



多传感器融合定位

第4章 点云地图构建及 基于地图的定位

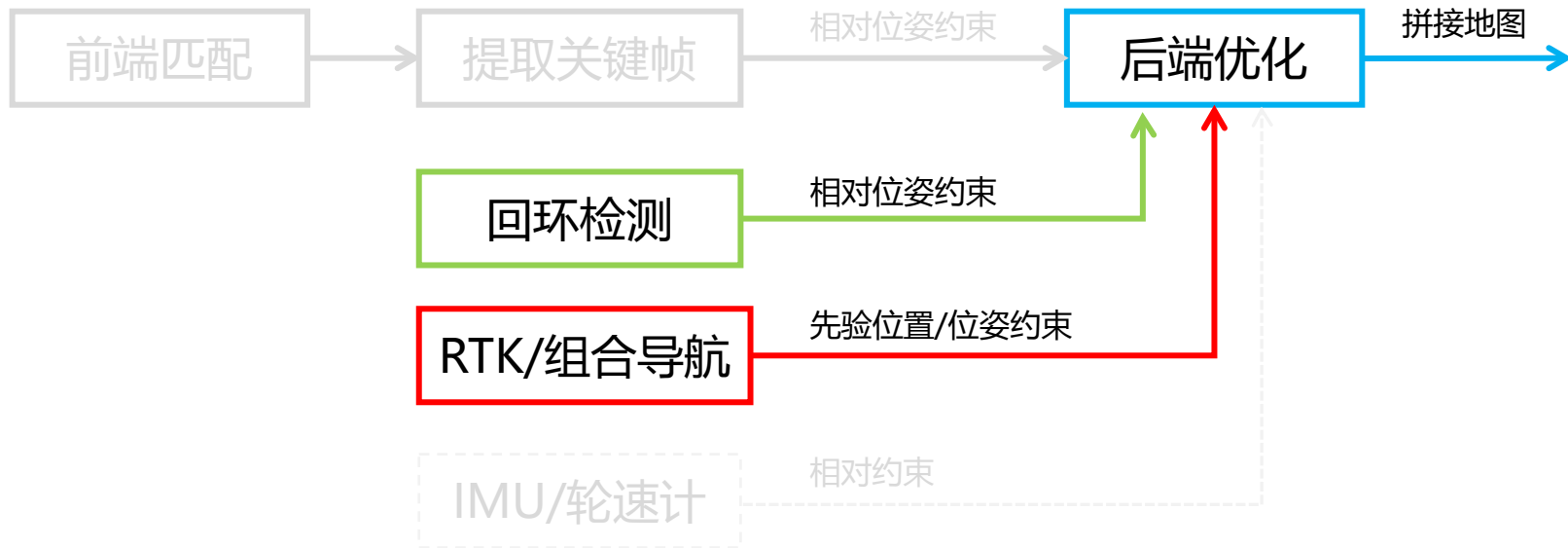
主讲人 任 乾

北京理工大学本硕
自动驾驶从业者





建图流程介绍





目录



1. 回环检测



2. 后端优化



3. 点云地图建立



4. 基于地图的定位



5. LeGO-LOAM介绍



目录



1. 回环检测



2. 后端优化



3. 点云地图建立



4. 基于地图的定位



5. LeGO-LOAM介绍

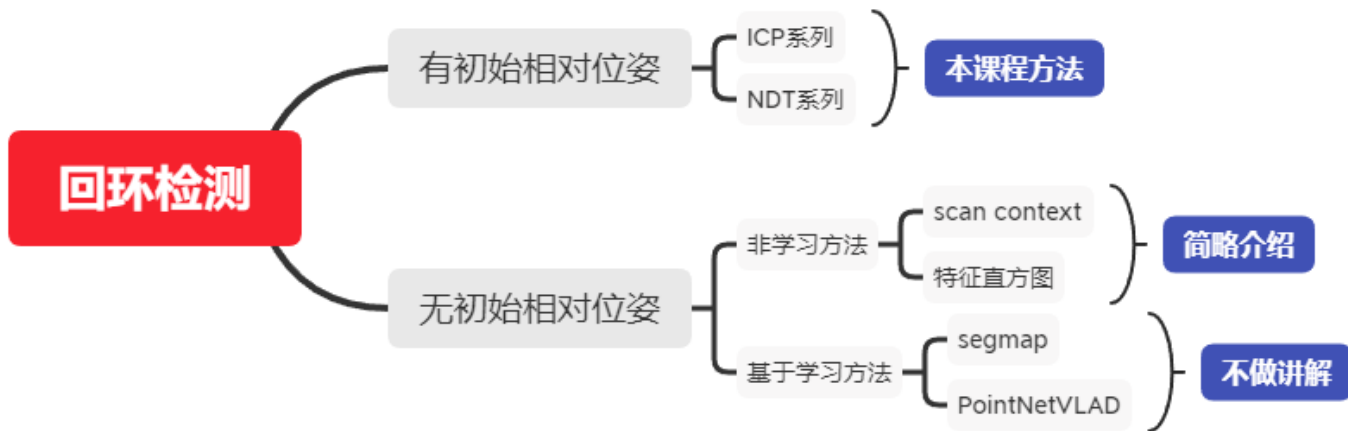


回环检测

目的：消除累计误差，并提高地图一致性。

方法：从历史帧中找出相似帧（即物理世界中相近位置且有充足共视区域的帧），并给出二者的相对位姿。

用法：在后端优化中，作为一个相对位姿约束的边，加入图优化系统中（后面详细讲述）。





回环检测

基于Scan Context

方法原理:

