

多传感器融合定位

第11讲 多传感器时空标定

主讲人 任 乾

北京理工大学本硕
自动驾驶从业者





目录



1. 多传感器标定简介



2. 内参标定



3. 外参标定



4. 时间标定



目录



1. 多传感器标定简介



2. 内参标定



3. 外参标定

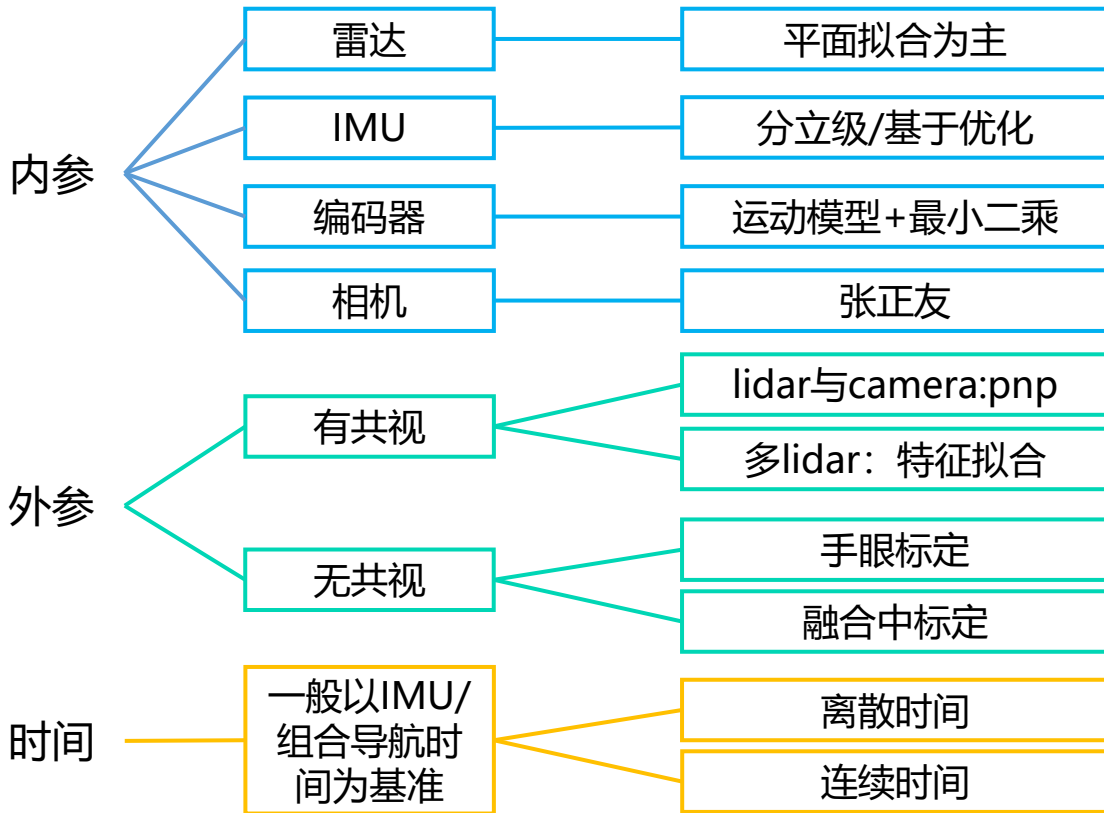


4. 时间标定



多传感器标定简介

1. 标定内容及方法





多传感器标定简介

2. 讲解思路

- 1) 以思路讲解为主，并给出参考文献和开源代码，不做过多细节展开；
- 2) 对已有方法做汇总分析，以求能在新的任务中掌握标定方案设计思路。



目录



1. 多传感器标定简介



2. 内参标定



3. 外参标定



4. 时间标定



内参标定

1. 雷达内参标定

1) 目的

由于安装原因，线束之间的夹角和设计不一致，会导致测量不准。

2) 方法

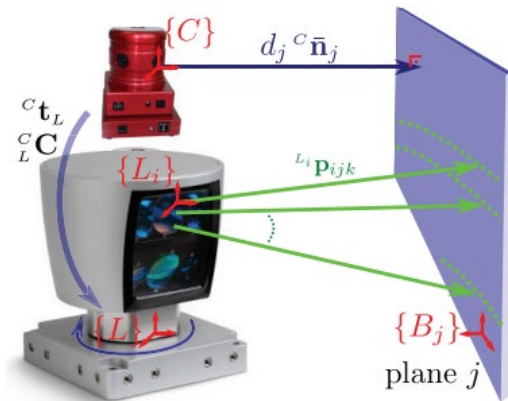
多线束打在平面上，利用共面约束，求解夹角误差。

3) 参考

论文: Calibration of a rotating multi-beam Lidar

论文: Improving the Intrinsic Calibration of a Velodyne LiDAR Sensor

论文: 3D LIDAR-camera intrinsic and extrinsic calibration: Identifiability and analytical least-squares-based initialization





