

# 智能驾驶算法基础(B02)

## 定位建图技术—2



傅春耘 副教授

fuchunyun@cqu.edu.cn

卓越工程师学院

# 主要内容

1

建图的作用及常用地图种类

2

高精地图

3

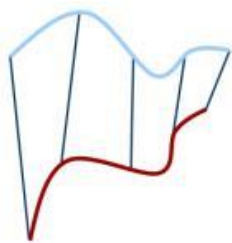
典型建图算法介绍——ICP、NDT和LOAM

4

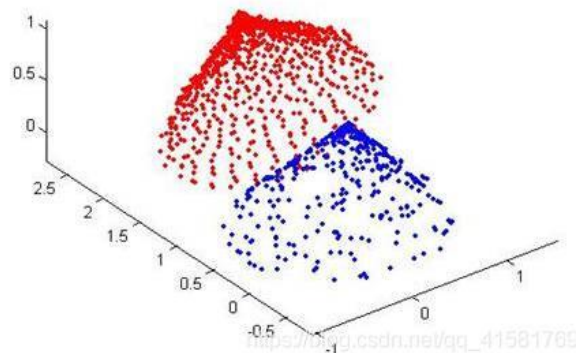
实验内容

### (1) 点云配准的背景

点云配准是点云处理研究领域中的一个典型问题和技术难点，其目的在于比较或融合针对同一对象在不同条件下获取的点云，例如点云会来自不同的采集设备，取自不同的时间，不同的采集视角等等，有时也需要处理针对不同对象的点云配准问题。具体地说，对于某一点云数据集中的两簇点云，通过寻找一种空间变换把一簇点云映射到另一簇点云，使得两簇点云中对应于空间同一位置的点一一对应起来，从而达到信息融合的目的。



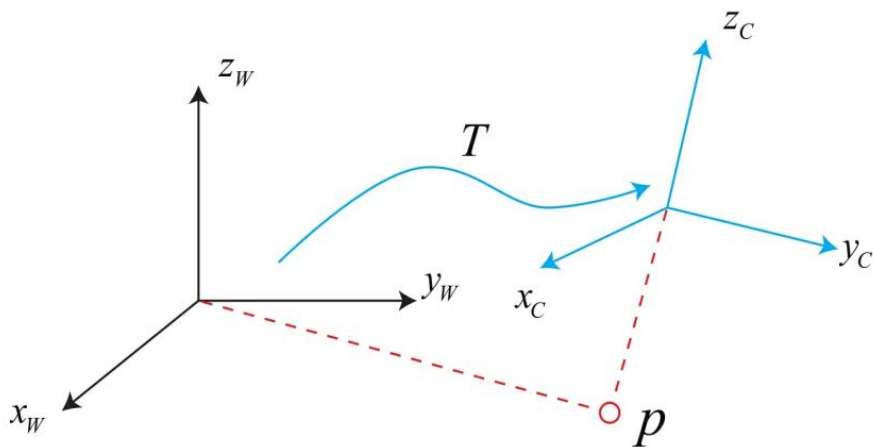
a. 二维线段配准



b. 三维点云配准

### (2) 点云配准的实质

简单点说，点云配准（Point Cloud Registration）指的是输入两簇点云  $P$  (source) 和  $Q$  (target)，输出一个变换  $T$ ，使得  $T(P)$  和  $Q$  的重合程度尽可能高。或者说，对于两个不同视角下的坐标系，比如世界坐标系或相机坐标系或激光雷达坐标系，我们要求出一个变换  $T$  使得两个坐标系变换到统一视角下。这里我们只考虑**刚性变换**，即变换只包括**旋转和平移**。

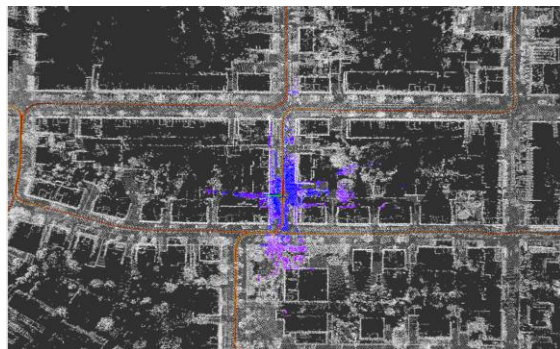


### (3) 点云配准的应用

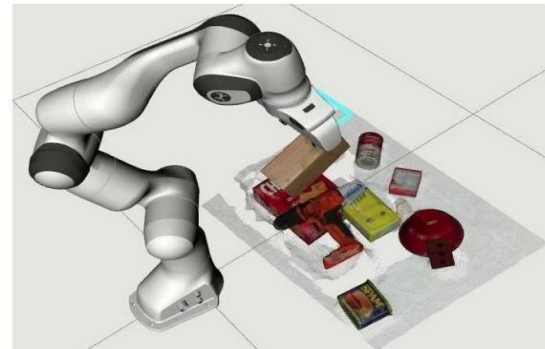
- **三维重建**：生成完整的三维场景是各种计算机视觉应用的基础和重要技术，包括自动驾驶中的高精度三维地图重建、机器人技术中的三维环境重建和用于地下矿井监控的三维重建。
- **三维定位**：点云配准可以将当前的实时三维点云精确匹配到所属的三维环境中，提供高精度的定位服务。
- **姿态估计**：将一个点云 A（3D实时视图）与另一个点云 B（3D环境）对齐，可以生成与点云 B 相对的点云 A 的姿态信息，这些姿态信息可用于机器人的决策。例如，可以获得机器人手臂的姿势信息，从而决定移动到哪里以准确地抓取对象。



a. 三维重建



b. 三维定位



c. 位姿估计

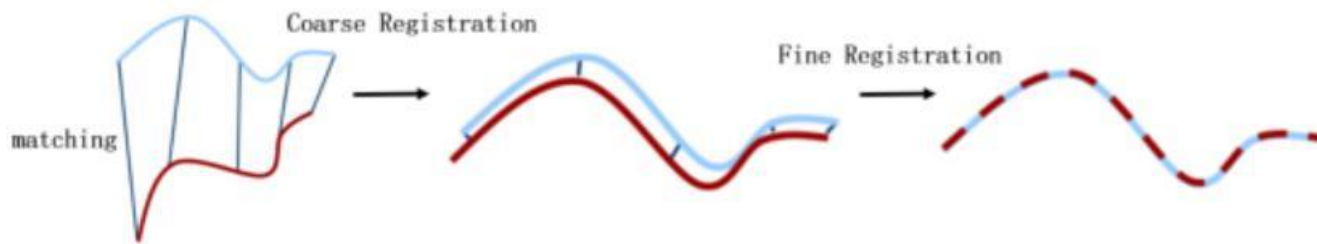


### (3) 点云配准的应用



点云配准可以分为**粗配准**（Coarse Registration）和**精配准**（Fine Registration）两步。粗配准指的是在两幅点云之间的变换完全未知的情况下进行较为粗糙的配准，目的主要是为精配准提供较好的变换初值；精配准则是给定一个初始变换，进一步优化得到更精确的变换。总而言之，点云配准的过程就是求取变换矩阵的过程。

