

PROJECT REPORT

This document is intended to summarize my work as an SDE-AI intern at Machani Robotics Private Limited.

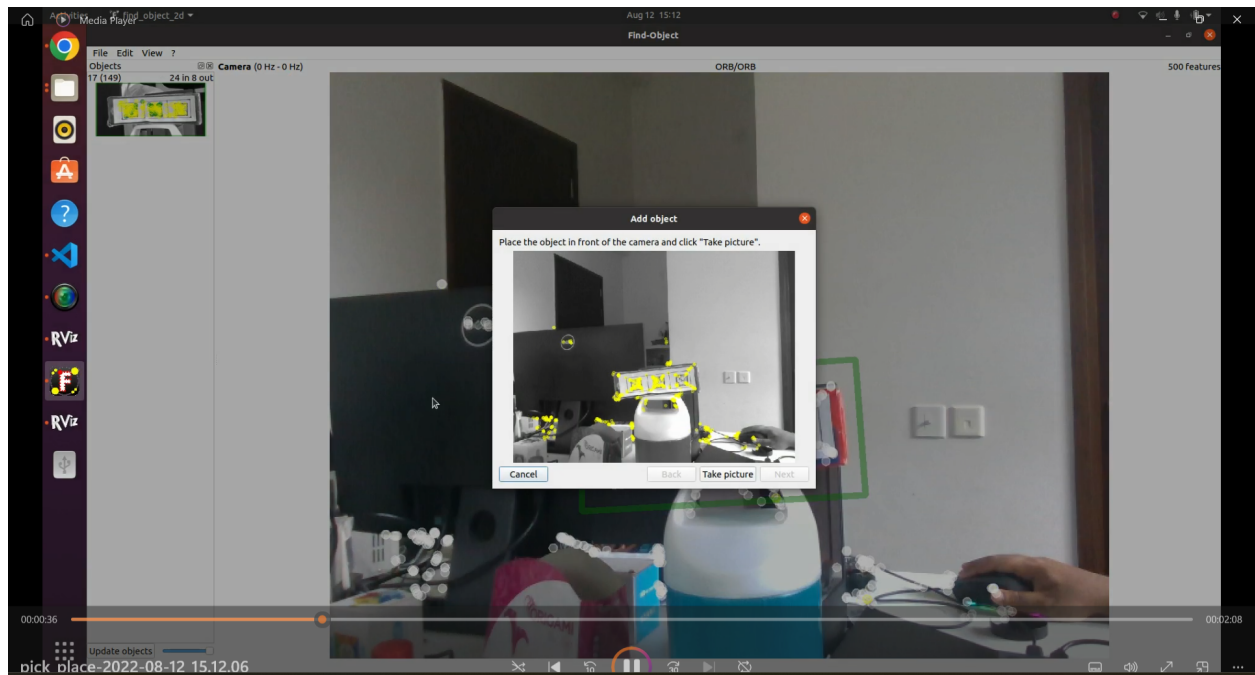
During the first week as an intern, I was told to watch lecture videos on Knn's, Linear Classification, Backpropagation, Neural Networks, and Convolutional Neural networks. I made a PDF report on back propagation using the help of <https://hmkcode.com/ai/backpropagation-step-by-step/>. I was given different values and did the necessary calculations to get the correct output. I have attached the report below

