Homework #7

ECE661

Name:Chengzhang Zhong

Email:zhongc@purdue.edu

Step 1. Convert depth image into point cloud

For a depth image, let $\mathrm{u} = (\mathrm{x}, \mathrm{y}, 1)$ be the homogeneous coordinate representation of the image pixel at (x, y). Also, let $\mathrm{D}(\mathrm{u})$ be the depth value at $\mathrm{u} = (\mathrm{x}, \mathrm{y}, 1)$. And K represents the intrinsic camera matrix, which is fixed as:

$$\mathrm{K} = \begin{bmatrix} 365 & 0 & 256 \\ 0 & 365 & 212 \\ 0 & 0 & 1 \end{bmatrix}$$

Thus, the 3D point **P(u)** of the point in the point cloud corresponding to **u** calculated as:

$$\mathbf{P}(\mathbf{u}) = \mathbf{D}(\mathbf{u})\mathbf{K}^{-1}\mathbf{u}$$

Notice that, if the pixel value in the depth image equals to 0, the point in point cloud is marked as (0,0,0).

Step 2. Implement ICP algorithm

Suppose **P** and **Q** are the two point cloud constructed for the same object from two different viewpoints. Now we apply ICP algorithm to find the matrix set **[R t]** which makes the common congruent of **P** and **Q**.

1. Iterate through every point in **P**, find the point in **Q** which has the least Euclidean distance to it as the corresponding point. If the distance is greater than a threshold $\delta = 0.1$, it is rejected as corresponding pair. Moreover, if one of the point in **P** or **Q** is (0,0,0), then we also reject it as corresponding pair.

The result point cloud are $\mathbf{P}^{'}$ and $\mathbf{Q}^{'}$, which have dimensions 3xN, where N is the total number of corresponding pairs.

2. Now, we need to estimate the rotation and translation matrices.

First, calculate the centroid of two point clouds by summing over all points in the point cloud and dividing by N.

$$\mathbf{P_c} = \sum_{\mathbf{i}}^{\mathbf{N}} \frac{\mathbf{P}'(\mathbf{i})}{\mathbf{N}}$$

$$\mathbf{Q_c} = \sum_{\mathbf{i}}^{\mathbf{N}} \frac{\mathbf{Q}'(\mathbf{i})}{\mathbf{N}}$$

Then, re-compute these two point clouds by subtracting each centroid:

$$\mathbf{M_P} = \{\mathbf{P}'(\mathbf{i}) - \mathbf{P_c}\}_{\mathbf{i=1}}^{\mathbf{N}}$$

$$\mathbf{M_Q} = \{\mathbf{Q}'(\mathbf{i}) - \mathbf{Q_c}\}_{\mathbf{j=1}}^{\mathbf{N}}$$

$\mathbf{M_P}, \mathbf{M_Q}$ also have the dimensions of 3xN.

Moreover, the correlation matrix C is computed as:

$$\mathbf{C} = \mathbf{M_Q} * \mathbf{M_P}^{\mathbf{T}}$$

Applying SVD on C as $\mathbf{C} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathbf{T}}$, and matrix R as $\mathbf{R} = \mathbf{V} * \mathbf{U}^{\mathbf{T}}$.

Translation matrix is computed as $\mathbf{t} = \mathbf{P_c} - \mathbf{R} * \mathbf{Q_c}$.

3. Finally, the transform matrix T is computed as:

$$\mathbf{T} = \begin{pmatrix} \mathbf{R} & \mathbf{t}^{\mathbf{T}} \\ \mathbf{0} & \mathbf{1} \end{pmatrix}$$

And the transformed point cloud is computed as:

$$\mathbf{Q_{trans}} = \mathbf{T} * \mathbf{Q}$$

4. Repeat the above steps for the refined point cloud Q for number of iterations M. In this experiment, the M is set to 10.

