计算机视觉life

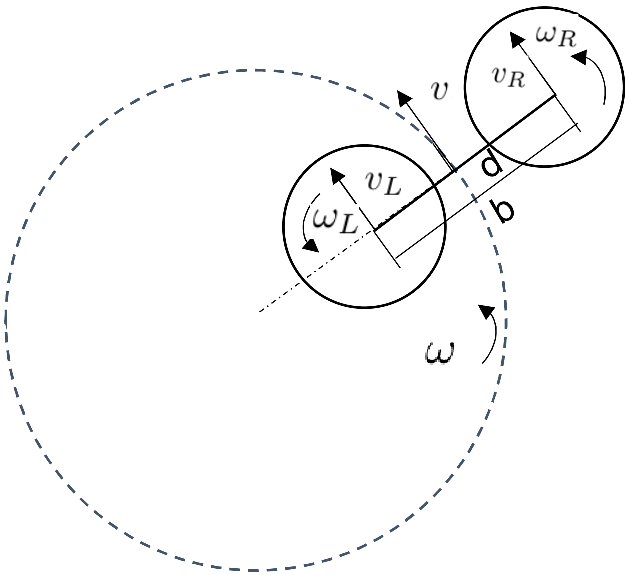
激光SLAM数据预处理

曹秀洁
2020.05.24

资料来源：Probabilistic robotics , Sebastian Thrun
激光slam课程ppt，深蓝学院第二期，曾书格

目录

1. 两轮差速底盘的运动学模型



ω_L ω_R 分别代表左轮和右轮的角速度

v_L v_R 分别代表左轮和右轮的线速度

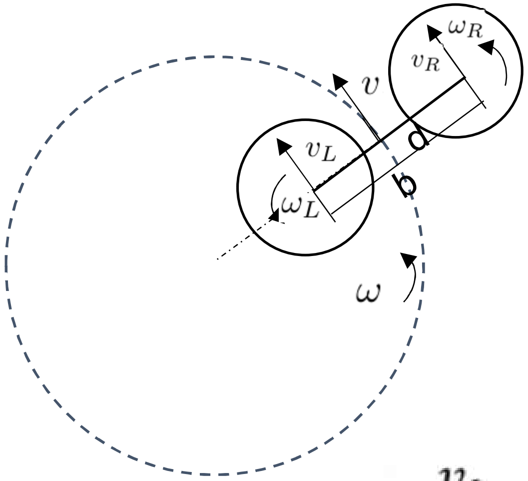
v ω 分别代表小车中心线速度和角速度

D为轮子离底盘中心的距离，b为两轮之间的距离

$$\mathrm{v} = \frac{\mathrm{v}_R + v_L}{2} = \frac{\omega_R \cdot r_R + \omega_L \cdot r_L}{2} \qquad \omega = \frac{\mathrm{v}_R - v_L}{2\mathrm{d}} = \frac{\omega_R \cdot r_R - \omega_L \cdot r_L}{2d}$$

$$\begin{pmatrix} \mathrm{v} \\ \omega \end{pmatrix} = \begin{bmatrix} \frac{\mathrm{r}_L}{2} & \frac{r_R}{2} \\ -\frac{r_L}{b} & \frac{r_R}{b} \end{bmatrix} \begin{pmatrix} \omega_L \\ \omega_R \end{pmatrix} = J \cdot \begin{pmatrix} \omega_L \\ \omega_R \end{pmatrix}$$