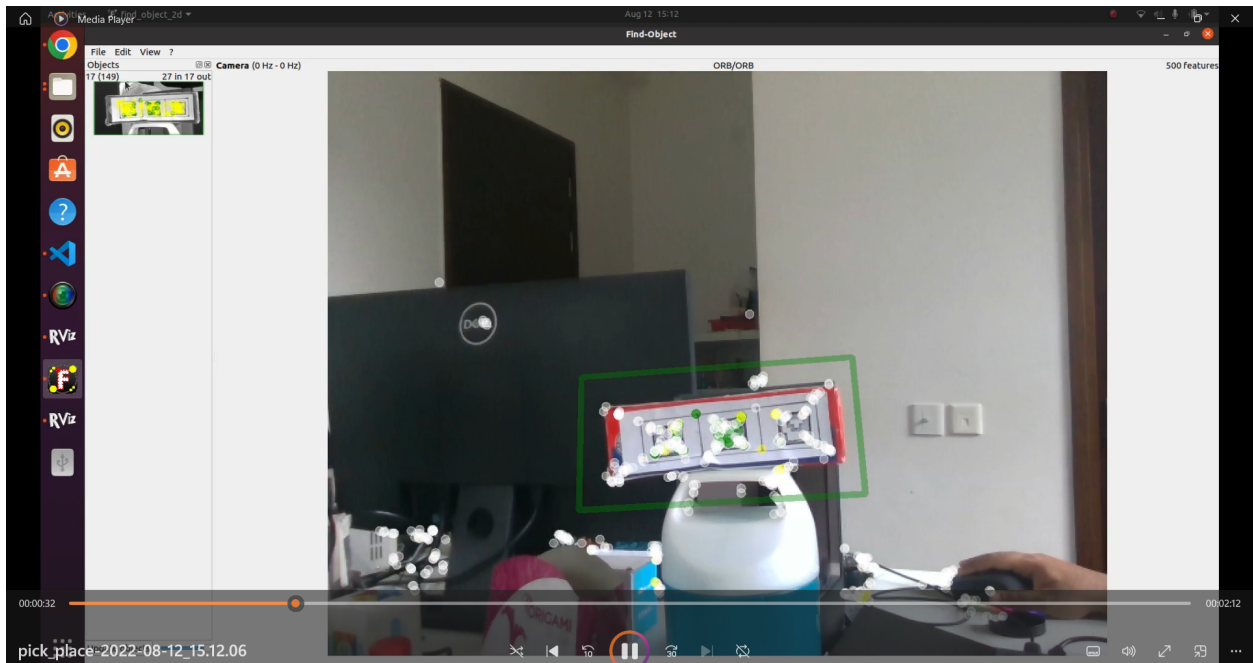
Robotic arm

The next week as an intern I started to learn about ROS(Robot Operating System) and Moveit. I started watching tutorials on the above 2 topics. I then started working on the robotic arm. I was provided with the URDF file for the robotic arm. Using the moveit setup assistant, I was able to configure the arm for my preference and was able to generate the configuration package for the robotic arm. I then started to move the arm in simulation using rviz moveit. I initially controlled the arms by manipulating the joint angles and by specifying the start and end goal. I was later able to control the movement of the arm through code. Given a 3D coordinate system, I was able to move the arm to that particular position using inverse kinematics to calculate the path. Using rviz I was able to add objects such as cubes, cones, and spheres to the scene. I then coded the arm to make it move to the object. Later I was able to add objects to the scene through code and make the arm move towards that object.

Object Detection using intel Real sense 3D camera:

Using an intel real sense D435i camera, I was able to grab the feature points of all the objects in a scene and capture any object I want it to detect. If an object is detected, it would be highlighted in green as shown below. This is essential for the pick and place operation as i was able to move the robotic arm to the detected object and place it to a specified 3D coordinate





Pick and place Operations:

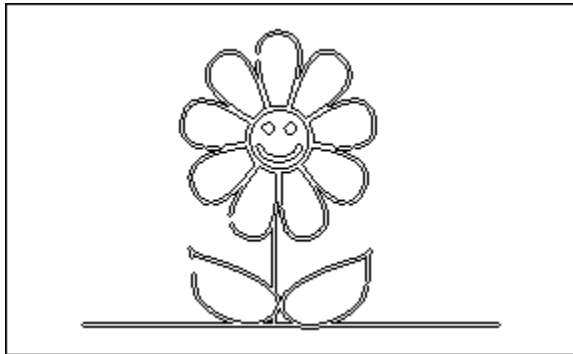
The goal of this operation was to add an object to a given place and make the robotic arm move towards the object, pick the object and place it at a given 3-dimensional coordinate system. In my case, my object was named 'object17'.

I have attached the link to the video demonstration of the simulation below.

<https://youtu.be/MzN32W4qHA1>

Drawing using the robotic arm

The goal of this operation was to make the robotic arm draw any image. This was achieved using an interactive marker array. I first converted the image to an array of coordinates that the compiler can read. In order to do that, I first got the contours of a given image using openCv and Canny software. This is what the output looks like.



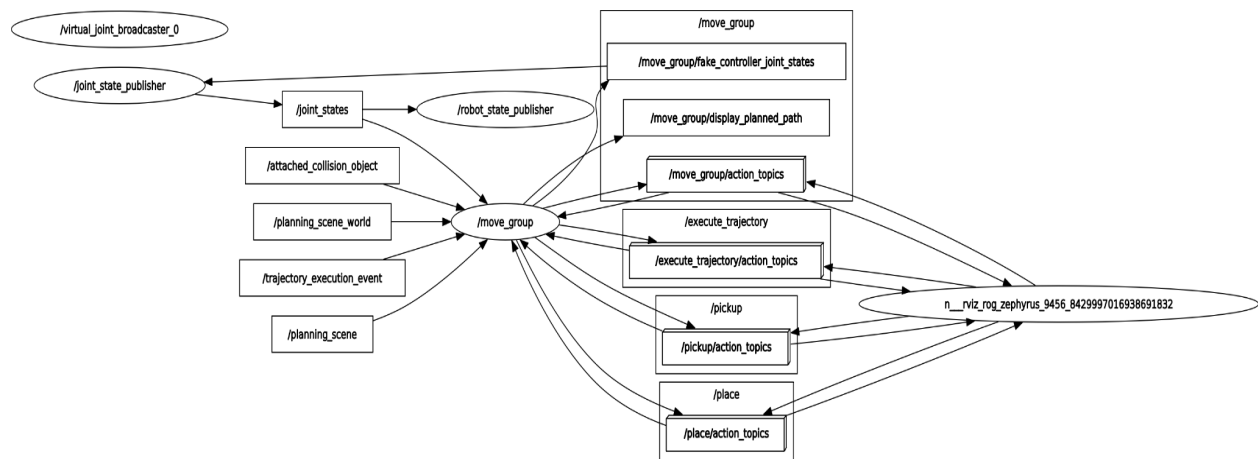
Then I converted this image into an array of coordinates that the compiler can read. After getting the coordinates, I supply it to another program that draws the image using interactive marker arrays. The arm then follows the trajectory and draws the image.

This is the link to the simulation: <https://youtu.be/b7WWIPUxYbM>

ROS Graph generated:

I have used rqt_graph to get how the nodes and topics are working.

rqt_graph provides a GUI plugin for visualizing the ROS computation graph. This graph shows all the topics and nodes which I have created. It shows all the publishers and subscribers I have created



CAN Communication :

Once I had everything running in simulation, I started learning about CAN Communication. The robotic arm was fitted with 6 RMD motors which uses CAN communication. I initially worked only on 1 motor. I used a PEAK-USB as a transceiver and was able to control various aspects of the motor. I was able to control the speed and position of the motor. I was also able to move the motor to a specific angle. I then connected another motor in parallel and I was able to run them both simultaneously as well. After working on these 2 motors, I started to work on the robotic arm.

I was able to do a couple of test runs but unfortunately one of the motors on the arm had stopped working.