目前应用最广泛的点云精配准算法是**迭代最近点**（Iterative Closest Point, ICP）算法和**正态分布变换**（Normal Distributions Transform, NDT）算法。



### (1) ICP配准基础理论

a. **变换矩阵**：点云配准的最终目的是通过一定的**旋转**和**平移**变换将不同坐标系下的两组或者多组点云数据统一到**同一坐标系**下。这个过程，可以通过一组映射来完成。假设映射变换为  $T$ ，则  $T$  可以用以下的公式来表示：

$$T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ v_x & v_y & v_z & s \end{bmatrix} \rightarrow T = \begin{bmatrix} R_{3 \times 3} & t_{3 \times 1} \\ v_{1 \times 3} & s_{1 \times 1} \end{bmatrix}$$

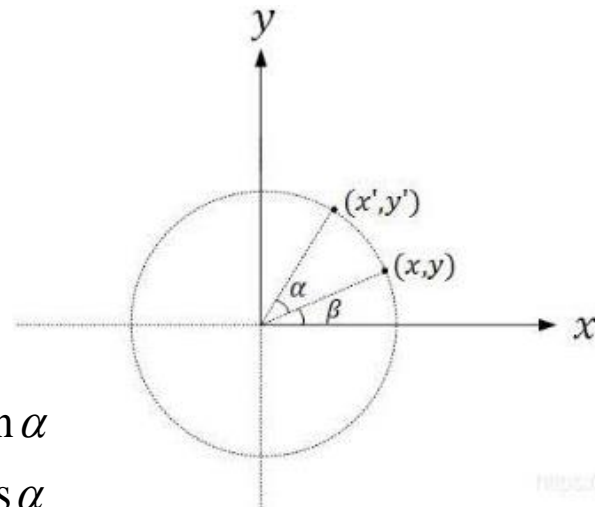
其中  $R$  代表旋转矩阵， $t$  代表平移向量， $v$  代表透视变换向量， $s$  代表整体的比例因子。在本课程中只存在旋转和平移变换，不存在形变，所以将  $v$  设为零向量，比例因子  $s = 1$ 。



### (1) ICP配准基础理论

b. **旋转矩阵理论推导**：以坐标系的三个坐标轴  $x, y, z$  分别作为旋转轴，则点实际上只在垂直坐标轴的平面上作二维旋转。此时用二维旋转公式就可以直接推出三维旋转变换矩阵。规定在右手坐标系中，物体**旋转的正方向是右手螺旋方向**，即从该轴正半轴向原点看是逆时针方向。

二维坐标旋转推导：绕坐标原点，以半径  $r$  旋转一定的角度  $\alpha$ ，如右图，点  $(x, y)$  通过逆时针旋转  $\alpha$  角度变换为  $(x', y')$ ，已知旋转角度  $\alpha$  和  $(x, y)$  点坐标，可计算出点  $(x', y')$ ：

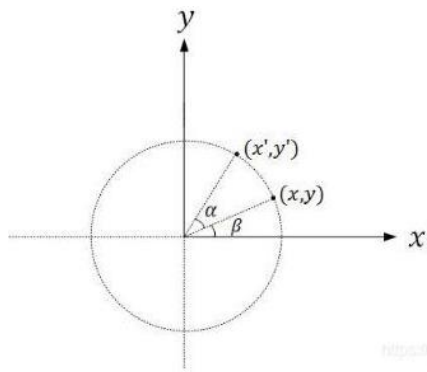


$$x' = r \cdot \cos(\alpha + \beta) = r \cdot \cos \alpha \cdot \cos \beta - r \cdot \sin \alpha \cdot \sin \beta = x \cdot \cos \alpha - y \cdot \sin \alpha$$

$$y' = r \cdot \sin(\alpha + \beta) = r \cdot \sin \alpha \cdot \cos \beta + r \cdot \cos \alpha \cdot \sin \beta = x \cdot \sin \alpha + y \cdot \cos \alpha$$

### (1) ICP配准基础理论

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



下面直接给出右手系中绕三轴的基本旋转矩阵，旋转角度分别为 $\alpha, \beta, \gamma$ :

$$R_x(\gamma) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \gamma & -\sin \gamma \\ 0 & \sin \gamma & \cos \gamma \end{bmatrix}$$

$$R_y(\beta) = \begin{bmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{bmatrix}$$

$$R_z(\alpha) = \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

通用旋转矩阵 $R$ 为:

$$R = R_x(\gamma)R_y(\beta)R_z(\alpha)$$

这里的 $R$ 描述的旋转过程为: 先绕 $z$ 轴旋转 $\alpha$ 角, 然后绕 $y$ 轴旋转 $\beta$ 角, 最后绕 $x$ 轴旋转 $\gamma$ 角。

### (1) ICP配准基础理论

c. 坐标变换：将两个不同坐标系下的点  $X$  和  $X'$  进行坐标变换时，可以通过以下公式来实现转换：

$$X' = R_{3 \times 3}X + t_{3 \times 1}$$

其中,  $X = [x_i \ y_i \ z_i]^T$ ,  $X' = [x_i' \ y_i' \ z_i']^T$ , 将  $X$  和  $X'$  带入上式可以得到：

$$\begin{bmatrix} x_i' \\ y_i' \\ z_i' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \gamma & -\sin \gamma \\ 0 & \sin \gamma & \cos \gamma \end{bmatrix} \begin{bmatrix} \cos \beta & 0 & \sin \beta \\ 0 & 1 & 0 \\ -\sin \beta & 0 & \cos \beta \end{bmatrix} \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$$

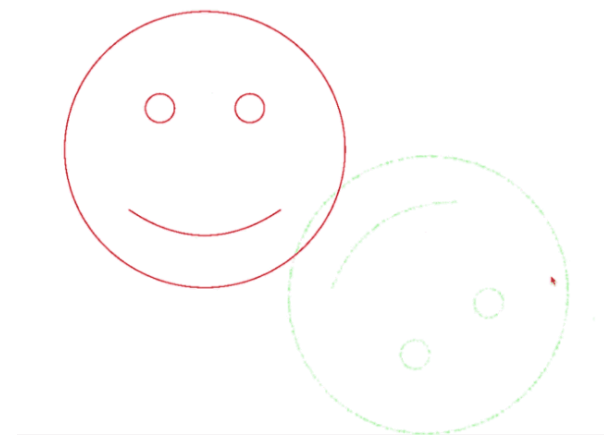
刚性变换矩阵中要唯一确定这六个未知参数，需要六个线性方程，即至少需要在待匹配点云重叠区域找到3组不能共线的对应点对，才可以得到这几个未知数的值。通常情况下，人们会选择尽可能多的对应点对，进一步提高刚性变换矩阵的参数估计精度。

