激 光 SLAM 前 端

Least-Squares Rigid Motion Using SVD

An ICP variant using a point-to-line metric

NICP: Dense Normal Based Point Cloud Registration

张涵

2020.06.28

激光SLAM流程

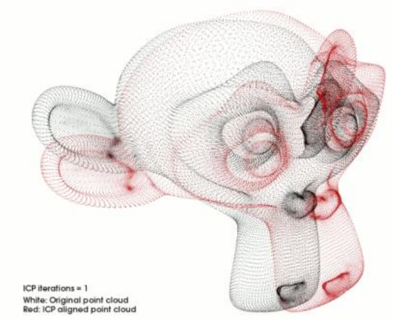
前端（配准/优化）

后端（非线性优化）

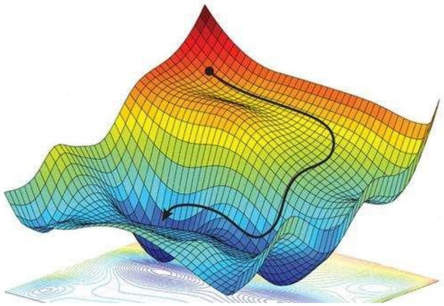
激光数据

回环检测

建图



基于点云配准



基于优化方法

Catalogue

I ICP
Iterative Closest Point

I Iterative Closest Point -ICP

计算机视觉life

1 Problem statement

- 算法的输入：参考点云和目标点云，停止迭代的标准
- 算法的输出：转换矩阵
- 假设待配准的两帧点云，分别为

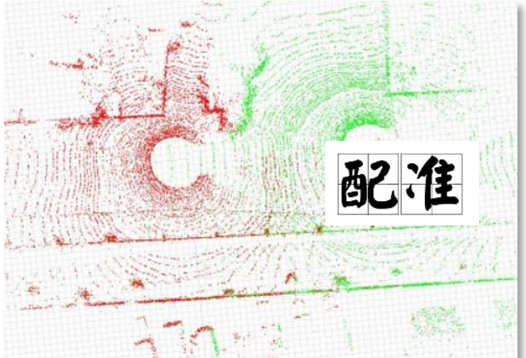
$$\mathcal{P} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_n\} \qquad \mathcal{Q} = \{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_n\}$$

- 目标函数（最小二乘）

$$\boxed{(R, \mathbf{t})} = \underset{R \in SO(d),\ \mathbf{t} \in \mathbb{R}^d}{\operatorname{argmin}} \sum_{i=1}^{n} \boxed{w_i} \|(R\mathbf{p}_i + \mathbf{t}) - \mathbf{q}_i\|^2$$

每对点云的权重

$$w_i > 0$$



http://www.robots.ox.ac.uk/~avsegal/generalized_icp.html

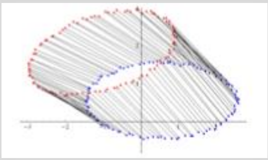
2 Computing the translation

- 简化成关于F的函数 $\boxed{F(\mathbf{t})} = \sum_{i=1}^{n} w_i \|(R\mathbf{p}_i + \mathbf{t}) - \mathbf{q}_i\|^2$

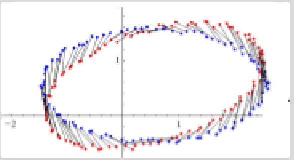
- 固定R,求解T $0 = \frac{\partial F}{\partial \mathbf{t}} = \sum_{i=1}^{n} 2w_i (R\mathbf{p}_i + \mathbf{t} - \mathbf{q}_i)$

$$= 2\mathbf{t}\left(\sum_{i=1}^{n} w_i\right) + 2R\left(\sum_{i=1}^{n} \boxed{w_i\mathbf{p}_i}\right) - 2\sum_{i=1}^{n} \boxed{w_i\mathbf{q}_i}$$

点云质心 $\bar{\mathbf{p}} = \frac{\sum_{i=1}^{n} w_i\mathbf{p}_i}{\sum_{i=1}^{n} w_i}$ $\bar{\mathbf{q}} = \frac{\sum_{i=1}^{n} w_i\mathbf{q}_i}{\sum_{i=1}^{n} w_i}$



平移



- 代入整理得T $\boxed{\mathbf{t} = \bar{\mathbf{q}} - R\bar{\mathbf{p}}}$

- 将T代入目标函数

$$\sum_{i=1}^{n} w_i \|(R\mathbf{p}_i + \mathbf{t}) - \mathbf{q}_i\|^2 = \sum_{i=1}^{n} w_i \|R\underbrace{(\mathbf{p}_i - \bar{\mathbf{p}})}_{\mathbf{x}_i} - \underbrace{(\mathbf{q}_i - \bar{\mathbf{q}})}_{\mathbf{y}_i}\|^2$$

