Mini Competition 2020

Sensing and Intelligent System

Teaching Staff

Scenario

In a warehouse, a robot needs to pick a cargo from its initial location to the target destination to place.

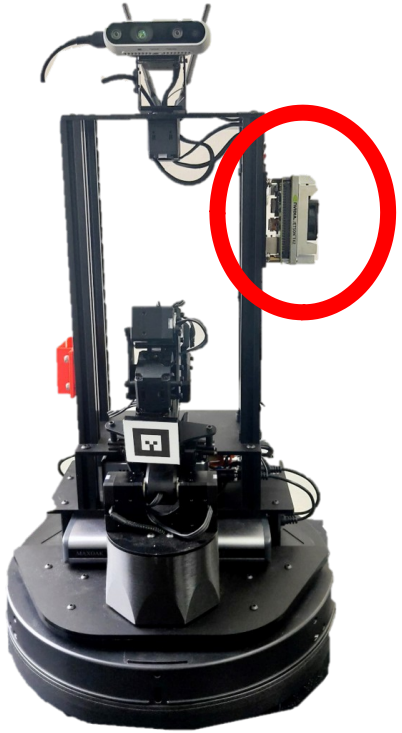
Now we have a robot which have only some basic functions and complete hardwares, please finish the rest of the algorithms to make sure it can works fine.

Platform

1. LocoBot
    a. WidowX 200 arm
    b. D435
2. TX2
3. UWB module

LoCoBot

TX2



Computing Unit

INTEL® NUC MINI PC

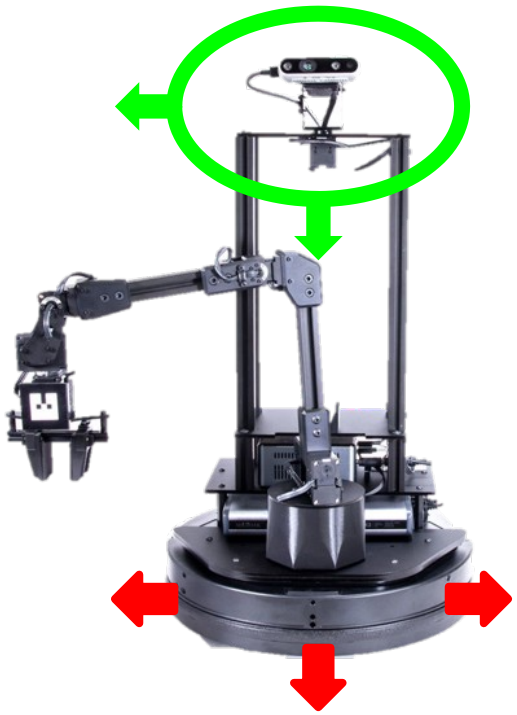
- CPU - Intel® Core™ i5-10210U
- RAM - 16 GB DDR4
- Storage - 256 GB

NVIDIA Jetson TX2

- GPU - NVIDIA Pascal™
- CPU - ARM® A57 Complex
- RAM - 8 GB L128 bit DDR4
- Storage - 32 GB eMMC 5.1 Flash

Mobility

- LoCoBot mobility
  - Forward
  - Backward
  - Left turn
  - Right turn
- Camera
  - Vertical turn
  - Horizontal turn



