

Objective: In this project, our objective was to use Hough transform on the image captured from the Duckiebot camera and draw yellow and white line edges for the lane lines and compare it with the default edges with lines.

Implementation details:

A local ROS package containing the image processing node which filters the yellow and white lines from the lane and draw edges for these lines by using the Hough transform. The node subscribe to the camera image from indefinite navigation and publishes to images with yellow lines and image with white lines respectively. Two launch files were written to launch the local ROS package along with the indefinite navigation launch package. I have a local launch file which includes the local ROS package and the custom image processing node and a master launch file which do remapping and including the local ROS package and launch file.

first I tested the Opencv color filtering and Hough transformation in the lab with some random yellow and red colored objects. After that, the node was tested in the duckietown. The Hogh transform lane filter worked fine but the image streaming latency was too high $>1s$ so it can not operate in real time. However is scene changed and image updated the rqt_image view of the transformed image also updated accordingly.

Results:

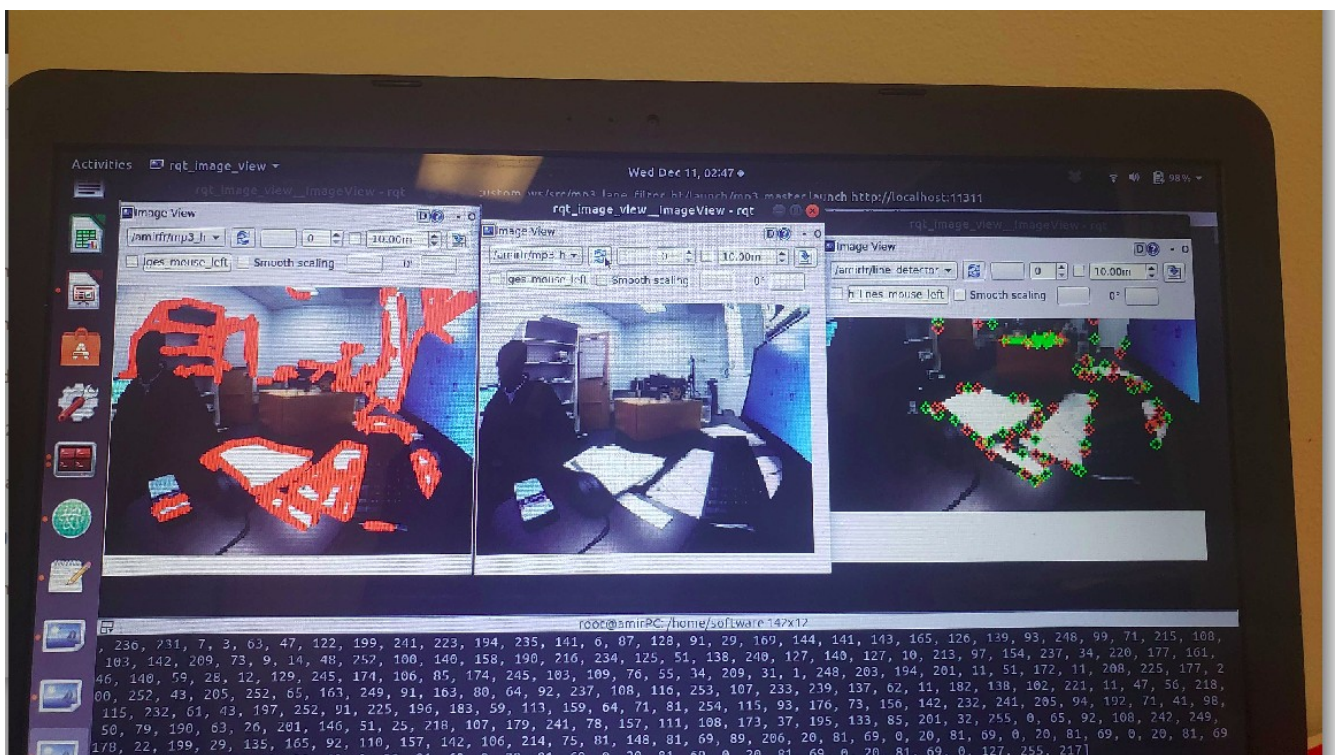


Illustration 1: Testing with random white and yellow objects

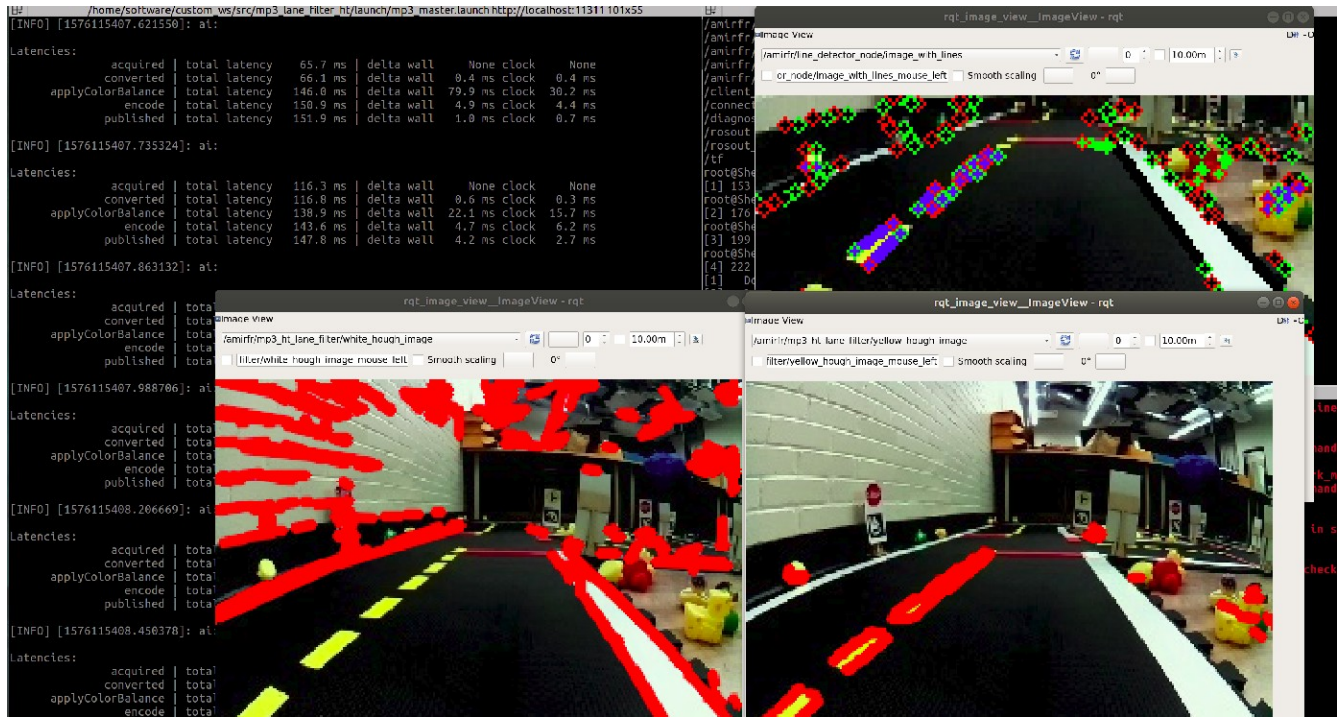


Illustration 2: Final results during an indefinite navigation in the duckietown

Conclusion:

The lane filtering algorithm using Opencv and Hough transform looks a powerful method for this purpose. However, we need to have better processor or improved algorithm to reduce latency and facilitates operation in real time.