Autonomous Golf Cart

John Garay
Cedarville University, johndgaray@cedarville.edu

Andrew Ross
Cedarville University, andrewdross@cedarville.edu

Luke Schwan
Cedarville University, lukerschwan@cedarville.edu

Ian Steptoe
Cedarville University, ianbsteptoe@cedarville.edu

Danielle Fredette
Cedarville University, daniellefredette@cedarville.edu

Obstacle Avoidance for Autonomous Golf Cart

John Garay, Andrew Ross, Luke Schwan, Ian Steptoe, Danielle Fredette
School of Engineering and Computer Science, Cedarville University, Cedarville Ohio

Abstract

The Cedarville University autonomous golf cart senior design team has built on the success of previous teams to add functionality and robustness towards the goal of a fully autonomous golf cart for campus use. This poster outlines this year's accomplishments, centering around obstacle avoidance and improvements in sensing and decision-making algorithms.

OBJECTIVES

1. Develop, test, and validate a robust method of object detection using LIDAR.
2. Develop, test, and validate decision logic to determine if an obstacle warrants car action.

ROS Structure

- In order to facilitate the number of concurrently running processes necessary for the car to operate in real time, we implemented the code to run in the Robot Operating System (ROS).
- ROS facilitates the message passing between discrete nodes.
- The nodes run independently of each other and will pass data to one another using a publisher-subscriber model shown with arrows in Figure 1.
- As an example, one can see the lidar_publisher node publishes the LIDAR data to the lidar_processing node.

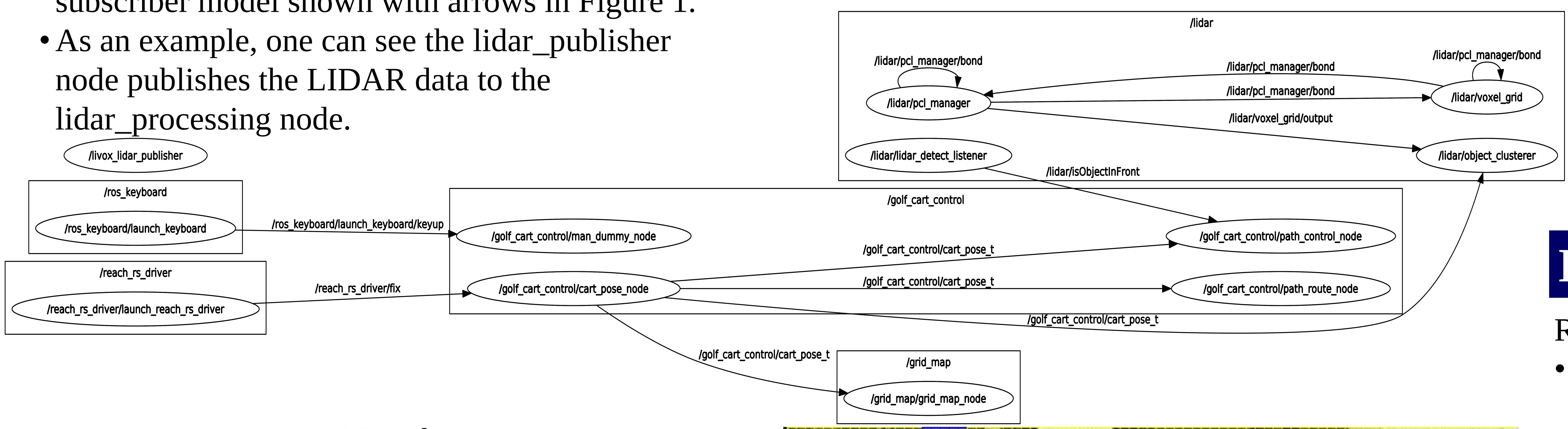

Figure 1: ROS Node Diagram

Grid Mapping

- The Grid-map is a data structure to synthesize all incoming data.
- If an obstacle is on a valid path that the cart needs to drive on, the car stops.

