Portfolio: Brief overview of relevant work

Contents

Path Planning and Patrolling for a team of Car-like Robots in a Campus Environment (YouTube link)

Technical skills: Python, Robot operating system (ROS), Webots Simulator, Matlab

For self-driving cars to traverse roads, it requires a robust path planning system. The planning system is responsible for high-level path planning such as route planning (which roads/ route to take to reach from point A to point B), path generation (generating a smooth curve to avoid abrupt acceleration/ deceleration and jerky behavior of car), and low-level path planning such as behavior-selection (selecting behavior such as lane-keeping, intersection handling, traffic light handling), motion planning (generating the reference path along with reference speeds) and obstacle avoidance (reactive avoiding obstacles in its path).

During my work at Indian Institute of Technology, Bombay, I have worked towards designing and developing several modules of a reactive path planning system for autonomous mobile vehicles, allowing the vehicles to navigate predefined areas from the world map. Figure 1 displays multiple vehicles traversing roads in our simulated environment.

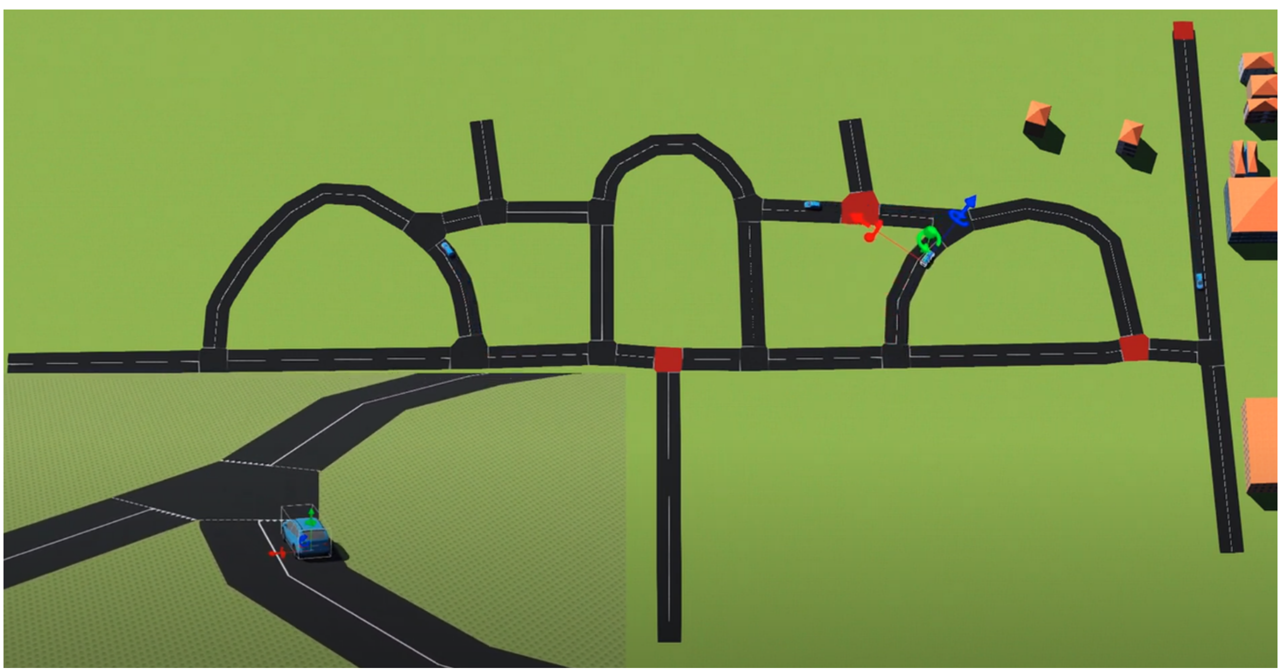
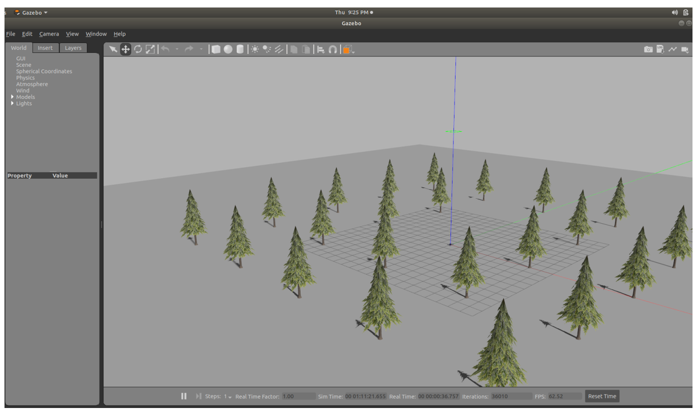