内参标定

2. IMU内参标定

1) 目的

由于加工原因，产生零偏、标度因数误差、安装误差。

2) 方法

分立级标定：基于转台；

迭代优化标定：不需要转台。

3) 参考

论文：A Robust and Easy to Implement Method for IMU Calibration without External Equipments

代码：https://github.com/Kyle-ak/imu_tk



内参标定

3. 编码器内参标定

1) 目的

用编码器输出解算车的位移增量和角度增量，需已知轮子半径和两轮轴距。

2) 方法

以车中心雷达/组合导航做观测，以此为真值，反推模型参数。

3) 参考

论文： Simultaneous Calibration of Odometry and Sensor Parameters for Mobile Robots



内参标定

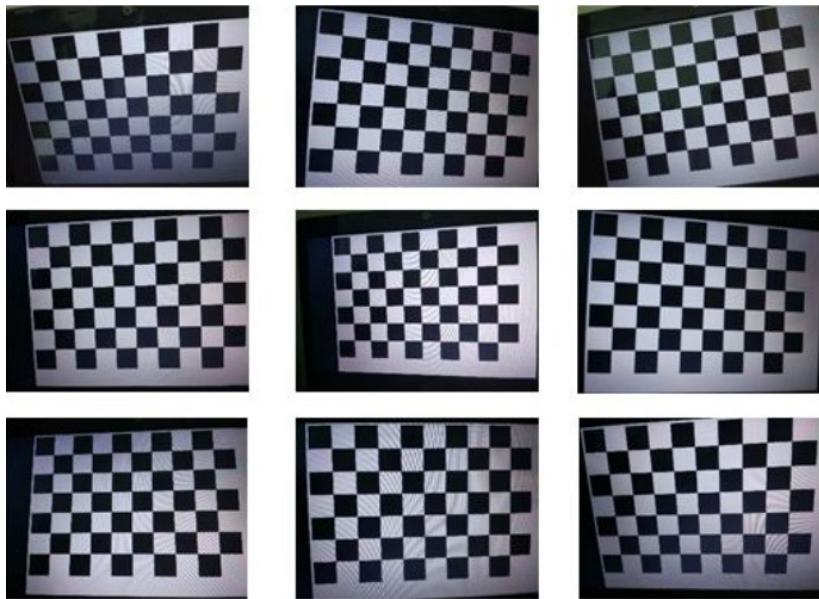
4. 相机内参标定

1) 目的

相机与真实空间建立关联，需已知其内参。

2) 方法

张正友经典方法





目录



1. 多传感器标定简介



2. 内参标定



3. 外参标定



4. 时间标定



外参标定

1. 雷达和相机外参标定

1) 目的

解算雷达和相机之间的相对旋转和平移。

2) 方法

PnP是主流，视觉提取特征点，雷达提取边缘，建立几何约束。

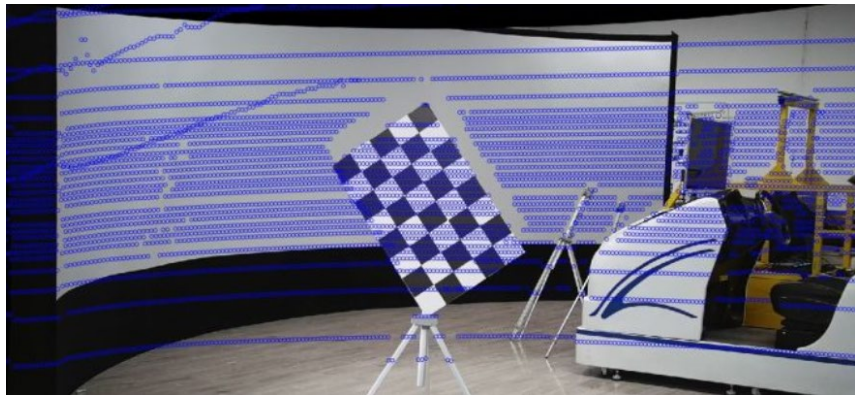
3) 参考

论文：LiDAR-Camera Calibration using 3D-3D Point correspondences

代码：https://github.com/ankitdhall/lidar_camera_calibration

论文：Automatic Extrinsic Calibration for Lidar-Stereo Vehicle Sensor Setups