- 旋转矩阵R可以表示为：

$$R = \underset{R \in SO(d)}{\operatorname{argmin}} \sum_{i=1}^{n} w_i \|R\mathbf{x}_i - \mathbf{y}_i\|^2$$

3 Computing the rotation

◆ 简化表达式： $\|R\mathbf{x}_i - \mathbf{y}_i\|^2$

$$
\begin{aligned}
&= (R\mathbf{x}_i - \mathbf{y}_i)^\top (R\mathbf{x}_i - \mathbf{y}_i) = (\mathbf{x}_i^\top R^\top - \mathbf{y}_i^\top)(R\mathbf{x}_i - \mathbf{y}_i) \\
&= \mathbf{x}_i^\top R^\top R\mathbf{x}_i - \mathbf{y}_i^\top R\mathbf{x}_i - \mathbf{x}_i^\top R^\top \mathbf{y}_i + \mathbf{y}_i^\top \mathbf{y}_i \\
&= \mathbf{x}_i^\top \mathbf{x}_i - \mathbf{y}_i^\top R\mathbf{x}_i - \mathbf{x}_i^\top R^\top \mathbf{y}_i + \mathbf{y}_i^\top \mathbf{y}_i
\end{aligned}
$$

其中 $\mathbf{x}_i^\top$ $1 \times d$　$R^\top$ $d \times d$　$\mathbf{y}_i$ $d \times 1$

◆ 标量合并

$$\|R\mathbf{x}_i - \mathbf{y}_i\|^2 = \mathbf{x}_i^\top \mathbf{x}_i - 2\mathbf{y}_i^\top R\mathbf{x}_i + \mathbf{y}_i^\top \mathbf{y}_i$$

◆ 最小化上式

$$
\begin{aligned}
&\underset{R\in SO(d)}{\operatorname{argmin}} \sum_{i=1}^{n} w_i \|R\mathbf{x}_i - \mathbf{y}_i\|^2 = \underset{R\in SO(d)}{\operatorname{argmin}} \sum_{i=1}^{n} w_i (\mathbf{x}_i^\top \mathbf{x}_i - 2\mathbf{y}_i^\top R\mathbf{x}_i + \mathbf{y}_i^\top \mathbf{y}_i) \\
&= \underset{R\in SO(d)}{\operatorname{argmin}} \left(\sum_{i=1}^{n} w_i \mathbf{x}_i^\top \mathbf{x}_i - 2\sum_{i=1}^{n} w_i \mathbf{y}_i^\top R\mathbf{x}_i + \sum_{i=1}^{n} w_i \mathbf{y}_i^\top \mathbf{y}_i \right) \\
&= \underset{R\in SO(d)}{\operatorname{argmin}} \left(-2\sum_{i=1}^{n} w_i \mathbf{y}_i^\top R\mathbf{x}_i \right)
\end{aligned}
$$

$$
\begin{bmatrix} w_1 & & & \\ & w_2 & & \\ & & \ddots & \\ & & & w_n \end{bmatrix}
\begin{bmatrix} - & \mathbf{y}_1^\top & - \\ - & \mathbf{y}_2^\top & - \\ & \vdots & \\ - & \mathbf{y}_n^\top & - \end{bmatrix}
\begin{bmatrix} & R & \end{bmatrix}
\begin{bmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \dots & \mathbf{x}_n \\ | & | & & | \end{bmatrix} =
$$

$$
= \begin{bmatrix} - & w_1\mathbf{y}_1^\top & - \\ - & w_2\mathbf{y}_2^\top & - \\ & \vdots & \\ - & w_n\mathbf{y}_n^\top & - \end{bmatrix}
\begin{bmatrix} | & | & & | \\ R\mathbf{x}_1 & R\mathbf{x}_2 & \dots & R\mathbf{x}_n \\ | & | & & | \end{bmatrix}
= \begin{bmatrix} w_1\mathbf{y}_1^\top R\mathbf{x}_1 & & & * \\ & w_2\mathbf{y}_2^\top R\mathbf{x}_2 & & \\ & & \ddots & \\ * & & & w_n\mathbf{y}_n^\top R\mathbf{x}_n \end{bmatrix}
$$

I Iterative Closest Point -ICP

◆ 求解

$$\underset{R\in SO(d)}{\operatorname{argmin}}\left(-2\sum_{i=1}^{n} w_i \mathbf{y}_i^\top R\mathbf{x}_i\right)=\underset{R\in SO(d)}{\operatorname{argmax}}\sum_{i=1}^{n} w_i \mathbf{y}_i^\top R\mathbf{x}_i$$

$$\sum_{i=1}^{n} w_i \mathbf{y}_i^\top R\mathbf{x}_i=\operatorname{tr}\left(WY^\top RX\right)$$