### (1) ICP配准基础理论

ICP算法本质上是基于**最小二乘法**的最优配准方法。该算法**重复**选择对应的关系点对，并计算**最优的刚体变换**，直到满足正确配准的收敛精度要求（**这是一个迭代过程**）。ICP算法的主要目的就是找到**旋转**矩阵和**平移**向量，将两个不同坐标系下的点云，以其中一个点云坐标系为全局坐标系，另一个点云经过旋转和平移后使两组点云重合部分完全重叠。

算法的输入：源点云和目标点云，停止迭代的标准。

算法的输出：旋转矩阵  $R$  和平移向量  $t$ ，即转换矩阵  $T$ 。

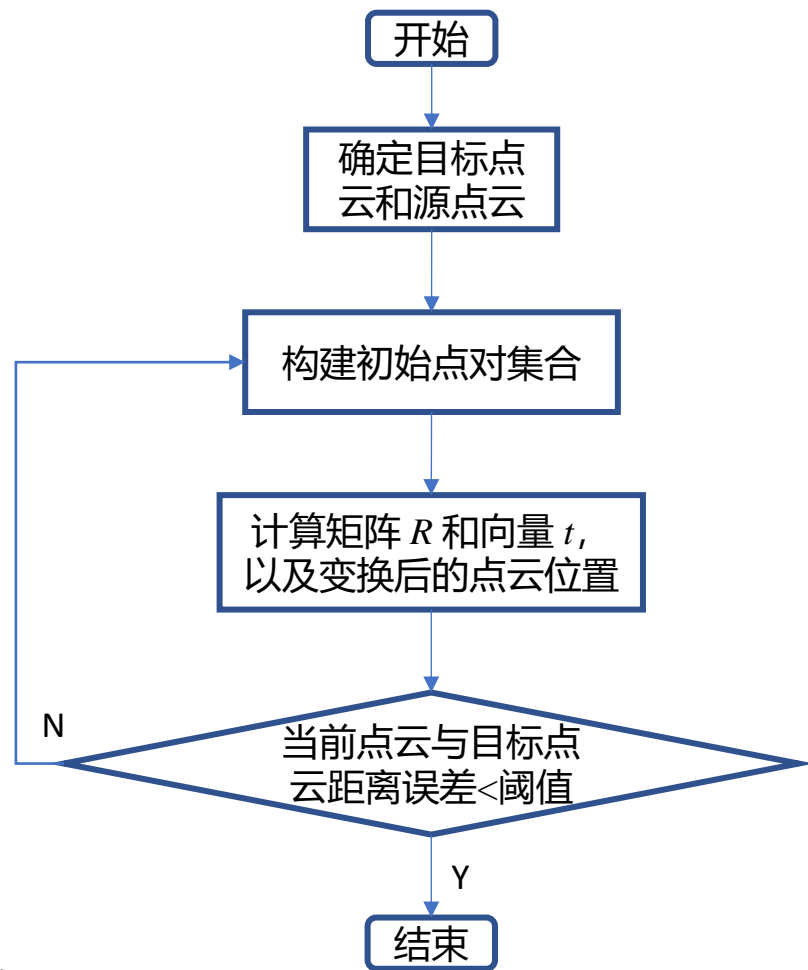


### (2) ICP基本原理与推导

ICP算法的基本流程：

- ① 确定源点云  $P$  和目标点云  $Q$ ；
- ② 寻找对应点（最近点）；
- ③ 根据对应点，计算  $R$  和  $t$ ；
- ④ 对点云进行变换，并计算距离误差；
- ⑤ 不断迭代，直至误差小于某一阈值。

**课后拓展：**如果点云中点的数量过大，可以在步骤①之后进行**降采样**处理；如果步骤②中存在大量错误的对应点对，可以在步骤②之后进行**错误点对的去除**。



### (2) ICP基本原理与推导

#### ① 确定源点云 $P$ 和目标点云 $Q$

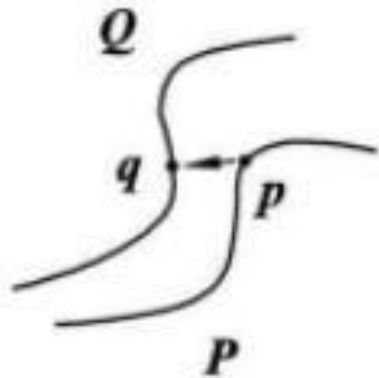
这一步就是确定你想配准的两帧点云数据是哪两帧，以及谁是目标点云、谁是源点云。

#### ② 寻找对应点（最近点）

本课程中主要讲解点到点（point-to-point）的方法，其他方法大家也可课后自行查阅，例如点到线、点到面等等。

点到点的方法通过直接搜索一簇点云中的点在另一簇点云中的最近点来确定对应点集，数学表达如下：

$$\{d_{\min} | d_{\min} = \|p_i - q_j\|, p_i \in P, q_j \in Q\}$$



点到点



### (2) ICP基本原理与推导

#### ③ 根据对应点，计算矩阵 $R$ 和向量 $t$

在我们利用最近点原则确定了点对集合之后，采用**最小二乘法**迭代求解两簇点云间的最优坐标变换。首先定义以矩阵  $R$  和向量  $t$  为自变量的**误差函数**  $E(R, t)$ :

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

其中  $N$  表示源点云  $P$  中点的数量。上式通过最小化误差函数  $E(R, t)$ ，求旋转矩阵  $R$  和平移向量  $t$ 。由于有两个未知量，所以我们先假设  $R$  已知，求出此时  $t$  的值。

### (2) ICP基本原理与推导

假设  $R$  已知后, 对  $t$  求偏导:

$$\frac{\partial E(R, t)}{\partial t} = -\frac{2}{N} \sum_{i=1}^N q_i + \frac{2}{N} \sum_{i=1}^N R p_i + \frac{2}{N} \sum_{i=1}^N t$$

令偏导数为 0, 可求出使误差函数  $E(R, t)$  最小的  $t$ :

$$\frac{1}{N} \sum_{i=1}^N t = \frac{1}{N} \sum_{i=1}^N q_i - \frac{1}{N} \sum_{i=1}^N R p_i$$

即:

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

质心:

$$\bar{q} = \frac{1}{N} \sum_{i=1}^N q_i$$

$$\bar{p} = \frac{1}{N} \sum_{i=1}^N p_i$$



### (2) ICP基本原理与推导

把  $t = \bar{q} - R\bar{p}$  代入  $E(R, t)$  中:

$$E(R, t) = \frac{1}{N} \sum_{i=1}^N \|q_i - Rp_i - t\|^2 = \frac{1}{N} \sum_{i=1}^N \|q_i - \bar{q} - R(p_i - \bar{p})\|^2$$

令:  $q_i' = q_i - \bar{q}$  及  $p_i' = p_i - \bar{p}$

则有:  $E(R, t) = \frac{1}{N} \sum_{i=1}^N \|q_i' - Rp_i'\|^2$

至此, 我们需要求出使  $E(R, t)$  最小时的  $R$ , 即求解:  $\operatorname{argmin}_R \sum_{i=1}^N \|q_i' - Rp_i'\|^2$