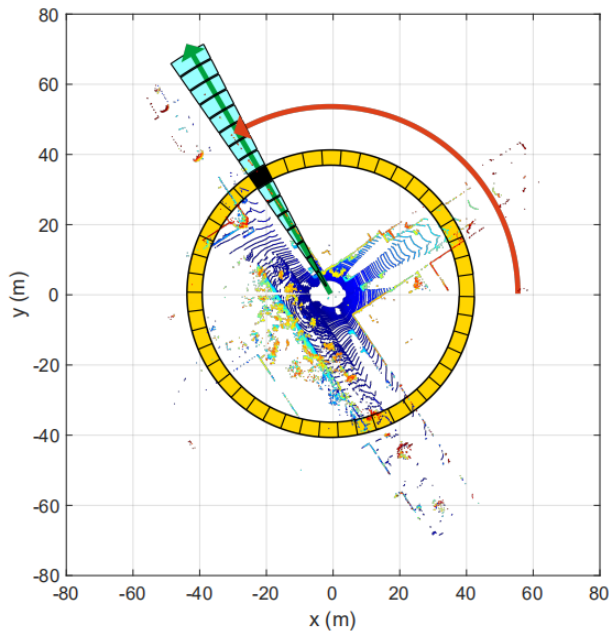
论文题目: Scan Context: Egocentric Spatial Descriptor for Place Recognition within 3D Point Cloud Map

代码地址: <https://github.com/irapkaist/scancontext>

应用案例:

方法: LeGO-LOAM + scan context 闭环检测

代码: <https://github.com/irapkaist/SC-LeGO-LOAM>





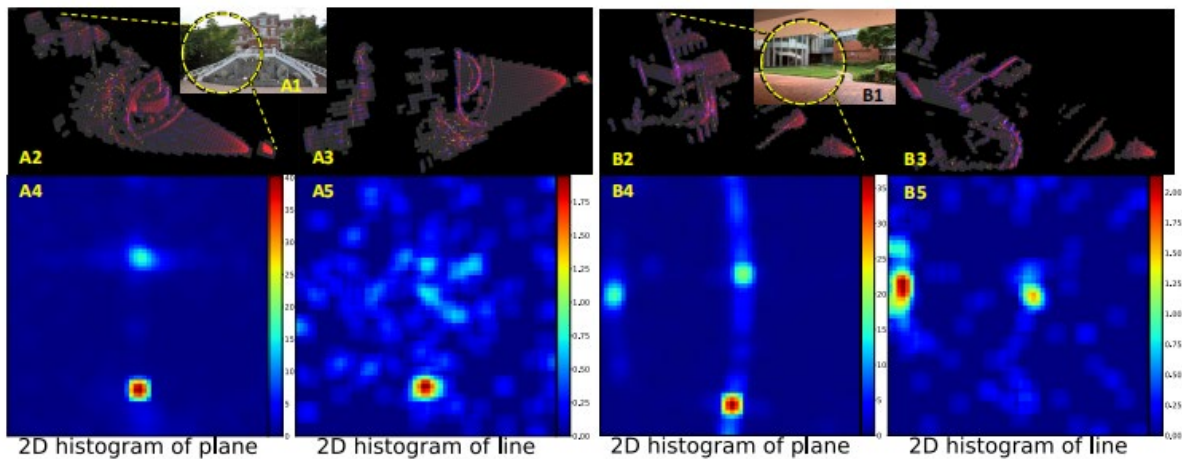
回环检测

基于直方图

论文题目: A fast, complete, point cloud based loop closure for LiDAR odometry and mapping

