NDT - TKU

heaton

What is ndt_tku

- **A 3-D Scan Matching using Improved 3-D Normal Distributions Transform for Mobile Robotic Mapping**
- 名古屋大學 竹內先生(TAKEUCHI)提出的 NDT 優化版
- **Autoware**/ros/src/computing/perception/localization/packages/ndt_localizer/nodes/**ndt_matching_tku/**

Why is ndt_tku

- 是指導教授提出的
- PCL 的 cuda 化極度麻煩，工程師表示不如自幹一套，然後 cuda 化

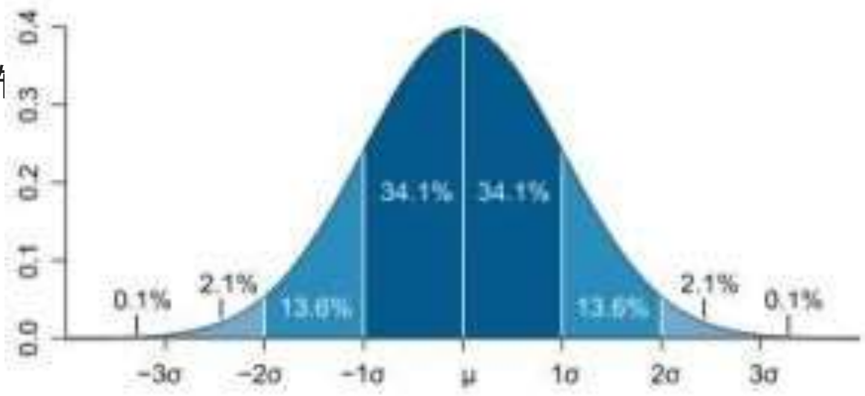
Outline ndt_tku

- ND
- NDT in slam
- NDT_TKU

Normal Distribution

- 正態分佈 - 是一個在數學、物理及工程等領域都非常重要的機率分佈，由於這個分布函數具有很多非常漂亮的性質.使得其在諸多涉及統計科學離散科學等領域的許多方面都有著重大的影響力.
- 符合
  - 台灣收入分佈
- 不符合
  - 骰子各面機率，我跟連家人的收入分佈

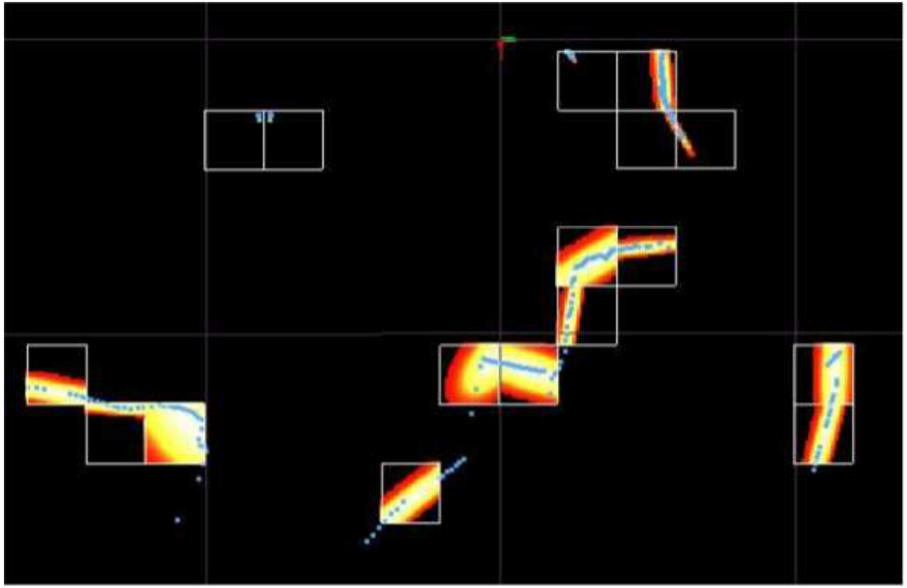
$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