### (2) ICP基本原理与推导

展开  $\|q'_i - Rp'_i\|^2$  :

$$\begin{aligned}\|q'_i - Rp'_i\|^2 &= (q'_i - Rp'_i)^T (q'_i - Rp'_i) \\ &= (q_i'^T - p_i'^T R^T) (q'_i - Rp'_i) \\ &= q_i'^T q'_i - p_i'^T R^T q'_i - q_i'^T Rp'_i + p_i'^T R^T Rp'_i\end{aligned}$$

注意: 1) 旋转矩阵  $R$  是正交阵, 故有  $R^T R = I$

2)  $p_i'^T R^T q'_i$  项是标量, 固有  $p_i'^T R^T q'_i = (p_i'^T R^T q'_i)^T = q_i'^T Rp'_i$

故有:  $\|q'_i - Rp'_i\|^2 = q_i'^T q'_i - 2q_i'^T Rp'_i + p_i'^T p'_i$

### (2) ICP基本原理与推导

因此：

$$\begin{aligned} & \operatorname{argmin}_R \sum_{i=1}^N \|q_i' - R p_i'\|^2 \\ &= \operatorname{argmin}_R \sum_{i=1}^N (q_i'^T q_i' - 2q_i'^T R p_i' + p_i'^T p_i') \\ &= \operatorname{argmin}_R \left( \sum_{i=1}^N q_i'^T q_i' - 2 \sum_{i=1}^N q_i'^T R p_i' + \sum_{i=1}^N p_i'^T p_i' \right) \\ &= \operatorname{argmin}_R \left( -2 \sum_{i=1}^N q_i'^T R p_i' \right) = \operatorname{argmax}_R \sum_{i=1}^N q_i'^T R p_i' \end{aligned}$$

### (2) ICP基本原理与推导

然而:

$$\sum_{i=1}^N q_i'^T R p_i' = q_1'^T R p_1' + q_2'^T R p_2' + \dots + q_N'^T R p_N'$$

$$= \text{tr} \left( \begin{bmatrix} q_1'^T R p_1' & & & \\ & q_2'^T R p_2' & & \\ & & \dots & \\ & & & q_N'^T R p_N' \end{bmatrix} \right)$$

$$= \text{tr} \left( \begin{bmatrix} q_1'^T \\ q_2'^T \\ \vdots \\ q_N'^T \end{bmatrix} [R] \begin{bmatrix} p_1' & p_2' & \dots & p_N' \end{bmatrix} \right)$$

$$= \text{tr}(Q'^T R P')$$



$$\begin{aligned} Q' &= [q_1' & q_2' & \dots & q_N'] \\ P' &= [p_1' & p_2' & \dots & p_N'] \end{aligned}$$



### (2) ICP基本原理与推导

由矩阵的迹的性质  $\text{tr}(AB) = \text{tr}(BA)$ , 可得:

$$\sum_{i=1}^N q_i'^T R p_i' = \text{tr}(Q'^T R P') = \text{tr}(R P' Q'^T)$$

令  $S = P' Q'^T$ , 并对矩阵  $S$  进行[SVD分解](#), 有:

$$S = U \Sigma V^T$$

其中  $U$ 、 $V$  为两个正交矩阵,  $\Sigma$  为对角阵。

**那么  $U$ 、 $V$ 、 $\Sigma$  三个矩阵该如何求取呢?**



### (2) ICP基本原理与推导

矩阵  $U$ 、 $V$ 、 $\Sigma$  求取方法如下：

构造矩阵  $SS^T$ ，这是一个**对称阵**，将其对角化可得：

$$SS^T = UDU^T$$

其中  $D$  是**对角阵**，其元素为矩阵  $SS^T$  的  $n$  个**特征值**； $U$  是一个由矩阵  $SS^T$  的**特征向量**组成的矩阵。

同理，对于矩阵  $S^TS$ ，将其对角化可得：

$$S^TS = VDV^T$$

其中  $D$  是**对角阵**，其元素为矩阵  $S^TS$  的  $n$  个**特征值**； $V$  是一个由矩阵  $S^TS$  的**特征向量**组成的矩阵。

注意：矩阵  $SS^T$  与  $S^TS$  具有相同的特征值，并令  $\Sigma = \sqrt{D}$

### (2) ICP基本原理与推导

由以上结果可得：

$$\sum_{i=1}^N q_i'^T R p_i' = \text{tr}(R P' Q'^T) = \text{tr}(R S) = \text{tr}(R U \Sigma V^T)$$

由矩阵的迹的性质  $\text{tr}(AB) = \text{tr}(BA)$ ，可得：

$$\sum_{i=1}^N q_i'^T R p_i' = \text{tr}(\Sigma V^T R U)$$

又有：

$$V^T R U (V^T R U)^T = V^T R U U^T R^T V = I$$

即，矩阵  $V^T R U$  为**正交阵**。

### (2) ICP基本原理与推导

令  $M = V^T R U$ , 则有:

$$\begin{aligned} \sum_{i=1}^N q_i'^T R p_i' &= \text{tr}(\Sigma M) \\ &= \text{tr} \left( \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \end{bmatrix} \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1n} \\ m_{21} & m_{22} & \cdots & m_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ m_{n1} & m_{n2} & \cdots & m_{nn} \end{bmatrix} \right) \\ &= \sum_{k=1}^n \sigma_k m_{kk} \end{aligned}$$

### (2) ICP基本原理与推导

由于  $M = V^T R U$  是正交矩阵, 则有:

$$M M^T = I$$

即:

$$\begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1n} \\ m_{21} & m_{22} & \cdots & m_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ m_{n1} & m_{n2} & \cdots & m_{nn} \end{bmatrix} \begin{bmatrix} m_{11} & m_{21} & \cdots & m_{n1} \\ m_{12} & m_{22} & \cdots & m_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ m_{1n} & m_{2n} & \cdots & m_{nn} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$$

则有:

$$m_{11}^2 + m_{12}^2 + \cdots + m_{1n}^2 = 1$$

故:

$$m_{11}^2 \leq 1$$

同理可得:

$$m_{22}^2 \leq 1, \cdots, m_{nn}^2 \leq 1$$

### (2) ICP基本原理与推导

令  $M = V^T R U$ , 则有:

$$\sum_{i=1}^N q_i'^T R p_i' = \sum_{k=1}^n \sigma_k m_{kk} \leq \sum_{k=1}^n \sigma_k$$

