ILR08

Name: Aman Agarwal
Team E: Wholesome Robotics
Teammates: Aaditya Saraiya, Hillel Hochsztein, Dung-han Lee, John Macdonald
ILR08
Date: October 10, 2019

Individual Progress

Map Builder Integration

The map representation used by the system is the start and end points of the rows of plants. Previously these points were collected by recording GPS data from the start to the approx. end of a plant row and then trimming the recorded rosbag to obtain the start and end GPS location of the row. The map builder was built to make this process easier so that we can continuously collect data during the visit and mark the start and end of row via User Input.

The user-input selected was via the joystick which is already available. The waypoint is collected when the user presses a particular button on the joystick. In order to avoid multiple waypoints from being collected at every button press, button debouncing was also added to the map builder.

GPS data was collected in UTM coordinates as they are quite accurate in the range that the robot will traverse per field. Further, this allowed us to transform the coordinates from UTM frame to a frame in the field (field_frame). The field_frame is defined as the frame with origin at the starting point of the first row and its x-axis pointing along the first row, further the z-axis is pointing upward thereby defining the frame. Set of helper functions developed by John were used for this transform. In order to use the same map for different tasks, the map of the field is saved in UTM coordinates. The pipeline for the map builder can be seen in the following flowchart.
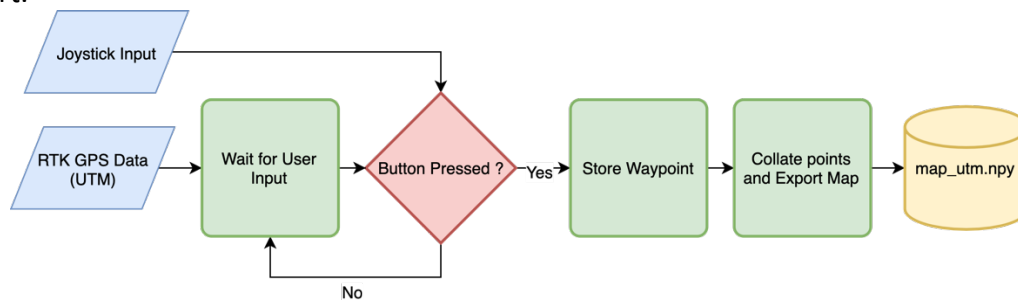

Figure 1 Map-Builder Data Flow

This map builder is now integrated into the remaining navigation stack. Previously, since we were not using GPS, the map file was a set of points in the field frame which were generated after manually measuring the average row width and average row length. Thus, this model failed if there was a high variability in the row length and width. Now with the GPS waypoints being collected we can account for this variability and generate a more feasible coverage plan.

Every-time the navigation package is started, it loads the map in UTM coordinates and transforms it into the field_frame. Then this transformed map representation is used by the coverage planner to build a global path for the robot, which in turn is used by the controller to ensure this generated trajectory is followed with acceptable accuracy. The controller uses the current location in field_frame to decide the velocity commanded to the controllers. The global path generated is also visible for visualization.The entire pipeline is presented in the following flow-chart.
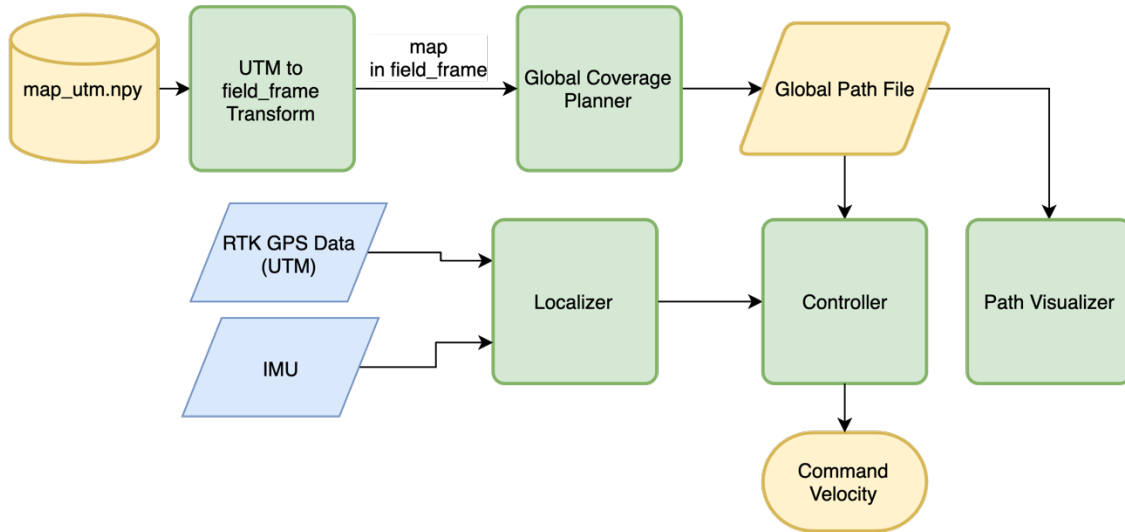
Figure 2 Navigation sub-system data flow

Challenges

Mismatch in map representation

The helper functions developed by John assumed a slightly different representation of the map file due to an error in communication. This led the transform functions to output just one coordinate. Identifying this error took significant time. However, finally the error was corrected, and the entire pipeline was tested.

Teamwork

Aaditya:

Depth estimation using semi-global block matching and integration for visualizer.

Dung-han Lee:

Integration for visualization and attempted UNet for end-end binary classification

Hillel:

Implemented Clustering of data and made a preliminary version of the plant guards

John:

Evaluated use of row-detection for localization and added motion constraint on the GPS data.

Future Plans

Team:

       Complete Monitoring MVP

       Complete Navigation MVP

Individual:

- Fuse data from IMU, RTK GPS and Visual Odometry