Figure 1: Cars traversing roads in an area selected in world map

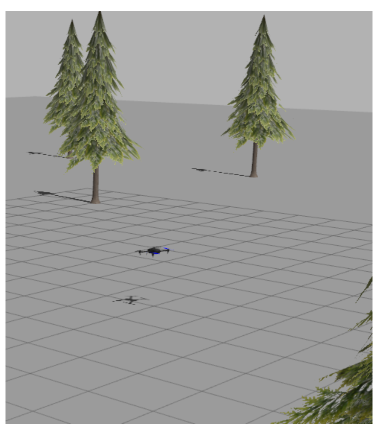
Development of motion planning algorithms for quadcopters for precision agriculture
(YouTube link)

Technical skills: Python, Robot operating system (ROS), Gazebo Simulator, PX4 autopilot, Matlab

After the completion of cars projects, I continued working under the guidance of Prof Arpita Sinha at IIT, Bombay; for developing coverage path planning for quadcopters in precision agriculture.
So far we have simulated a farm in gazebo environment and implemented route planning for IRIS quadcopter using PX4 autopilot module and ROS.



(a) simulated farm



(b) IRIS quadcopter navigating the farm

Figure 2: Implementation of route planning for an IRIS quadcopter in simulated farm enviornment

Object detection and tracking (YouTube link)

Technical skills: C++, OpenCV, Python, Tensorflow, PyTorch

At AitoeLabs, I worked with a team of engineers to build efficient video analytic solutions for internal security. To ensure ATM security we developed deep learning models and algorithmic codes on the live video stream.

I was affiliated with the object detection and tracking group. Our implemented human detection algorithm's performance is displayed in 3.

Figure 3: Detecting multiple objects (people) in live stream

Object tracking deals with tracking detected objects by assigning IDs to them. A theft or vandalizing of an ATM generally happens when there is a group of people (more than the allowed number) in the ATM or if a person is loitering around inside the ATM for a prolonged duration. Using object tracking methodology we developed algorithms to detect this kind of behavior and on detection alerted the security personnel/ company.

Programming robotic manipulator for pick and place task
(YouTube link)

Technical skills: Python, ROS, Gazebo Simulator

At my job at Dhristi works, we aimed to developed garbage bots for beach cleaning. My focus was on developing algorithms for a robotic arm for pick and place tasks. The application would be used to identify and pick garbage on the beach and place it on the bag/bin on top of a mobile rover (which was parallelly being developed by other team members).

I used ROS to set up an interface between the actual robot(purchased 5DOF robotic arm) and gazebo simulator. Thus controlling the robot both in the simulated and real environment. Figure below displays the robot being controlled by ROS in both real and simulated environments.

