RUTGERS
WINLAB | Wireless Information
Network Laboratory

Active AI Driver Assistant

By: Atharva Belsare, Lucas III Cordova, Kishan Patel, and Christopher Sawicki

Meet The Team

Lucas Cordova III
School of Engineering
Team Leader

Kishan Patel
Rutgers Business School +
Computer Science

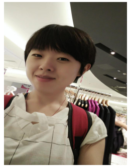
Atharva Belsare
School of Engineering

Chris Sawicki
School of Engineering

ADVISOR

Dr. Jorge Ortiz

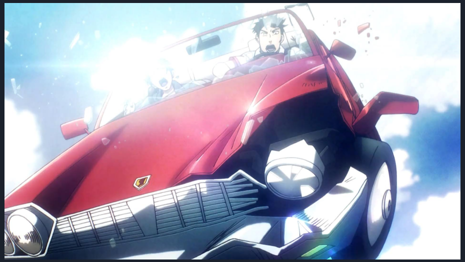
RESEARCH MENTOR



Sonya Yuan Sun

Introduction 👨‍💼

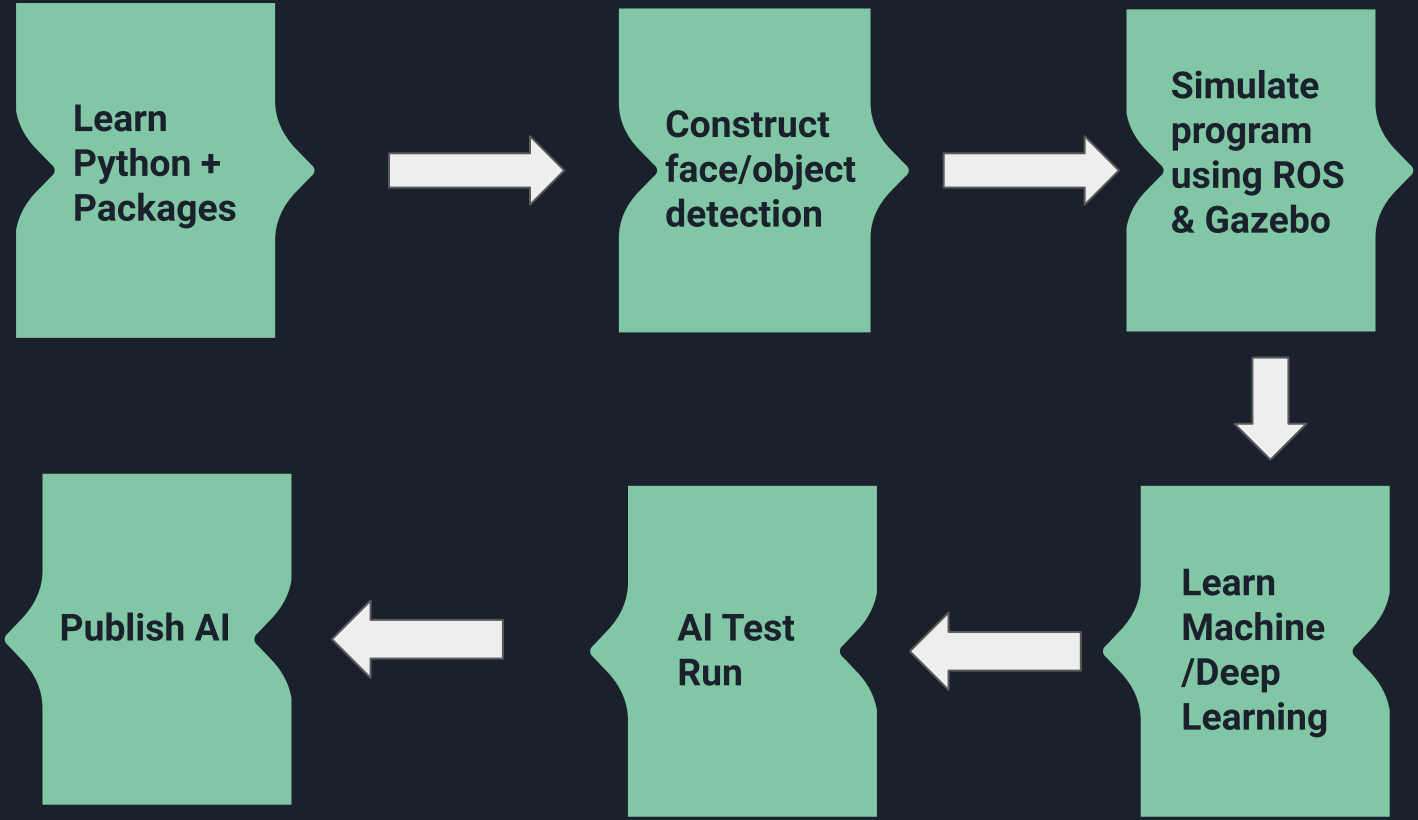
Nowadays, traffic accidents are mainly caused by human operational mistakes such as inattention, misbehavior or distractions. Hence, driver assistance was created to assist drivers for operation in order to lessen traffic incidents and promote safe driving.



Objectives

For this project, a naturalistic driving monitoring and intervention system will be develop. It will focus in assessing driving condition and performance that will draw the driver’s attention using a voice-based interface (AI) to maximize driving safety and quality.

