Autonomous Driving Robot

SAKTHI RAMANTHAN (A53070191)

DEBJIT ROY (A53094436)

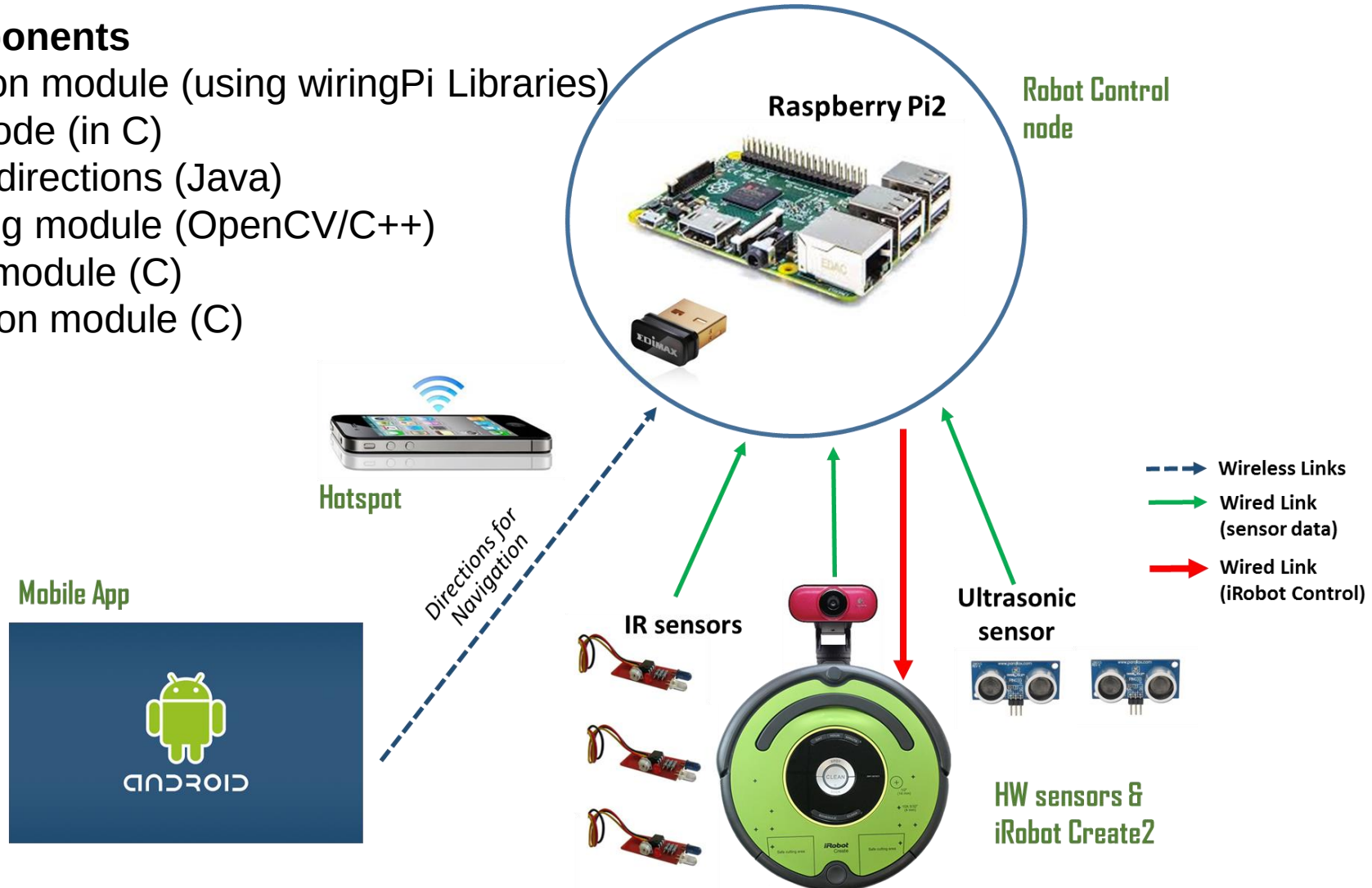
Motivation & Related Work

- **Motivation**: to analyze the challenges of self driving cars and build a low complexity model to understand how to overcome them in real world.
- **Related Work**: Google's converted Toyota Prius is a fully autonomous car that uses cameras to detect traffic lights, and radars to detect pedestrians. And they have successfully completed over 300,000 miles of accident-free autonomous driving.
- **Objectives**:
  - Self-driving on the track (auto-turning, lane changing)
  - Traffic Lights detection *{** stretch goal completed}*
  - Collision Avoidance