1. 两轮差速底盘的运动学模型



假设：

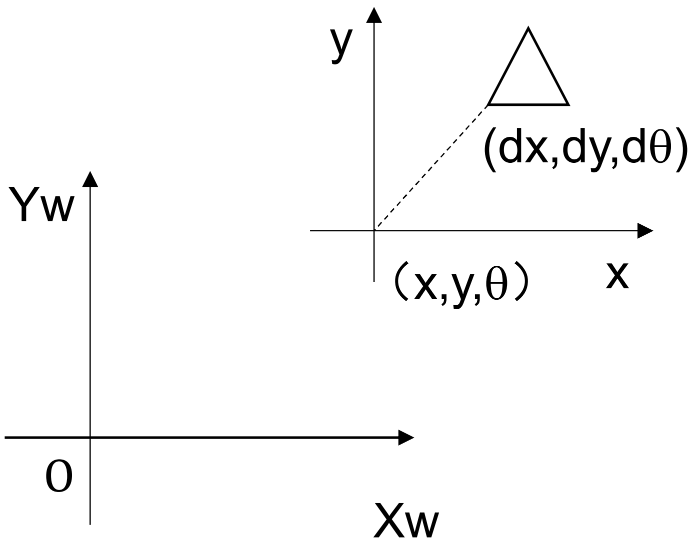
- 刚体小车做圆弧运动
- 欠驱动系统：运动耦合

R为底盘中心圆弧运动的半径

$$\frac{v_L}{r-d}=\frac{v_R}{r+d}$$
$$v_L(\mathrm{r}+\mathrm{d})=v_R(r-d)$$
$$(v_R-v_L)\mathrm{r}=(v_R+v_L)\mathrm{d}$$
$$\mathrm{r}=\frac{(v_R+v_L)\mathrm{d}}{(v_R-v_L)}$$

$$\mathrm{v}=\omega*\mathrm{r}=\frac{(v_R-v_L)}{2d}\frac{(v_R+v_L)\mathrm{d}}{(v_R-v_L)}=\frac{v_R+v_L}{2}$$
$$\mathrm{v}=\frac{v_R+v_L}{2}$$

1. 坐标变换



- (x, y, θ)为底盘当前位姿
- $(dx, dy, d\theta)$为运动学解算增量

$$\begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} dx \\ dy \\ d\theta \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} dx + \varepsilon_x \\ dy + \varepsilon_y \\ d\theta + \varepsilon_\theta \end{bmatrix}$$

线性最小二乘的基本原理

Ax表示A的列向量空间S，

无解意味着向量b不在S中

最近的解即为向量b在空间S中的投影

设Ax^*为向量b在空间S中的投影，显然$(b-Ax^*)$垂直于空间S。

$(b-Ax^*)$跟矩阵A的每一个列向量都垂直

设：

$$A=[a_1,a_2,\cdots a_n]$$

a_i表示矩阵A的第i个列向量

则：

$$a_i^T(b-Ax^*)=0$$

可得：

$$A^T(b-Ax^*)=0$$

$$A^Tb=A^TAx^*$$

$$x^*=(A^TA)^{-1}A^Tb$$

直接线性方法

- 通用性强
- 实现简单
- 精度不高

基于模型的方法

- 特异性高
- 实现复杂
- 精度高

2. 里程计标定——直接线性方法

计算机视觉life

- 用激光雷达的scan-match数据作为真值u_i^*
- 里程计测量得到的数据为u_i
- 假设成线性关系$u_i^* = X * u_i$

其中:

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix}$$

对于每一组数据，可得：

$$u_{ix} * x_{11} + u_{iy} * x_{12} + u_{i\theta} * x_{13} = u_{ix}^*$$

$$u_{ix} * x_{21} + u_{iy} * x_{22} + u_{i\theta} * x_{23} = u_{iy}^*$$

$$u_{ix} * x_{31} + u_{iy} * x_{32} + u_{i\theta} * x_{33} = u_{i\theta}^*$$

$$\begin{bmatrix} u_{ix} & u_{iy} & u_{i\theta} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & u_{ix} & u_{iy} & u_{i\theta} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & u_{ix} & u_{iy} & u_{i\theta} \end{bmatrix} \begin{bmatrix} x_{11} \\ \vdots \\ x_{33} \end{bmatrix} = \begin{bmatrix} u_{ix}^* \\ u_{iy}^* \\ u_{i\theta}^* \end{bmatrix}$$

$$A_i\vec{X} = b_i$$

$$A = \begin{matrix} A_1 \\ \vdots \\ A_n \end{matrix} \qquad b = \begin{matrix} b_1 \\ \vdots \\ b_n \end{matrix} \qquad \vec{X} = (A^T A)^{-1} A^T b$$