即是说, 当  $m_{kk} = 1$  时, 上式左边最大。又由于  $M$  为正交阵, 故当  $m_{kk} = 1$ , 矩阵其余元素均为 0, 即  $M$  必为**单位阵  $I$** , 即:

$$M = I = V^T R U$$

故有:

$$R = V U^T$$

将求出的  $R$  代回  $t = \bar{q} - R\bar{p}$ , 即可求出平移向量  $t$ 。



### (2) ICP基本原理与推导

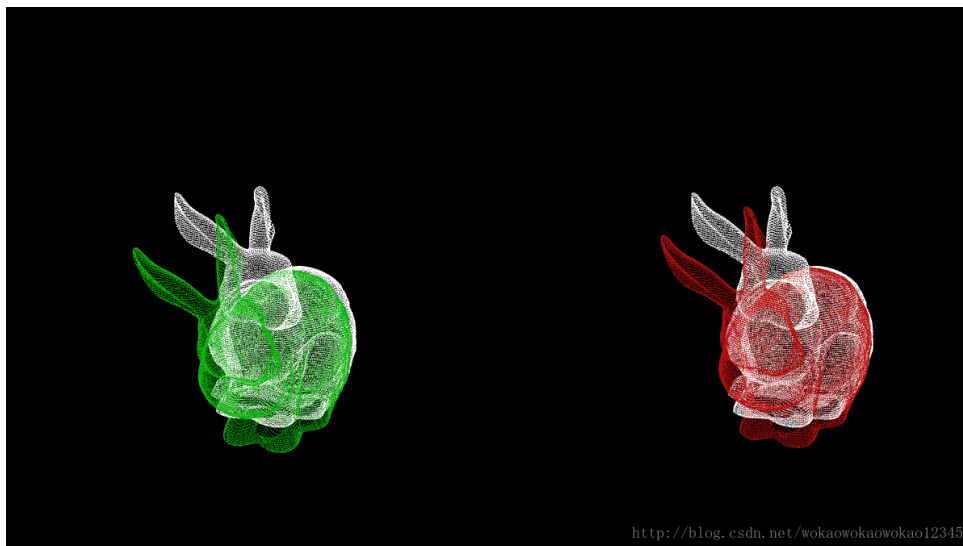
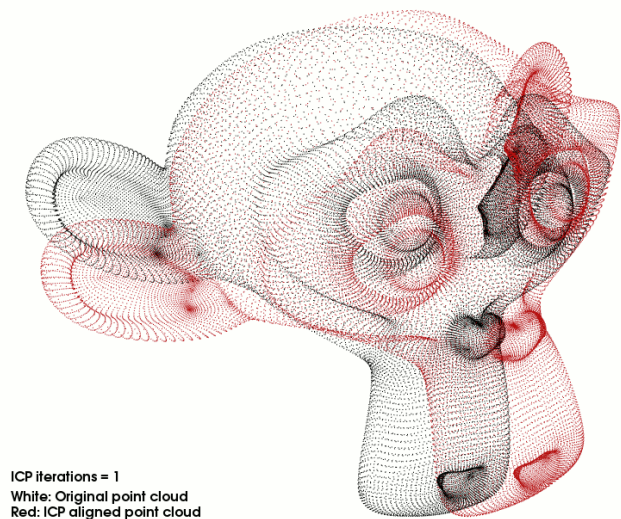
#### ④ 对点云进行转换，计算误差

得到  $R$  和  $t$  之后，对点云进行坐标变换，再计算对应点云间的误差函数  $E(R, t)$ 。

#### ⑤ 不断迭代，直至误差小于某一个值

如果误差  $E(R, t)$  小于某一给定的阈值，或者迭代次数大于预设的最大迭代次数，则停止迭代计算；否则返回第2步，直到满足迭代停止条件为止。

### (3) ICP配准效果演示



### (4) ICP算法的优缺点

#### ICP 配准优点:

- 简单，不必对点云进行分割和特征提取
- 初值较好情况下，精度和收敛性都不错

#### ICP 配准缺点:

- 寻找最近对应点的计算量较大
- 只考虑了点与点距离，缺少对点云结构信息的利用
- 两帧点云变换较大时，匹配容易陷入局部极值

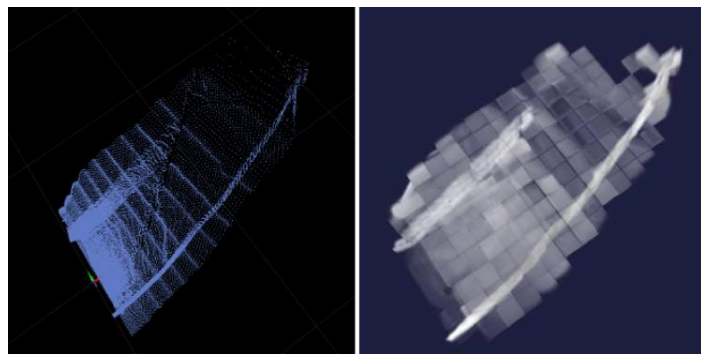
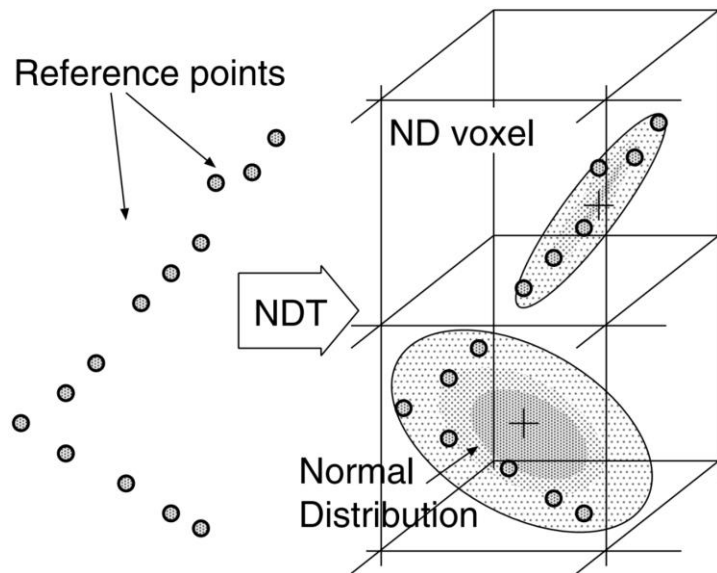
### (5) 基于ICP算法的一些改进

原始的 ICP 算法计算成本大，对初始变换敏感，容易陷入局部最优解。因此自 ICP 提出以来，有许多学者提出了基于 ICP 的改进算法，举例如下：

- Point-to-Plane ICP，该算法是考虑“源点”到“目标点所在平面”的距离，比起直接计算点到点距离，考虑了点云的局部结构，精度更高，不容易陷入局部最优；但 point-to-plane 的优化是一个非线性问题，速度比较慢，一般使用其线性化近似；
- Plane-to-Plane ICP，该算法考虑源点云的局部结构，计算面到面的距离；
- Generalized ICP (GICP)，综合考虑 point-to-point、point-to-plane 和 plane-to-plane 三种策略，精度、鲁棒性都有所提高；
- Normal Iterative Closest Point (NICP)，考虑法向量和局部曲率，更进一步利用了点云的局部结构信息，其论文中实验结果比 GICP 的性能更好。

