驶一圈回到初始出发点。在实际轨迹起点和终点重合的情况下,依据 SLAM 算定位轨迹的出发点和终点位置之间的距离评价定位精度。为了确保算法对比实验条件的一致性,首先录制低照度场景下机器人移动的图像数据,在同样的图像数据下测试 VINS-Fusion 和 VINS-GNN 算法的效果。

2)低照度场景视觉 SLAM 定位实验分析

为了从速度上区分实验,将 3 次实验用“低速”、“中速”和“快速”名称区分。具体速度取值参考 EuRoC 数据集中“简单”,“中等”和“困难”序列中的线速度和角速度。

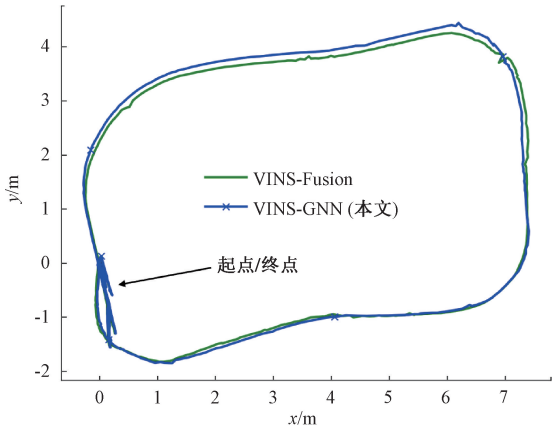
表 4 可以发现,除了光照条件外,载体移动的线速度和角速度对定位误差的影响也较大。这是由于机器人快速运动时图像会发生运动模糊,低照度、低纹理、运动模糊场景对传统算法特征点提取和光流跟踪算法来说是一种挑战。

表 4 低照度场景实验中小车不同速度的定位误差

Table 4 Localization error of the robot at different speeds in low-light scenarios (m)

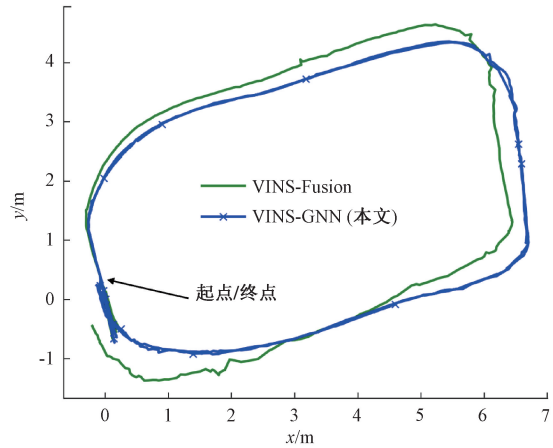
	速度	最大线速度 ($m \cdot s^{-1}$)	最大角速度 (°/s)	VINS-Fusion	VINS-GNN
1	低速	1.1	30	0.045	**0.020**
2	中速	2.1	42	0.927	**0.171**
3	快速	3.5	61	定位失败	**0.223**

特征点精度和数量决定了视觉 SLAM 前端的定位精度,进一步使用图形化的方式对比 VINS-GNN 和 VINS-Fusion 算法前端特征点提取质量和数量的情况,圆点为算法提取的特征点位置。如图 15 所示,左侧两图为 VINS-GNN 的特征点可视化,右侧两图为 VINS-Fusion 算法的特征点可视化。从特征点提取的质量上分析,一般情况下,特征点应分布在图像中的物体的边缘或图像梯度变化明显区域的附近,通过观察特征点分布可以发现左侧图片中特征点检测的质量高于右侧图片中的质量。从特征点数量上分析,VINS-GNN 特征点的数量显著高



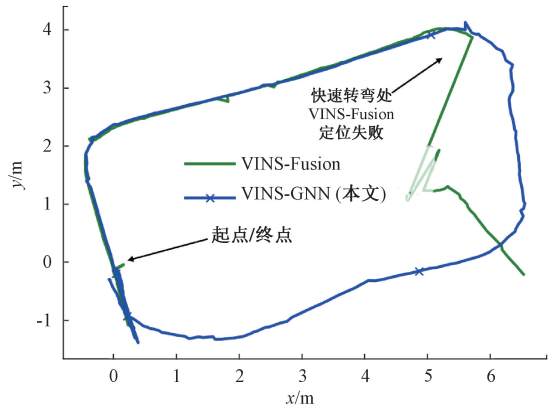
(a) 低照度场景的低速轨迹对比

(a) Comparison of low-speed trajectories in low-light scenarios



(b) 低照度场景的中速轨迹对比

(b) Comparison of low-speed trajectories in low-light scenarios



(c) 低照度场景的快速轨迹对比

(c) Comparison of low-speed trajectories in low-light scenarios

图 14　不同条件下算法定位轨迹对比

Fig. 14　Comparison chart of algorithm positioning trajectories

于 VINS-Fusion。从定位精度和特征点提取可视化对比 VINS-GNN 与传统 VINS-Fusion 算法的性能，验证了 VINS-GNN 算法在包含低照度、低纹理等挑战性环境下，具有较好的定位精度。