HW & SW components

Software Components

Sensor interaction module (using wiringPi Libraries)
Server Socket code (in C)
Android App for directions (Java)
Image processing module (OpenCV/C++)
Lane Following module (C)
Obstacle detection module (C)

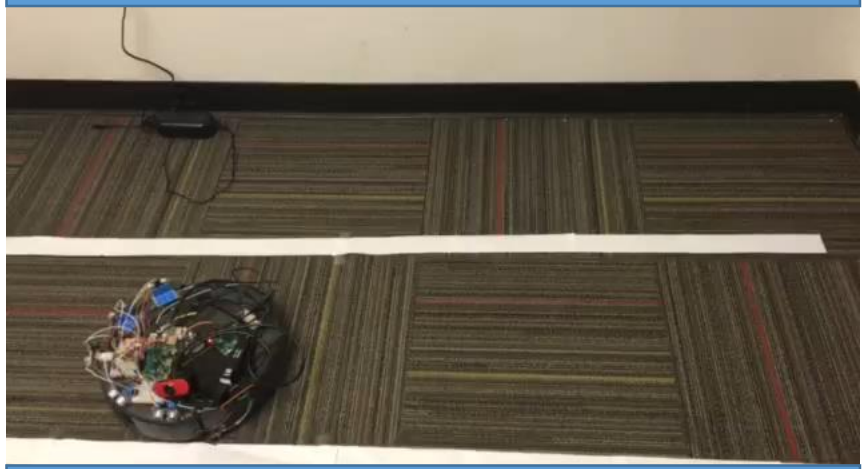


Issues Faced

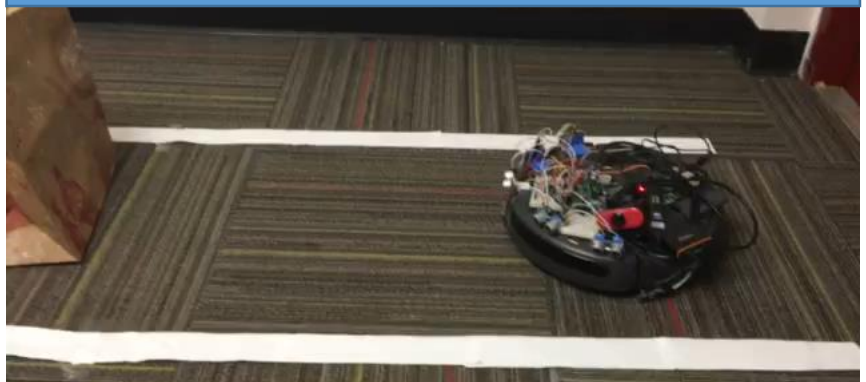
1. Ultrasound Sensors:
    - HC SR04 sensors operate at 5V. But RaspberryPi GPIO support only 3.3V, so we had to use a voltage converter (74lvc245) to interface the sensors' echo pins.
    - Axis of obstacle had to be perpendicular to sensor for correct operation, so we had faced unreliability during obstacle avoidance scenarios.
2. Left Bias of Robot:
    - During testing we found that the iRobot had an unpredictable left bias. We were unable to fix this problem using the encoder values obtained from the bot and so had to add heuristics to our code to accommodate for this.
    - We had faced serial communication problems when continuous stream of data was sent. So we fixed using delays.
3. Image processing:
    - The traffic lights detection was done by using thresholding over HSV values corresponding to the colors. So due to lighting condition variations, we faced issues in getting reliable detection working.
    - To avoid overhead on the remaining system, we used another raspberry pi for image processing.
4. Issues with Odroid XU-4 & ROS:
    - We had difficulties in finding FTDI drivers for Odroid. So we had to use Raspberry Pi2 for the processing/control node.
    - ROS was an unnecessary overhead for such a system with low complexity.

Highlights

Traffic Signal Detection



Lane changing



Obstacle Avoidance

Obstacle Avoidance