

MTRN4010 – Project #2

Basic Data Fusion

In this task we commence using (and processing) data from multiple sensors, simultaneously. We do it in a simple way: deterministically. These are preliminary steps, which are useful for solving the localization problem through stochastic Sensor Data Fusion, which will be applied in subsequent project 3.

This task involves processing multiple sensing capabilities: LIDAR, wheel encoder and gyroscope (yaw rate, from IMU).

All the modules which are implemented in this task will be used in a subsequent project. Properly solving this task does not only give you a good mark in this task, but also facilitates the solution of a subsequent project.

One of the purposes of this task is for understanding and implementing localization by applying “dead-reckoning”. The second purpose is getting used to using different coordinate frames and for implementing a Data Association process. All of them are necessary for the subsequent project.

Part A.

Implement a function for estimating the attitude of the platform, based on the measurements of the IMU’s gyroscopes. You may assume that the platform is always operating in a 2D context; consequently, you can assume that the pitch and roll are always =0; which allows processing the integration by simply integrating the yaw angular rate ω_z . The measurements, to be processed in this task, are provided in the file “IMU_dataC.mat”.

Verification of results: For verifying the performance of your implementation, you will use the data provided in the file “IMU_dataC.mat”. You will integrate the yaw gyroscope’s measurements, and plot the estimated attitude, for the full duration of the test (whose duration was about 250 seconds). You will compare it (by simply inspection) with a solution provided by the lecturer, and, also, by verifying that the initial and final conditions “make sense”, by visually inspecting the laser scanner images (which were taken from the vehicle, during the same test). The laser scanner data is provided in the file “Laser__2C.mat”.

Relevance of this part: 22% (of the full mark of this project)

Part B.

You are required to implement a “dead-reckoning” process, based on the kinematic model of the platform and the measurements provided by sensors (speed encoder and gyroscope). The kinematic model was explained in the document “[AAS_2019s1]_KinematicModels.pdf”. The way for implementing it is explained in that document, as well.

The inputs of the process model are the angular rate ω_z and the speed encoder measurements. The necessary data is contained in the Matlab data files “IMU_dataC.mat” and “speed_dataC.mat”. These files do also include useful comments (in addition to the necessary data and sample times) for explaining the data format.

Validation: You will compare the estimated path (by simply inspection) with a solution provided by the lecturer. Additional validation of these results will be done in Part C.

Relevance of this part: 26%

Part C

At each time, when a laser scanner measurement is available, you will perform the following processing:

- 1) Obtain an estimate of the position and orientation of the platform at that time (provided by part B).
- 2) Perform feature extraction, for detecting the OOIs (as implemented in Project1)
- 3) Express the currently detected OOIs' positions in a global coordinate frame (using the results obtained in (1) and (2))

We assume as “global coordinate frame” the one aligned with the platform at time t_0 . I.e. we say that the platform's position and heading at time t_0 are $(x=0 \text{ m}, y=0 \text{ m})$ and $(\text{heading}=90 \text{ degrees})$ respectively (based on the coordinate frame convention, shown in figure 1).

For proper processing, you must consider the position of the laser scanner (on the platform), as it is shown in figure 1. The displacement, d , is 46 cm (approximately), longitudinally.