代码：https://github.com/beltransen/velo2cam_calibration





外参标定

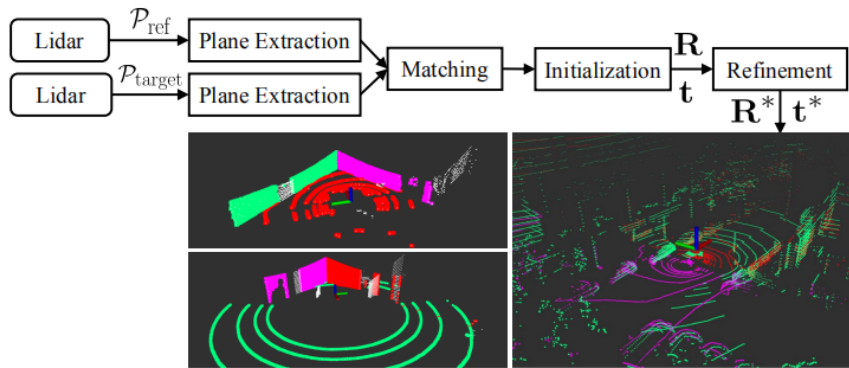
2. 多雷达外参标定

1) 目的

多雷达是常见方案，使用时将点云直接拼接，但前提是已知雷达之间的外参（相对旋转和平移）。

2) 方法

基于特征(共面)建立几何约束，从而优化外参。



3) 参考

论文: A Novel Dual-Lidar Calibration Algorithm Using Planar Surfaces

代码: https://github.com/ram-lab/lidar_appearance_calibration



外参标定

3. 手眼标定

1) 目的

手眼标定适用于所有无共视，但是能输出位姿的传感器之间标定。包括：

- 无共视的相机、雷达，或雷达与雷达之间；
- 相机与IMU，或雷达与IMU之间(前提是IMU要足够好，或直接使用组合导航)。

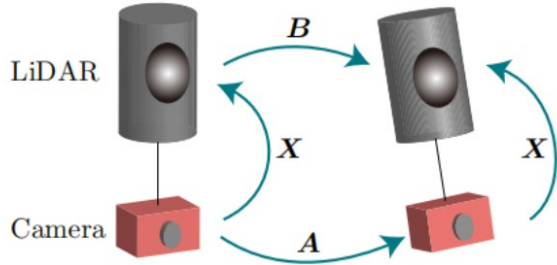
2) 方法

均基于公式 $AX = XB$

3) 参考

论文：LiDAR and Camera Calibration using Motion Estimated by Sensor Fusion Odometry

代码： https://github.com/ethz-asl/lidar_align





外参标定

4. 融合中标定

1) 目的

- 脱离标靶，实现在线标定；
- 某些器件无法提供准确位姿(如低精度IMU)，不能手眼标定。

2) 方法

在融合模型中，增加外参作为待估参数。

3) 参考

众多vio/lio系统，如vins、lio-mapping、M-Loam 等



外参标定

5. 总结

1) 这些方法中，推荐优先级从高到低为：

- a. 基于共视的标定
- b. 融合中标定
- c. 手眼标定

2) 建议

应在良好环境下标定，尽量避免不分场景的在线标定。良好环境指观测数据优良的场景，例如：