运动学模型

$$\begin{pmatrix} v \\ \omega \end{pmatrix} = \begin{bmatrix} \frac{r_L}{2} & \frac{r_R}{2} \\ -\frac{r_L}{b} & \frac{r_R}{b} \end{bmatrix} \begin{pmatrix} \omega_L \\ \omega_R \end{pmatrix} = J \cdot \begin{pmatrix} \omega_L \\ \omega_R \end{pmatrix}$$

$$\theta(t) = \int \omega(t) dt$$

$$x(t) = \int v(t) \cos(\theta(t)) dt$$

$$y(t) = \int v(t) \sin(\theta(t)) dt$$

匀速运动假设

$$\omega(t) = \omega = J_{21}\omega_L + J_{22}\omega_R$$

$$v(t) = v = J_{11}\omega_L + J_{12}\omega_R$$

$$v(t) = v = \frac{b}{2}(-J_{21}\omega_L + J_{22}\omega_R)$$

已知两个轮子的角速度，需要求解两个轮子的半径$r_{L,}$,r_R和两轮之间的距离b。

2. 里程计标定——基于模型的方法

计算机视觉life

- ◆ 假设激光雷达位于车体正中心
- ◆ 激光雷达匹配值作为观测值
- ◆ 里程计的积分值作为预测值
- ◆ 通过最小化预测值和观测值的差即可得到里程计的参数
- ◆ 里程计积分值用r_x, r_y, r_{theta}表示，激光雷达的匹配值用s_x, s_y, s_{theta}表示。

- 角度积分表达式

$$r_\theta(t) = \int \omega(t)\mathrm{d}t = \int J_{21}\omega_L + J_{22}\omega_R dt$$

$$r_\theta(t) = \begin{pmatrix} \omega_L \cdot \Delta T & \omega_R \cdot \Delta T \end{pmatrix} \begin{pmatrix} J_{21} \\ J_{22} \end{pmatrix}$$

- 罗列方程组

$$\begin{bmatrix} \omega_{L0} \cdot \Delta T_0 & \omega_{R0} \cdot \Delta T_0 \\ \omega_{L1} \cdot \Delta T_1 & \omega_{R1} \cdot \Delta T_1 \\ \vdots & \vdots \\ \omega_{Ln} \cdot \Delta T_n & \omega_{Rn} \cdot \Delta T_n \end{bmatrix} \bullet \begin{bmatrix} J_{21} \\ J_{22} \end{bmatrix} = \begin{bmatrix} S_{\theta 0} \\ S_{\theta 1} \\ \vdots \\ S_{\theta n} \end{bmatrix}$$

$$Ax = b$$

2. 里程计标定——基于模型的方法

计算机视觉life

- 套用线性最小二乘的求解公式：

$$\begin{bmatrix} J_{21} \\ J_{22} \end{bmatrix} = (A^T A)^{-1} A^T b$$

- 在已经J_21和J_22的情况下，里程计的位置积分和参数b呈线性关系，推导过程如下：

$$r_x(t) = \int v(t)\cos(\theta(t))dt$$

$$= \frac{b}{2}(-J_{21}\omega_L + J_{22}\omega_R)\int \cos(\theta(t))dt = c_x b$$

$$r_y(t) = \int v(t)\sin(\theta(t))dt$$

$$= \frac{b}{2}(-J_{21}\omega_L + J_{22}\omega_R)\int \sin(\theta(t))dt = c_y b$$

- 罗列方程组

$$\begin{bmatrix} c_{x0} \\ c_{y0} \\ \vdots \\ \vdots \\ c_{xn} \\ c_{yn} \end{bmatrix} \bullet b = \begin{bmatrix} s_{x0} \\ s_{y0} \\ \vdots \\ \vdots \\ s_{xn} \\ s_{yn} \end{bmatrix}$$