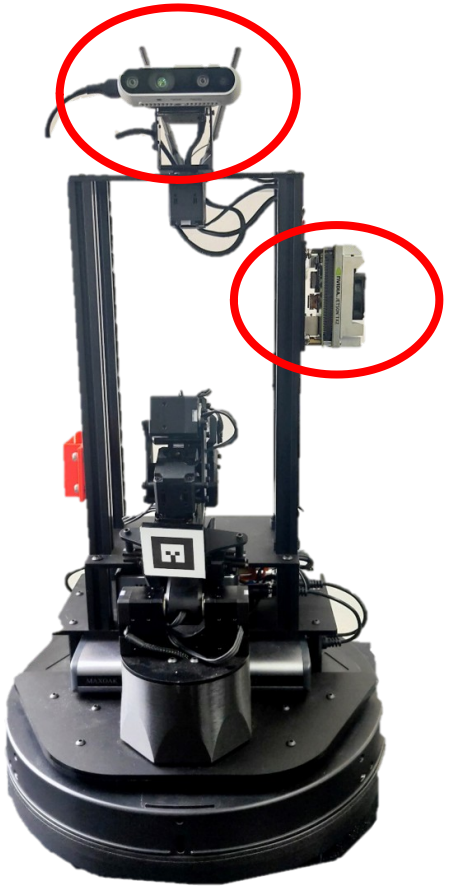
Perception

Camera : D435

Computer : TX2

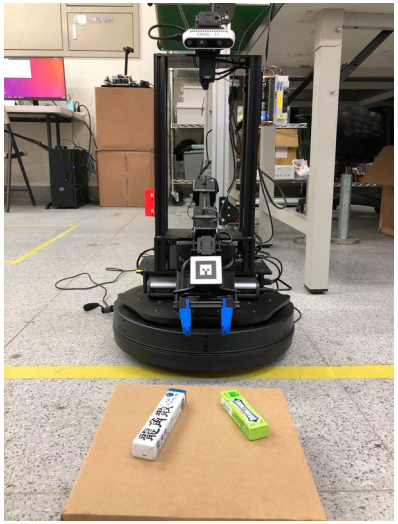
Use D435 to get images and predict bounding box by TX2.

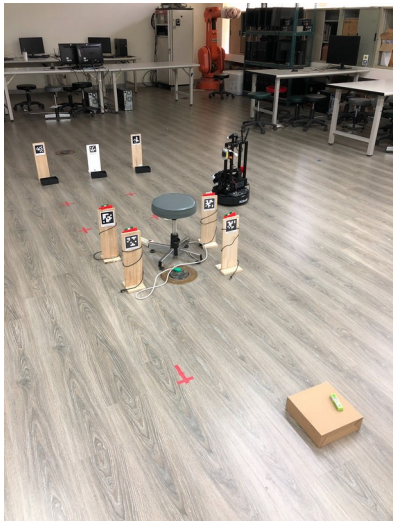
Data transmission via ROS.



Tasks: 4 Basic Tasks

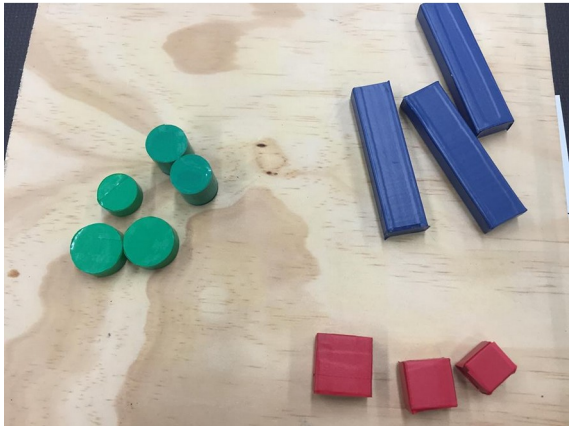
Object detection -> Pose estimation + Pick -> Move to distimation -> Place to box






Task 1: Object Detection

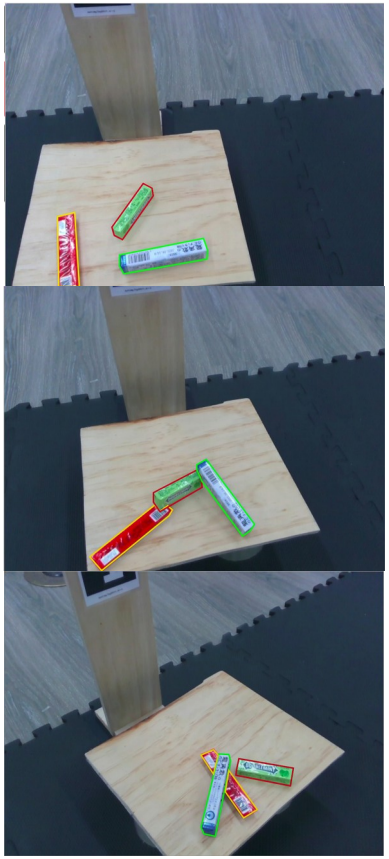
- Scenario: There are a certain number of objects in a platform, robot needs to detect its shape and draw a bounding box or contour for it.
- Used module: OpenCV, PyTorch
- Objects:
  - Dataset1 (24): cuboid, cube, cylinder which have same color, different size.
  - Dataset2 (10): 青箭口香糖，健達巧克力，龍角散，黃色小鴨，樂高人
  - We only provide the first three datasets for 青箭口香糖、健達巧克力、龍角散, and you have to log the rest of the datas for 黃色小鴨、樂高人