◆ 根据矩阵的迹 $\operatorname{tr}(AB)=\operatorname{tr}(BA)$

$$\operatorname{tr}\left(WY^\top RX\right)=\operatorname{tr}\left((WY^\top)(RX)\right)=\operatorname{tr}\left(RXWY^\top\right)$$

SVD分解 $\underset{d\times d}{S}=XWY^\top$ $S=U\Sigma V^\top$

$$\operatorname{tr}\left(RXWY^\top\right)=\operatorname{tr}(RS)=\operatorname{tr}\left(RU\Sigma V^\top\right)=\operatorname{tr}\left(\Sigma V^\top RU\right)$$

正交阵 $M=V^\top RU$

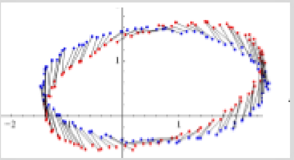
◆ 正交阵元素性质

$$1=\mathbf{m}_j^\top\mathbf{m}_j=\sum_{i=1}^{d} m_{ij}^2 \;\Rightarrow\; m_{ij}^2\le 1 \;\Rightarrow\; |m_{ij}|\le 1$$

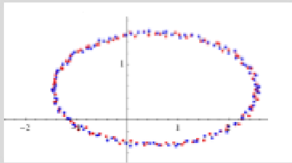
◆ 求下列矩阵迹的最大值 $tr(\Sigma M)$

$$=\begin{pmatrix}\sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_d\end{pmatrix}\begin{pmatrix}m_{11} & m_{12} & \dots & m_{1d}\\ m_{21} & m_{22} & \dots & m_{2d}\\ \vdots & \vdots & \vdots & \vdots\\ m_{d1} & m_{d2} & \dots & m_{dd}\end{pmatrix}=\sum_{i=1}^{d}\sigma_i m_{ii}\le\sum_{i=1}^{d}\sigma_i$$

◆ R的解 $I=M=V^\top RU \;\Rightarrow\; V=RU \;\Rightarrow\; R=VU^\top$



旋转



Least-Squares Rigid Motion Using SVD

I Iterative Closest Point -ICP小结

- ✓ ICP的解：

$$\mathbf{t} = \bar{\mathbf{q}} - R\bar{\mathbf{p}}$$

$$I = M = V^\top R U \;\Rightarrow\; V = RU \;\Rightarrow\; R = VU^\top$$

$$R = V \begin{pmatrix} 1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & 1 & \\ & & & & \boxed{\det(VU^\top)} \end{pmatrix} U^\top$$

- ✓ ICP算法的缺点

① 迭代效果很大程度依赖初值

② 收敛到局部最小值

③ 一阶收敛，收敛速度慢

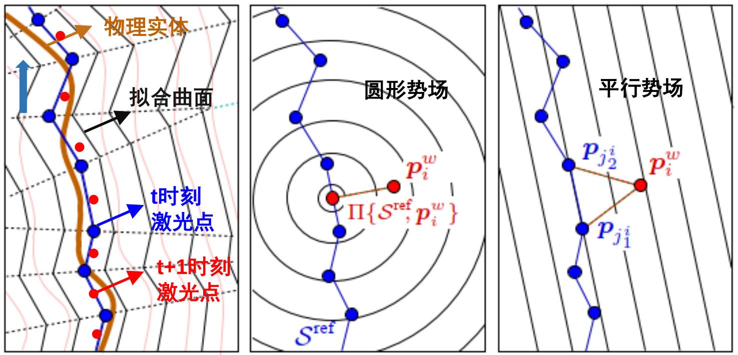
④ 离群点以及噪声

- ✓ 算法流程：

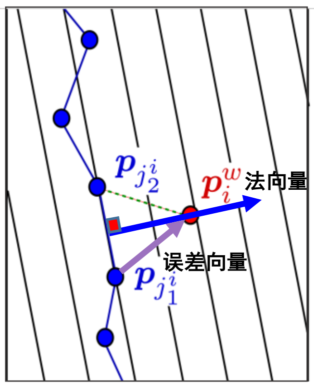
计算最近点 ⇒ 极小化目标函数计算位姿 ⇒ 通过位姿作用点云 ⇒ 迭代计算匹配点 ⇒ 直到误差小于设定阈值 ⇒ 输出