应用案例: loam_livox

开源代码: https://github.com/hku-mars/loam_livox





目录



1. 回环检测



2. 后端优化



3. 点云地图建立



4. 基于地图的定位



5. LeGO-LOAM介绍



1. 后端优化基本原理

目的：利用回环检测结果和惯导先验位姿修正里程计误差。

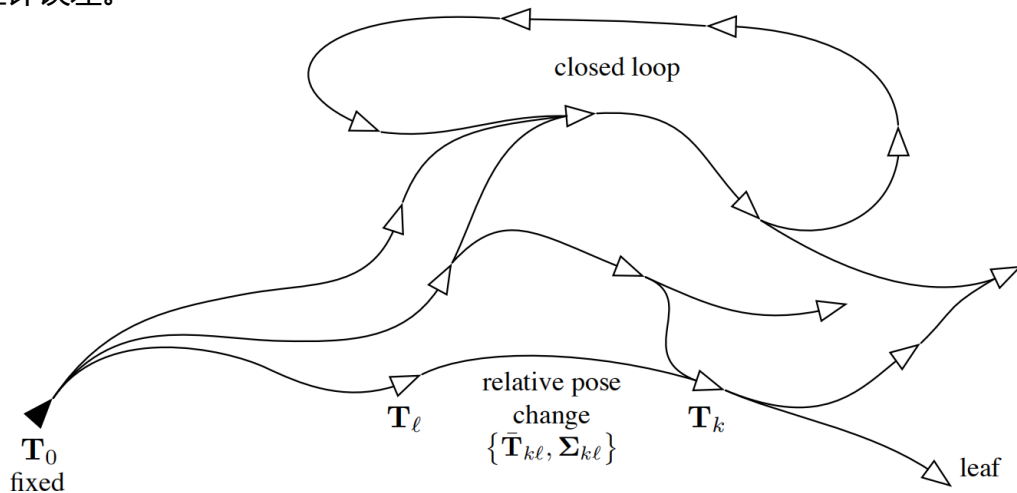
回环在此处提供的是两帧之间的相对位姿。

观测：

- 1) 连续两帧的相对位姿观测；
- 2) 回环匹配得到的相对位姿观测；
- 3) 组合导航提供的先验位姿观测。

使用方式：

- a. 1)和2)的观测构成了基于回环的位姿修正；
- b. 1)和3)的观测构成了基于先验观测的位姿修正；
- c. 1) 2) 3) 也可以同时使用。

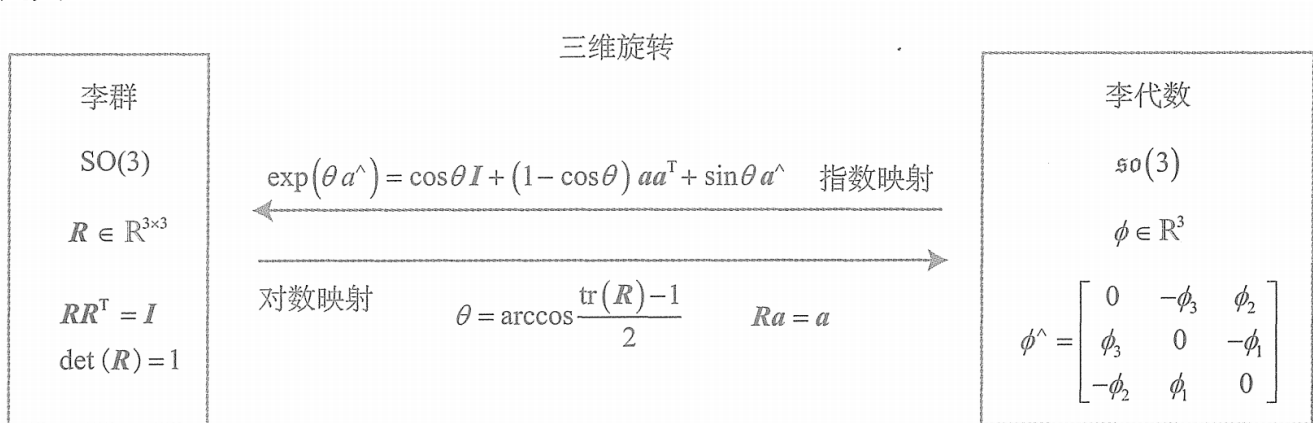


后端优化示意



2. 李群、李代数基本知识

三维旋转上的定义：



常用转换：

$$R = \exp(\phi^\wedge)$$

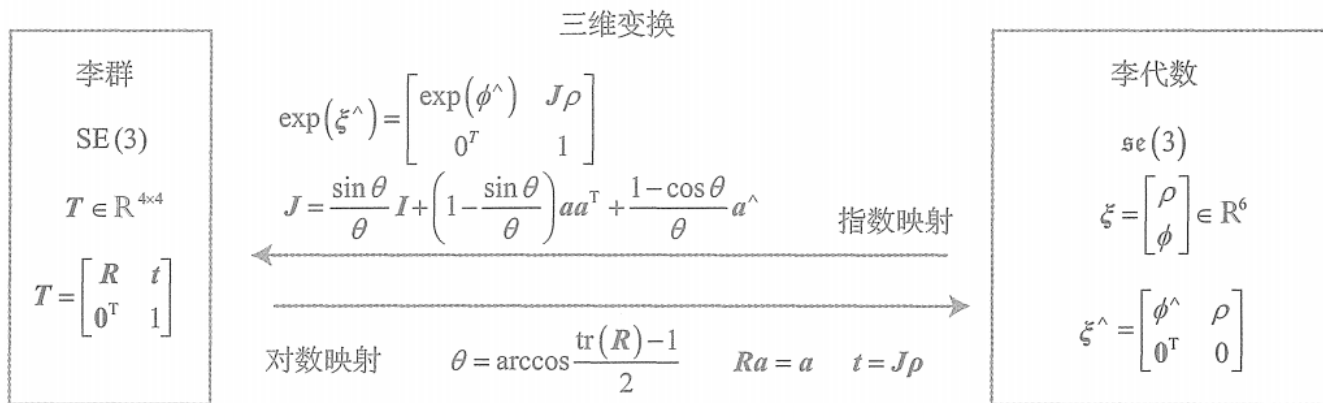
$$R^{-1} = \exp((- \phi)^\wedge)$$

$$\phi = \ln(\exp(\phi^\wedge))^\vee = \ln(R)^\vee$$



2. 李群、李代数基本知识

三维变换上的定义：



常用转换：

$$T = \exp(\xi^\wedge)$$

$$T^{-1} = \exp((- \xi)^\wedge)$$

$$\xi = \ln(\exp(\xi^\wedge))^\vee = \ln(T)^\vee$$



2. 李群、李代数基本知识

BCH公式

$$\ln(\exp(A) \exp(B)) = A + B + \frac{1}{2}[A, B] + \frac{1}{12}[A, [A, B]] - \frac{1}{12}[B, [A, B]] + \dots$$

SO(3)下的李括号定义为

$$[\phi_1, \phi_2] = (\Phi_1 \Phi_2 - \Phi_2 \Phi_1)^\vee$$

SE(3)下的李括号定义为

$$[\xi_1, \xi_2] = (\xi_1^\wedge \xi_2^\wedge - \xi_2^\wedge \xi_1^\wedge)^\vee$$

其中

$$\Phi = \phi^\wedge = \begin{bmatrix} 0 & -\phi(3) & \phi(2) \\ \phi(3) & 0 & -\phi(1) \\ -\phi(2) & \phi(1) & 0 \end{bmatrix}$$



2. 李群、李代数基本知识

SO(3)对应的BCH公式

$$\ln \left(\exp \left(\phi_1^\wedge \right) \exp \left(\phi_2^\wedge \right) \right)^\vee \approx \begin{cases} J_l \left(\phi_2 \right)^{-1} \phi_1 + \phi_2, & \text{当 } \phi_1 \text{ 为小量} \\ J_r \left(\phi_1 \right)^{-1} \phi_2 + \phi_1, & \text{当 } \phi_2 \text{ 为小量} \end{cases}$$

