激光的前端配准算法

主讲人 曾书格

越凡创新技术负责人
电子科技大学硕士

课程内容

帧间匹配算法

1、ICP匹配方法

2、PL-ICP匹配方法

3、NICP匹配方法

4、IMLS-ICP匹配方法

课程内容

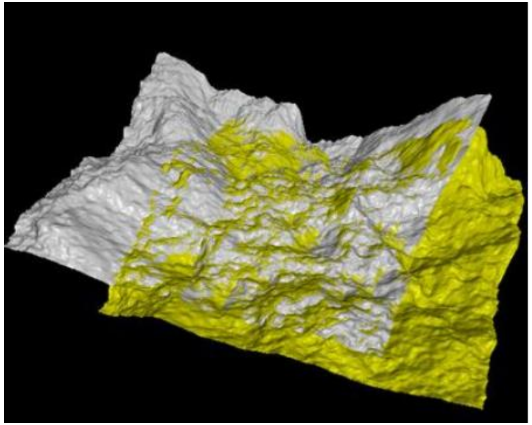
帧间匹配算法

- **1、ICP匹配方法**
- 2、PL-ICP匹配方法
- 3、NICP匹配方法
- 4、IMLS-ICP匹配方法

运动畸变去除—ICP(Iterative Closest Point)方法介绍

目的

ICP方法是用来求解两个点云集合转换关系的最通用的方法。



数学描述

- 给定两个点云集合:

$$X = \{x_1, x_2, \cdots, x_{N_x}\}$$
$$P = \{p_1, p_2, \cdots, p_{N_p}\}$$

其中,

x_i和p_i表示点云的坐标;

N_x和N_p表示点云的数量。

- 求解旋转矩阵R和平移向量t，使得下式最小:

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

运动畸变去除—ICP方法介绍

已知对应点的求解方法

$$u_x = \frac{1}{N_p}\sum_{i=1}^{N_p} x_i$$ u_x表示点云集合X的几何中心

$$u_p = \frac{1}{N_p}\sum_{i=1}^{N_p} p_i$$ u_p表示点云集合P的几何中心

去中心化

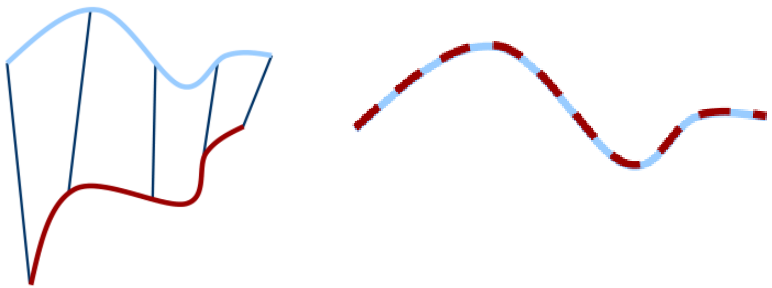
$$x_i' = x_i - u_x$$
$$p_i' = p_i - u_p$$

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

则ICP的解为：

$$R = VU^T$$

$$t = u_x - Ru_p$$



ICP方法介绍

已知对应点的求解方法—证明

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

$$=\frac{1}{N_p}\sum_{i=1}^{N_p}\left\|x_i-Rp_i-t-u_x+Ru_p+u_x-Ru_p\right\|^2$$

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

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

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

等于0，详细推导见附录I

ICP方法介绍

已知对应点的求解方法—证明

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

$$=\underbrace{\frac{1}{N_p}\sum_{i=1}^{N_p}\|x_i-u_x-R(p_i-u_p)\|^2}_{E_1(R,t}+\underbrace{\|u_x-Ru_p-t\|^2}_{E_2(R,t}$$

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

可转变为：

$$minE_1(R,t)=\frac{1}{N_p}\sum_{i=1}^{N_p}\|x_i-u_x-R(p_i-u_p)\|^2$$

$$E_1(R,t)=\frac{1}{N_p}\sum_{i=1}^{N_p}\|x_i-u_x-R(p_i-u_p)\|^2 \longrightarrow$$ 只跟R有关

$$E_2(R,t)=\|u_x-Ru_p-t\|^2 \longrightarrow$$ 对于任意R，总能得到$t=u_x-Ru_p$，使得$E_2(R,t)$取最小值0