Clustering

- In order to segment the image, we tested a wide variety of clustering algorithms to go from non-ground point cloud data to groups of points that represent discrete objects as seen in the images.
- We selected the DBScan clustering algorithm to cluster the points. DBScan is both efficient and will propose cluster quantity.
- Other clustering algorithms require the programmer to specify the number of objects, something that is problematic when one does not know the number of objects that exist.

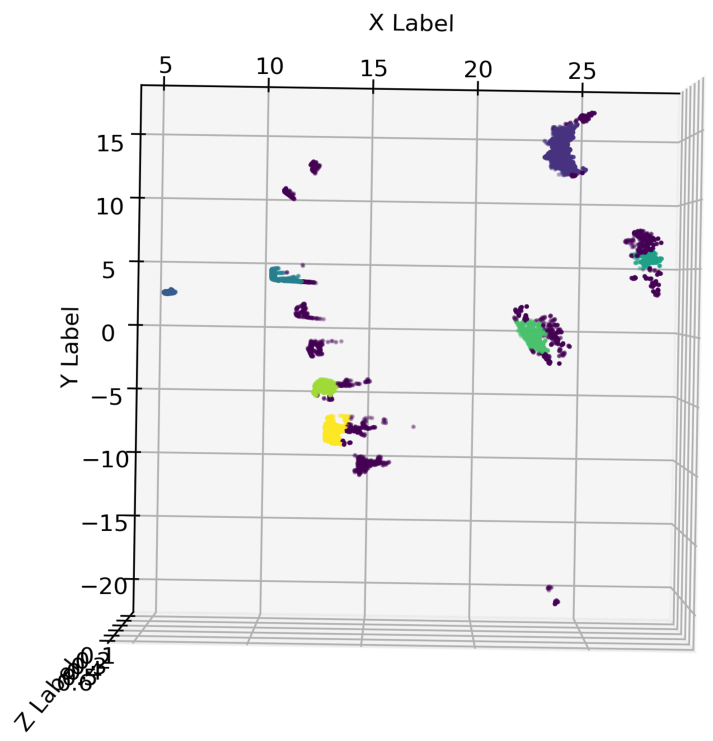

Figure 2: DBScan Clustered Obstacles from Above

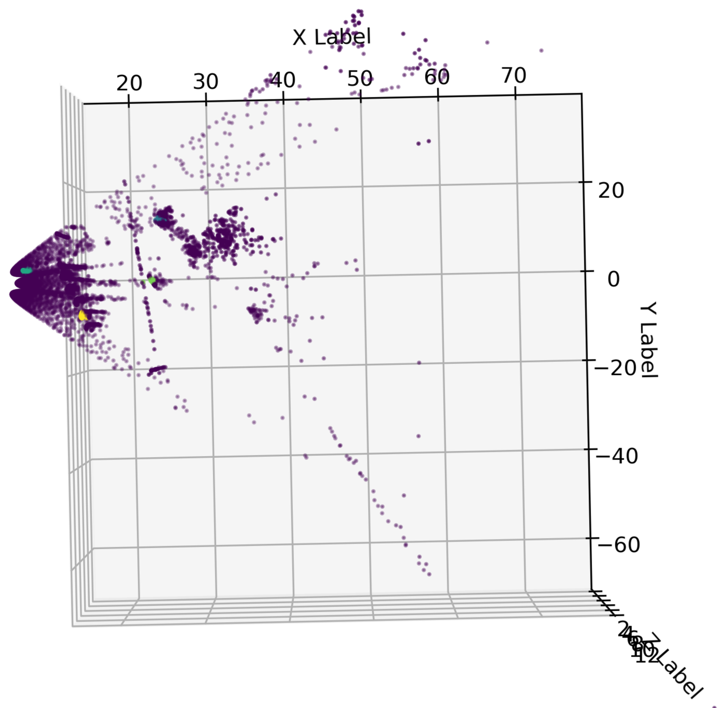