其中左乘雅可比为

$$J_l = \frac{\sin \theta}{\theta} I + \left(1 - \frac{\sin \theta}{\theta} \right) a a^T + \frac{1 - \cos \theta}{\theta} a^\wedge$$

即

$$J_l^{-1} = \frac{\theta}{2} \cot \frac{\theta}{2} I + \left(1 - \frac{\theta}{2} \cot \frac{\theta}{2} \right) a a^T - \frac{\theta}{2} a^\wedge$$

右乘雅可比仅需要在左乘雅可比的基础上对自变量取负号，即

$$J_r(\phi) = J_l(-\phi)$$



2. 李群、李代数基本知识

SE(3)对应的BCH公式

$$\ln \left(\exp \left(\xi_1^\wedge \right) \exp \left(\xi_2^\wedge \right) \right)^\vee \approx \begin{cases} \mathcal{J}_\ell \left(\xi_2 \right)^{-1} \xi_1 + \xi_2, & \text{当 } \xi_1 \text{ 为小量} \\ \mathcal{J}_r \left(\xi_1 \right)^{-1} \xi_2 + \xi_1, & \text{当 } \xi_2 \text{ 为小量} \end{cases}$$

由于SE(3)上的雅可比形式过于复杂，此处直接给出本章所用到的近似形式如下。详细内容可参考《机器人中的状态估计》中公式7.83的推导过程。

$$\mathcal{J}_r \left(\xi \right)^{-1} \approx I + \frac{1}{2} \begin{bmatrix} \phi^\wedge & \rho^\wedge \\ 0 & \phi^\wedge \end{bmatrix}$$

若 ξ_1 和 ξ_2 非常小，则左右雅可比均可以直接约等于单位阵，此时有

$$\ln \left(\exp \left(\xi_1^\wedge \right) \exp \left(\xi_2^\wedge \right) \right)^\vee \approx \ln \left(\exp \left(\xi_1^\wedge + \xi_2^\wedge \right) \right)^\vee$$



2. 李群、李代数基本知识

SO(3)上的伴随性质

$$\exp(\phi^\wedge) R = R \exp((R^{-1}\phi)^\wedge)$$

SE(3)上的伴随性质

$$\exp(\xi^\wedge) T = T \exp((\text{Ad}(T^{-1})\xi)^\wedge)$$

其中伴随矩阵的定义如下

$$\text{Ad}(T) = \begin{bmatrix} R & t^\wedge R \\ 0 & R \end{bmatrix}$$



3. 基于回环的位姿修正

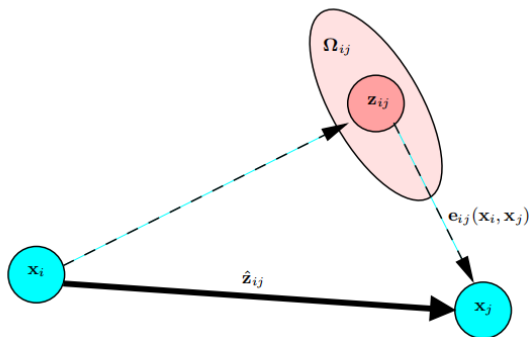
位姿图优化是把所有的观测和状态放在一起优化，残差项是前面所讲的残差项的总和。

在实际使用中，各残差会被分配一个权重，也就是信息矩阵，它相当于对残差进行加权。

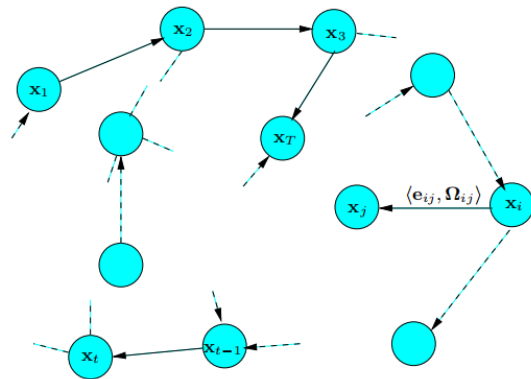
考虑信息矩阵后，总的残差项可以表示为：

$$F(\mathbf{x}) = \sum_{\langle i,j \rangle \in \mathcal{C}} \underbrace{\mathbf{e}_{ij}^T \boldsymbol{\Omega}_{ij} \mathbf{e}_{ij}}_{F_{ij}}$$

此时优化问题可以表示为： $\mathbf{x}^* = \underset{\mathbf{x}}{\operatorname{argmin}} F(\mathbf{x})$



单个残差项



多个残差项组成位姿图



3. 基于回环的位姿修正

第 i 帧和第 j 帧之间的相对位姿，在李群 $SE(3)$ 上可以表示为

$$\mathbf{T}_{ij} = \mathbf{T}_i^{-1} \mathbf{T}_j$$

也可以在李代数上表示为

$$\begin{aligned} \xi_{ij} &= \ln (\mathbf{T}_i^{-1} \mathbf{T}_j)^\vee \\ &= \ln (\exp ((-\xi_i)^\wedge) \exp (\xi_j^\wedge))^\vee \end{aligned}$$

若位姿没有误差，则上面两个式子是精确相等的，但当位姿有误差存在时，便可以使用等式的左右两端计算残差项。

$$\begin{aligned} e_{ij} &= \ln (\mathbf{T}_{ij}^{-1} \mathbf{T}_i^{-1} \mathbf{T}_j)^\vee \\ &= \ln (\exp ((-\xi_{ij})^\wedge) \exp ((-\xi_i)^\wedge) \exp (\xi_j^\wedge))^\vee \end{aligned}$$