Step 3. Result demonstration

The point cloud P – green points

The point cloud Q – red points

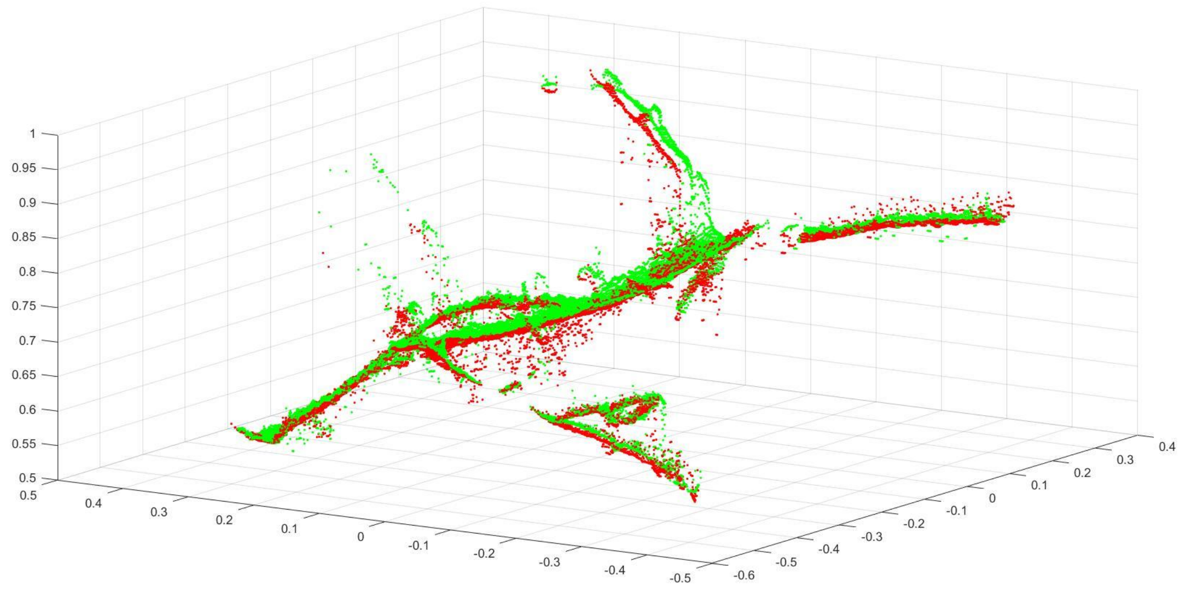

Figure 1.1 The P and Q before alignment

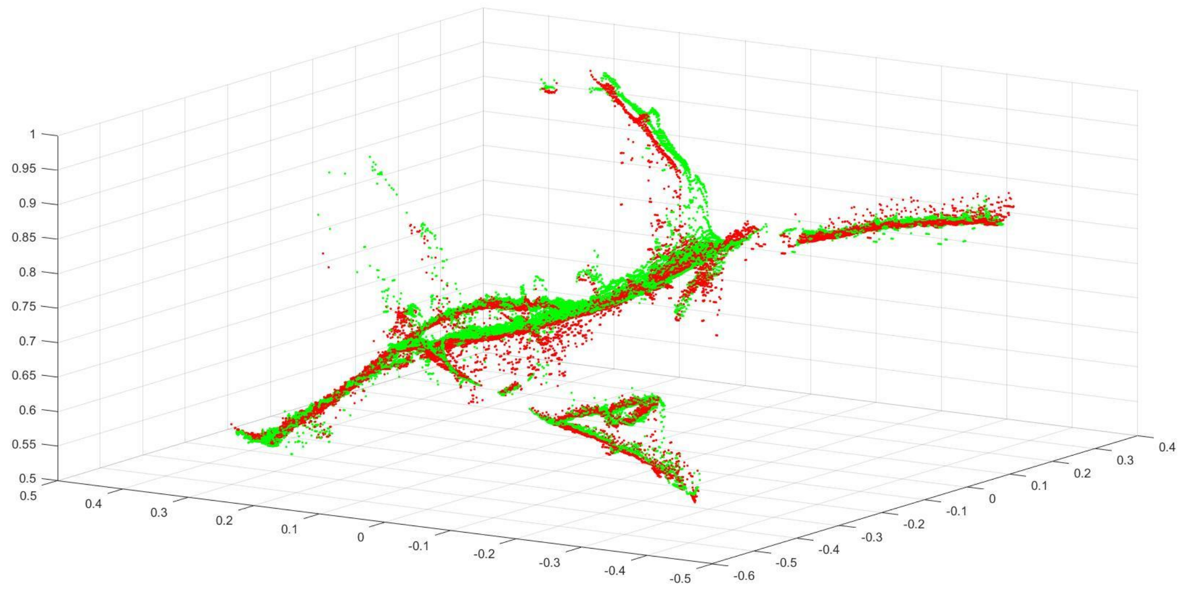

Figure 1.2 The P and Q after 1st iteration