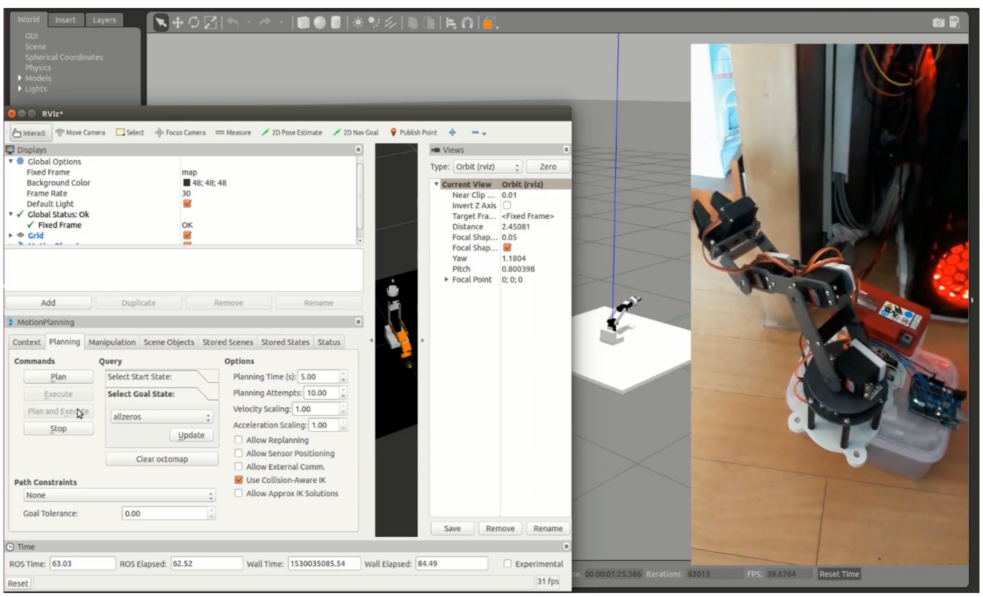


Figure 4: Robot mimicking position via commands from ROS

To grasp household objects, I compared and analyzed several available grasping techniques. And interfaced the selected technique with our simulated robot.
Figure 5 displays the scene creation in simulated environment consisting of robot and household objects.

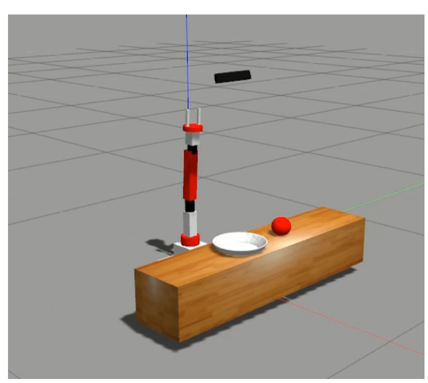


Figure 5: Robotic arm grasping household objects

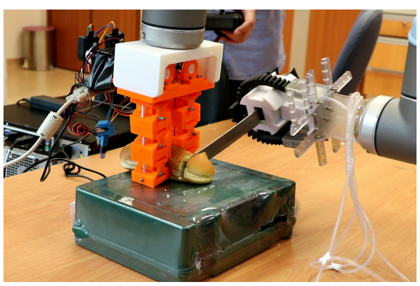
Overview of my work at Singapore Institute of Neurotechnology (SINAPSE), National University of Singapore

During my tenure as a Research assistant at SINAPSE laboratory, I have worked on

- Development of motion planning algorithms for robots,
- Slip control experimental setup design, and
- Tactile and haptic glove interface.

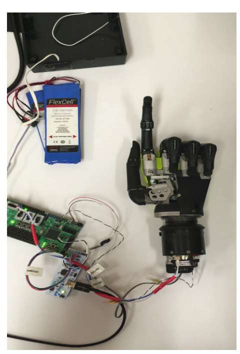
4.1 Development of motion planning algorithms for robots (YouTube link)

Technical Skills: C++



(a) Bread cutting task

(b) Corkscrew unlocking task



(c) Programmatic controlling of a robotic hand

Figure 6: Motion planning and control of robots

This part of my work was affiliated with the office of naval research Singapore (abbreviated ONR). Here I worked with a group of research scientists to make robots perform day to day tasks such as bread cutting and opening the corkscrew for wine bottle. My work consisted of motion planning and control of the robots. I programmed the robot to perform zig-zag, spiral motion as well as build an interface to control the digits of a robotic hand.