Mini competition dataset

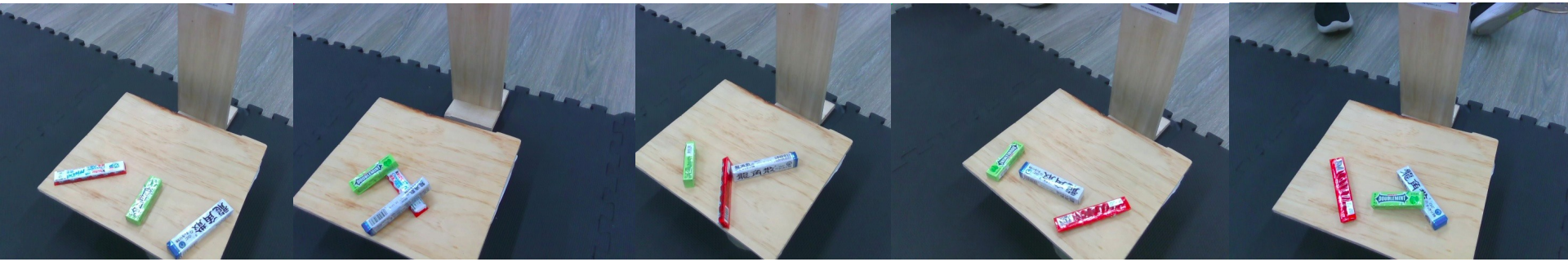
- 3 Products: 青健，健達，龍角散
  - Totally 744 images
  - Used Sensor: Realsense SR300
  - Pixel: 640*480
- Collect method:
  - About 18 viewpoint
  - Multi product placing method （41scenes）
- Label image format
  - Classes: "doublemint": 1, "kinder": 2, "kusan": 3