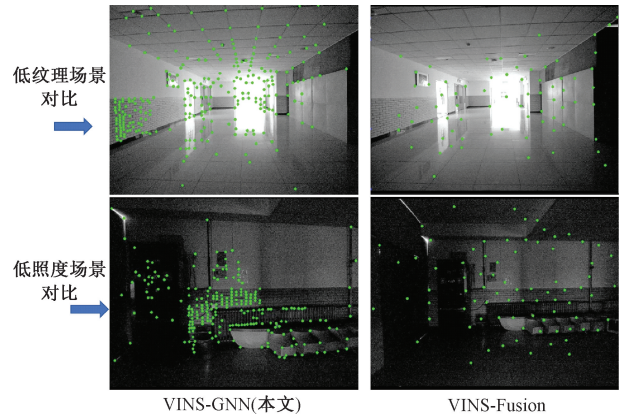


图 15　低照度、低纹理条件下 VINS-GNN 与 VINS-Fusion 特征点提取对比

Fig. 15　Comparison between VINS-GNN and VINS Fusion feature point extraction under low illumination and low texture conditions

3　神经网络推理加速技术及算法资源占用评估

为了减少显存占用并提高神经网络推理帧率，本文对 VINS-GNN 中的神经网络进行模型加速。模型加速具体流程如图 16 所示。

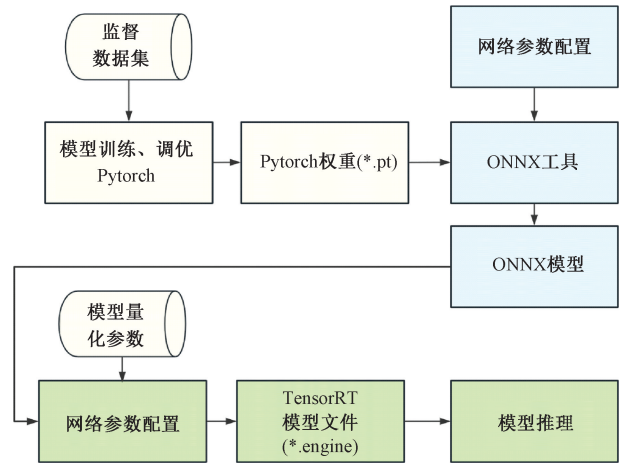


图 16　VINS-GNN 深度学习模型部署流程

Fig. 16　Diagram of the deployment process for the VINS-GNN deep learning model

CNN 与 GNN 网络首先使用 Pytorch 框架训练和调优模型，输出 Pytorch 格式的模型权重文件；接着使用开放式神经网络交换（open neural network exchange，ONNX）工具将 Pytorch 模型权重文件转换为通用的模型格式文件，在本文中 ONNX 模型从动态计算图转换为静态计算图，提高了模型的推理速度。TensorRT 工具导入 ONNX

和量化参数配置,使用模型量化、剪枝、压缩等方法对模型进行量化加速,输出 TensorRT 格式的模型文件。在实际中使用 TensorRT 的 context 加载模型文件,对输入张量计算得到输出。

使用如表 5 所示的计算机分别测试 Pytorch、ONNX 和 TensorRT 模型在相同的 480×640 尺寸图像序列输入条件下的推理速度和显存占用情况,具体性能参数取 100 帧图像推理后的平均值。

表 5 实验计算机性能参数

Table 5 Experimental computer performance parameters

名称	参数
CPU 型号	i5-10600KF
GPU 型号	RTX3070
内存大小	16 GB

如表 6 所示,CNN 特征点提取网络中使用 TensorRT 的推理相比于 Pytorch 的推理时间降低了 83.9%,显存占用减小了 22.4%。GNN 特征点匹配网络中使用 TensorRT 的推理相比于 Pytorch 的推理时间降低了 89%,显存占用减小了 41.7%,CNN 特征点检测与 GNN 特征点匹配每帧所用时间仅为 16 ms,满足视觉 SLAM 算法对实时性的要求。

表 6 深度学习模型加速部署效果对比

Table 6 Comparison of accelerated deployment effects of deep learning models

性能参数	Pytorch	ONNX	TensorRT
CNN 推理时间(平均)/ms	31	7	5
GNN 推理时间(平均)/ms	100	30	11
CNN 显存占用(平均)/GB	6.7	5.8	5.2
GNN 显存占用(平均)/GB	1.2	0.8	0.7