4.2 Slip control experimental setup design (YouTube link)

Technical Skills: Matlab, Simulink, Python

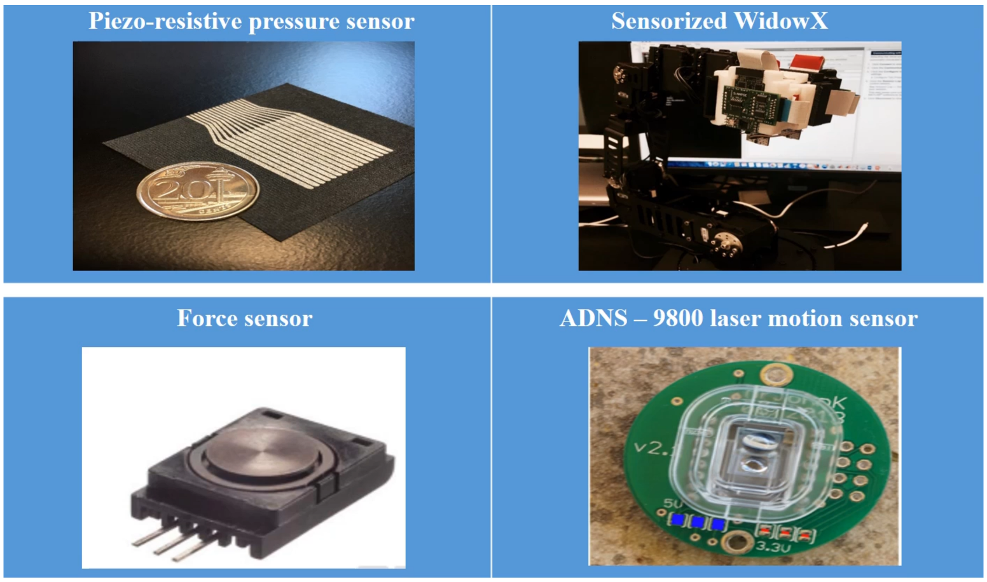


Figure 7: Slip control experimental setup

To study the occurrence of slippage when mass is added to an object gripped by a robotic end-effector, I designed an experimental setup to recreate slip conditions and test friction models developed by the post-doctoral fellow.

Slip conditions can be interpreted by normal and tangential forces of the object on the end- effector. I worked on developing a high density tactile sensor array to fit the surface area of WidowX's (purchased robot) parallel gripper, as well as built circuitry and designed algorithms to read tactile, force and optical sensor data every 0.01 second using Simulink software. This part of the setup would provide magnitude and direction of the normal force and would serve as input to friction model/ controller designed by the post-doctoral fellow.

To control the orientation of the robot and adjust the gripper position, I developed an interface which permitted the gripper to perform micromotions every 0.01 seconds. The high speed micromotions enables the WidowX to promptly control the grip of the robot once slip event is detected by the friction model/ controller.

4.3 Tactile and haptic glove interface (Youtube link)

Technical Skills: Python, Altium Designer (PCB Design Software)

This interface is an telerobotics application and consists of following 3 parts:

- Development of tactile glove: We developed a fabric tactile sensor array in the form of glove that would be fitted to a robotic hand. The output of the tactile sensor is inversely proportional to the pressure applied by an external object.
- Development of haptic glove: To facilitate the remote human operator to perceive and understand distant and inaccessible environment, we developed haptic glove with an aim to render and replicate the sense of touch.
- Interface: Worked on developing first-generation interface that enables the user to feel the object gripped by the robot, by determining the amount of pressure applied to the sensorized tactile glove and equivalently modulating vibratory haptic feedback on the haptic glove worn by the user.