ICP方法介绍

已知对应点的求解方法—证明

$$
\begin{aligned}
minE_1(R,t) &= \frac{1}{N_p}\sum_{i=1}^{N_p}\left\|x_i - u_x - R(p_i - u_p)\right\|^2 \\
&= \frac{1}{N_p}\sum_{i=1}^{N_p}\|x_i' - Rp_i'\|^2 \\
&= \frac{1}{N_p}\sum_{i=1}^{N_p} x_i'^T x_i' + p_i'^T R^T R p_i' - 2x_i'^T R p_i' \\
&= \frac{1}{N_p}\sum_{i=1}^{N_p} x_i'^T x_i' + p_i'^T p_i' - 2x_i'^T R p_i' \\
&= \frac{1}{N_p}\sum_{i=1}^{N_p} -2x_i'^T R p_i' + \underbrace{\frac{1}{N_p}\sum_{i=1}^{N_p}(x_i'^T x_i' + p_i'^T p_i')}_{\text{与}R\text{无关}}
\end{aligned}
$$

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

已知对应点的求解方法—证明

$$x_i'^T R p_i' = \begin{pmatrix} x_{i0}' & x_{i1}' & x_{i2}' \end{pmatrix} R p_i' = \begin{pmatrix} x_{i0}' & x_{i1}' & x_{i2}' \end{pmatrix} \begin{pmatrix} \bar{R}_0 \\ \bar{R}_1 \\ \bar{R}_2 \end{pmatrix} p_i'$$

$$= x_{i0}' \bar{R}_0 p_i' + x_{i1}' \bar{R}_1 p_i' + x_{i2}' \bar{R}_2 p' = \boxed{\bar{R}_0 x_{i0}' p_i'} + \boxed{\bar{R}_1 x_{i1}' p_i'} + \boxed{\bar{R}_2 x_{i2}' p_i'}$$

$$R p_i' x_i'^T = \begin{pmatrix} \bar{R}_0 \\ \bar{R}_1 \\ \bar{R}_2 \end{pmatrix} \begin{pmatrix} x_{i0}' p_i' & x_{i1}' p_i' & x_{i2}' p_i' \end{pmatrix} = \begin{bmatrix} \boxed{\bar{R}_0 x_{i0}' p_i'} & \bar{R}_0 x_{i1}' p_i' & \bar{R}_0 x_{i2}' p_i' \\ \bar{R}_1 x_{i0}' p_i' & \boxed{\bar{R}_1 x_{i1}' p_i'} & \bar{R}_1 x_{i2}' p_i' \\ \bar{R}_2 x_{i0}' p_i' & \bar{R}_2 x_{i1}' p_i' & \boxed{\bar{R}_2 x_{i2}' p_i'} \end{bmatrix}$$

$$\Rightarrow \quad x_i'^T R p_i' = Trace(R p_i' x_i'^T)$$

$$max \sum_{i=1}^{N_p} x_i'^T R p_i' = max \sum_{i=1}^{N_p} \mathrm{Trace}(R p_i' x_i'^T) = max\mathrm{Trace}(R(\underbrace{\sum_{i=1}^{N_p} p_i' x_i'^T}_{\text{令其为}H})) = maxTrace(RH)$$

ICP方法介绍

已知对应点的求解方法—证明

目标 $\max\ Trace(RH)$

- 假设矩阵A为正定对称矩阵，则对于任意的正交矩阵B，都有 $Trace(A) \geq Trace(BA)$

- H并非正定对称矩阵，那么如何应用上述性质呢？

  Step 1 对H进行SVD分解，即$H = U\Lambda V^T$
  Step 2 构建正交矩阵X，令$X = VU^T$
  Step 3 $XH = VU^TU\Lambda V^T = V\Lambda V^T$，为正定对称矩阵