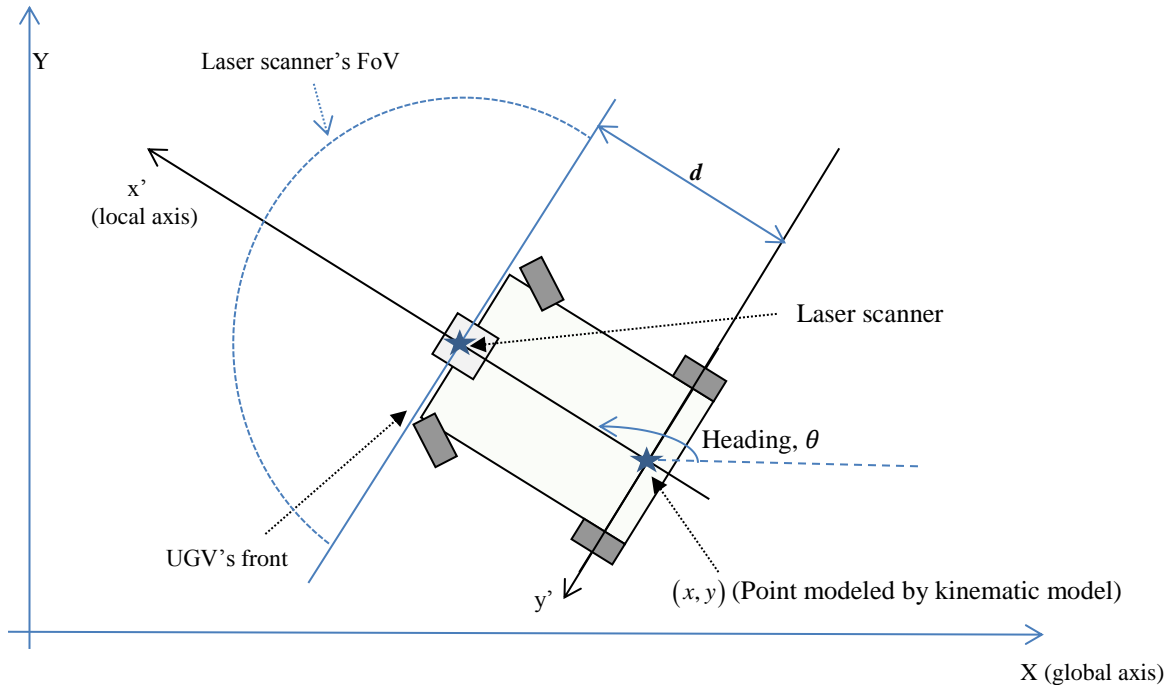


Figure 1

Expected result: When the currently detected OOIs are shown in the global coordinate frame, they would appear near the original OOIs, detected at the initial time. The discrepancy will be increasing as the platform travels, due to the cumulated error in the platform's pose estimates (an error which usually grows with the time).

Validation/Visualization of results: For showing the performance of this process, your program will plot (dynamically, as you have done in previous tasks), the OOIs detected at the initial time and the OOIs currently being detected. Both sets of OOIs will be shown in the global coordinate frame. Use different colors and/or symbols for each set of OOIs.

Relevance of this part: 26%

Part D

Implement a Data Association process.

Based on the previous parts of this task; implement a Data Association (DA) process. The DA concept is discussed in class (week 4).

The DA process would be able to identify (and keep identifying) the OOIs that are detected at each scan. The operation of the module is expected to be the following one:

1) At scan #0, the ranges are processed for inferring the OOIs which are present in the image. Those are given identities, i.e. a unique number for each of them, and their positions registered.

2) Each subsequent scan is also processed for extracting the OOIs occurring in it. The DA process will infer the identity of each currently detected OOI, based on the positions of the OOIs obtained in the first scan and on the estimated global positions of the currently detected OOIs. A tolerance of 40 cm will be considered for the matching process.

For visualizing the performance of the approach, you will use Part C, adding text nearby each currently detected OOI, for indicating its inferred ID.

Expected performance: In this case, considering that the localization process is simply based on the dead reckoning of the platform, we expect you to achieve a successful DA during, at least, the first 10 meters of the trip.

Relevance of this part: 26%

(note: this part can be demonstrated on week 6, or on week 7, or even jointly demonstrated with Project 3, on week 8)

Deadline: Demonstration of this task (except part D) will be on Week 6, during your lab session. In the demonstration, you will briefly show your solutions working. You may need to explain parts of your program, and answer questions, as part of the demonstration.

Quiz: A short quiz, about this task, may take place during your lab session (the day of the demonstration). It will be announced in week 5; consequently, you will know about it, in advance. If the quiz is announced to take place, you must attend the quiz of your session. If you are not able to attend your quiz, you will need to apply for SC.

Report: There is no report associated to this task.

Exam: Questions related to this task may be part of the final exam.

Explanation: An explanation about the project will be given during the lecture on week 4. The lecturer will show his solution to the problem, running in class. A video link, showing the program running, will be publicly available, in Youtube, searching for the keywords: “Project_02”, “Advanced Autonomous Systems” and “AAS2019.P2”. A similar video will be available through the lecture’s video recording.



Questions: Via Moodle Forum or email to lecturer (j.guivant@unsw.edu.au)