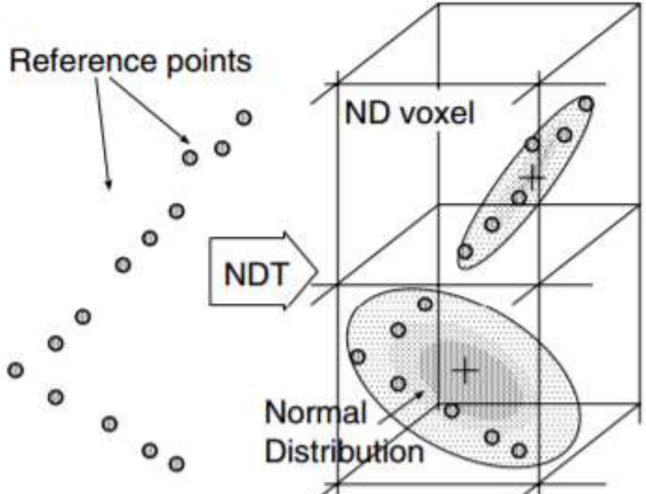


Normal Distribution Transform

Subdivide the space occupied by the scan into a grid of cells.

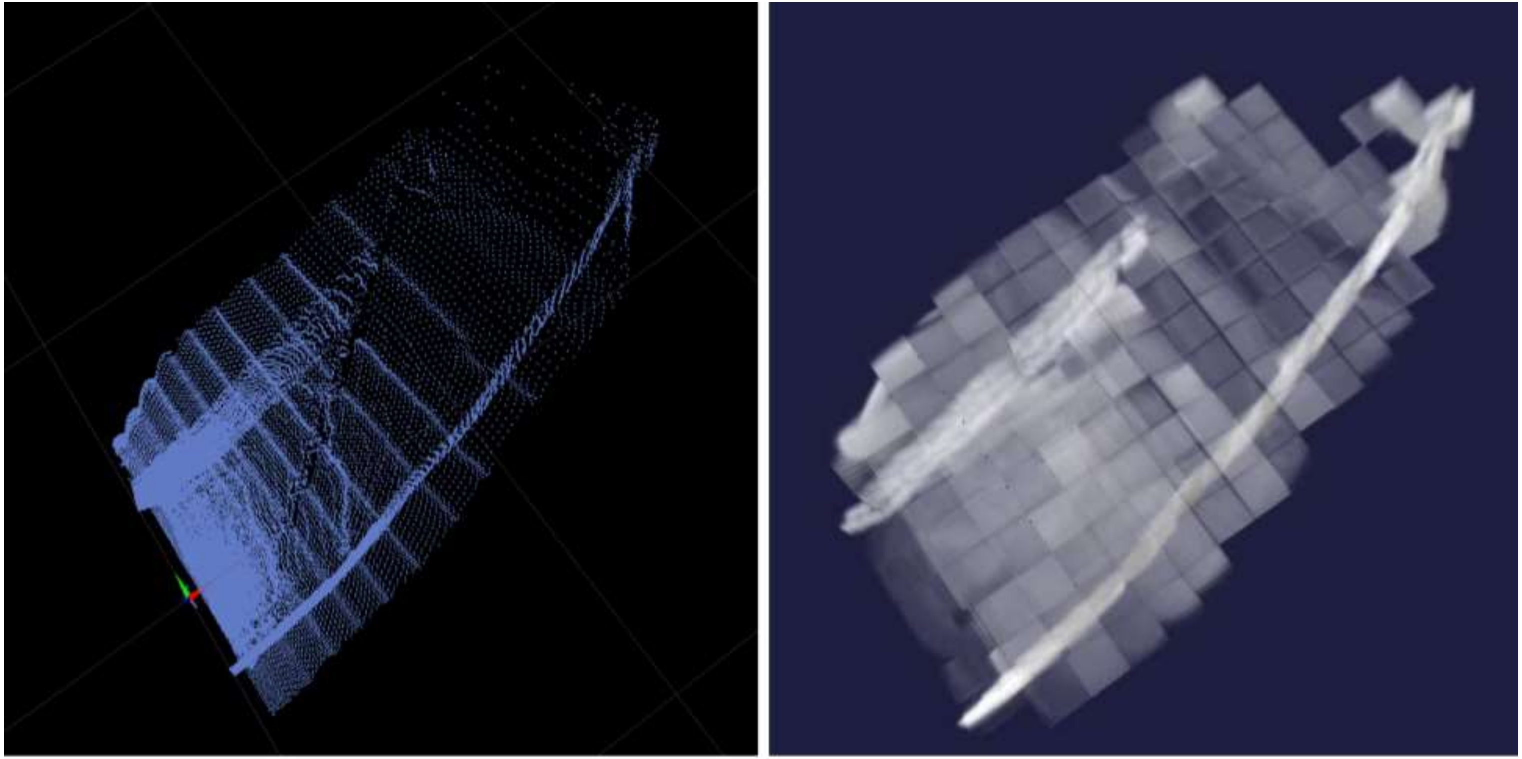
A PDF is computed for each cell, based on the point distribution within the cell