位姿图优化的思想是通过调整状态量(即位姿)，使残差项的值最小化，这就需要用残差项对位姿求雅可比，才能使用高斯牛顿方法进行优化。

求雅可比的方式是对位姿添加扰动，此时残差表示为：

$$\hat{e}_{ij} = \ln (\mathbf{T}_{ij}^{-1} \mathbf{T}_i^{-1} \exp ((-\delta \xi_i)^\wedge) \exp (\delta \xi_j^\wedge) \mathbf{T}_j)^\vee$$



3. 基于回环的位姿修正

进一步对前面的式子进行化简

$$\begin{aligned}
 \hat{e}_{ij} &= \ln \left(T_{ij}^{-1} T_i^{-1} \exp \left((-\delta \xi_i)^\wedge \right) \exp \left(\delta \xi_j^\wedge \right) T_j \right)^\vee \\
 &\stackrel{①}{=} \ln \left(T_{ij}^{-1} T_i^{-1} \exp \left((-\delta \xi_i)^\wedge \right) T_j \exp \left(\left(\text{Ad} \left(T_j^{-1} \right) \delta \xi_j \right)^\wedge \right) \right)^\vee \\
 &\stackrel{②}{=} \ln \left(T_{ij}^{-1} T_i^{-1} T_j \exp \left(\left(-\text{Ad} \left(T_j^{-1} \right) \delta \xi_i \right)^\wedge \right) \exp \left(\left(\text{Ad} \left(T_j^{-1} \right) \delta \xi_j \right)^\wedge \right) \right)^\vee \\
 &\stackrel{③}{\approx} \ln \left(T_{ij}^{-1} T_i^{-1} T_j \exp \left(\left(-\text{Ad} \left(T_j^{-1} \right) \delta \xi_i \right)^\wedge + \left(\text{Ad} \left(T_j^{-1} \right) \delta \xi_j \right)^\wedge \right) \right)^\vee \\
 &\stackrel{④}{=} \ln \left(\exp \left(e_{ij}^\wedge \right) \exp \left(\left(-\text{Ad} \left(T_j^{-1} \right) \delta \xi_i \right)^\wedge + \left(\text{Ad} \left(T_j^{-1} \right) \delta \xi_j \right)^\wedge \right) \right)^\vee \\
 &\stackrel{⑤}{\approx} e_{ij} - \mathcal{J}_r^{-1} \left(e_{ij} \right) \text{Ad} \left(T_j^{-1} \right) \delta \xi_i + \mathcal{J}_r^{-1} \left(e_{ij} \right) \text{Ad} \left(T_j^{-1} \right) \delta \xi_j
 \end{aligned}$$

- ① 三维变换下的伴随性质
- ② 三维变换下的伴随性质
- ③ BCH公式，且是两个李代数都比较小情况下的近似
- ④ 残差的定义
- ⑤ 三维变化下的伴随性质

上面的式子表明，残差关于 T_i 的雅可比为

$$A_{ij} = \frac{\partial e_{ij}}{\partial \delta \xi_i} = -\mathcal{J}_r^{-1} \left(e_{ij} \right) \text{Ad} \left(T_j^{-1} \right)$$

残差关于 T_j 的雅可比为

$$B_{ij} = \frac{\partial e_{ij}}{\partial \delta \xi_j} = \mathcal{J}_r^{-1} \left(e_{ij} \right) \text{Ad} \left(T_j^{-1} \right)$$

其中

$$\mathcal{J}_r^{-1} \left(e_{ij} \right) \approx I + \frac{1}{2} \begin{bmatrix} \phi_e^\wedge & \rho_e^\wedge \\ 0 & \phi_e^\wedge \end{bmatrix}$$



3. 基于回环的位姿修正

按照高斯牛顿法的流程，需要对残差进行一阶泰勒展开，即求雅可比

$$\begin{aligned} \mathbf{e}_{ij}(\mathbf{x}_i + \Delta \mathbf{x}_i, \mathbf{x}_j + \Delta \mathbf{x}_j) \\ &= \mathbf{e}_{ij}(\mathbf{x} + \Delta \mathbf{x}) \\ &\approx \mathbf{e}_{ij} + \mathbf{J}_{ij} \Delta \mathbf{x} \end{aligned}$$

其中 \mathbf{J}_{ij} 即为前面推导的残差关于位姿的雅可比组成的矩阵

$$\mathbf{J}_{ij} = (0 \cdots 0 \underbrace{\mathbf{A}_{ij}}_{\text{node } i} 0 \cdots 0 \underbrace{\mathbf{B}_{ij}}_{\text{node } j} 0 \cdots 0)$$

对于每一个残差块，便有

$$\begin{aligned} \mathbf{F}_{ij}(\mathbf{x} + \Delta \mathbf{x}) \\ &= \mathbf{e}_{ij}(\mathbf{x} + \Delta \mathbf{x})^T \Omega_{ij} \mathbf{e}_{ij}(\mathbf{x} + \Delta \mathbf{x}) \\ &\approx (\mathbf{e}_{ij} + \mathbf{J}_{ij} \Delta \mathbf{x})^T \Omega_{ij} (\mathbf{e}_{ij} + \mathbf{J}_{ij} \Delta \mathbf{x}) \\ &= \underbrace{\mathbf{e}_{ij}^T \Omega_{ij} \mathbf{e}_{ij}}_{c_{ij}} + 2 \underbrace{\mathbf{e}_{ij}^T \Omega_{ij} \mathbf{J}_{ij}}_{\mathbf{b}_{ij}^T} \Delta \mathbf{x} + \Delta \mathbf{x}^T \underbrace{\mathbf{J}_{ij}^T \Omega_{ij} \mathbf{J}_{ij}}_{\mathbf{H}_{ij}} \Delta \mathbf{x} \\ &= c_{ij} + 2\mathbf{b}_{ij}^T \Delta \mathbf{x} + \Delta \mathbf{x}^T \mathbf{H}_{ij} \Delta \mathbf{x} \end{aligned}$$

