Institute for Aerospace Studies
UNIVERSITY OF TORONTO

AER1217: Winter 2022 Lab 4
March 20, 2025

AER1217: Development of Unmanned Aerial Vehicles
Lab 4: Stereo Camera-based Visual Odometry

1 Introduction

This is the fourth lab in a series designed to complement the lecture material and help you gain experiences in the development of autonomous systems. In this lab, you are required to implement a stereo camera-based visual odometry (VO) to estimate the pose of an vehicle. The Kitti dataset will be used for the VO implementation. We provide the `2011_09_26_drive_0005 city data` and a code repository in here to help you get started. For detailed information about the format of Kitti dataset and the vehicle setup, please refer to the official website.

1.1 Lab Objectives

The lab is split up into two parts:

1. Implement a RANSAC algorithm for outlier rejection,
2. Implement a scalar-weighted point cloud alignment for pose estimation.

1.2 Requirements

For this lab, you are required to estimate the pose of the vehicle with respect to the first frame based on stereo camera visual information. The ground truth is provided by a RTK GPS localization system. A start code is provided for you which includes feature detection, corresponding feature matching, and visualization.

The requirements are:

- **Outlier rejection:** Implement the RANSAC algorithm for outlier rejection and visualize the inlier features on the right image plane.
- **Pose estimation:** Implement the scalar-weighted point cloud alignment to estimate the pose of the vehicle.
- **Discussion:** Compare the pose estimation results with respect to the ground truth and comment on the VO performance. What's the limitation of VO-based navigation? How can we improve the overall performance?

2 Lab Components

As shown in the tutorial, the VO pipeline involves *(1)* image de-wrap and rectification, *(2)* keypoint detection and matching, *(3)* outlier rejection, and *(4)* pose estimation. In lab4, the raw image has been pre-processed and the keypoints detection and matching algorithm is provided. Therefore, the main components in this lab are outlier rejection and pose estimation.

In function `find_feature_correspondences`, the corresponding features between previous frame and current frame are computed through a brute-force matching. The output is a $N \times 8$ feature correspondence matrix where N indicates the number of matched features and the i-th row indicates the i-th matched feature points expressed in (1) previous left image frame (`prev_l_x, prev_l_y`), (2) previous right image frame (`prev_r_x, prev_r_y`), (3) current left image frame (`cur_l_x, cur_l_y`), and (4) current right image frame (`cur_r_x, cur_r_y`).

As a general guide, the rest a stereo-camera based visual odometry process involves the following:

1. Use the inverse stereo camera model to convert the matched feature points into 3D point clouds expressed in previous frame $\{\mathbf{p}_a^j\}$ and current frame $\{\mathbf{p}_b^j\}$, respectively.
2. Start the RANSAC algorithm by randomly selecting three corresponding points to fit the model
$$\mathbf{p}_b^j = \mathbf{C}_{ba}\mathbf{p}_a^j + \mathbf{r}_b^{ab} \tag{1}$$
3. Based on the computed model $\{\mathbf{C}_{ba}, \mathbf{r}_b^{ab}\}$, test the rest of the corresponding feature points and classify inlier/outlier based on residual errors.
4. Repeat the RANSAC algorithm for a large number of trails (or stop the algorithm by a pre-defined termination criteria) and keep the largest set of inliers across all trials.
5. With the filtered point clouds, compute the center of each point cloud and construct the $\mathbf{W}_{ba}$ matrix.
$$\mathbf{p}_a = \frac{1}{w}\sum_{j=1}^{J} w^j \mathbf{p}_a^j, \qquad \mathbf{p}_b = \frac{1}{w}\sum_{j=1}^{J} w^j \mathbf{p}_b^j, \qquad w = \sum_{j=1}^{J} w^j \tag{2}$$
$$\mathbf{W}_{ba} = \frac{1}{w}\sum_{j=1}^{J} w^j \left(\mathbf{p}_b^j - \mathbf{p}_b\right)\left(\mathbf{p}_a^j - \mathbf{p}_a\right)^T \tag{3}$$
6. Apply SVD to the $\mathbf{W}_{ba}$ matrix
$$\mathbf{V}\mathbf{S}\mathbf{U}^T = \mathbf{W}_{ba}, \qquad \mathbf{U}^T\mathbf{U} = \mathbf{V}^T\mathbf{V} = \mathbf{1} \tag{4}$$
7. Compute the unknown rotation matrix $\mathbf{C}_{ba}$
$$\mathbf{C}_{ba} = \mathbf{V}\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \det\mathbf{U}\det\mathbf{V} \end{bmatrix}\mathbf{U}^T, \qquad \mathbf{r}_a^{ba} = -\mathbf{C}_{ba}^T\mathbf{p}_b + \mathbf{p}_a \tag{5}$$
8. Compute the pose $\mathbf{T}_{ba}$
$$\mathbf{T}_{ba} = \begin{bmatrix} \mathbf{C}_{ba} & \mathbf{r}_b^{ab} \\ \mathbf{0}^T & 1 \end{bmatrix} \tag{6}$$

3 Deliverable and Grading

Your deliverable for this lab includes (1) a short report of **maximum four pages** documenting your results, and (2) the source code that your team wrote or modified. You are encouraged to write code in a modular way.

This lab is worth 15%. Please submit the following in a .zip format, to Quercus, latest by 11:59PM April 3rd 2025 :

- **Code** (8%):
  - RANSAC algorithm, (3%)
  - A video with the inliers visualized on the right image plane, (1%)
  - Pose estimation. (4%)

Please comment your code appropriately and provide a `readme.txt` with proper explanation of the code structure.

- **PDF report** (7%) containing the following:
  - Overview of the algorithms for RANSAC outlier rejection and point cloud alignment, including the equations used, (3%)
  - Summary of estimated results comparing against the ground truth, (3%)
  - Comments on the limitation of visual odometry and how we can improve the estimation performance without an external localization system like Vicon system or GPS. (1%)