(a) NDT convertes reference scan into Normal Distribusion on each ND voxels

NDT in tunnel - 3D



(a) Original point cloud. (b) NDT representation.

如何表示點雲的機率分布

- multivariate probability function p(~x)l

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

- mean

$$\vec{\mu} = \frac{1}{m}\sum_{k=1}^{m}\vec{y}_k,$$

- covariance

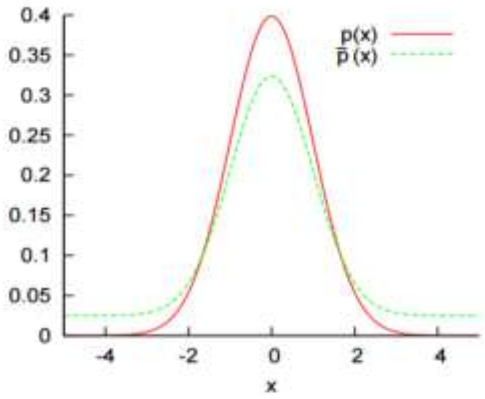
$$\mathbf{\Sigma} = \frac{1}{m-1}\sum_{k=1}^{m}(\vec{y}_k-\vec{\mu})(\vec{y}_k-\vec{\mu})^{\mathrm{T}},$$

Scan registration

- The current scan is represented as a point cloud X = {~x1, . . . , ~xn}. Assume that there is a spatial transformation function T(~p, ~x) that moves a point ~x in space by the pose ~p.

- Given some PDF p(~x) for scan points , the best pose ~p should be the one that maximises the likelihood function

$$\Psi = \prod_{k=1}^{n} p(T(\vec{p}, \vec{x}_k))$$



(a) Likelihood

Scan registration

- Given a set of points X = {~x1, . . . , ~xn}, a pose ~p, and a transformation function T(~p, ~x) to transform point ~x in space by ~p, the NDT score function s(~p) for the current parameter vector is

$$s(\vec{p}) = -\sum_{k=1}^{n} \tilde{p}\left(T(\vec{p}, \vec{x}_k)\right),$$

- Using such a Gaussian approximation, the influence of one point from the current scan on the NDT score function is

$$\tilde{p}(\vec{x}_k) = -d_1 \exp\left(-\frac{d_2}{2}(\vec{x}_k - \vec{\mu}_k)^{\mathrm{T}} \boldsymbol{\Sigma}_k^{-1} (\vec{x}_k - \vec{\mu}_k)\right),$$

Newton’s algorithm for

- Newton’s algorithm can be used to find the parameters ~p that optimise s(~p)
- Newton’s method iteratively solves the equation H∆~p = −~g
- g and H are partial differential and second order partial differential of

$$g_i = \frac{\delta s}{\delta p_i} = \sum_{k=1}^{n} d_1 d_2 \vec{x}_k'^{\mathrm{T}} \boldsymbol{\Sigma}_k^{-1} \frac{\delta \vec{x}_k'}{\delta p_i} \exp\left(\frac{-d_2}{2} \vec{x}_k'^{\mathrm{T}} \boldsymbol{\Sigma}_k^{-1} \vec{x}_k'\right).$$