- a. GNSS 信号良好；
- b. 点云面特征丰富，没有特征退化；
- c. 动态物体较少

方法的起因是实际需求，而不是想象中的问题，更不是刻意制造的困难



目录



1. 多传感器标定简介



2. 内参标定



3. 外参标定



4. 时间标定

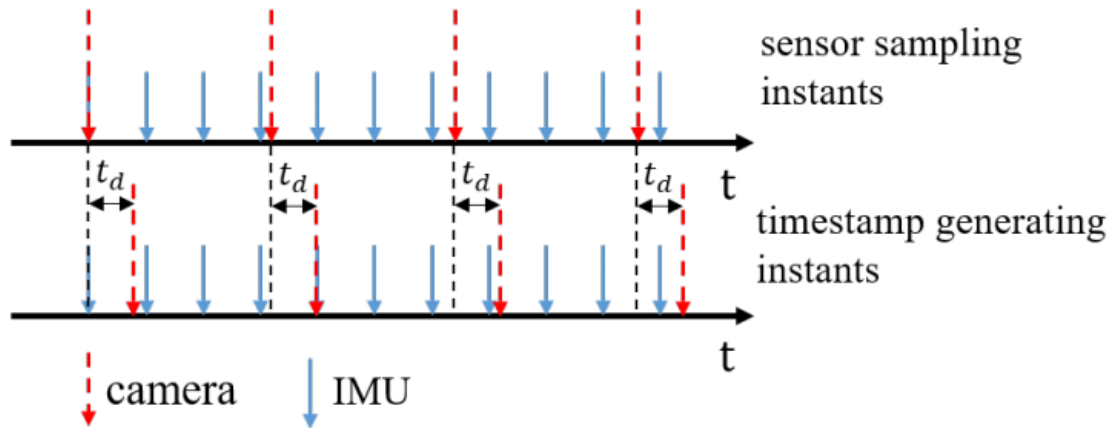


时间标定

1. 离散时间

1) 目的

在原有离散时间融合模式下，简单地解决时间同步问题。





时间标定

1. 离散时间

2) 方案 I

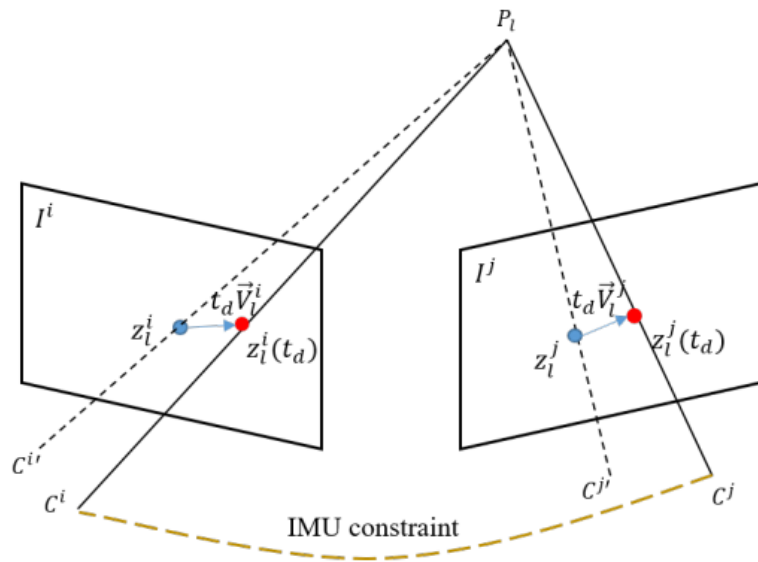
简单但巧妙的策略：IMU时间保持不变，图像上特征点基于匀速运动模型修改位置。

与不考虑时间误差时相比，架构不变，使用极小的改动，实现了期望的效果。

$$\mathbf{e}_l^k = \mathbf{z}_l^k - \pi(\mathbf{R}_{c_k}^{w^T} (\mathbf{P}_l - \mathbf{p}_{c_k}^w))$$
$$\mathbf{z}_l^k = [u_l^k \ v_l^k]^T.$$



$$\mathbf{e}_l^k = \mathbf{z}_l^k(t_d) - \pi(\mathbf{R}_{c_k}^{w^T} (\mathbf{P}_l - \mathbf{p}_{c_k}^w))$$
$$\mathbf{z}_l^k(t_d) = [u_l^k \ v_l^k]^T + \boxed{t_d \mathbf{V}_l^k}.$$





时间标定

1. 离散时间

2) 方案 II

在滤波中计算相机位姿时，直接按时间差对积分区间进行调整。

状态量
$$\mathbf{x} = [\mathbf{x}_I^T \quad {}^C_I \hat{\mathbf{q}}^T \quad {}^C_I \mathbf{p}_I^T \quad t_d \quad \mathbf{c}_1^T \quad \cdots \quad \mathbf{c}_M^T]^T$$