Depending on the amount of pressure applied to the sensorized robotic hand equivalent vibration will be experience by the user. Figure 7 displays that high pressure is applied to the robotic hand and equivalent intensity vibration will be felt by the remote user wearing the haptic glove.

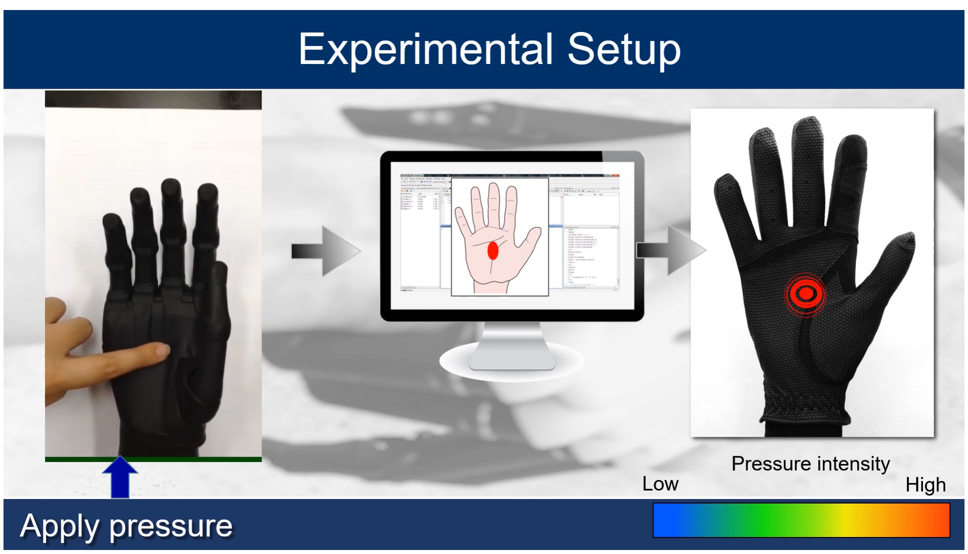


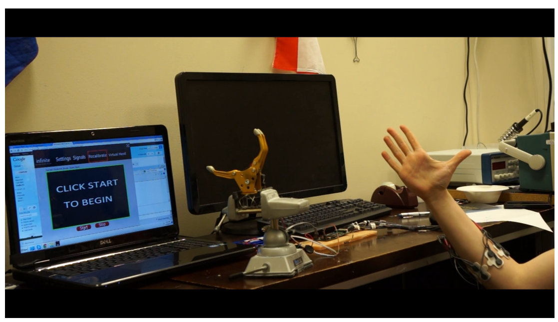
Figure 8: Tactile and haptic glove interface

Pattern recognition based prosthetic arm (YouTube link)

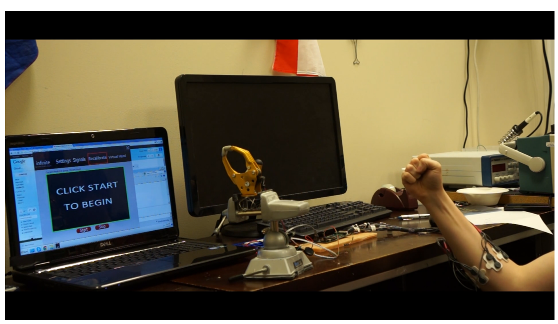
Technical skills: C++

The following section entails my work at Infinite biomedical technologies (a John Hopkins affiliated research lab in USA). Here I worked as a project manager, and with my team developed a stand-alone device that non-invasively records 8-channels of the subject's EMG signals (in real-time) to differentiate, determine and control prosthetic hand positions. We successfully implemented hand-open, close, flex, extend, pronate, supinate and hook hand positions.

The images below (as well as the YouTube video) display the working of our interface for hand open and close positions. The EMG patterns generated when the subject opens or closes hand are analyzed by our device and based on the classification result the robotic hand mimics the subject's hand position.