The entries H_{ij} of the Hessian matrix $\mathbf{H}$ are

$$H_{ij} = \frac{\delta^2 s}{\delta p_i \delta p_j} =$$

$$\sum_{k=1}^{n} d_1 d_2 \exp\left(\frac{-d_2}{2} \vec{x}_k'^{\mathrm{T}} \boldsymbol{\Sigma}_k^{-1} \vec{x}_k'\right) \left(-d_2 \left(\vec{x}_k'^{\mathrm{T}} \boldsymbol{\Sigma}_k^{-1} \frac{\delta \vec{x}_k'}{\delta p_i}\right) \left(\vec{x}_k'^{\mathrm{T}} \boldsymbol{\Sigma}_k^{-1} \frac{\delta \vec{x}_k'}{\delta p_j}\right) + \right.$$

$$\left. \vec{x}_k'^{\mathrm{T}} \boldsymbol{\Sigma}_k^{-1} \frac{\delta^2 \vec{x}_k'}{\delta p_i \delta p_j} + \frac{\delta \vec{x}_k'}{\delta p_j}^{\mathrm{T}} \boldsymbol{\Sigma}_k^{-1} \frac{\delta \vec{x}_k'}{\delta p_i} \right). \quad (6.13)$$

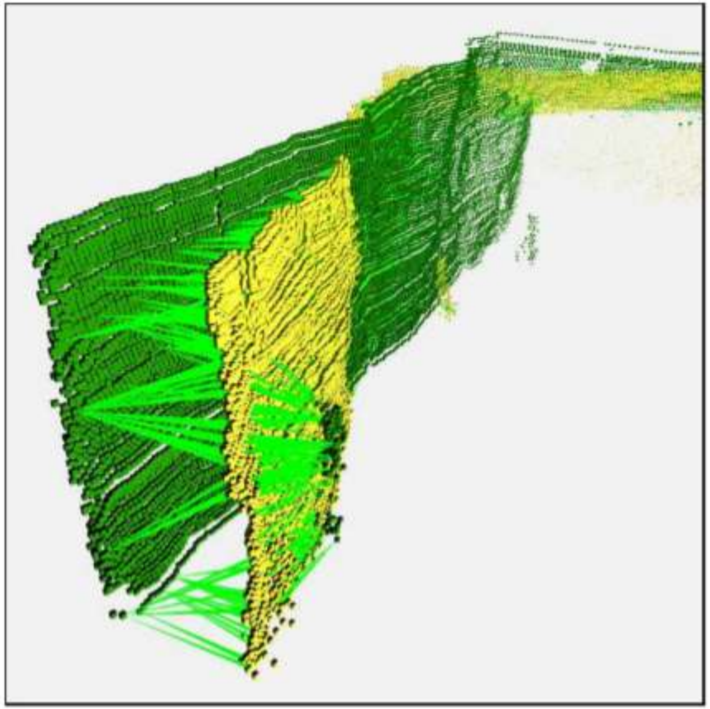
流程

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

我知道大家看數字很痛苦

但我只是要說算這個很麻煩。



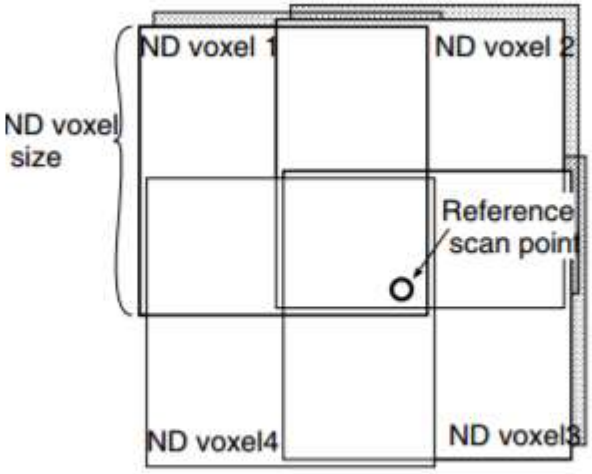
About ND Voxel size

- 太小
  - 運算量大，memory 消耗大
  - 匹配精確
  - 但小於五個點，則很難形成正態分佈
- 太大
  - 運算量少
  - 匹配不精確

NDT - TKU version

格子重疊

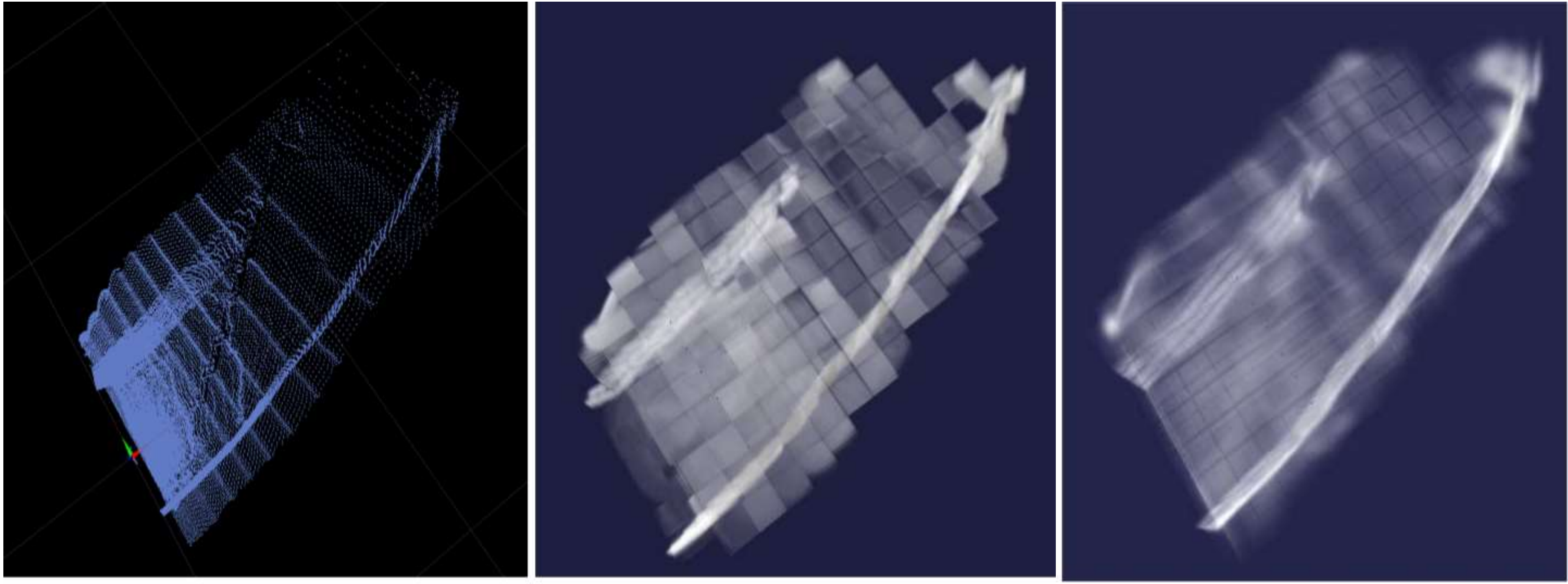
- 提高精確度
- 運算量提高，一點點雲會有八個格子
- Trilinear interpolation



(b) One reference scan point falls into eight overlapped ND voxels

$$\hat{p}(\vec{x}) = \sum_{b=1}^{8} -d_{1(b)} w(\vec{x}, \vec{\mu}_b) \exp\left(-\frac{d_{2(b)}}{2} (\vec{x} - \vec{\mu}_b)^{\mathrm{T}} \Sigma_b^{-1} (\vec{x} - \vec{\mu}_b) \right),$$

圖片有感



(a) Original point cloud.　　(b) NDT representation.

