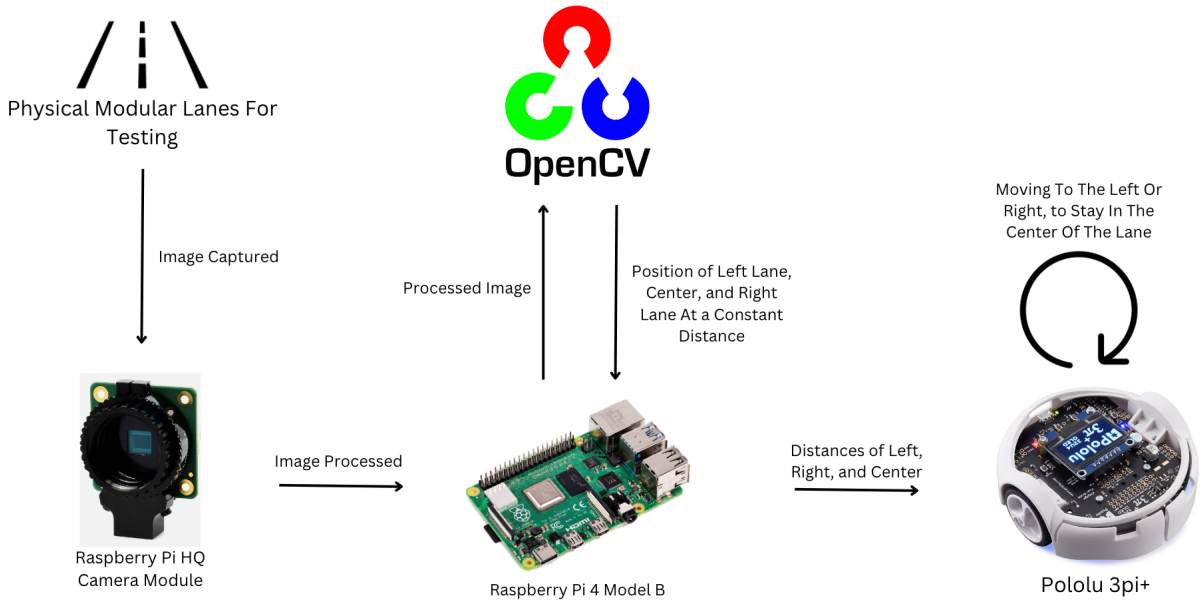
Milestone 1: Pololu Lane Keeping

Group 5: Adrian Botvinik, Megan Joseph, Jesus Medina

EECS 149/249A Final Project Fall 2023

Architecture Drawing



- We will use the camera and Opencv to get image data which will be sent to the Raspberry Pi 4. On it, we will do image processing to send the lane positions to the Pololu. Based on the information given, the robot will either stay straight, turn, or make a lane change.

Progress

- CAD of Raspberry Pi to mount to the Pololu
    - On Github Here
- Determined Algorithm for Lane Detection and Lane Keeping
    - Use Pixel Position of Left and Right Lanes to determine if lane is straight or turning Left or Right, Control Robot based on this Data
    - When Distances Between the Lanes shrinks below a threshold, Initiate a Lane Change Using an Interrupt
- Ordered hardware
- Designed Tracks for demos and testing
- Architecture Drawing and Prototype Modal Model

Modifications to Goals/Scope

Based on the feedback we received, we decided to add elements to our project. We will be incorporating lane changes in the form of merges. Our testing road will also incorporate turns so that we are not only testing if it'll stay in a straight line.

List of Resources

- Hardware
    - Raspberry Pi 12MP Camera
    - Raspberry Pi 4
    - Arducam Ultra Wide Angle Lens
        - We initially went with a telephoto lens but decided a wide angle lens would be a better fit for our goals.
- Starter code
    - https://www.instructables.com/Autonomous-Lane-Keeping-Car-Using-Raspberry-Pi-and/
    - https://medium.com/@mrhwick/simple-lane-detection-with-opencv-bfeb6ae54ec0

Schedule and Member Assignments

Date	Deliverables
11/8/2023 - 11/20/2023	• Have lane tracking down • Connect Raspberry Pi and Pololu ○ Pololu can take some input and move based on that
11/?/2023	Milestone 2
11/?/2023 - 12/13/2023	• Have lane changes done • Fine Tuning (testing on edge cases), making sure everything is connected and works • Make poster, work on report
12/13/2023	Poster and demo

Major Risks

A concern we have is how fast image processing can be done on the Raspberry Pi. One solution we came up with in regards to this is to have the Pololu move at a slower pace, in the event that image processing is slower than we anticipate.

Github

https://github.com/abotvinik/Pololu-Lane-Keeping