label



Different point of view

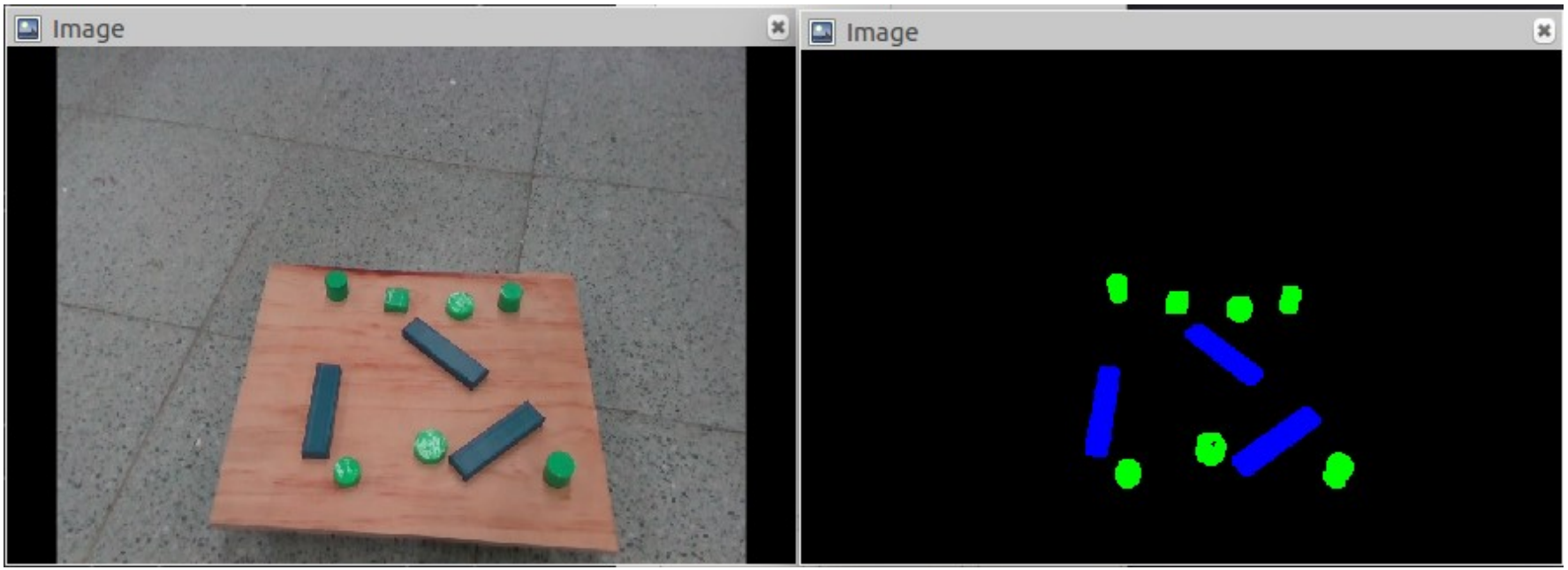
41 scene



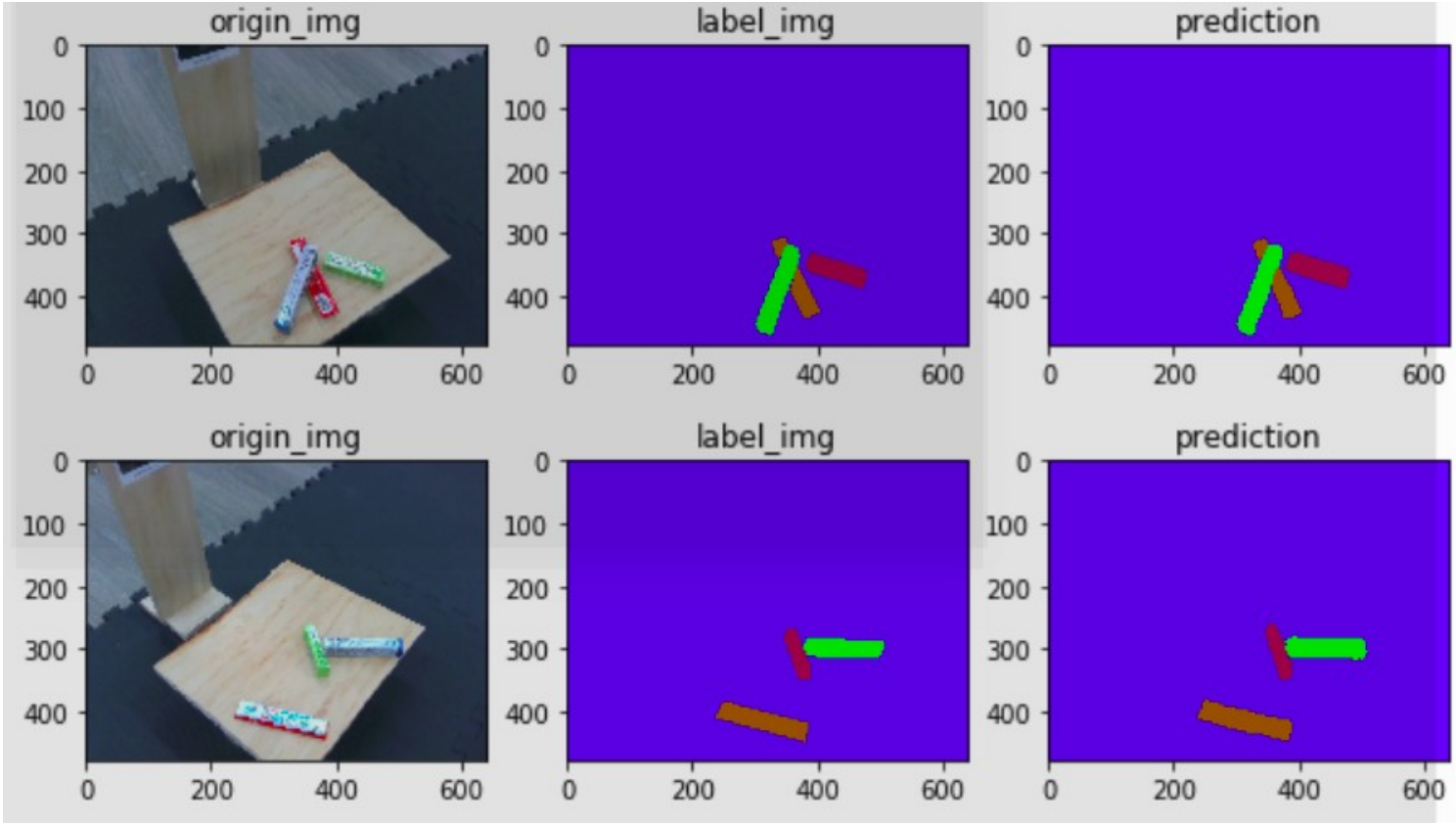
Task 1: Object Detection

- Input: Image (Realsense ROS message)
- Output: Bounding box or pixel wise prediction
- Score: Dataset1 accuracy (24) + Dataset2 accuracy (10)
- Baseline: Lab 5 and Deep learning section