TKU - ND Voxel size

將收斂流程分成兩階段

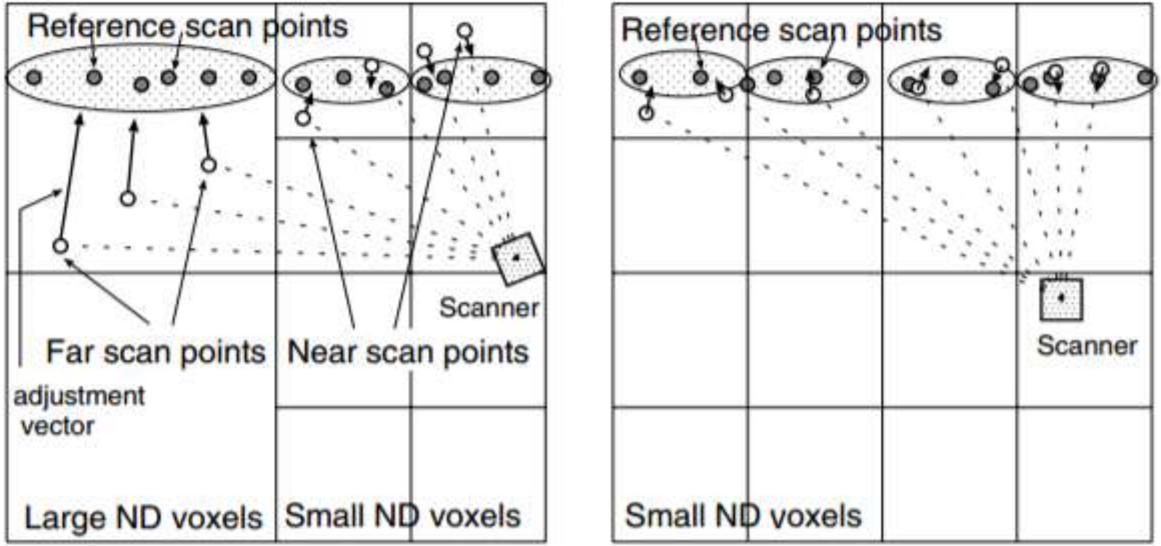
- Converging state

  按照距離切分 ND Voxel size，並運算

- Adjust state

  到一定次數後則通通用

  最小格子來運算



(a) Converging state　　(b) Adjust state

ENDING

THANKs FOR YOUR ATTENTION.

Reference

1. A 3-D Scan Matching using Improved 3-D Normal Distributions Transform for Mobile Robotic Mapping(網路上不公開)

2. The Three-Dimensional Normal-Distributions Transform — an Efficient Representation for Registration, Surface Analysis, and Loop Detection

3. The Normal Distributions Transform:A New Approach to Laser Scan Matching

Other

1. Parameter
    a. voxel size
    b. step size
    c. iterative times

score detail

rameters d_i by requiring that $\tilde{p}(x)$ should behave like $p(x)$ for $x = 0$, $x = \sigma$, and $x = \infty$:

$$\begin{aligned} d_3 &= -\log(c_2), \\ d_1 &= -\log(c_1 + c_2) - d_3, \\ d_2 &= -2\log\left(\left(-\log(c_1 \exp(-1/2) + c_2) - d_3\right)/d_1\right). \end{aligned} \tag{6.8}$$

Using such a Gaussian approximation, the influence of one point from the current scan on the NDT score function is

$$\tilde{p}(\vec{x}_k) = -d_1 \exp\left(-\frac{d_2}{2}(\vec{x}_k - \vec{\mu}_k)^{\mathrm{T}} \mathbf{\Sigma}_k^{-1} (\vec{x}_k - \vec{\mu}_k)\right), \tag{6.9}$$