- 对于任意的正交矩阵B，根据上述性质，可得 $Trace(XH) \geq Trace(BXH)$

- 因为B为任意正交矩阵，因此BX可以取遍所有的正交矩阵，当然，也包括需要求解的旋转矩阵R，因此：

$$Trace(RH) \leq Trace(XH)$$

- 当$R = X$时，等式成立，因此得 $R = X = VU^T$， $t = u_x - Ru_p$

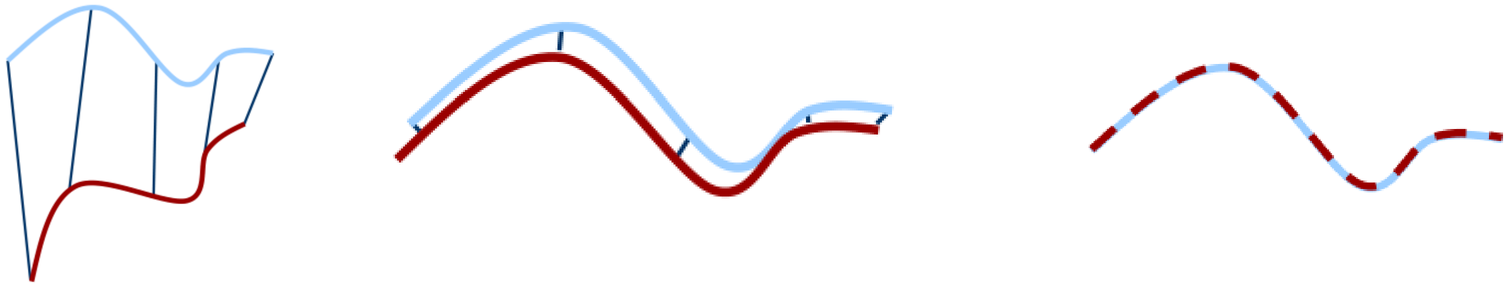
ICP方法介绍

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（找最近的点）
- 根据对应点，计算R和t
- 对点云进行姿态变换，计算误差
- 不断迭代，直至误差小于某一个值



课程内容

帧间匹配算法

1、ICP匹配方法

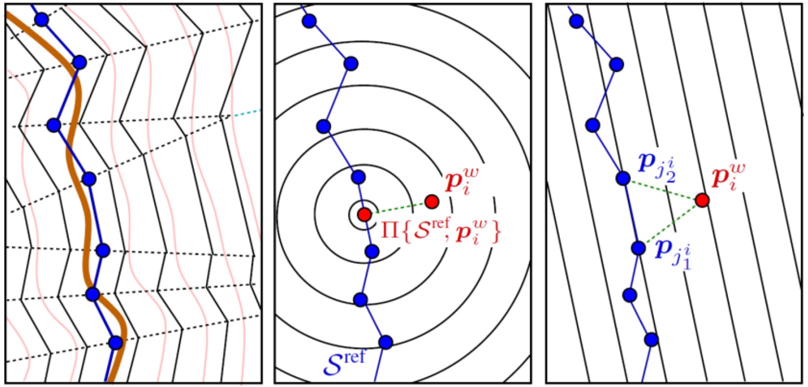
2、PL-ICP匹配方法

3、NICP匹配方法

4、IMLS-ICP匹配方法

PL-ICP方法介绍

示意图



(a) Distance to curve and to polyline (b) Point-to-point metric (c) Point-to-line metric

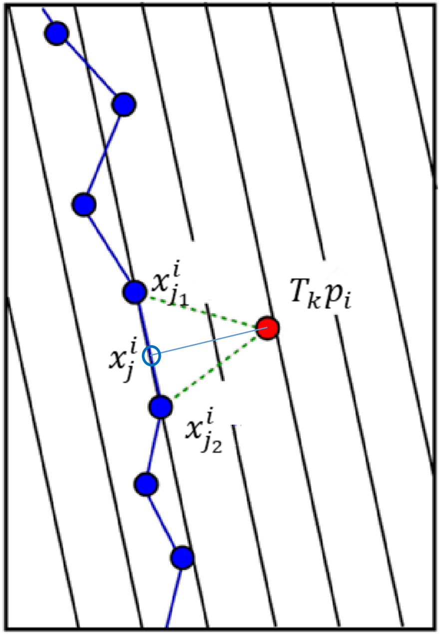
PL-ICP方法示意图：图(a)中棕色的曲线表示实际场景的墙，蓝色是某时刻的激光扫描点。

基本思想

- 激光点是对实际环境中曲面的离散采样；
- 重要的不是激光点，而是隐藏在激光点中的曲面；
- 最好的误差尺度为当前激光点到实际曲面的距离；关键的问题在于如何恢复曲面；
- PL-ICP的思想：用分段线性的方法来对实际曲面进行近似，用激光点到最近两点连线的距离来模拟实际激光点到曲面的距离。

PL-ICP方法介绍

数学描述



- 变量说明

  p_i：点云集合P中点i的三维坐标
  $x_{j_1}^i, x_{j_2}^i$：点云集合X中，距离p_i最近的两个点
  n_i^T：点$x_{j_1}^i, x_{j_2}^i$组成的直线的法向量

- PL-ICP的目标函数

$$\min_{T_k} \sum_i \left\| n_i^T (x_{j2}^i - T_k p_i) \right\|^2 \Leftrightarrow \min_{T_k} \sum_i \left\| x_j^i - T_k p_i \right\|^2$$

