CS210 PROJECT REPORT

Problem(Motivation): For my research project, I need to find the Absolute Transformation Error (ATE) of my estimated camera trajectory of a drone trajectory to its ground truth camera trajectory, where the trajectory is stored as an n-by-3 matrix expressing the x,y,z coordinate of the camera. To get the ATE, the estimated camera trajectory should be mapped to the ground truth trajectory with a least square error. This can be realized by the SVD and Least Squares (LS) problem. Because the key points of the camera trajectory are chosen differently, sometimes, its vertical coordinate will dominate the trajectory estimation. However, when plotting the estimated trajectory, we only care about the horizontal plane, so it is meaningful to test the trajectory mapping in 2D rather than 3D. In the bonus project, I map the camera trajectory of the drone in 2D and 3D with the ground truth, and compare the results.

Method: The key point of solving this problem can be decomposite as such:

We have 2 trajectories:

- Trajectory D which is the ground truth trajectory
- Trajectory M which is the estimated trajectory

The estimated trajectory should be similar to the ground truth trajectory, but with some errors. Their relationship can be explained as:

$$d_i = \mathbf{R}\, m_i + \mathbf{T} + V_i \qquad (1)$$

where **R** is a standard 3 × 3 rotation matrix, **T** is a 3-D translation vector and V_i is a noise vector.

We want to solve the optimal transformation {**R**, **T**} so that we could map the set {mi} onto {di}. This process is equivalent to solving a least squares error problem.

$$\Sigma^2 = \Sigma \, || \, d_i - \mathbf{R} m_i - \mathbf{T} \, || \qquad (2)$$

After equation 2, all the points in {di} and {mi} will have the same centroid. Then equation 2 can be reduced to solving a problem that minimize the function

$$\sum (d_{ci}^T d_{ci} + m_{ci}^T m_{ci} - 2 d_{ci}^T \, \mathbf{R} \, m_{ci}) \qquad (3)$$

Equation 3 is minimized when $2d_{ci}^{T}$ **R** m_{ci} is maximized, which is the same to maximizing Trace(**R H**), where **H** is

$$\sum (m_{ci}\ d_{ci}{}^{T}{}_{i}) \tag{4}$$

Using SVD function in python, we can get the SVD of **H = U**$\wedge$**V**^{T}, then the optimal rotation matrix **R** is

$$\mathbf{R} = \mathbf{V}\mathbf{U}^{\mathbf{T}} \tag{5}$$

File Explanation: The python files read in the estimated trajectory and the ground truth trajectory, and align these two trajectories in the function in align(model,data), using the above method:

- evaluate_ate_scale_dataset_2D.py evaluates the trajectory in 2D;
- evaluate_ate_scale_dataset.py evaluates the trajectory in 3D;

Results and scripts:

- The CS210_script.txt has the receipt to run the python code.
- The ground truth trajectory is stored in the folder “xyzGroundTruth” and the camera trajectory is stored in the folder “camera trajacterEstimation ”.
- The pdf results are saved in the “2D pdf results” and “3D pdf results” folders;
- The ATE is compared in the ATEresults.ods

Results: In the ATEresults.ods file, the ATE is compared in 2D and 3D. Since the same key points are chosen, (which means the input data is the same) Most results in 2D are better than the results in 3D. This is actually consistent with the problem in reality. In the 3D version, since we need to reduce the ATE of each point, we need to compensate for the error in the vertical plane. However, in the 2D case, since we only focus on the data on the horizontal plane, less error will be introduced when transferring and rotating the graph.

Reference

Eggert, David W., Adele Lorusso, and Robert B. Fisher. "Estimating 3-D rigid body transformations: a comparison of four major algorithms." *Machine vision and applications* 9.5 (1997): 272-290.