### (1) NDT基本原理与推导

NDT算法的基本思想是将点云所占的空间划分成指定大小的**网格单元 (Cell)** 或者**体素 (Voxel)** , 然后基于每个网格 (或体素) 中点的具体分布情况, 计算出每个网格 (或体素) 的**正态分布参数**, 即均值和协方差矩阵。我们把这一步骤称为对点云进行NDT化, 其示意图如下:



目标点云网格划分示意图

想了解更多细节可参阅: [The Normal Distributions Transform: A New Approach to Laser Scan Matching](#)

### (1) NDT基本原理与推导

对点云进行NDT化之后，先根据目标点云 (Target) 来构建多维变量的正态分布。如果**变换参数**能使得两幅激光点云数据匹配得很好，那么源点云 (Source) 在目标点云中的**联合概率密度**将会很大。基于此原理，可以利用合理的**优化方法**求出使得联合概率密度之和最大的变换参数，从而实现两幅激光点云数据的最优匹配效果。

算法的**输入**：源点云和目标点云，停止迭代的标准。

算法的**输出**：旋转矩阵和平移向量，即转换矩阵。

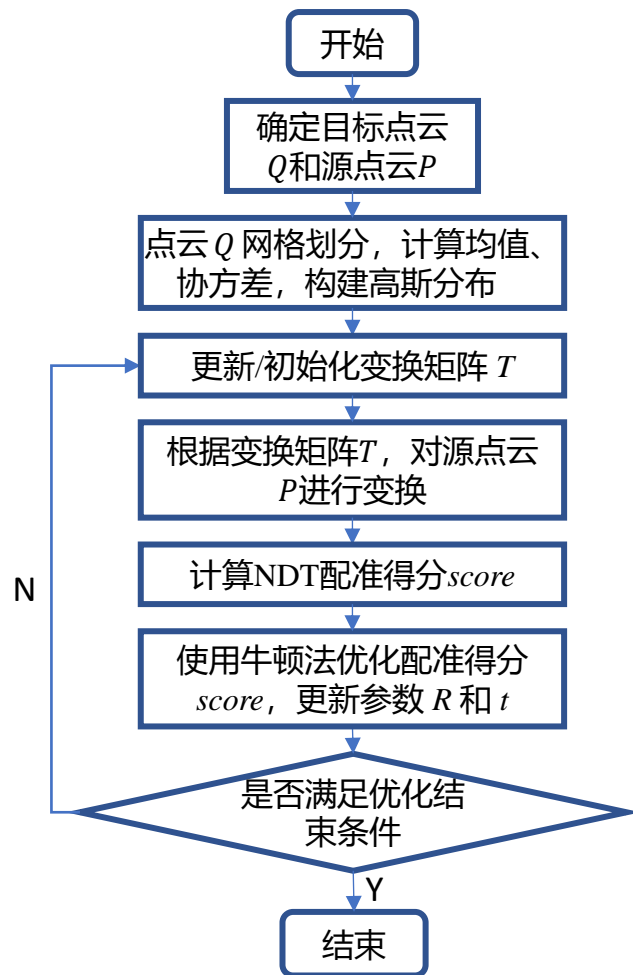




#### (1) NDT基本原理与推导

NDT 算法的基本流程：

- ① 确定源点云  $P$  和目标点云  $Q$ ;
- ② 对目标点云  $Q$  进行**网格划分**，并计算各网格的均值、协方差，**构建高斯分布**;
- ③ 更新/初始化变换矩阵  $T(R, t)$ ;
- ④ 根据变换矩阵  $T$  对源点云  $P$  进行**变换**;
- ⑤ 计算**NDT配准得分  $score$** ;
- ⑥ 通过牛顿优化算法，优化**配准得分  $score$** ，求得**优化增量**;
- ⑦ 不断**迭代**，直至满足最优解的条件。



### (1) NDT基本原理与推导

#### ① 确定源点云 $P$ 和目标点云 $Q$

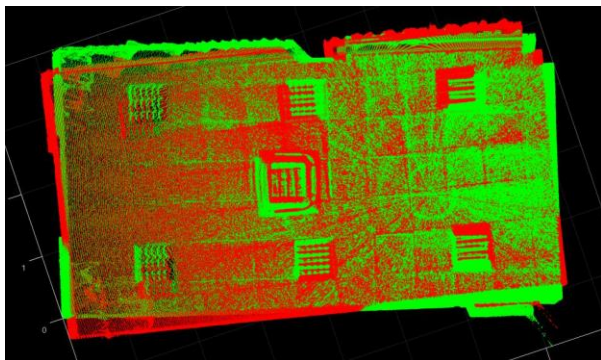
这一步工作就是明确用于配准的两帧点云数据是哪两帧，以及谁是目标点云、谁是源点云。

源点云  $P$ :  $P = \{p_1, p_2, \dots, p_{N_p}\}$

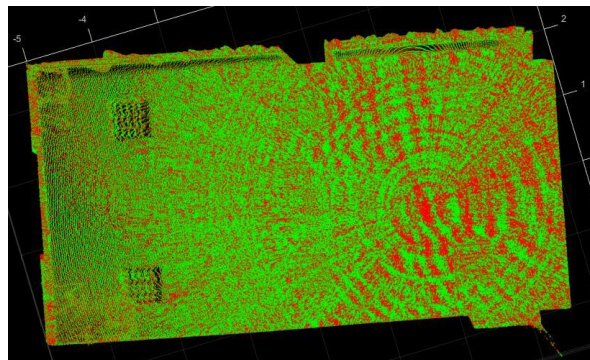
目标点云  $Q$ :  $Q = \{q_1, q_2, \dots, q_{N_q}\}$

变换矩阵  $T$ :  $T(R, t)$

优化目标:  $\arg \max_T \prod_{i=1}^{N_p} f(q_i, T(p_i))$



两帧点云配准前



两帧点云配准后

### (1) NDT基本原理与推导

② 对目标点云  $Q$  进行网格划分，并计算每个网格的均值、协方差，构建多维高斯分布

均值:  $\mu = \frac{1}{N_Q} \sum_{i=1}^{N_Q} q_i$     协方差:  $\Sigma = \frac{1}{N_Q} \sum_{i=1}^{N_Q} (q_i - \mu)(q_i - \mu)^T$

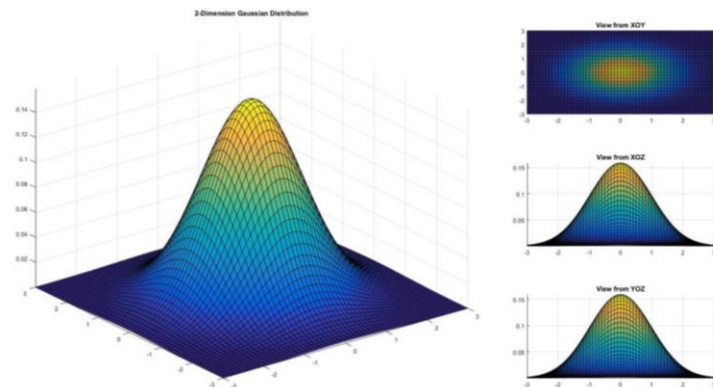
多维高斯分布:  $f(x) = \frac{1}{(2\pi)^{3/2} \cdot |\Sigma|^{1/2}} \exp\left(-\frac{(x - \mu)^T \Sigma^{-1} (x - \mu)}{2}\right)$