其中，$T_k = \begin{bmatrix} R_k & t_k \\ 0 & 1 \end{bmatrix}$ $\quad T_k p_i = R_k p_i + t_k$

PL-ICP的目标函数实际上表示点到曲面的距离，即点到直线的距离

PL-ICP方法求解

已知数据

- 当前激光帧 P
- 参考激光帧 X
- 初始位姿 T_0

待求数据

- 两帧激光之间的相对位姿 T^*

算法流程

1. 把当前帧的数据p_i根据初始位姿进行变换，得到$T_k p_i$；
2. 对于当前帧中的点 $T_k p_i$，在参考帧中找到最近的两个点 $x_{j_1}^i, x_{j_2}^i$，并计算直线法向量 n_i^T 和投影点 x_j^i；
3. 计算点$T_k p_i$与投影点x_j^i的距离，去除误差过大的点；
4. 最小化误差函数 $\min_{T_k}\sum_i \left\| x_j^i - T_k p_i \right\|^2$

PL-ICP方法介绍

跟ICP的区别

- 1. 误差函数的形式不同，ICP为点对点的距离作为误差，PL-ICP为点到线的距离作为误差；PL-ICP的误差形式更符合实际情况。

- 2. 收敛速度不同，ICP为一阶收敛，PL-ICP为二阶收敛。

$$\|T_k - T_\infty\| < k\|T_{k-1} - T_\infty\| \qquad \|T_k - T_\infty\|^2 < k\|T_{k-1} - T_\infty\|^2$$

- 3. PL-ICP的求解精度高于ICP，特别是在结构化环境中。

- 4. PL-ICP对初始值更敏感；不单独使用，与里程计、CSM等一起使用。

课程内容

帧间匹配算法

1、ICP匹配方法

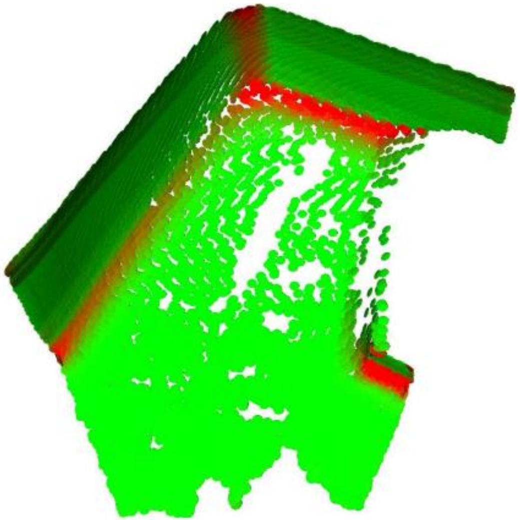
2、PL-ICP匹配方法

3、NICP匹配方法

4、IMLS-ICP匹配方法

NICP(Normal ICP)匹配方法

NICP特征示意图



NICP特征示意图

基本思想

- 替换原始ICP方法中的对应点匹配(point correspondences)方法；
- 充分利用实际曲面的特征来对错误的点匹配进行滤除，主要的特征为法向量和曲率；
- 误差项除了考虑对应点的欧氏距离之外，同时还考虑对应点法向量的角度差。

NICP(Normal ICP)匹配方法

数学描述

p_i, p_j：点云集合P^C, P^r中点的三维坐标

n_i：点i处曲面的法向量

σ_i：点i处曲面的曲率

T：欧氏变换矩阵

$\mathcal{C}$：两个点云P^C, P^r中对应点对的集合

$$\tilde{p}_i = \begin{pmatrix} p_i \\ n_i \end{pmatrix}$$

$$T \oplus \tilde{p}_i = \begin{pmatrix} Rp_i + t \\ Rn_i \end{pmatrix}$$

- 误差定义为：

$$\mathbf{e}_{ij}(\mathbf{T}) = \left(\tilde{\mathbf{p}}_i^{\mathrm{c}} - \mathbf{T} \oplus \tilde{\mathbf{p}}_j^{\mathrm{r}}\right)$$

- 目标函数的定义为：

$$\sum_{\mathcal{C}} \mathbf{e}_{ij}(\mathbf{T})^T \tilde{\mathbf{\Omega}}_{ij} \mathbf{e}_{ij}(\mathbf{T})$$

$$\tilde{\mathbf{\Omega}}_{ij} = \begin{pmatrix} \mathbf{\Omega}_i^{\mathrm{s}} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Omega}_i^{\mathrm{n}} \end{pmatrix}$$

