16-833 SLAM HW4 Dense SLAM with Point-based Fusion

2 Iterative Closest Point (ICP)

2.2 Linearization

Given:

$$\delta R = \begin{bmatrix} 1 & -\gamma & \beta \\ \gamma & 1 & -\alpha \\ -\beta & \alpha & 1 \end{bmatrix}, \quad \delta t = [t_x, t_y, t_z], \quad \sum_{i \in \Omega} r_i^2(\delta R, \delta t) = \left\| n_{q_i}^T((\delta R)p_i' + \delta t - q_i) \right\|^2, \quad p_i' = R^0 p_i + t^0$$

Find:

$$r_i(\alpha, \beta, \gamma, t_x, t_y, t_z) = A_i \begin{bmatrix} \alpha \\ \beta \\ \gamma \\ t_x \\ t_y \\ t_z \end{bmatrix} + b_i$$

A_i is 6×1

b_i is scalar

Solution:

$$r_i(\delta R, \delta t) = n_{q_i}^T((\delta R)p_i' + \delta t - q_i) \rightarrow r_i(\delta R, \delta t) = n_{q_i}^T((\delta R)(R^0 p_i + t^0) + \delta t - q_i)$$

$$r_i(\delta R, \delta t) = n_{q_i}^T \left(\begin{bmatrix} 1 & -\gamma & \beta \\ \gamma & 1 & -\alpha \\ -\beta & \alpha & 1 \end{bmatrix} p_i' + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} - q_i \right)$$

where $i = (1, 2, 3)$

$$r_i(\delta R, \delta t) = [n_{q_1} \; n_{q_2} \; n_{q_3}] \left(\begin{bmatrix} 1 & -\gamma & \beta \\ \gamma & 1 & -\alpha \\ -\beta & \alpha & 1 \end{bmatrix} \begin{bmatrix} p_1' \\ p_2' \\ p_3' \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix} - \begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} \right)$$

$$r_i(\delta R, \delta t) = [n_{q_1} \; n_{q_2} \; n_{q_3}] \begin{bmatrix} p_1' - \gamma p_2' + \beta p_3' + t_x - q_1 \\ \gamma p_1' + p_2' - \alpha p_3' + t_y - q_2 \\ -\beta p_1' + \alpha p_2' + p_3' + t_z - q_3 \end{bmatrix} = \begin{bmatrix} n_{q_1}(p_1' - \gamma p_2' + \beta p_3' + t_x - q_1) \\ n_{q_2}(\gamma p_1' + p_2' - \alpha p_3' + t_y - q_2) \\ n_{q_3}(-\beta p_1' + \alpha p_2' + p_3' + t_z - q_3) \end{bmatrix}$$

$$r_i(\alpha, \beta, \gamma, t_x, t_y, t_z) = \underbrace{\begin{bmatrix} -n_{q_2} p_3' + n_{q_3} p_2' \\ n_{q_1} p_3' - n_{q_3} p_1' \\ -n_{q_1} p_2' + n_{q_2} p_1' \\ n_{q_1} \\ n_{q_2} \\ n_{q_3} \end{bmatrix}}_{A_i \, (6 \times 1)} \begin{bmatrix} \alpha \\ \beta \\ \gamma \\ t_x \\ t_y \\ t_z \end{bmatrix} + \underbrace{\left[n_{q_1}(p_1' - q_1) + n_{q_2}(p_2' - q_2) + n_{q_3}(p_3' - q_3) \right]}_{b_i \text{ (scalar)}}$$

#

2.3 Optimization

Given:

$$\sum_{i=1}^{n} r_i^2(\alpha, \beta, \gamma, t_x, t_y, t_z) = \sum_{i=1}^{n} \left\| A_i \begin{bmatrix} \alpha \\ \beta \\ \gamma \\ t_x \\ t_y \\ t_z \end{bmatrix} + b_i \right\|^2$$

Find: write down the linear system that provides a closed form solution

Solution:

$$delta = (A^T A)^{-1} A^T b \quad , \quad A = \sum_{i=1}^{N} A_i^T A_i \quad , \quad b = \sum_{i=1}^{N} A_i^T b_i \ \#$$

\# Note: use $-b$ in Python solvers.

3 Point-based Fusion

3.2 Merge

Given: $p \in \mathbb{R}^3$, w, $q \in \mathbb{R}^3$, $w=1$

Find: the weight average of the positions in terms of p, q, R_c^w, t_c^w, w

the weight average of normals in terms of n_p, n_q, R_c^w, w

Equation: $$W = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$$

Solutions: $w_i = (w+1)$

$$W_{positions} = \frac{w(p) + 1[R_c^w(q) + t_c^w]}{w+1} \#$$

$$W_{normals} = \frac{w(n_p) + 1[R_c^w(n_q)]}{w+1} \#$$

3.4 Results

Given: width = 680, height = 480, # of total points = 1,363,469

Find: the compression ratio

Equation: $$CR = \frac{\text{\# of total points}}{(H/2 \cdot W/2 \cdot 200)}$$

Solution:

$$CR = \frac{1{,}363{,}469}{\left(\frac{480}{2} \cdot \frac{680}{2} \cdot 200\right)} \approx 0.08355 \text{ or } 8.355\% \#$$