③ 初始化变换矩阵  $T(R, t)$  (通常可赋予零值或利用轮速里程计数据赋值)

$$T(R, t) = \begin{bmatrix} R_{3 \times 3} & t_{3 \times 1} \\ v_{1 \times 3} & s_{1 \times 1} \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ v_x & v_y & v_z & s \end{bmatrix}$$



$R$  代表旋转矩阵,  $t$  代表平移向量,  $v$  代表透视变换向量,  $s$  代表整体的比例因子。



### (1) NDT基本原理与推导

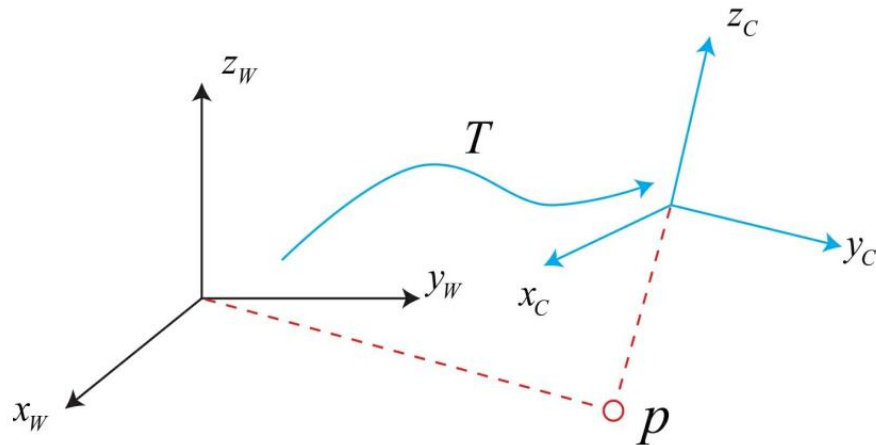
#### ④ 基于变换矩阵 $T(R, t)$ , 对源点云 $P$ 进行变换

对于源点云  $P$  中的每个点,  $p_i \in P = \{p_1, p_2, \dots, p_{N_p}\}$  进行如下变换:

$$p'_i = R p_i + t$$

即有:

$$p'_i = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} p_i + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}$$



#### ⑤ 计算NDT配准得分score

基于矩阵  $T(R, t)$ , 将  $p_i$  变换为  $p'_i$  后, 可计算点  $p'_i$  在点云  $Q$  的多维高斯分布中的概率:

$$f(p'_i) = \frac{1}{(2\pi)^{3/2} \cdot |\Sigma_i|^{1/2}} \exp\left(-\frac{(p'_i - \mu_i)^T \Sigma_i^{-1} (p'_i - \mu_i)}{2}\right)$$



这个概率的意义是什么?

### (1) NDT基本原理与推导

将所有点的概率相加，得到用于**衡量配准性能**的NDT配准**得分**— $score$ ：

$$\begin{aligned} score(R, t) &= \sum_i^{N_p} \frac{1}{(2\pi)^{3/2} \cdot |\Sigma_i|^{1/2}} \exp\left(-\frac{(p'_i - \mu_i)^T \Sigma_i^{-1} (p'_i - \mu_i)}{2}\right) \\ &= \sum_i^{N_p} \frac{1}{(2\pi)^{3/2} \cdot |\Sigma_i|^{1/2}} \exp\left(-\frac{(Rp_i + t - \mu_i)^T \Sigma_i^{-1} (Rp_i + t - \mu_i)}{2}\right) \end{aligned}$$

原理：基于变换矩阵  $T(R, t)$ ，将原点云中的每一个点  **$p_i$  变换为  $p'_i$** ，则  $p'_i$  将落入目标点云NDT化之后的**某一网格之中**。假设源点云  $P$  与目标点云  $Q$  能够很好地匹配，即是说点  $p'_i$  所处的位置与  $p_i$  所处的位置，在全局坐标系下对应于三维空间中的**同一结构**。因此，点  $p'_i$  在该网格所对应的高斯分布中产生**较大的概率密度**，而在其他网格所对应的高斯分布中产生很小的概率密度。

### (1) NDT基本原理与推导

#### ⑥ 通过牛顿优化算法，优化NDT配准得分 $score$ ，求得优化增量

利用牛顿优化算法，对NDT配准得分  $score$  进行优化，求取  $score$  **最大**时对应的  $R$  和  $t$ ：

$$T(R, t) = \arg \max_{R, t} \sum_i^{N_p} \frac{1}{(2\pi)^{3/2} \cdot |\Sigma_i|^{1/2}} \exp \left( -\frac{(Rp_i + t - \mu_i)^T \Sigma_i^{-1} (Rp_i + t - \mu_i)}{2} \right)$$

牛顿优化算法通过不断迭代求取  $R$  和  $t$ ，即该方法每次计算出增量  $\Delta R$  和  $\Delta t$ ，并用增量不断更新现有参数，即有：

$$R \leftarrow R + \Delta R$$

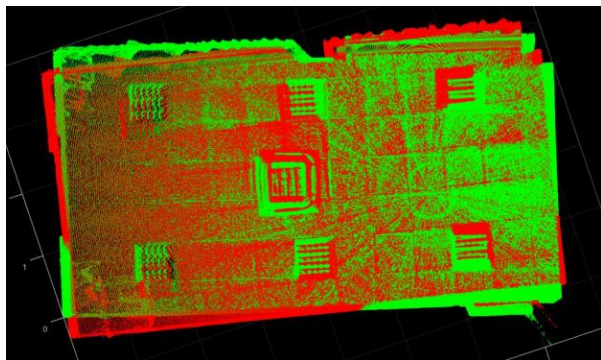
$$t \leftarrow t + \Delta t$$

注意：1. 实际优化过程中，一般是最小化  $-score$ ，而非最大化  $score$ 。

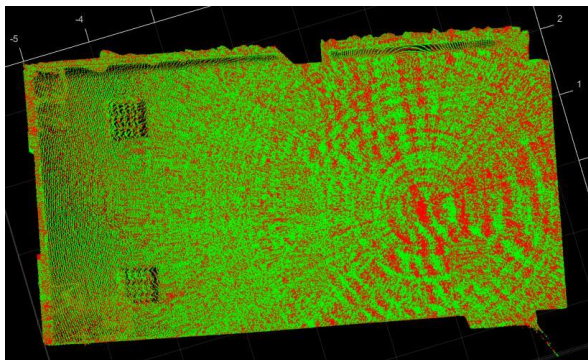
2. 牛顿优化算法细节请参阅：<https://www.cnblogs.com/21207-iHome/p/8039741.html>