I PL-ICP
Point-Line Iterative Closest Point

II Point-to-Line ICP



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



PL-ICP思想：

a) 利用激光点的离散性

b) 更关注激光点中隐藏的曲面

c) 采用分段线性曲面拟合

d) 通过定义激光点到曲面距离

向量

$$(p \oplus (t, \theta) \triangleq \mathbf{R}(\theta) p + t)$$

目标函数：

$$\min_{q_{k+1}} \sum_i \left(n_i^{\mathrm{T}} \left[p_i \oplus q_{k+1} - \underbrace{\Pi\{\mathcal{S}^{\mathrm{ref}}, p_i \oplus q_k\}}_{\text{K时刻激光点在参考曲面投影的欧式距离}} \right] \right)^2$$

法向量

PL-ICP特点：

a) 二阶收敛

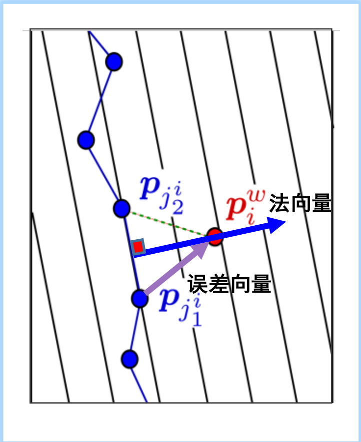
b) 有限步内收敛

c) 比ICP更接近真实曲面

II Point-to-Line ICP

目标函数：

$$\min_{\boldsymbol{q}_{k+1}} \sum_i \left(\boldsymbol{n}_i^{\mathrm{T}} \left[\boldsymbol{p}_i \oplus \boldsymbol{q}_{k+1} - \Pi\{\mathcal{S}^{\mathrm{ref}}, \boldsymbol{p}_i \oplus \boldsymbol{q}_k\}\right]\right)^2$$



将当前帧激光转换到参考框架下

↓

根据阈值，在参考帧当中选择最邻近的两个点

↓

修剪去除异常点（Trimmed ICP）

↓

改写误差函数

↓

求解位姿转换矩阵

$$\boldsymbol{p}_i^w \triangleq \boldsymbol{p}_i \oplus \boldsymbol{q}_k = \mathbf{R}(\theta_k)\boldsymbol{p}_i + \boldsymbol{t}_k$$

$$\langle i, j_1^i, j_2^i \rangle$$

LTS(the least trimmed squares)

$$J(\boldsymbol{q}_{k+1}, \boldsymbol{C}_k) = \sum_i \left(\boldsymbol{n}_i^{\mathrm{T}} \left[\mathbf{R}(\theta_{k+1})\boldsymbol{p}_i + \boldsymbol{t}_{k+1} - \boldsymbol{p}_{j_1^i}\right]\right)^2$$

III NICP
Normal Iterative Closest Point

III Normal ICP

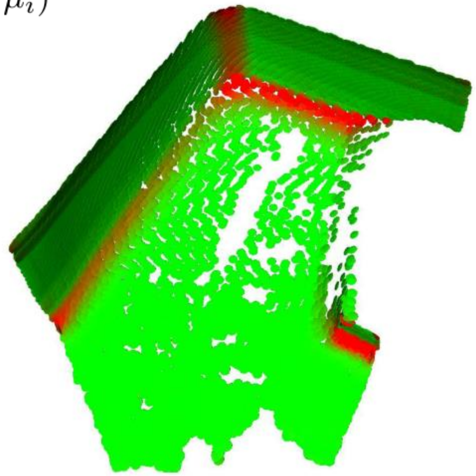
Differently from ICP, NICP considers each point together with the **local features of the surface** (normal and curvature) and it takes advantage of the 3D structure around the points for the determination of the data association between two point clouds. Moreover, it is based on a **least squares formulation** of the alignment problem, that minimizes an **augmented error metric** depending not only on the point coordinates but also on these surface characteristics.

III Normal ICP-流程

计算机视觉life

- 高斯拟合，计算以 $\mathbf{p}_i$ 为中心，R为半径点集 $\mathcal{V}_i$ 的均值与协方差（integral images）

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



- 特征值分解 $$\mathbf{\Sigma}_i^s = \mathbf{R}\begin{pmatrix}\lambda_1 & 0 & 0\\ 0 & \lambda_2 & 0\\ 0 & 0 & \lambda_3\end{pmatrix}\mathbf{R}^T$$

- 曲率与特征值关系 $$\sigma_i = \lambda_1 / (\lambda_1+\lambda_2+\lambda_3)$$

- 通过调整椭球长轴长度削减误差 $$\mathbf{\Sigma}_i^s \leftarrow \mathbf{R}\begin{pmatrix}\epsilon & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{pmatrix}\mathbf{R}^T$$