其中

$$\mathbf{H}_{ij} = \begin{pmatrix} \ddots & & & \\ & \mathbf{A}_{ij}^T \Omega_{ij} \mathbf{A}_{ij} & \cdots & \mathbf{A}_{ij}^T \Omega_{ij} \mathbf{B}_{ij} \\ & \vdots & \ddots & \vdots \\ & \mathbf{B}_{ij}^T \Omega_{ij} \mathbf{A}_{ij} & \cdots & \mathbf{B}_{ij}^T \Omega_{ij} \mathbf{B}_{ij} \\ & \ddots & & \end{pmatrix} \quad \mathbf{b}_{ij} = \begin{pmatrix} \vdots \\ \mathbf{A}_{ij}^T \Omega_{ij} \mathbf{e}_{ij} \\ \vdots \\ \mathbf{B}_{ij}^T \Omega_{ij} \mathbf{e}_{ij} \\ \vdots \end{pmatrix}$$

此后便可以使用高斯牛顿法进行优化。



4. 基于先验观测的位姿修正

先验观测是一元边，它不像前面所述的帧间观测连接两个位姿状态，而是只连接一个位姿状态量，它直接给出的就是该状态量的观测值，因此它对应的残差就是观测值与状态量之间的差异，即

$$\begin{aligned} e_i &= \ln \left(Z_i^{-1} T_i \right)^\vee \\ &= \ln \left(\exp \left((-\xi_{zi})^\wedge \right) \exp \left(\xi_i^\wedge \right) \right)^\vee \end{aligned}$$

对残差添加扰动，可得

$$\hat{e}_i = \ln \left(Z_i^{-1} \exp \left(\delta \xi_i^\wedge \right) T_i \right)^\vee$$

利用伴随性质和BCH公式进行化简，可得

$$\begin{aligned} \hat{e}_i &= \ln \left(Z_i^{-1} T_i \exp \left(\left(\text{Ad} \left(T_i^{-1} \right) \delta \xi_i \right)^\wedge \right) \right)^\vee \\ &= \ln \left(\exp \left(e_i^\wedge \right) \exp \left(\left(\text{Ad} \left(T_i^{-1} \right) \delta \xi_i \right)^\wedge \right) \right)^\vee \\ &\approx e_i + \mathcal{J}_r^{-1} \left(e_i \right) \text{Ad} \left(T_i^{-1} \right) \delta \xi_i \end{aligned}$$

因此残差关于 T_i 的雅可比为

$$\frac{\partial e_i}{\partial \delta \xi_i} = \mathcal{J}_r^{-1} \left(e_i \right) \text{Ad} \left(T_i^{-1} \right)$$

其中

$$\mathcal{J}_r^{-1} \left(e_i \right) \approx I + \frac{1}{2} \begin{bmatrix} \phi_e^\wedge & \rho_e^\wedge \\ 0 & \phi_e^\wedge \end{bmatrix}$$

