Individual Project: Build a SLAM Application

SLAM (simultaneous localization and mapping) is a method often used for autonomous vehicles that lets you build a map and localize your vehicle in that map at the same time. SLAM algorithms allow the vehicle to map out unknown environments. Engineers use the map information to carry out tasks such as path planning and obstacle avoidance.

Two of the often seen SLAM methods are visual SLAM (or vSLAM) and lidar SLAM. For this project, you will be working on visual SLAM. Visual SLAM can be implemented at low cost with relatively inexpensive cameras. In addition, since cameras provide a large volume of information, they can be used to detect landmarks (previously measured positions). Landmark detection can also be combined with graph-based optimization, achieving flexibility in SLAM implementation.

Deliverables for this project is a GitHub Repo that include:

1. A project report: summarize your work and results.
2. Dataset, all the codes, as well as all the results files.
3. Other typical GitHub Repo components.
4. Anything else you want me to know.

Example: Monocular Visual Simultaneous Localization and Mapping

Link to project example: https://www.mathworks.com/help/vision/ug/monocular-visual-simultaneous-localization-and-mapping.html

This example shows how to process image data from a monocular camera to build a map of an indoor environment and estimate the trajectory of the camera. The example uses ORB-SLAM, which is a feature-based vSLAM algorithm.

There are two steps to this project:

1. Use the dataset provided by TUM RGB-D benchmark (link can be found in the webpage above) and their camera intrinsic parameters. The dataset is taken within an office, the scene and expect results can be found in Figure 1.

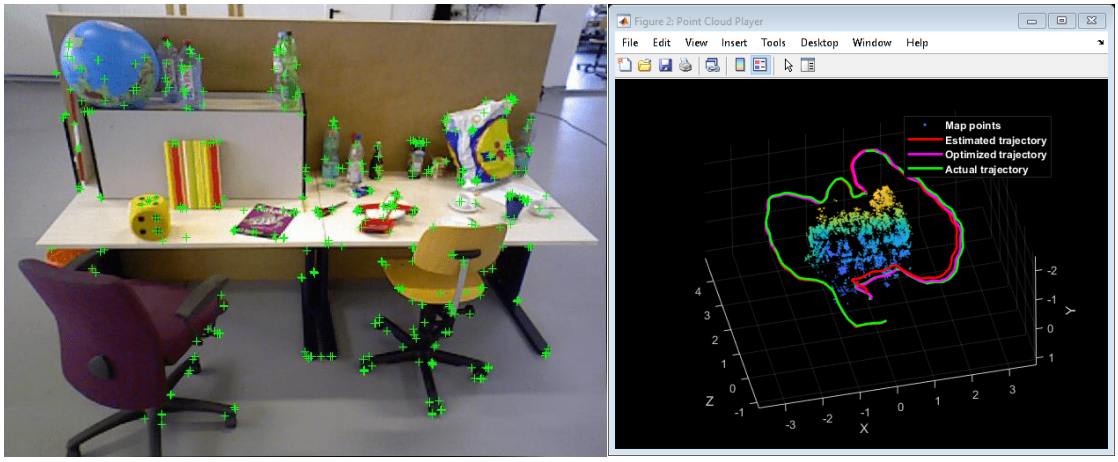

Figure 1, left: office scene with key points, right: reconstruct scene, optimized camera trajectory and ground truth

Deliverables for step one:

- Read the webpage of the example and summarize the main steps of how their Visual-SLAM is implemented
- Run the example code and include the results in your report
- Discuss how would you implement a visual-SLAM application yourself

2. Create a dataset yourself about one scene of your choice, it could be the classroom, your bedroom or anywhere you like (provide a picture of you in the scene separately to prove that this scene data is acquired by yourself). Use the sample provided, reconstruct the scene from image, predict and camera trajectory and compare it with the ground truth (if you can acquire that). If you decide to do this, you need to know the camera parameters, there are MATLAB toolboxes and example codes you can use to measure the camera parameters.

Alternative for step 2: If you prefer not to use this MATLAB example, **you can use any resources online (no restriction on algorithms or programing languages)**. As long as it is a Monocular Visual-SLAM and you can produce a reconstructed 3D scene and a camera trajectory.

Final remark (stating the obvious): this is an individual assignment; you should be working on this independently. Cheating (which include sharing codes) will result in 0 for both party, as well as an academic misconduct report to SIUE.

Appendix I: Grading Rubrics for Visual SLAM GitHub

Quality of the 3D reconstruction from Step 2	40 pts
Step 1 is finished, and the discussion is convincing	10 pts
Well written “summary reports” are included	10 pts
Include all the typical GitHub content, such as description, Readme, Tags, License, and other things you find necessary.	5 pts
The GitHub repo is well organized, the organizations methods fits the nature of the work and it is easy to navigate and find the information	5 pts
Use appropriate methods to display your content (codes, images, data, summary reports) and the content is easy to read. For example, MS word is not readable on GitHub website, you should avoid that!	5 pts
Overall, the GitHub Repo is professional looking	5 pts