- 法向量：最小特征值对应的特征向量

III Normal ICP-点的剔除策略

计算机视觉life

- Either $\mathbf{p}_i^{\mathrm{c}}$ or $\mathbf{p}_j^{\mathrm{r}}$ do not have a well defined normal

最初是一个数学上的概念，和ambiguous相对，是指某一表达式只有一种解释或取值唯一。

- The distance between the point in the current cloud and the reprojected point in the reference cloud is larger than a threshold

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

- The magnitude of the log ratio of the curvatures of the points is greater than a threshold

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

- The angle between the normal of the current point and the reprojected normal of the reference point is greater than a threshold

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

III Normal ICP-根据对应关系确定变换

计算机视觉life

- 数学描述 $\tilde{\mathbf{p}}' = \mathbf{T} \oplus \tilde{\mathbf{p}} = \begin{pmatrix} \mathbf{Rp} + \mathbf{t} \\ \mathbf{Rn} \end{pmatrix}$

  $\mathbf{Rp} + \mathbf{t}$ → 对激光点旋转+平移

  $\mathbf{Rn}$ → 对法向量旋转

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

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

  $\tilde{\mathbf{\Omega}}_{ij}$ → 信息矩阵

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

- 非线性最小二乘，LM方法求解 $(\mathbf{H} + \lambda \mathbf{I}) \Delta \mathbf{T} = \mathbf{b}$

  Hessian: $\mathbf{H} = \sum \mathbf{J}_{ij}^T \tilde{\mathbf{\Omega}}_{ij} \mathbf{J}_{ij}$

  residual: $\mathbf{b} = \sum \mathbf{J}_{ij}^T \tilde{\mathbf{\Omega}}_{ij} \mathbf{e}_{ij}(\mathbf{T})$

- 不断迭代求解 $\mathbf{T} \leftarrow \Delta \mathbf{T} \oplus \mathbf{T}$

IV IMLS ICP
Implicit Moving Least ICP

IV IMLS ICP—scan to model

算法思想：

a. 选点：选择具有代表性(结构化以及精确定义)的激光点来进行匹配,.