- 求解此方程组，得到参数b。

2. 里程计标定——基于模型的方法

计算机视觉life

- 已知参数b和J_21和J22，可以得到：

$$J_{21} = -\frac{r_L}{b} \qquad J_{22} = \frac{r_R}{b}$$

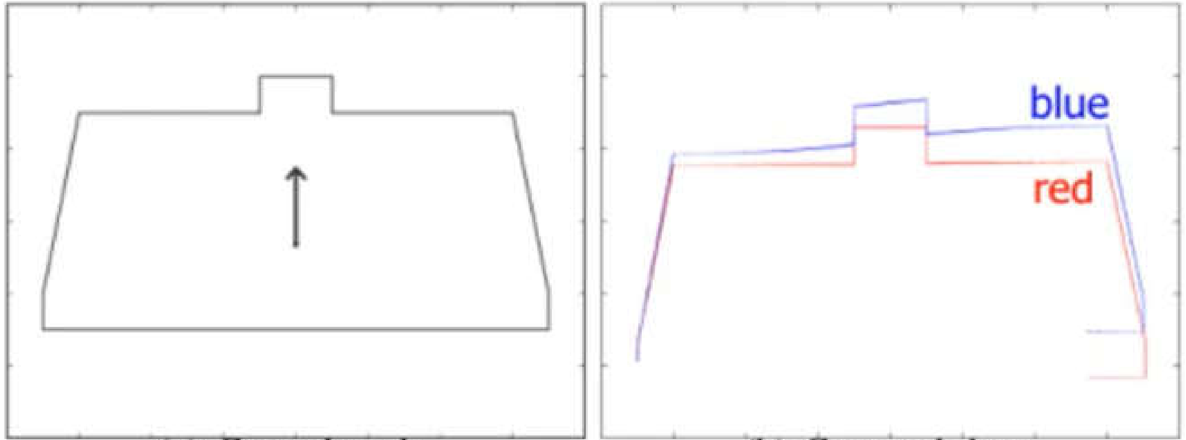
- 因此，可得：

$$r_L = -J_{21} \cdot b \qquad r_R = J_{22} \cdot b$$

- 两轮直径和轮间距都已知，求解完毕

- 收集n段数据，每段数据包含两个轮子的角速度(w_L,w_R)、该段数据持续的时间dT以及激光雷达的匹配值
- 按照上面介绍的公式，计算中间变量J_21和J_22
- 按照上面介绍的公式，计算轮间距b
- 用轮间距b、J_21和J_22，计算两个轮子的半径。

运动畸变示意图



(a) Ground truth (b) Captured data

- 激光点数据不是瞬时获得
- 激光测量时伴随着机器人的运动

！激光帧率较低时，机器人的运动不能忽略

4. 运动畸变消除——位姿表示及坐标转换

- 机器人B在坐标系O中的坐标：

$$(x, y, \theta)$$

- 坐标系B到坐标O的转换矩阵：

$$T_{OB} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & x \\ \sin(\theta) & \cos(\theta) & y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}$$

- 坐标系O到坐标B的转换矩阵：

$$T_{BO} = \begin{bmatrix} R^T & -R^T t \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R^{-1} & -R^{-1} t \\ 0 & 1 \end{bmatrix}$$

- 机器人A在坐标系O的坐标：

$$(x_A, y_A, \theta_A), \begin{bmatrix} \cos(\theta_A) & -\sin(\theta_A) & x_A \\ \sin(\theta_A) & \cos(\theta_A) & y_A \\ 0 & 0 & 1 \end{bmatrix}$$