(a) hand open



(b) hand close

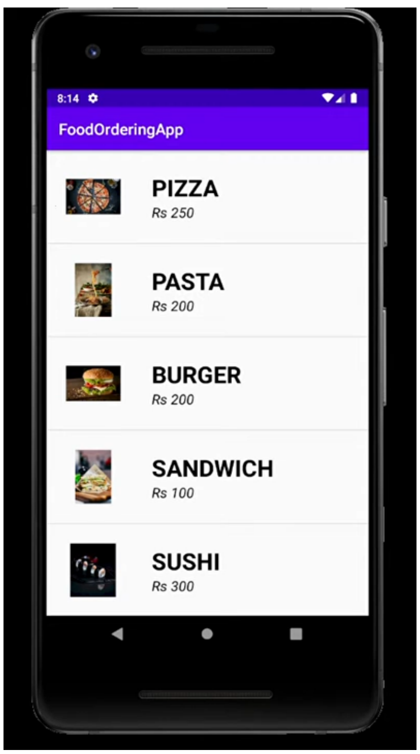
Figure 9: Controlling prosthetic hand based on subject's EMG patterns

Android development: Food ordering app (YouTube link)

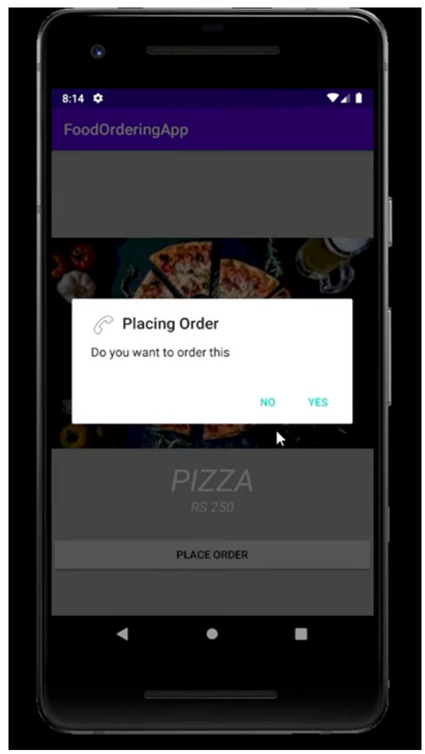
Technical skills: Android Studio, Java

I had enrolled at a one month long mobile app development course conducted by Robotech labs organized by IIT, Bombay.

Following is the github link for the apps that I developed during the course. Figure 11 displays snippets of food ordering and payment application that I successfully developed during the course.



(a) food item list



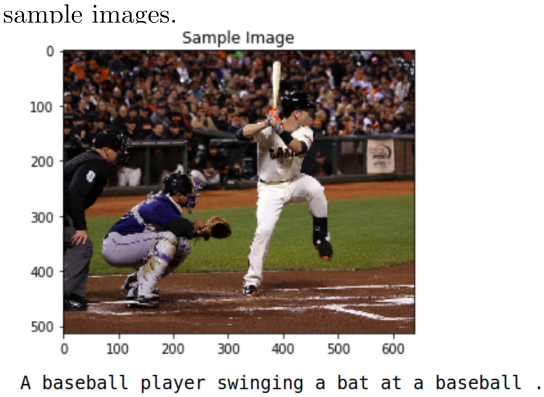
(b) payment page

Figure 10: Food ordering and payment app

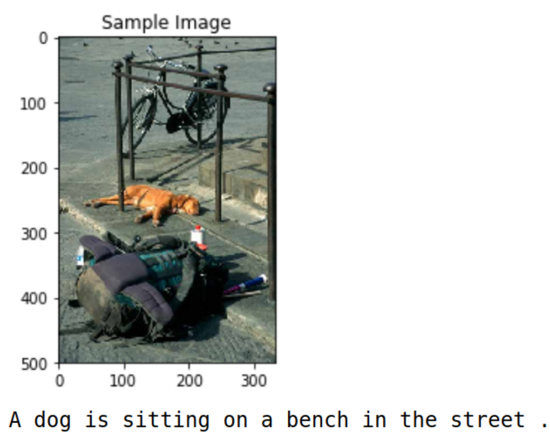
Image captioning
(GitHub Project Link)