$\tilde{\Omega}_{ij}$为信息矩阵，详细定义见下一页

NICP(Normal ICP)匹配方法

法向量和曲率的计算

- 找到点p_i周围半径R球形空间内的所有点V_i

$$\mu_i^s = \frac{1}{|\mathcal{V}_i|} \sum_{\mathbf{p}_j \in \mathcal{V}_i} \mathbf{p}_i$$

$$\boldsymbol{\Sigma}_i^s = \frac{1}{|\mathcal{V}_i|} \sum_{\mathbf{p}_j \in \mathcal{V}_i} (\mathbf{p}_i - \mu_i)^T (\mathbf{p}_i - \mu_i)$$

$$\Sigma_i^s = U \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} U^T \qquad \lambda_2 > \lambda_1$$

- 曲率的定义：

$$\sigma_{\mathrm{i}} = \frac{\lambda_1}{\lambda_1 + \lambda_2}$$

- 法向量的定义：最小特征值对应的特征向量。

- 信息矩阵$\tilde{\Omega}_{ij}$的定义：

$$\Omega_i^s = \left(\Sigma_i^s\right)^{-1}$$

$$\Omega_i^n = \begin{cases} R \begin{bmatrix} \frac{1}{\varepsilon} & 0 \\ 0 & 1 \end{bmatrix} R^T & \text{曲率足够小} \\ I & \end{cases}$$

$\varepsilon << 1$　　R为旋转矩阵

NICP(Normal ICP)匹配方法

点匹配规则

- 如果没有well define的法向量，则拒绝。
- 两点间的距离大于阈值，则拒绝。

$$\|\mathbf{p}_i^{\mathrm{c}} - \mathbf{T} \oplus \mathbf{p}_j^{\mathrm{r}}\| > \epsilon_d$$

- 两点的曲率之差距大于阈值，则拒绝。

$$|\log \sigma_i^{\mathrm{c}} - \log \sigma_j^{\mathrm{r}}| > \epsilon_\sigma$$

- 两点的法向量角度之差大于阈值，则拒绝。

$$\mathbf{n}_i^{\mathrm{c}} \cdot \mathbf{T} \oplus \mathbf{n}_j^{\mathrm{r}} < \epsilon_n$$

NICP(Normal ICP)匹配方法

目标函数的求解

- 目标函数的定义为:

$$\sum_{\mathcal{C}} \mathbf{e}_{ij}(\mathbf{T})^{T} \tilde{\boldsymbol{\Omega}}_{ij} \mathbf{e}_{ij}(\mathbf{T})$$

- 非线性最小二乘问题，通过LM方法进行求解。

$$(\mathbf{H} + \lambda \mathbf{I}) \Delta \mathbf{T} = \mathbf{b}.$$

$$H = \sum J_i^T J_i$$

$$J_{\mathrm{i}} = \frac{\partial e_{ij}(T)}{\partial T} \qquad \mathbf{T} \leftarrow \Delta \mathbf{T} \oplus \mathbf{T}$$

- 当迭代过程收敛即得到需要的解。
- 由于在寻找点匹配的过程中，考虑了环境曲面的法向量和曲率，因此可以提前排除一些明显是错误的匹配。
- 在误差定义中，除了考虑欧氏距离之外，还考虑了法向量之间的距离，因此具有更加准确的角度。
- 在开源领域，效果最好的ICP匹配方法。