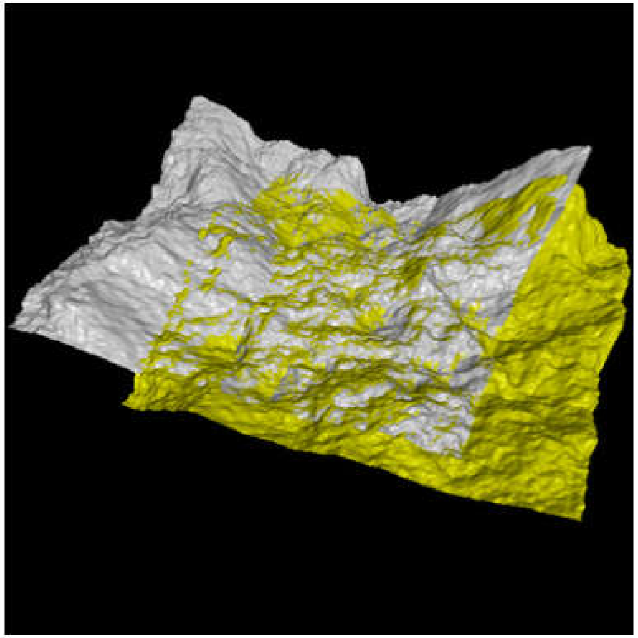
- 机器人B在坐标系O的坐标：

$$(x_B, y_B, \theta_B), \begin{bmatrix} \cos(\theta_B) & -\sin(\theta_B) & x_B \\ \sin(\theta_B) & \cos(\theta_B) & y_B \\ 0 & 0 & 1 \end{bmatrix}$$

- 机器人A在机器人B中的坐标：

$$T_{BA} = T_{OB}^{-1} \bullet T_{OA}$$

$$(x, y, \theta) = (T_{BA}(0,2), T_{BA}(1,2), a\tan 2(T_{BA}(1,0), T_{BA}(0,0)))$$



- Given two point sets:

  $Q = \{\boldsymbol{q}_1, \ldots, \boldsymbol{q}_N\} \qquad P = \{\boldsymbol{p}_1, \ldots, \boldsymbol{p}_M\}$

  with correspondences $C = \{(i, j)\}$
- Wanted: Translation $\boldsymbol{t}$ and rotation R that minimize the sum of the squared errors:

$$E(R, \boldsymbol{t}) = \sum_{(i,j)\in C} \|\boldsymbol{q}_i - R\boldsymbol{p}_j - \boldsymbol{t}\|^2$$

$$u_x = \frac{1}{N_p}\sum_{i=1}^{N_p} x_i \qquad u_p = \frac{1}{N_p}\sum_{i=1}^{N_p} p_i$$

$$X' = \{x_i - u_x\} = \{x_i'\}$$

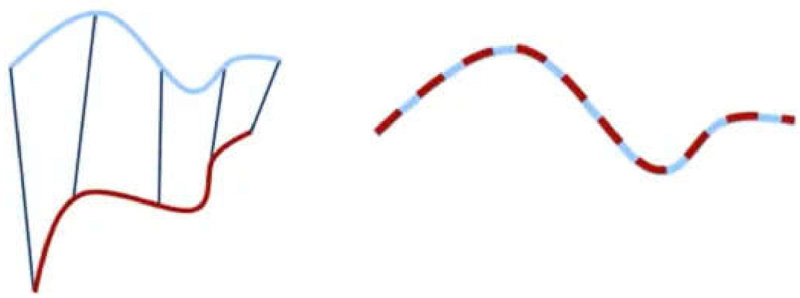
$$P' = \{p_i - u_p\} = \{p_i'\}$$

$$W = \sum_{i=1}^{N_p} x_i' p_i'^T = U\begin{bmatrix}\sigma_1 & 0 & 0\\ 0 & \sigma_2 & 0\\ 0 & 0 & \sigma_3\end{bmatrix}V^T$$