b. 曲面重建：点云中隐藏着真实得曲面，从参考帧点云中把曲面重建出来

c. 曲面重建的越准确，对真实世界描述越准确，匹配的精度更高

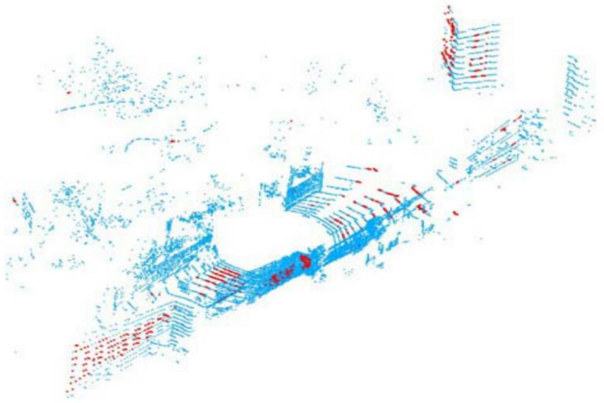
选点策略：

a. 特征丰富的点

b. 结构化的点-具备良好的曲率和法向量的定义

c. 曲率越小的点越好

d. 均衡选取激光点

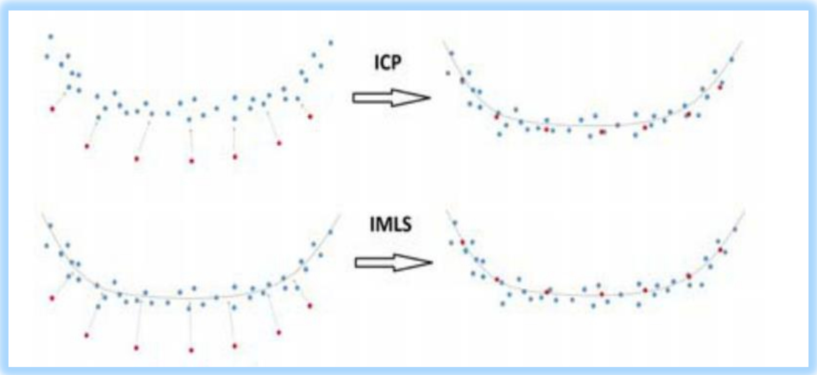


IV IMLS ICP

计算机视觉life

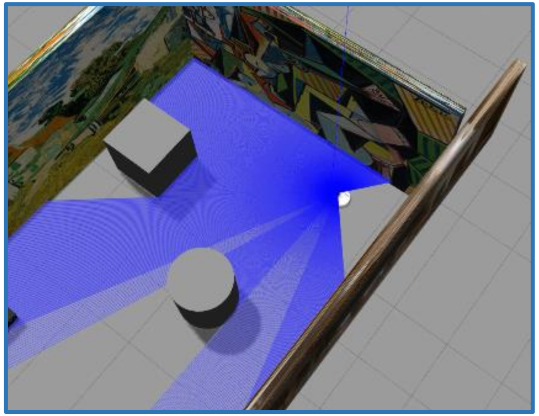
- 点到隐式曲面的距离： $I^{P_k}(x) = \frac{\sum_{p_i \in P_k} W_i(x)((x - p_i) \cdot \vec{n_i})}{\sum_{p_j \in P_k} W_j(x)}$ → 曲面重建
- 权重表达式： $W_i(x) = e^{-\|x-p_i\|^2/h^2}$ 搜索范围： $B(x, r)$
- 位姿计算差值： $\sum_{x_j \in \tilde{S_k}} |I^{P_k}(Rx_j + t)|^2$
- 定义投影点： $y_j = x_j - I^{P_k}(x_j)\vec{n_j}$
- 位姿计算差值改写为：

→ $\sum_{x_j \in \tilde{S_k}} |\vec{n_j} \cdot (Rx_j + t - y_j)|^2$



V 高斯牛顿优化

V 高斯牛顿优化-hector SLAM



Hector SLAM

- 对硬件要求高
- 利用高斯牛顿解决scan-matching问题
- 不需要里程计
- 没有回环
- 使用多分辨率地图

梯度下降方法	最速下降	牛顿法	高斯牛顿法	L-M方法	Dogleg
特点	计算速度慢； 收敛慢	收敛慢； 求解Hessian 矩阵复杂	用Jacobi矩阵相乘 代替Hessian矩阵 $J^T J = H$	通过 λ 调整收敛速度 $J^T J + \lambda I$	高斯牛顿 +最速下降

V 高斯牛顿优化-推导

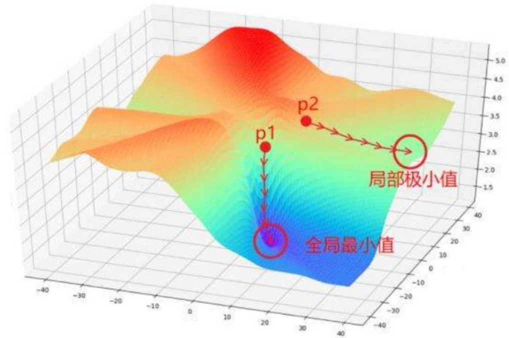
- 对于一个非线性最小二乘问题 $x = \arg\min_{x} \frac{1}{2} \parallel f(x) \parallel^2$
- 把 $f(x)$ 利用泰勒展开，取一阶线性项近似 $f(x+\Delta x) = f(x) + f'(x)\Delta x = f(x) + J(x)\Delta x$
- 回代得：$\frac{1}{2} \parallel f(x+\Delta x) \parallel^2 = \frac{1}{2}\{f(x)^T f(x) + 2f(x)^T J(x)\Delta x + \Delta x^T J(x)^T J(x)\Delta x\}$
- 对上式求导，令导数为0 $J(x)^T J(x)\Delta x = -J(x)^T f(x)$

令 $H = J^T J$，$B = -J^T f$

- 问题转换为 $\boxed{H\Delta x = B}$

给定初值 → 进行k次迭代，计算Jacobi，以及增量x → 当x的增量小于设定阈值，停止迭代，否则进行更新 → 不断循环前两步骤，直到满足条件或者达到最大循环次数

V 高斯牛顿优化



- 目标函数：将帧间匹配问题转换为求解极值 $E(T)=\arg\min_{T}\sum\left[1-M\left(S_i(T)\right)\right]^2$

- Taylor展开，一阶近似 $1\text{-}M(S_i(T+\Delta T)) = 1\text{-}M(S_i(T))\text{-}\nabla M(S_i(T))\dfrac{\partial S_i(T)}{\partial T}\Delta T$

$$E(T+\Delta T)=\arg\min_{T}\sum\left[1-M\left(S_i(T)\right)-\nabla M\left(S_i(T)\right)\frac{\partial S_i(T)}{\partial T}\Delta T\right]^2$$