NICP(Normal ICP)匹配方法

算法流程总结

- 计算参考激光帧和当前激光帧中每一个点的法向量和曲率。

- 根据当前解，把当前激光帧的点转换到参考坐标系中，并且根据欧氏距离、法向量、曲率等信息来选择匹配点(也有可能没有匹配点)。

- 根据上面介绍的方法，用LM方法进行迭代求解，迭代收敛即可得到两帧激光数据之间的相对位姿。

课程内容

帧间匹配算法

1、ICP匹配方法

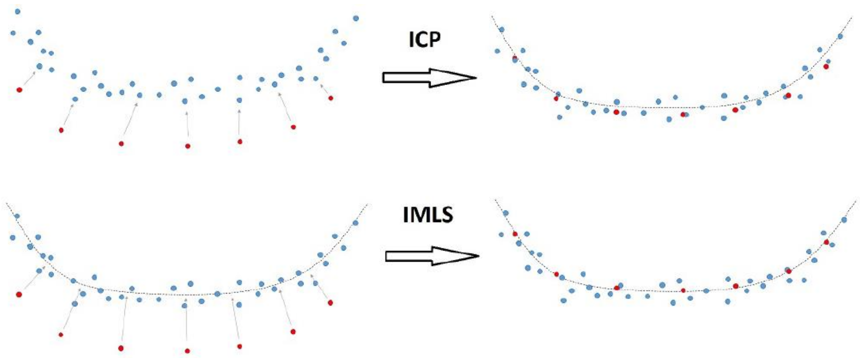
2、PL-ICP匹配方法

3、NICP匹配方法

4、IMLS-ICP匹配方法

IMLS(Implicit Moving Least Square)-ICP匹配方法

示意图



IMLS-ICP示意图

基本思想

- 选择具有代表性的激光点来进行匹配，既能减少计算量同时又能减少激光点分布不均匀导致的计算结果出现偏移。
- 点云中隐藏着真实的曲面，最好的做法就是能从参考帧点云中把曲面重建出来。
- 曲面重建的越准确，对真实世界描述越准确，匹配的精度就越高。

IMLS(Implicit Moving Least Square)-ICP匹配方法

代表点(Informative Point)的选取

- 具有丰富特征的点，即为结构化的点：具有良好的曲率和法向量的定义。

- 曲率越小的点越好，因为曲率为0代表着直线，代表着最结构化的点，也代表着具有非常好的法向量定义，能够提供足够的约束。

- 选点的时候需要注意选取的激光点的均衡以保证可观性，因为是平面匹配，不存在角度不可观的情况。只需要考虑X方向和Y方向的可观性。要保证两者的约束基本上是一致的，才能让结果不出现偏移。

IMLS(Implicit Moving Least Square)-ICP匹配方法

曲面重建

P_k：前n帧激光数据组成的子图；
n_i：点云集P_k中的点p_i的法向量；
$I^{P_k}(x)$：$\mathbb{R}^3$空间的点x到点云集合P_k隐藏曲面的距离。

- $I^{P_k}(x)$定义如下：

$$I^{P_k}(x) = \frac{\sum_{p_i \in P_k} W_i(x)((x - p_i) \cdot \vec{n_i})}{\sum_{p_j \in P_k} W_j(x)}$$

其中，权重$W_i(x)$被定义为：

$$W_i(x) = e^{-\|x - p_i\|^2 / h^2}$$

$I^{P_k}(x) = 0$表示点云集合P_k隐藏的曲面，证明见参考文献[1]。

IMLS(Implicit Moving Least Square)-ICP匹配方法

匹配求解

S_k： 新一帧激光数据；

$I^{P_k}(x_i)$： 当前帧S_k中的点x_i到曲面的距离；

$\overrightarrow{n_i}$： P_k中距离点x_i最近的点的法向量。

- 点x_i在曲面上的投影y_i为：

$$x_i' = Rx_i + t \qquad\qquad y_i = x_i' - I^{P_k}(x_i')\vec{n}_i$$

- 对于新一帧激光数据S_k，通过最小化以下函数，求解姿态变换：

$$\sum_{x_i \in S_k} ((Rx_i + t - y_i) \cdot \vec{n}_i)^2$$

参考资料

[1] Provably good moving least square

[2] An ICP variant using a point-to-line metric

[3] NICP:Dense Normal Based Point Cloud Registration

[4] IMLS:SLAM-scan-to-model matching based on 3D data

[5] Geometrically Stable Sampling for the ICP Algorithm

作业

- 详细见作业说明

结语

感谢聆听！

Thanks for Listening

附录

$$
\begin{aligned}
&\frac{1}{N_p}\sum_{i=1}^{N_p} 2\left(x_i - u_x - R(p_i - u_p)\right)^T (u_x - Ru_p - t) \\
&= 2(u_x - Ru_p - t)\ \frac{1}{N_p}\sum_{i=1}^{N_p}\left(x_i - u_x - R(p_i - u_p)\right)^T \\
&= 2(u_x - Ru_p - t)\left(\frac{1}{N_p}\sum_{i=1}^{N_p}\left(x_i - u_x - R(p_i - u_p)\right)\right)^T \\
&= 2(u_x - Ru_p - t)\left(\underbrace{\tfrac{1}{N_p}\textstyle\sum_{i=1}^{N_p} x_i}_{u_x} - u_x + Ru_p - \underbrace{\tfrac{1}{N_p}\textstyle\sum_{i=1}^{N_p} Rp_i}_{Ru_p}\right)^T \\
&= 0
\end{aligned}
$$