HW5 Report

Jinxi Xiao
2021533005
ShanghaiTech University

Abstract

This report contains my solutions to the given problems as well as some of my own understandings. In addition, the sections listed in this report does not one-to-one correspond to the problems, but they do overall cover the whole problem set. [1]

1. Dealing with clean data

The algorithm as well as the method for this part would be rather easy, for what we need to do is to follow the *Algorithm 1* in [1]. And here are my results:

For task **one**:

$$\mathbf{R}_l = \begin{bmatrix} 0.01386 & -0.95103 & -0.30879 \\ 0.90568 & -0.11893 & 0.40694 \\ -0.42374 & -0.28530 & 0.85968 \end{bmatrix}$$

$$\mathbf{u}_l = [0\ 1.12647\ -1.40540]^T$$

For task **two**:

$$\mathbf{R}_l = \begin{bmatrix} 0.67654 & -0.62614 & 0.38761 \\ -0.65838 & -0.75008 & -0.06252 \\ 0.32989 & -0.21290 & -0.91970 \end{bmatrix}$$

$$\mathbf{u}_l = [0\ 0.37342\ 0.19655]^T$$

2. Using RANSAC to handle noisy data

First of all, some detailed methods for RANSAC can be found in https://vnav.mit.edu/material/15-RANSAC-notes.pdf and https://en.wikipedia.org/wiki/Random_sample_consensus. My methods are based on these.

Notice that for data package 3, all points contain a certain degree of Gaussian White noises. However, due to the fact that all noises are within a small scale, all points can be regraded as inliers. As a result, there's no need to use RANSAC to filter outliers.

As a result, I combine the 50 clean data in data package 2 and 200 generated large noises into a new data package, and such 200 generated data can be regarded as outliers. Visualization results can be found in Figure 1. The computed results would still be valid.

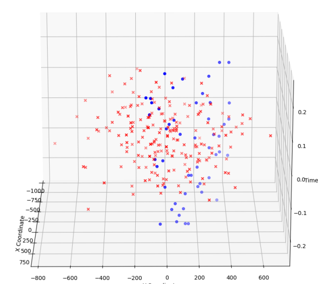

Figure 1. Self Generated data package. Blue dots are clean data and red crosses are outliers.

3. Validation for results

Suppose we have obtained the 3 DoF velocity in the reference frame as v_c, the basis matrix in the line frame expressed in the reference frame $\{e_1, e_2, e_3\}$, and the velocity in the line frame $[0\ u_y\ u_z]$.

Define

$$v_{est} = u_y e_2 + u_z e_3$$

and

$$v_{gt} = (I - e_1 e_1^T) v_c$$

After normalization, the following constraint should be satisfied

$$|v_{gt} \cdot v_{est}| = 1$$

[1] Codes are stored at my github repo

References

[1] L. Gao, D. Gehrig, H. Su, D. Scaramuzza, and L. Kneip, “An n-point linear solver for line and motion estimation with event cameras,” 2024.