- 展开，代入求解对 ΔT 的导数，并令其等于0 $\sum\left[\nabla M\left(S_i(T)\right)\dfrac{\partial S_i(T)}{\partial T}\right]^T\left[1-M\left(S_i(T)\right)-\nabla M\left(S_i(T)\right)\dfrac{\partial S_i(T)}{\partial T}\Delta T\right]=0$

$$\sum\left[\nabla M(S_i(T))\frac{\partial S_i(T)}{\partial T}\right]^T\left[1-M(S_i(T))\right]=\underbrace{\sum\left[\nabla M(S_i(T))\frac{\partial S_i(T)}{\partial T}\right]^T\left[\nabla M(S_i(T))\frac{\partial S_i(T)}{\partial T}\right]}_{H}\Delta T=0$$

$$\boxed{\Delta T=H^{-1}\sum\left[\nabla M\left(S_i(T)\right)\frac{\partial S_i(T)}{\partial \Delta T}\right]^T\left[1-M\left(S_i(T)\right)\right]}$$

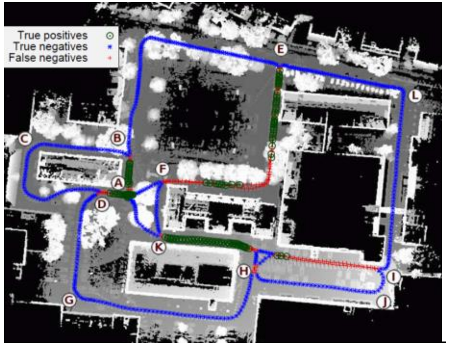
- 其中 $T=\left(T_x,T_y,T_\theta\right)$ $p_i=\left(p_{ix},p_{iy}\right)$表示第i个激光点的坐标

$$S_i(T)=\begin{bmatrix}\cos T_\theta & -\sin T_\theta & T_x\\ \sin T_\theta & \cos T_\theta & T_y\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}p_{ix}\\ p_{iy}\\ 1\end{bmatrix}\xrightarrow{\text{展开}} S_i(T)=\begin{bmatrix}\cos T_\theta * p_{ix}-\sin T_\theta * p_{iy}+T_x\\ \sin T_\theta * p_{ix}-\cos T_\theta * p_{iy}+T_y\\ 1\end{bmatrix}\xrightarrow{\text{求导}}\frac{\partial S_i(T)}{\partial T}=\begin{bmatrix}1 & 0 & -\sin T_\theta * p_{ix}-\cos T_\theta * p_{iy}\\ 0 & 1 & \cos T_\theta * p_{ix}-\sin T_\theta * p_{iy}\end{bmatrix}$$

- 同时 M对S_i的导数 采用拉格朗日双线性插值

VI Normal Distribution Transformation
NDT正态分布变换

VI NDT正弦距离转换



划分网格

↓

计算网格的PDF

↓

对每个点云数据，找到对应的网格点，并根据PDF和评分函数计算结果

↓

根据结果更新g和H，计算新的步长

↓

判断是否收敛或达到迭代次数，是则跳出，否则继续步骤3-5

- 2D情况下为正方形，3D情况下为立方体
- 计算概率分布函数PDF $p(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$
- 测量值 $\vec{x}$ 的似然函数

$$p(\vec{x}) = \frac{1}{(2\pi)^{D/2}\sqrt{|\Sigma|}} \exp\left(-\frac{(\vec{x}-\vec{\mu})^T \Sigma^{-1} (\vec{x}-\vec{\mu})}{2}\right)$$

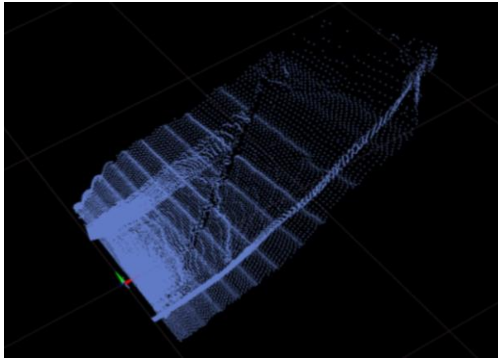
- 计算由 $\vec{x}$ 组成的网格的均值与协方差

$$\vec{\mu} = \frac{1}{m} \sum_{m=1}^{m} \vec{y_k} \qquad \Sigma = \frac{1}{m-1} \sum_{k=1}^{m} (\vec{y_k} - \vec{\mu})(\vec{y_k} - \vec{\mu})^T$$

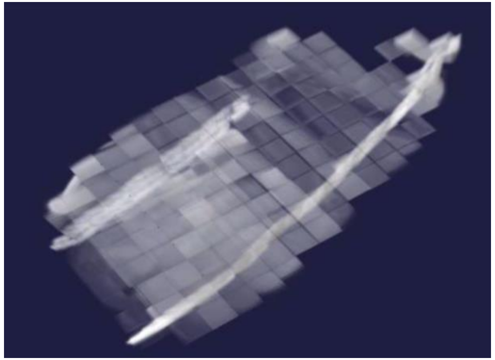
- 目标函数简化以及增加限制

$$\bar{p}(x) = c_1 exp\left(-\frac{\overrightarrow{(x-\mu)}^T \mathbf{\Sigma}^{-1} \overrightarrow{(x-\mu)}}{2}\right) + c_2 p_0$$