Technical skills: Python, Pytorch

Image Captioning is the process of generating textual description of an image. Basically image captioning is computer describing an input image in a sentence or paragraph. It uses concepts of Computer Vision and Natural Language Processing fields to generate the captions. Figure 11 displays my developed model's predictions on sample images.



(a) image with correct prediction



(b) image where model could have performed better

Figure 11: Image captioning examples

This project was a part of my computer vision nanodegree program at udacity. I had applied for the Secure and Private AI challenge, and was selected among the Top 300 performers. As a result of which I was awarded a full scholarship to Computer Vision Nanodegree program.

Face generation
(GitHub Link)

Technical skills: Python, Pytorch, GAN

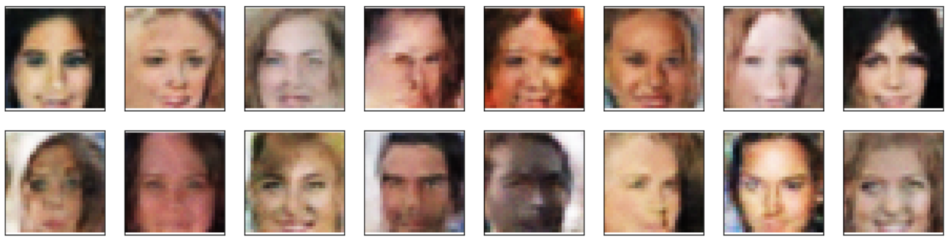


Figure 12: Face generation with GANs

I had applied to Bertelsmann Technology Scholarships and was among the Top 1,600 performers. As a result of which I was awarded a scholarship to Deep Learning Nanodegree program. A module of my program was dedicated to fake face generation using general adversarial networks (GANs). Figure 12 displays the generated samples (or generated faces) from my developed GAN model.

The generated samples do not exactly resemble human faces, rather seem to be scary caricatures. This was because the training data consists of most images that are clipped below the chin area and was biased towards white celebrity faces. To improve the model I am working towards creating a dataset comprising of high resolution images of complete faces of diverse ethnicity.

Self driving cars - Behavioral Cloning (YouTube link)

Technical skills: Python, Keras

I was selected for term 1 of self driving cars nanodegree through KPIT scholarship. In the nanodegree we implemented several projects like lane line detection, traffic signal classifier, behavioral cloning and sensor fusion via extended kalman filters. Following is the github link for the codes that I implemented during the course.
Of particular interest for me was behavioral cloning project. In this project our goal was to train a network using Keras which can successfully emulate human driving behavior autonomously in a simulated enviornment. I built the modified CNN based on NVIDIA's End to End Learning for Self-Driving Cars paper.

Figure 13: Car driving autonomously in simulator

Jewellery segmentation

Technical Skills: C++, OpenCV

In digital image processing and computer vision, image segmentation is a process of partitioning a digital image into multiple segments. It helps to simplify the image as well as the partitioned image is eventually assigned labels which with further processing play an important role in object detection within an image.

I worked as a Project Assistant at Multimodal Perception Laboratory, International Institute of Information Technology, Bangalore, India. My work here was to analyze different image segmentation algorithms and develop an algorithm that would segment the background in shared jewelry images. The output of our algorithm was to replace the original background with transparent background. Our vendors intended to replace the background, edit the image and use it on selling websites.

Figures 14 and 15 displays our algorithm's performance on original images shared by our vendors.

(a) original image (b) segmented image

Figure 14: Segmentation algorithm (best case)

(a) original image (b) segmented image

Figure 15: Segmentation algorithm (worst case)