Project Approach



Learn Python + Packages

Construct face/object detection

Simulate program using ROS & Gazebo

Learn Machine /Deep Learning

AI Test Run

Publish AI

Python Packages

Open CV - an open- source library that provides the machine to extract information and recognize face or object with a camera.

NumPy - a package library that supports large, multi-dimensional arrays and matrices, along with a large collection of high level mathematical functions that operates machine.

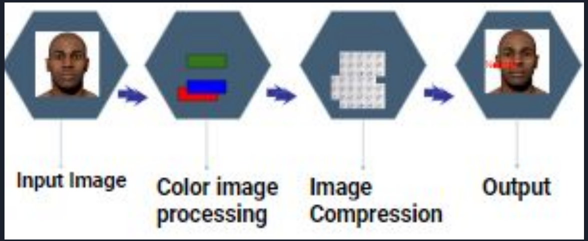
Pandas - Software library for data manipulation and analysis

Keras - Open-source software library that provides python interface for artificial neural networks

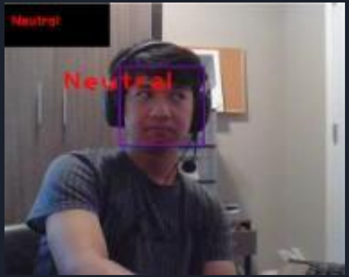
Face Detection

Utilizing the python packages and OpenCV we were able to create a face and emotion detection program. We installed JPEGS files using Keras Image Processing in order to perform operations on an image and enhanced itself to extract formation.

Overview of Face Detection



Results

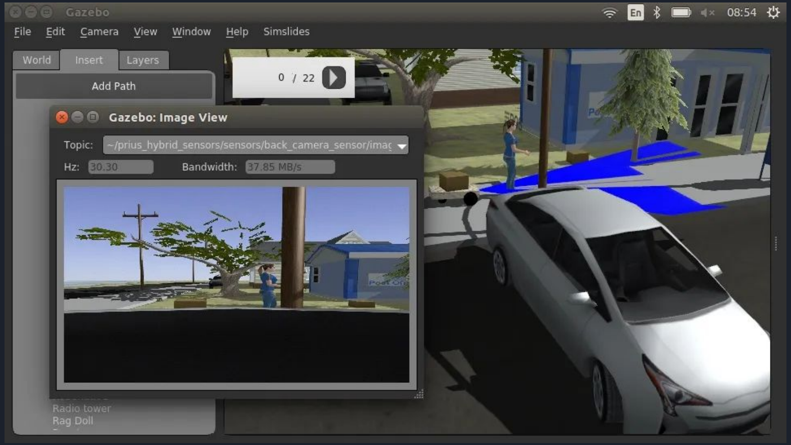


Road Sign Detection

Packages such as Numpy and OpenCV are useful in reading hundreds of images to detect stop signs and enhance image processing and object and facial recognition

ROS & Gazebo Simulation

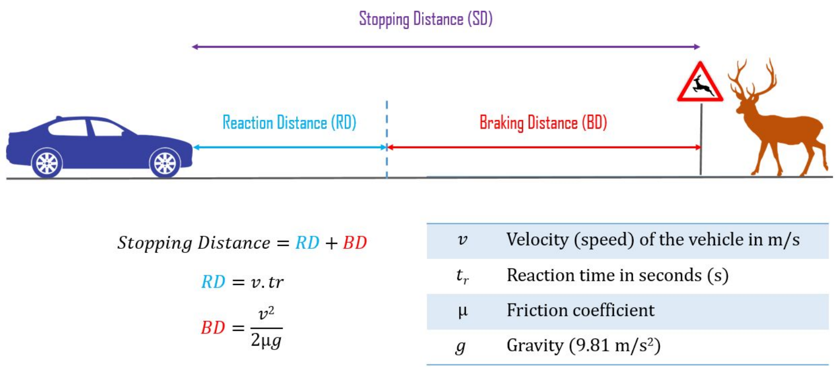
Work with ROS and Python to implement a program to estimate the total stopping distance of a vehicle based on its speed.



Topic: ~/prius_hybrid_sensors/sensors/back_camera_sensor/imag

Hz: 30.30

Bandwidth: 37.85 MB/s



$$Stopping\ Distance = RD + BD$$

$$RD = v.tr$$

$$BD = \frac{v^2}{2\mu g}$$

v	Velocity (speed) of the vehicle in m/s
t_r	Reaction time in seconds (s)
μ	Friction coefficient
g	Gravity (9.81 m/s^2)

Results

❏ A emotion detection program was created that can successfully read the emotion of a person, although in testing is was inaccurate due to low frame rate on the camera.

❏ A road sign detection program was created using object detection on a forward facing camera. This program can also calculate the distance from a road sign using a distance derivation.

❏ Using ROS and Gazebo, these two programs were tested and operated in a virtual car.

Conclusion

❑ Given face detection, ROS, Gazebo, and the stop sign detection code we combine these systems to deliver a product that can read facial emotion to give the user an alert on stopping earlier or later from a stop sign.

❑ Our progress thus far can monitor and detect everything needed for AI to actively assist a driver. To complete the AI we must incorporate the detection code into the AI to respond to any detected misbehavior.

❑ The AI has been tested and used in Gazebo, and the last step to complete our active AI driver assistant is to test hardware to make this project functional in a real car is it had worked in Gazebo

Any Questions?