则ICP的解为：

$$R = UV^T$$

$$\mathrm{t} = u_x - Ru_p$$



$$E(R,t)=\frac{1}{N_p}\sum_{i=1}^{N_p}\|x_i-Rp_i-t\|^2$$

$$E(R,t)=\frac{1}{N_p}\sum_{i=1}^{N_p}\|x_i-Rp_i-t-u_x+Ru_p+u_x-Ru_p\|^2$$

$$=\frac{1}{N_p}\sum_{i=1}^{N_p}\|x_i-u_x-R(p_i-u_p)+(u_x-Ru_p-t)\|^2$$

$$\frac{1}{N_p}\sum_{i=1}^{N_p}\|x_i-u_x-R(p_i-u_p)\|^2+\|u_x-Ru_p-t)\|^2$$

$$+2\left(x_i-u_x-R(p_i-u_p)\right)^T(u_x-Ru_p-t))$$

$$=\frac{1}{N_p}\sum_{i=1}^{N_p}\|x_i-u_x-R(p_i-u_p)\|^2+\|u_x-Ru_p-t)\|^2$$

$\|x_i-u_x-R(p_i-u_p)\|^2$ 只跟R有关

当已知R时可以通过u_x-Ru_p-t求解得到t

转换为最小化函数：

$$E(R,t)=\frac{1}{N_p}\sum_{i=1}^{N_p}\|x_i-u_x-R(p_i-u_p)\|^2$$

4.运动畸变去除——纯估计方法（ICP）

计算机视觉life

$$E(R,t)=\frac{1}{N_p}\sum_{i=1}^{N_p}\|x_i-u_x-R(p_i-u_p)\|^2$$

$$=\frac{1}{N_p}\sum_{i=1}^{N_p}\|x_i'-Rp_i'\|^2=\frac{1}{N_p}\sum_{i=1}^{N_p}x_i'^T x_i'+p_i'^T R^T R p_i'-2x_i'^T R p_i'$$

$$=\sum_{i=1}^{N_p}-2x_i'^T R p_i'$$

$$\sum_{i=1}^{N_p}x_i'^T R p_i'=\sum_{i=1}^{N_p}\mathrm{Trace}(Rx_i'p_i'^T)=Trace(RH)$$

$$H=\sum_{i=1}^{N_p}x_i'p_i'^T$$

- 假设矩阵A为正交对称矩阵，则对于任意的正交矩阵B，都有:

$$Trace(A)\geq Trace(BA)$$

$$H=U\Lambda V^T \qquad X=VU^T\text{--正交矩阵}$$

$$XH=VU^TU\Lambda V^T=V\Lambda V^T\text{--正定对称}$$

则：

$$Trace(XH)\geq Trace(BXH)$$

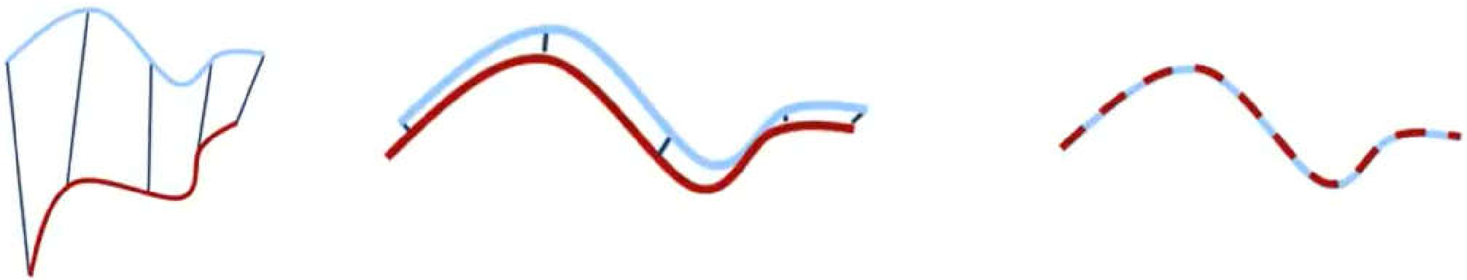
因此：$R=X=VU^T$

未知对应点的求解方法

- 实际中，不知道对应点匹配
- 不能一步到位计算出R和t
- 进行迭代计算
- EM算法的一个特例

算法流程：

- 寻找对应点
- 根据对应点，计算R和t
- 对点云进行转换，计算误差
- 不断迭代，直至误差小于某一个值



4.运动畸变去除——里程计辅助方法

ICP方法缺点

- 没有考虑激光的运动畸变
- 当前的激光数据时错误的

VICP方法缺点

- 第帧率激光（5Hz），匀速运动假设不成立
- 数据预处理和状态估计过程耦合

解决方法

- 尽可能准确的反应运动情况
- 实现预处理和状态估计的解耦

传感器辅助方法（Odom/IMU）

- 极高的位姿更新频率（200Hz），可以比较准确的反应运动情况
- 较高精度的局部位姿估计
- 跟状态估计的完全解耦

4.运动畸变去除——里程计辅助方法

- 惯性测量单元（IMU）
  - 直接测量角速度和线加速度
  - 具有较高的角速度测量精度
  - 测量频率极高（1kHz~8kHz）
  - 线加速度精度太差，二次积分在局部的精度依然很差