后面的推导过程，便与相对位姿做观测的推导过程完全一致。

此外，部分场合提供的观测只有位置，没有姿态，比如只有RTK，而没有组合导航，这里的残差便只剩下位置误差。相应的雅可比公式，可自行推导。



目录



1. 回环检测



2. 后端优化



3. 点云地图建立



4. 基于地图的定位



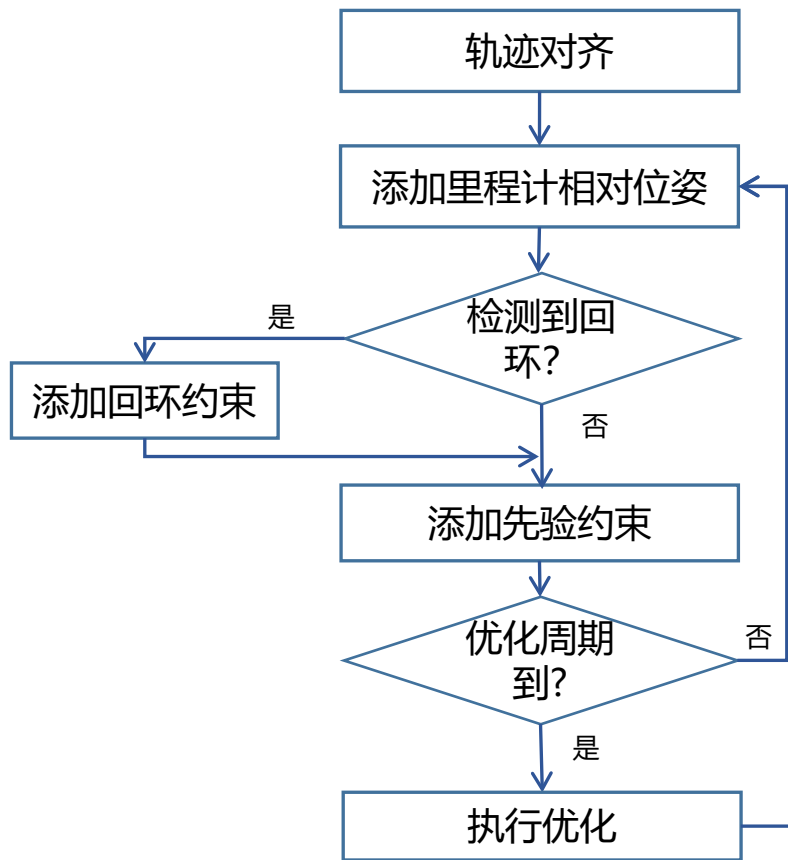
5. LeGO-LOAM介绍



点云地图建立

1. 整体流程

建图流程设计的核心原则是准确、高效地把里程计相对位姿、回环相对位姿、惯导先验位姿进行融合。



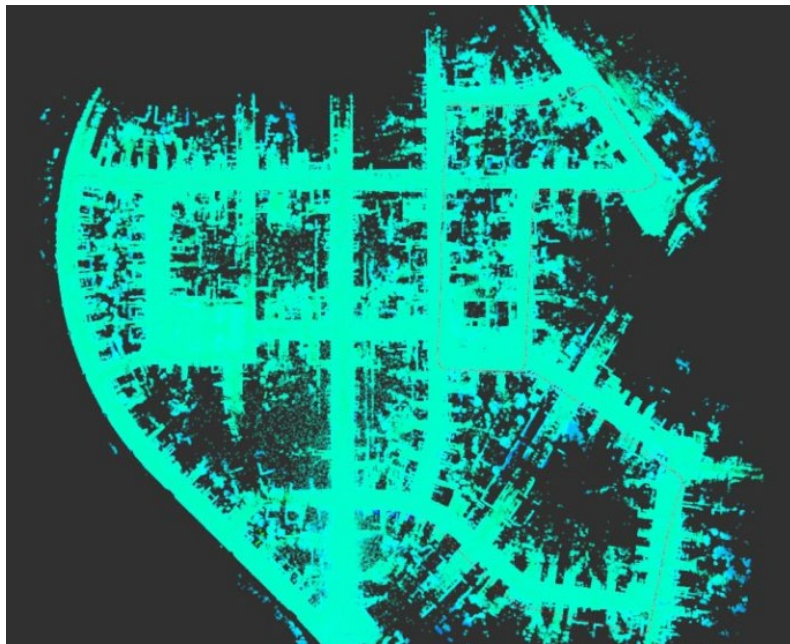
点云地图建立流程图



2. 建图流程代码讲解

“低耦合、高内聚”变成思想的运用。

最终建图效果





目录



1. 回环检测



2. 后端优化



3. 点云地图建立



4. 基于地图的定位



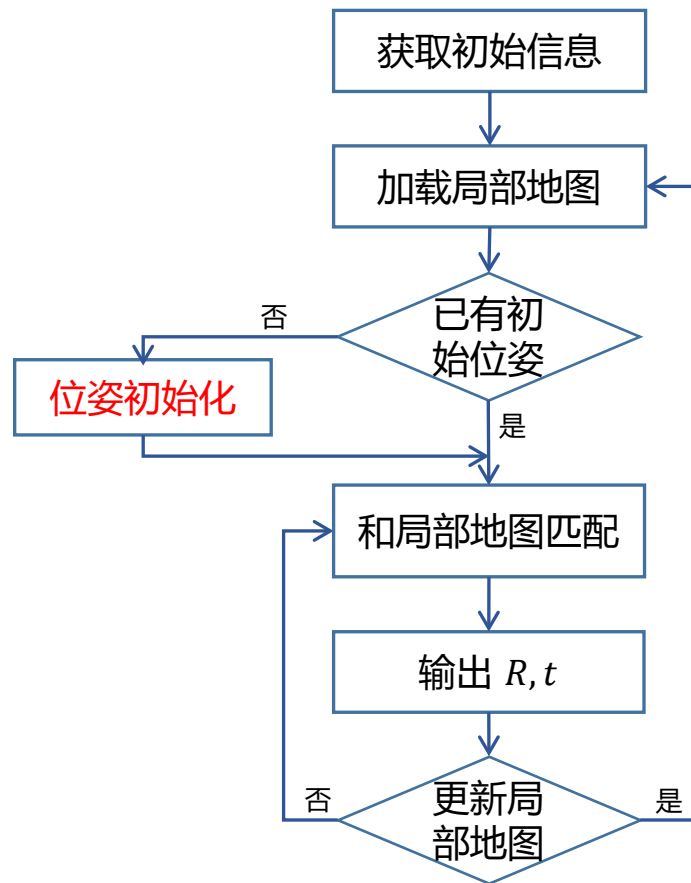
5. LeGO_LOAM介绍



基于地图的定位

1. 整体流程

在地图匹配中，鲁棒性和运行速度更加重要，因此实际使用中，基于NDT的匹配使用更广泛。



基于地图定位流程图



基于地图的定位

2. 位姿初始化

由于NDT匹配需要较准确的初始位姿，因此在定位之前需要初始化环节，给出载体的初始位姿。

按照难度由低到高，常见的初始化需求有这样几种：

- 1) 已知位姿的初始化
- 2) 位置已知而姿态位置的初始化
- 3) 位置和姿态均未知的初始化



目录



1. 回环检测



2. 后端优化



3. 点云地图建立



4. 基于地图的定位



5. LeGO-LOAM介绍

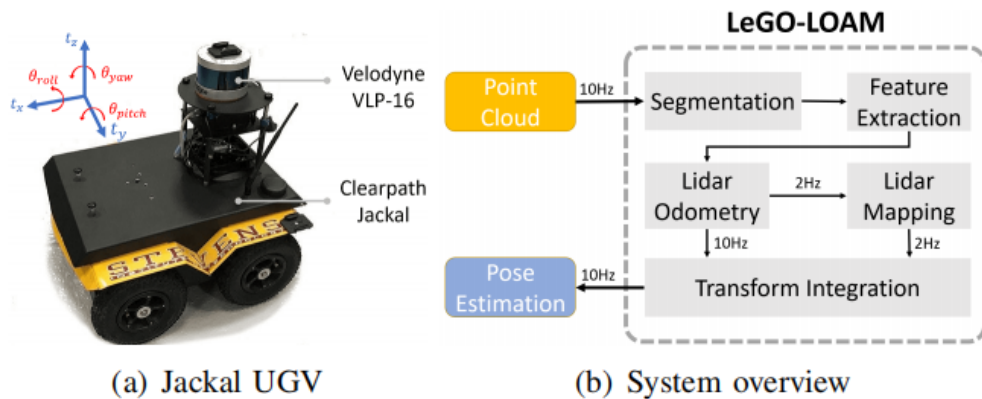


LeGO-LOAM介绍

LOAM系列—LeGO-LOAM

1. 主要特点

- 1) 对地面做了分割，减小了特征搜索范围；
- 2) 提取特征之前做了聚类，提高了特征质量；
- 3) 以帧为单位进行优化，使得全局地图可以多次调整，而不像LOAM那样不可修改；
- 4) 增加了回环修正。