Figure 3: Pre-clustering and Filtering LIDAR Data

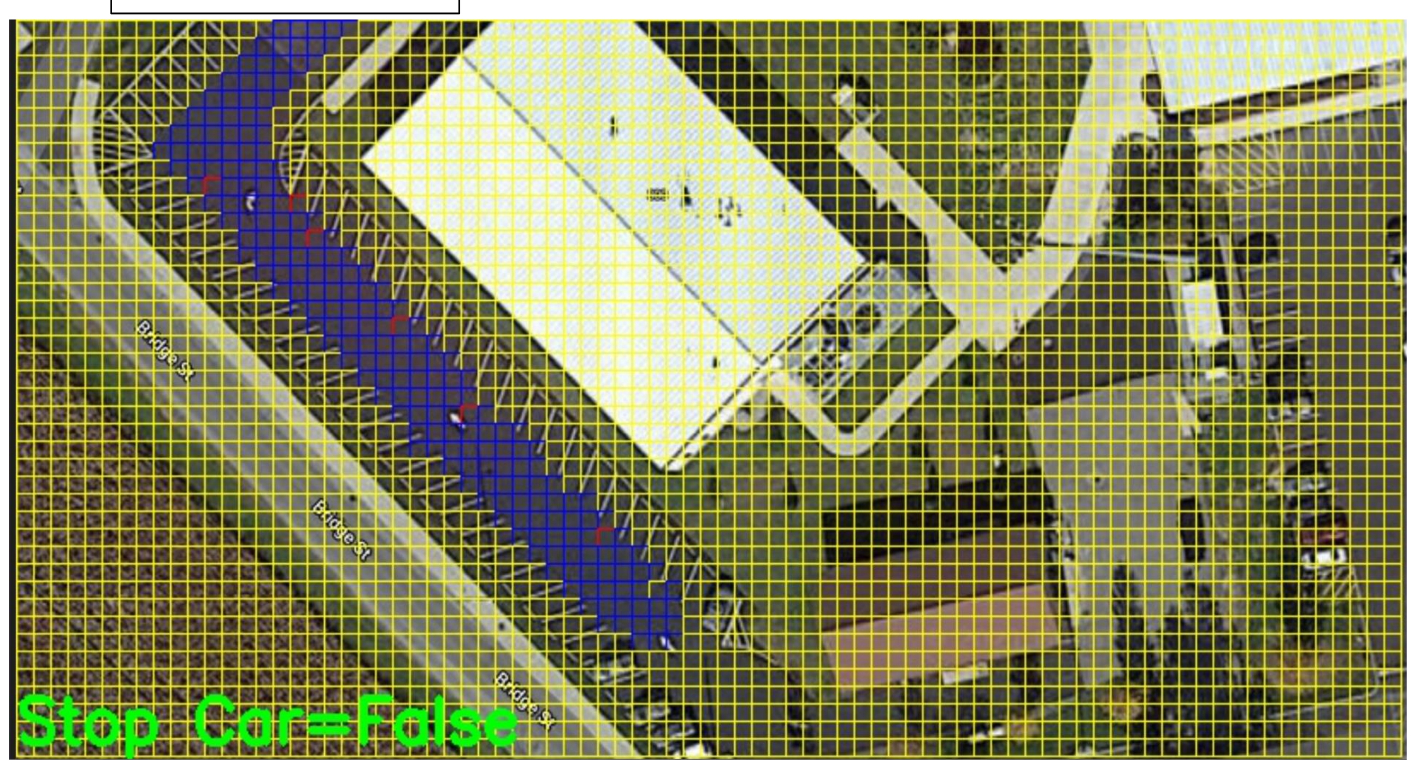

Figure 4: A graphical representation of the grid map

Obstacle Detection

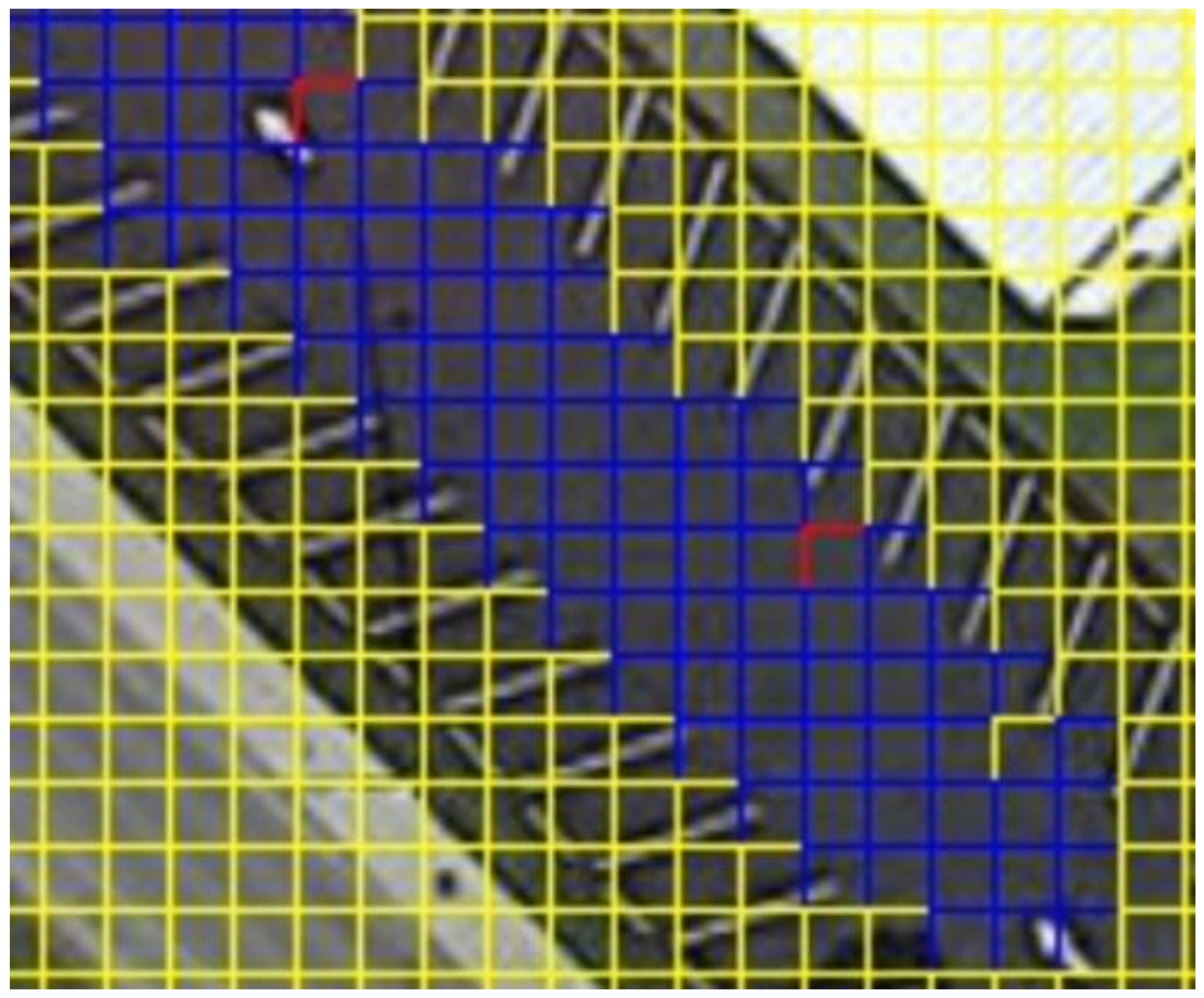

Figure 5: An Obstacle Plotted on The Grid Map (In Red)

Obstacle detection combines the work on both clustering and ground plane detection. It includes the following steps:

1. The ground plane detection node takes the points received from the LIDAR and filters out the ground plane.
2. The points that are determined to be not-ground are clustered together into groups (As seen in section: Clustering). These groups represent discrete objects.
3. The closest point from the object to the car is then taken and the obstacle is plotted on the grid map.
4. The grid map then is utilized to determine if the obstacle merits stopping.

DISCUSSION and Future Work

Results

- The cell map strategy allows us to effectively synthesize information in a way that renders navigation decisions trivial

Future work

- Find ways to increase robustness in electrical and mechanical systems as well as algorithms
- Refine and integrate ICP into the pose estimation
- Dynamic path rerouting
- Estimate the probability of an object moving onto the path
- Dynamic Object Avoidance
- Integrate a Kalman filter for pose estimation