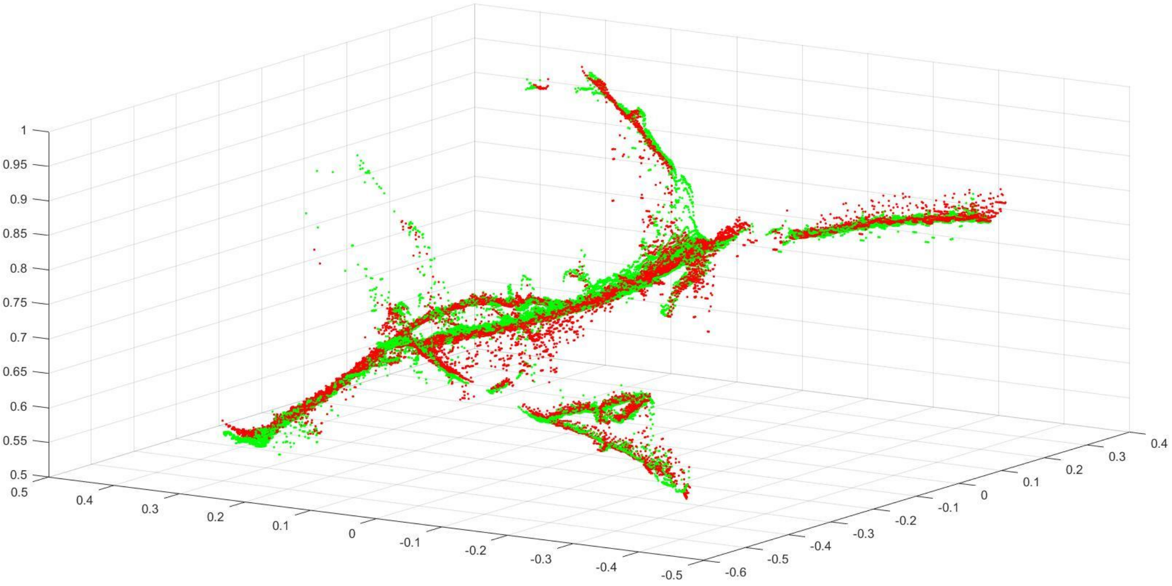

Figure 1.3 The P and Q after 5th iteration

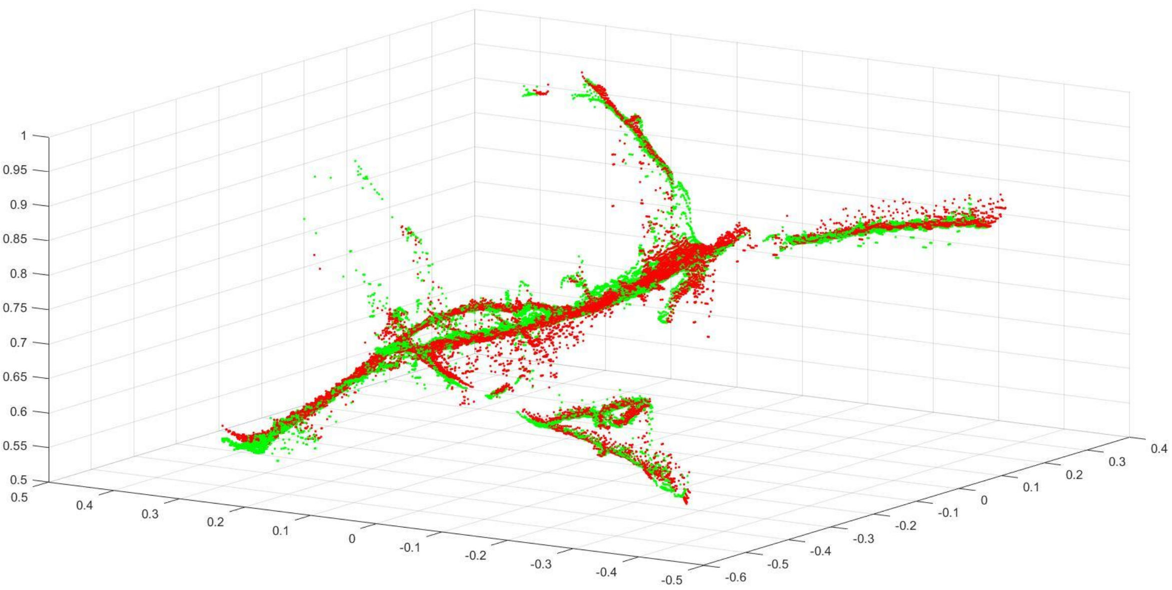

Figure 1.4 The P and Q after 10th iteration

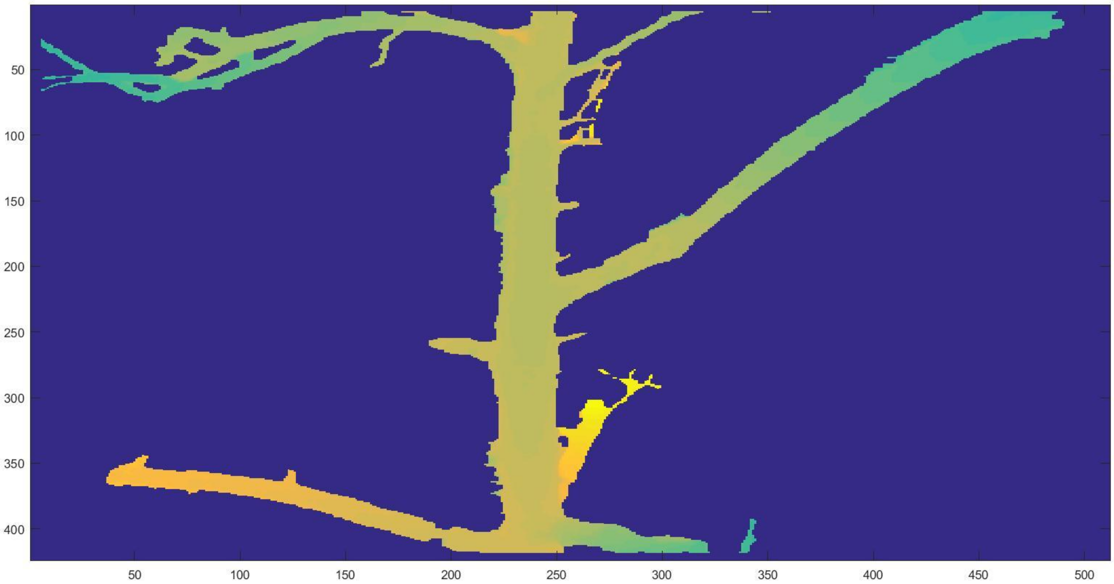

Figure 1.5 Depth image to generate P point cloud

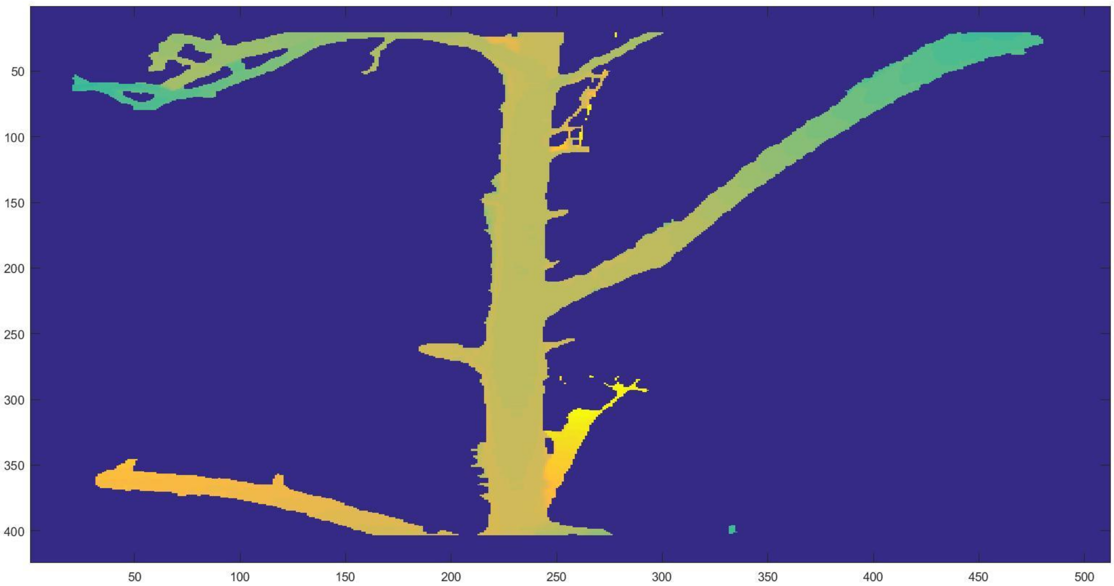

Figure 1.6 Depth image to generate Q point cloud

Conclusion: As the iteration number increases, the point from two point clouds are getting more and more closer to each other. You can view it on the Figure 1.1 to 1.4, especially on top left branch points.