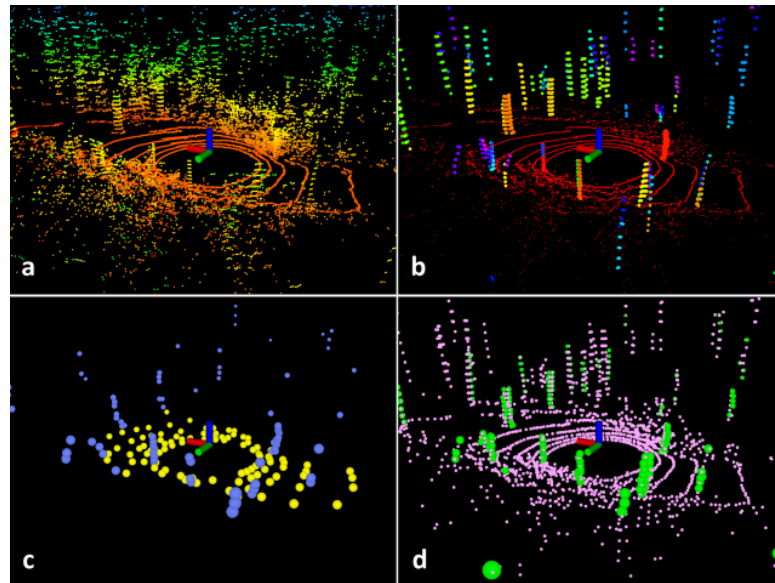


LeGO-LOAM介绍

LOAM系列—LeGO-LOAM

2. 特征提取

- 1) 根据线与线之间的夹角，以及点的曲率，筛选出地面点。所有用于匹配的平面点仅使用地面点。
- 2) 在非地面点中，使用广度优先搜索(BFS)做聚类，聚类中点的数量大于30，才用来筛选线特征。
- 3) 筛选线特征方法，与LOAM中相同。





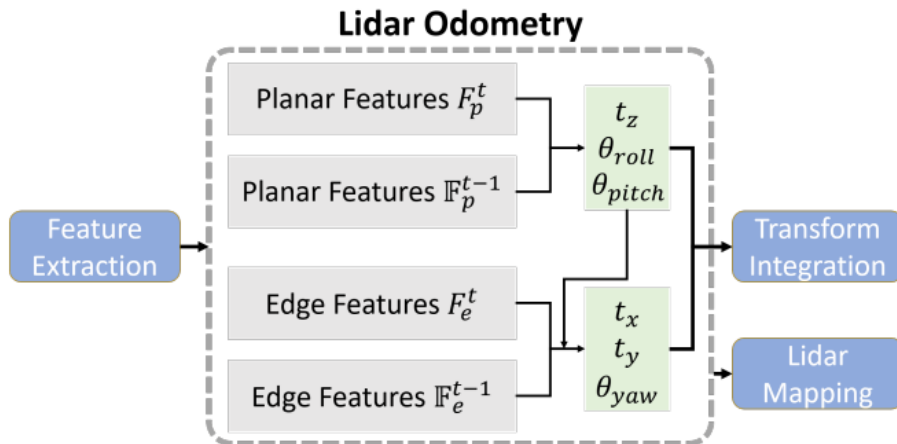
LeGO-LOAM介绍

LOAM系列—LeGO-LOAM

3. 位姿解算

- 1) 使用地面面特征优化高度和水平角;
- 2) 使用线特征优化水平位移和航向角。

注意：实际代码中，只有odom环节(scan-to-scan)使用了该模式，在mapping环节(scan-to-map)仍使用的是六自由度模型。





LeGO-LOAM介绍

LOAM系列—LeGO-LOAM

4. 回环检测与修正

- 1) 回环检测使用的初始位姿，采用前端里程计位姿
- 2) 匹配采用 ICP 方法
- 3) 优化库采用 gtsam

(讲解LeGO-LOAM代码)

论文: LeGO-LOAM: Lightweight and Ground-Optimized Lidar Odometry and Mapping on Variable Terrain

代码: <https://github.com/RobustFieldAutonomyLab/LeGO-LOAM>



作业

1) 构建点云地图

使用提供的工程代码和bag文件，跑通建图过程，并保存地图。

2) 基于点云地图定位

实际使用时，载体可能处于地图中任意位置，而当雷达点云还没有开始和地图匹配时，它所处的精确位姿是未知的，往往只有 gps 或组合导航提供的带有一定误差的粗略位姿。在任意位置，利用粗略位姿作为初始值，通过雷达点云和地图点云匹配，得到精确初始位姿的过程，叫做全局初始化。

提供的作业框架中，只能完成在地图原点的初始化，而不能实现全局初始化，全局初始化的实现为作业内容。

作业要求提供在 bag 时间的 100s、200s、300s、400s 处初始化成功的截图。

提示： bag 可以在指定时间点（例如第100s）播放，指令为 “`roslaunch bag xxx.bag -s 100`”

评价标准：

- 1) 及格：跑通建图流程、保存地图，并截图显示完整地图；
- 2) 良好：在建图的基础上，加载点云地图，实现在地图原点的初始化 (此功能提供的代码中已实现)；
- 3) 优秀：在建图的基础上，实现全局初始化的要求。

感谢聆听 !
Thanks for Listening