相机位姿估计
$$\hat{\mathbf{c}}_{new} = \begin{bmatrix} {}^C_G \widehat{\mathbf{q}}(t+\hat{t}_d) \\ {}^G_I \widehat{\mathbf{p}}_C(t+\hat{t}_d) \end{bmatrix} = \begin{bmatrix} {}^C_I \hat{\mathbf{q}} \otimes {}^I_G \hat{\mathbf{q}}(t+\hat{t}_d) \\ {}^G_I \hat{\mathbf{p}}_I(t+\hat{t}_d) + {}^I_G \hat{\mathbf{R}}(t+\hat{t}_d)^T {}^I_I \hat{\mathbf{p}}_C \end{bmatrix}$$



时间标定

2. 连续时间

1) 原因

预积分中把时间差作为待估状态量，对时间差进行建模，如下：

$$\Delta \mathbf{p}_m^i = \sum_{k=\kappa}^{i-1} \left(\Delta \mathbf{v}_m^k \Delta t_k + \frac{\Delta \mathbf{R}_m^k}{2} (\mathbf{f}(t_k - \delta_t^m) - \mathbf{b}_f^m) \Delta t_k^2 \right)$$

$$\Delta \mathbf{v}_m^i = \sum_{k=\kappa}^{i-1} \Delta \mathbf{R}_m^k (\mathbf{f}(t_k - \delta_t^m) - \mathbf{b}_f^m) \Delta t_k$$

$$\Delta \mathbf{R}_m^i = \prod_{k=\kappa}^{i-1} \text{Exp}((\boldsymbol{\omega}(t_k - \delta_t^m) - \mathbf{b}_\omega^m) \Delta t_k),$$

由于要对时间差求雅可比，因此插值函数必须可导，雅可比如下：

$$\begin{aligned} \Delta \mathbf{R}_m^i(\mathbf{b}_\omega, \delta_t) &\approx \Delta \mathbf{R}_m^i(\bar{\mathbf{b}}_\omega^m, \bar{\delta}_t^m) \text{Exp}\left(\frac{\partial \Delta \mathbf{R}_m^i}{\partial \mathbf{b}_\omega} \hat{\mathbf{b}}_\omega^m + \frac{\partial \Delta \mathbf{R}_m^i}{\partial \delta_t} \hat{\delta}_t^m\right) \\ \Delta \mathbf{v}_m^i(\mathbf{b}_f, \mathbf{b}_\omega, \delta_t) &\approx \Delta \mathbf{v}_m^i(\bar{\mathbf{b}}_f^m, \bar{\mathbf{b}}_\omega^m, \bar{\delta}_t^m) + \frac{\partial \Delta \mathbf{v}_m^i}{\partial \mathbf{b}_f} \hat{\mathbf{b}}_f^m \\ &\quad + \frac{\partial \Delta \mathbf{v}_m^i}{\partial \mathbf{b}_\omega} \hat{\mathbf{b}}_\omega^m + \frac{\partial \Delta \mathbf{v}_m^i}{\partial \delta_t} \hat{\delta}_t^m \\ \Delta \mathbf{p}_m^i(\mathbf{b}_f, \mathbf{b}_\omega, \delta_t) &\approx \Delta \mathbf{p}_m^i(\bar{\mathbf{b}}_f^m, \bar{\mathbf{b}}_\omega^m, \bar{\delta}_t^m) + \frac{\partial \Delta \mathbf{p}_m^i}{\partial \mathbf{b}_f} \hat{\mathbf{b}}_f^m \\ &\quad + \frac{\partial \Delta \mathbf{p}_m^i}{\partial \mathbf{b}_\omega} \hat{\mathbf{b}}_\omega^m + \frac{\partial \Delta \mathbf{p}_m^i}{\partial \delta_t} \hat{\delta}_t^m \end{aligned}$$



时间标定

2. 连续时间

2) 方法

把输入建立为连续时间函数，从而可以在任意时间求导。

3) 参考

a. kalibr 系列

论文: Continuous-Time Batch Estimation using Temporal Basis Functions

论文: Unified Temporal and Spatial Calibration for Multi-Sensor Systems

论文: Extending kalibr Calibrating the Extrinsic of Multiple IMUs and of Individual Axes

代码: <https://github.com/ethz-asl/kalibr>

b. 其他

论文: Targetless Calibration of LiDAR-IMU System Based on Continuous-time Batch Estimation

代码: https://github.com/APRIL-ZJU/lidar_IMU_calib



时间标定

3. 总结

- 1) 时间差估计，在某些情况下不得已而为之，实际中应尽量创造条件实现硬同步；
- 2) 不得不估计时，也应尽量在良好环境下估计。

感谢聆听 !
Thanks for Listening