VI NDT正弦距离转换



Original point cloud



NDT representation

Algorithm 2 Register scan $\mathcal{X}$ to reference scan $\mathcal{Y}$ using NDT.

ndt($\mathcal{X}, \mathcal{Y}, \vec{p}$)

{Initialisation:}
allocate cell structure $\mathcal{B}$
for all points $\vec{y}_k \in \mathcal{Y}$ **do**
  find the cell $b_i \in \mathcal{B}$ that contains $\vec{y}_k$
  store $\vec{y}_k$ in b_i
end for
for all cells $b_i \in \mathcal{B}$ **do**
  $\mathcal{Y}' = \{\vec{y}'_1, \ldots, \vec{y}'_m\} \leftarrow$ all points in b_i
  $\vec{\mu}_i \leftarrow \frac{1}{n}\sum_{k=1}^{m} \vec{y}'_k$
  $\Sigma_i \leftarrow \frac{1}{m-1}\sum_{k=1}^{m} (\vec{y}'_k - \vec{\mu})(\vec{y}'_k - \vec{\mu})^{\mathrm{T}}$
end for
{Registration:}
while not converged **do**
  $score \leftarrow 0$
  $\vec{g} \leftarrow 0$
  $\mathrm{H} \leftarrow 0$
  for all points $\vec{x}_k \in \mathcal{X}$ **do**
    find the cell b_i that contains $T(\vec{p}, \vec{x}_k)$
    $score \leftarrow score + \tilde{p}\,(T(\vec{p}, \vec{x}_k))$ (see Equation 6.9)
    update $\vec{g}$ (see Equation 6.12)
    update H (see Equation 6.13)
  end for
  solve $\mathrm{H}\Delta\vec{p} = -\vec{g}$
  $\vec{p} \leftarrow \vec{p} + \Delta\vec{p}$
end while

$$\boxed{\begin{aligned} H\Delta\vec{p} &= -\vec{g} \\ \vec{p} &\leftarrow \vec{p} + \Delta\vec{p} \end{aligned}}$$

$$\boxed{score_i = -\exp\left(-\frac{(x'_i - q_i)^T \Sigma_i^{-1} (x'_i - q_i)}{2}\right)}$$

改进：

- cell size
- Iterative discretization
- Adaptive clustering
- Linked cells
- Octree discretization
- Trilinear interpolation

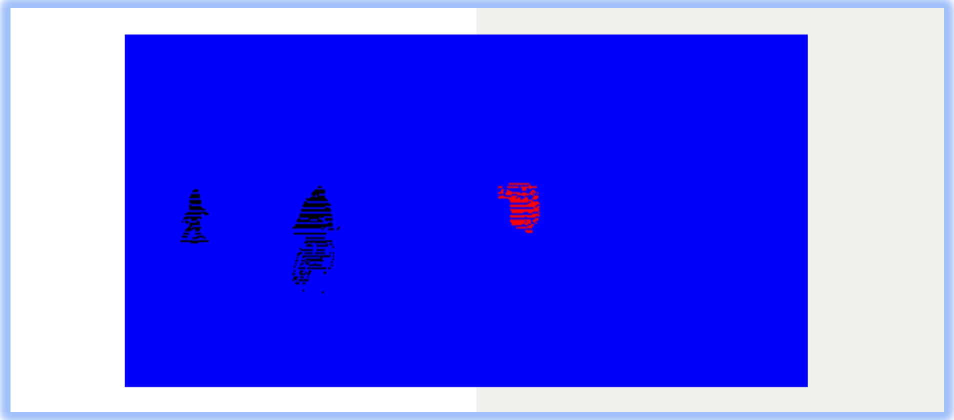
参考论文：The three-dimensional normal-distributions transform: an efficient representation for registration, surface analysis, and loop detection

V Other methods

VII. Other methods

- Iterative dual correspondences (IDC) algorithm--主要是采用极坐标代替笛卡尔坐标系进行最近点搜索
- Probabilistic iterative correspondence (PIC) method--考虑了噪声和初始位姿的不确定性
- Gaussian fields--采用高斯混合模型，类似NDT
- Conditional random fields (CRFs)--条件随机场
- Branch-and-bound strategy--分支定界法
- Registration using local geometric features--结合图像的局部特征进行匹配
- 结合深度学习

Thank you for your attention



什么是「 SLAM研习社」？

计算机视觉life

- 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、联络、宣传
- 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
- QQ群：1105241614