- 轮式里程计
  - 直接测量机器人位移和角度
  - 具有较高的局部角度测量精度
  - 具有较高的局部位置测量精度
  - 更新速度较高（100Hz~200Hz）

4.运动畸变去除——里程计辅助方法

已知数据

- 当前帧激光起始时间为t_s, t_e
- 两个激光束间的时间Δt
- 里程计数据按照时间顺序存储在一个队列中，队首的时间最早
- 最早的里程计数据的时间戳<t_s
- 最晚的里程计数据的时间戳>t_e

目标

- 求解当前帧激光数据中每一个激光点对应的机器人位姿，即求解$\{t_s, t_{s+\Delta t}, \cdots, t_e\}$时刻的机器人位姿
- 根据求解的位姿把所有激光点转换到同一坐标系下
- 重新封装成一帧激光数据，发布出去

求解t_s, t_e时刻的位姿p_s, p_e

- 里程计队列中正好和激光数据同步，假设第i和第j跟数据是时刻分别为t_s, t_e:

$$p_s = OdomList[i]$$

$$p_e = OdomList[j]$$

- 在t_s时刻没有对应的里程计位姿，则进行线性插值，设在l,k时刻有位姿，且l<s<k,则：

$$p_l = OdomList[l]$$

$$p_k = OdomList[k]$$

$$p_s = LinarInterp(p_l, p_k, \frac{s-l}{k-l})$$

二次插值

- 在一帧激光数据之间，认为机器人做匀加速运动。
- 机器人的位姿是关于时间t的二次函数。
- 设$t_m = \frac{t_s+t_e}{2}$，且l<m<k则：

$$p_m = LinarInterp(p_l, p_k, \frac{m-l}{k-l})$$

- 已知p_s, p_m, p_e，可以插值一条二次曲线：

$$P(t) = At^2 + Bt + C$$

$$t_s \le t \le t_e$$

二次曲线的近似

- 用分段线性函数对二次曲线进行近似
- 分段数大于3时，近似误差可以忽略不计
- 在t_s和t_e时间段内，一共取k个位姿$\{p_s, p_{s+1}, \cdots, p_{s+k-2}, p_e\}$
- 位姿通过线性插值获取，在这K个位姿之间，进行线性插值：

设p_s和p_{s+1}之间有N个位姿$\{p_s, p_{s1}, \cdots, p_{s(n-2)}, p_{s+1}\}$

则：

$$p_{si} = LinarInterp(p_l, p_k, \frac{si - s}{\Delta t})$$

坐标系统—&激光数据发布

- 一帧激光数据n个激光点，每个激光点对应的位姿$\{p_1, p_2, \cdots, p_n\}$通过上述介绍的方法插值得到
- x_i为转化之前的坐标，x_i'为转换之后的坐标，则：

$$x_i' = p_i^T x_i$$

- 把转换之后的坐标转换为激光数据发布出去：

$$x_i' = (p_x, p_y)$$
$$range = \sqrt{p_x * p_x + p_y * p_y}$$
$$angle = atan2(p_y, p_x)$$

轮式方法和匹配方法的结合

- 用里程计方法进行矫正，去除绝大部分的运动畸变。
- 认为里程计存在误差，但是误差值线性分布的。
- 用ICP的方法进行匹配，匹配的结果作为正确值，得到里程计的误差值。
- 把误差值均摊到每一个点上，重新进行激光点位置修正。
- 再一次进行ICP迭代，直到收敛为止。

什么是「 SLAM研习社」？

- 计算机视觉life读者自发组织，专注于SLAM的开源学习组织
- 宗旨：SLAM技术从基础到应用，懂原理会应用，完全搞透彻，不留死角
- 角色：主讲、嘉宾、联络、宣传、策划
- 报名：simiter@126.com
- 报名主讲/嘉宾不需要你是大牛，只要熟悉SLAM某个知识点即可
- SLAM研习社详细介绍、直播预告、视频回放链接
- https://github.com/electech6/LearnSLAM/blob/master/README.md

谢谢观看！

曹秀洁
Contact me
E-mail: caoxiujie123@163.com
37 Xueyuan Rd., Haidian District,
Beijing, P. R. China