4 结 论

本文提出一种图神经网络匹配前后帧特征点的视觉 SLAM 算法:VINS-GNN,在视觉 SLAM 前端,将 GNN 网络与特征点检测跟踪结合,提高了特征点匹配的数量和质量,进一步的,使用基于基础矩阵和随机采样一致性的匹配异常点剔除策略,剔除匹配外点。在视觉 SLAM 的后端设计了一种在回环检测环节融合多帧回环帧的回环重定位算法,使用李代数工具解决了位姿融合的正交约束问题,进一步优化了回环定位的精度。

在 EuRoC 公开数据集中,VINS-GNN 算法相比于 VINS-Fusion 算法定位精度提高了 17.33%,实际室内低照度实验中,在低速、中速、高速条件下,VINS-GNN 相比于 VINS-Fusion 在轨迹终点处均有显著的精度提示,实验结果表明 VINS-GNN 提出的策略在室内低照度、高速运动条件下的定位精度提升效果显著,最后本文使用神经网络加速技术对 VINS-GNN 中的神经网络进行推理加速,将神经网络的推理时间降低至 16 ms,满足了视觉 SLAM 的是实时性要求,综上所示,本文算法及技术方案对行人与移动机器人定位技术发展具有重要意义。

参考文献

[1] 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.

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

[3] 欧阳豪. 深度学习结合视觉 SLAM 的室内定位研究[D]. 大连:大连理工大学, 2021.
OUYANG H. Research on indoor localization combining deep learning with visual SLAM[D]. Dalian: Dalian University of Technology, 2021.

[4] DOSOVITSKIY A, FISCHER P, ILG E, et al. FlowNet: Learning optical flow with convolutional networks[C]. Proceedings of the IEEE International Conference on Computer Vision, 2015: 2758-2766.

[5] WANG S, CLARK R, WEN H K, et al. DeepVO: Towards end-to-end visual odometry with deep recurrent convolutional neural networks [C] 2017 IEEE International Conference on Robotics and Automation (ICRA), 2017: 2043-2050.

[6] ZHAN H, GARG R, WEERASEKERA C S, et al. Unsupervised learning of monocular depth estimation and visual odometry with deep convolutional networks[C] IEEE Conference on Computer Vision and Pattern Recognition, 2018: 340-349.

[7] LI R H, WANG S, GU D B. DeepSLAM: A robust monocular SLAM system with unsupervised deep learning[J]. IEEE Transactions on Industrial Electronics, 2021, 68(4): 3577-3587.

[8] CAO Y, DENG Z, LUO Z, FAN J. A Multi-sensor deep fusion SLAM algorithm based on TSDF map[J]. IEEE Access, 2024. DOI: 10.1109/ACCESS.2024.3415416.

[9] SARLIN P E, DETONE D, MALISIEWICZ T, et al. SuperGlue: Learning feature matching with graph neural

networks[C] Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020: 4938-4947.

[10] RAMCHOUN H, GHANOU Y, ETTAOUIL M, et al. Multilayer perceptron: Architecture optimization and training [C]. Proceedings of the 2nd International Conference on Big Data, Cloud and Applications, 2016: 1-6.

[11] YE L W, ROCHAN M, LIU ZH, et al. Cross-modal self-attention network for referring image segmentation[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019: 10502-10511.

[12] GOSHEN L, SHIMSHONI I. Balanced exploration and exploitation model search for efficient epipolar geometry estimation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008, 30(7): 1230-1242.

[13] 蒋郡祥. 基于图优化的视觉/惯性/GNSS 融合导航方法研究[D]. 武汉:武汉大学, 2021.
JIANG J X. Research on visual/inertial/GNSS integrated navigation method based on graph optimization [D]. Wuhan: Wuhan University, 2021.

[14] BURRI M, NIKOLIC J, GOHL P, et al. The EuRoC micro aerial vehicle datasets [J]. The International Journal of Robotics Research, 2016, 35(10): 1157-1163.

[15] MOURIKIS A I, ROUMELIOTIS S I. A multi-state constraint Kalman filter for vision-aided inertial navigation[C]. Proceedings of the 2007 IEEE International Conference on Robotics and Automation, 2007: 3565-3572.

[16] WANG W, HU Y, SCHERER S. TartanVO: A generalizable learning-based visual odometry [C]. Conference on Robot Learning. PMLR, 2021: 1761-1772.

[17] GAO Y H, ZHAO L. Coarse TRVO: A robust visual odometry with detector-free local feature[J]. Journal of Advanced Computational Intelligence and Intelligent Informatics, 2022, 26(5): 731-739.

作者简介

纪泽源，2023 年于北京理工大学获得硕士学位，现为北京机电工程研究所助理工程师，主要研究方向为飞行器综合电子电气总体设计。
E-mail: jzyuan567@163.com

Ji Zeyuan received his M. Sc. degree from Beijing Institute of Technology in 2023. He is currently the chief designer at Beijing Electro-mechanical Engineering Institute. His main research interest is general design of integrated electronics and electrics for aircraft.