#### (1) NDT基本原理与推导

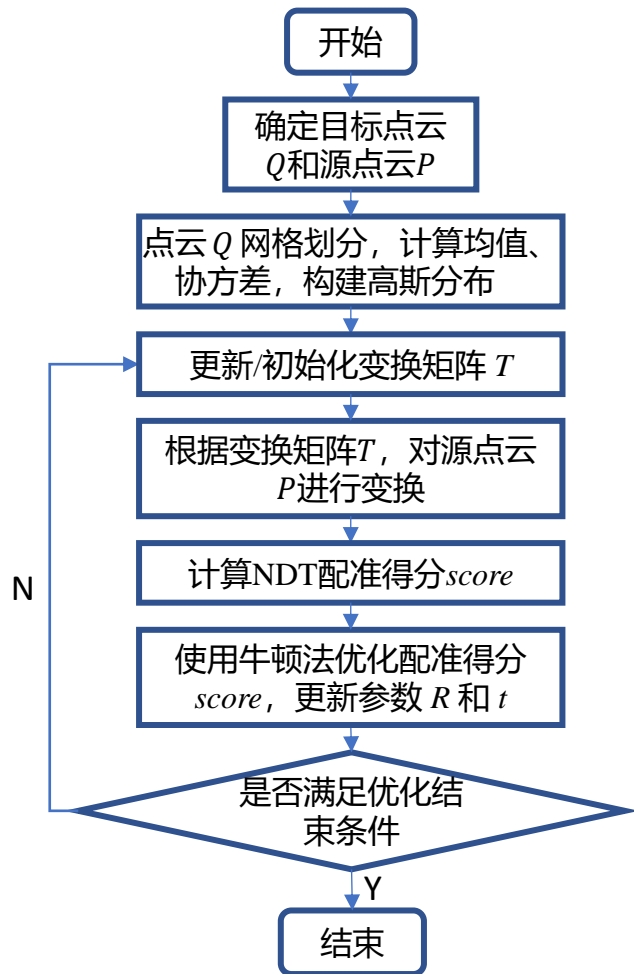
⑦ 利用更新后的  $R$  和  $t$  进行重新对源点云  $P$  进行变换，不断迭代优化，直到获得变换矩阵  $T$  的最优解（满足优化结束条件）。



两帧点云配准前

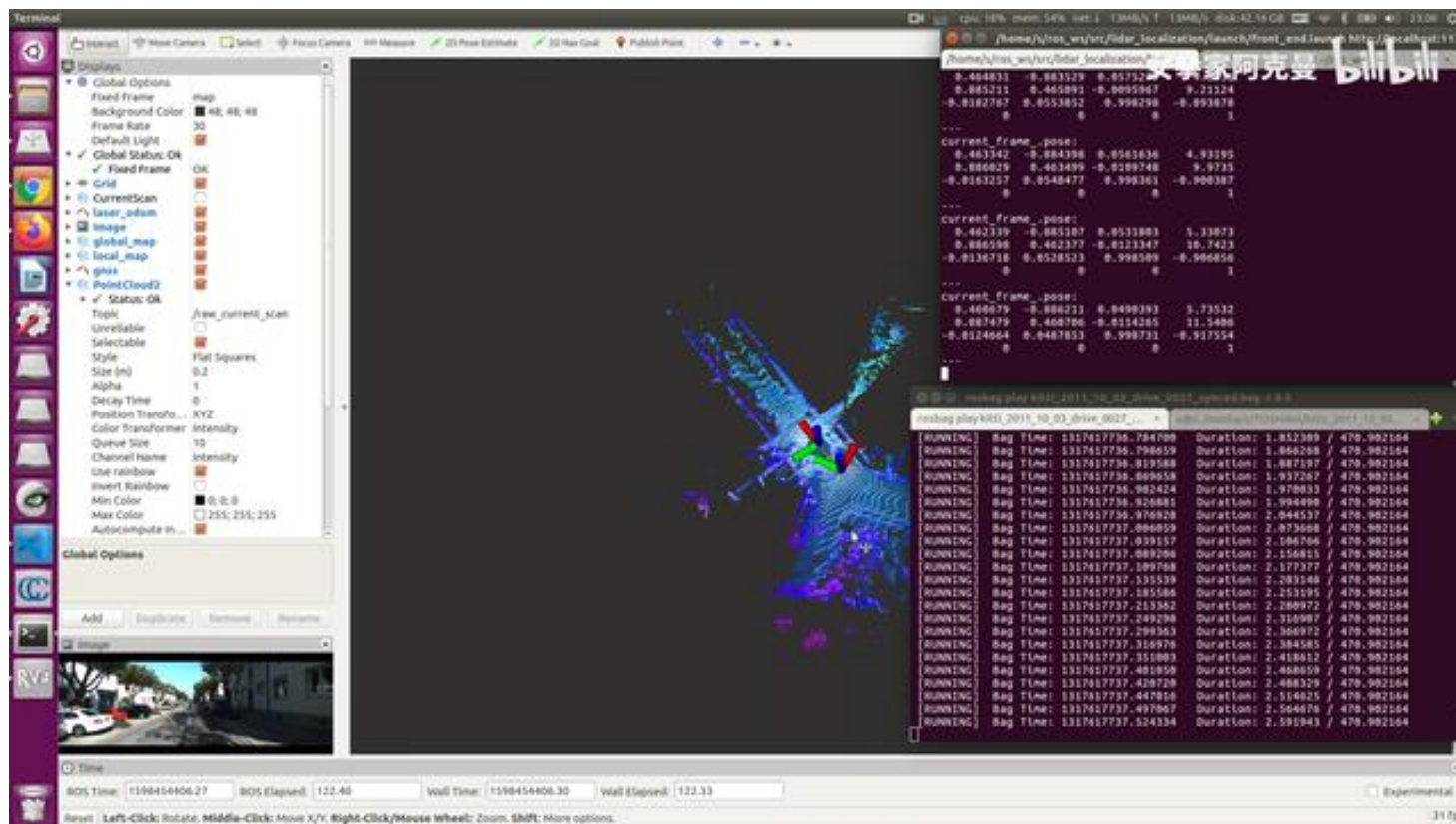


两帧点云配准后





## (2) NDT里程计性能演示





### (3) NDT算法的优缺点

#### NDT 配准优点:

- NDT耗时稳定, 配准结果**对初值不敏感**; 初值误差偏大时, 也能很好的纠正过来;
- 由于NDT算法采用了高效的计算方法, 因此在处理**大规模点云**数据时具有**较高的效率和精度**。

#### NDT 配准缺点:

- NDT算法受**参数设置**的影响较大, 需要合理地设置网格的分辨率、迭代优化的次数等;
- 在**点云数量较大**和两帧**点云相差较多**时, NDT也会出现匹配失误的情况。