Dataset 1 Expected Result



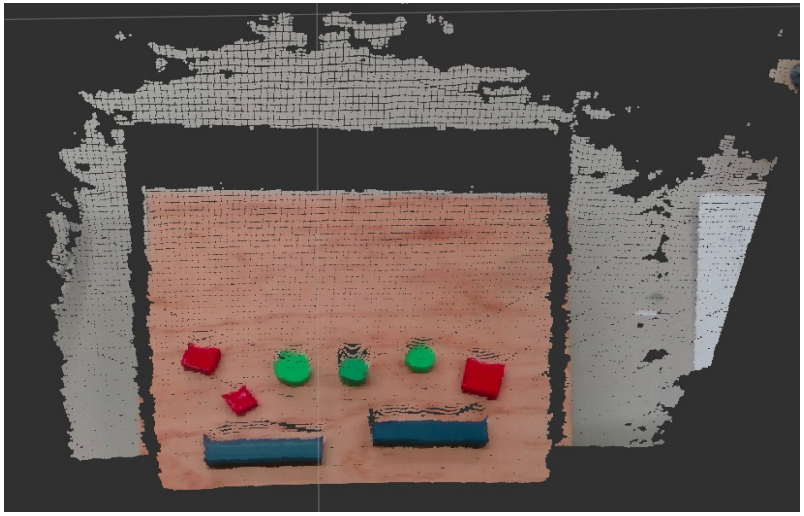
Dataset 2 Expected Result

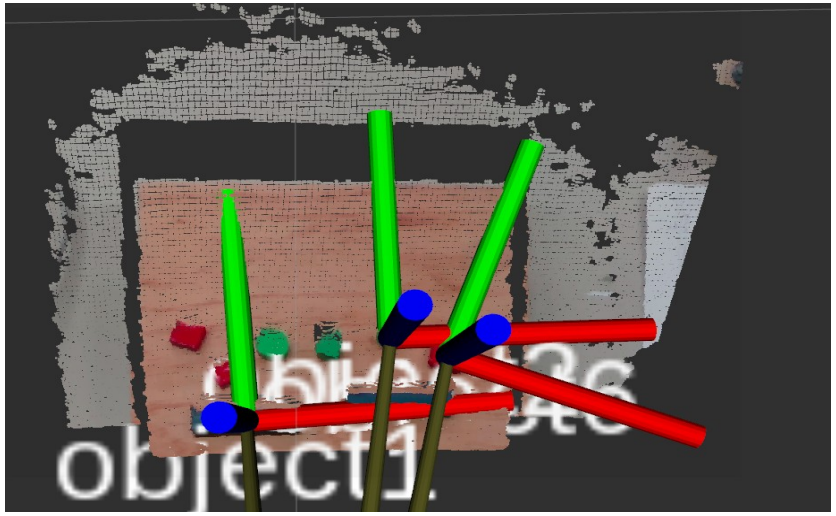


Task 2: Pose Estimation and Picking

- Scenario: Given object mask, robot needs to estimate all objects pose and pick one of a object.
- Used module: PCL, Moveit
- Input: Object mask
- Task: Pose Estimation and Pick
- Score: Pose accuracy (15) and pick result (9)
- Baseline: Lab 6, 7, and 13

Expected Result





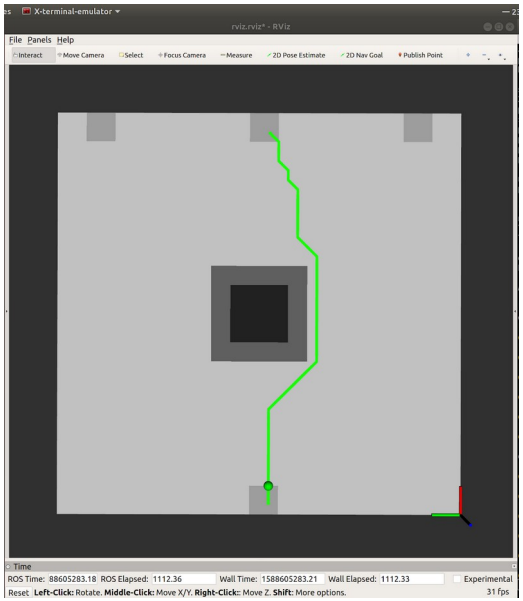
Task 3: Move to Destination

- Scenario: In a known map, given a destination from assigned Apriltag, park the platform to there
- Used Module: Navigation, AprilTag
- Input: Distance from AprilTag, Location and orientation from UWB module
- Task: move the platform to there
- Score: Set destination (15), moving (5), arriving (5)
- Baseline: Lab 4
- **Bonus:** If your robot finish parking in 30 seconds, get

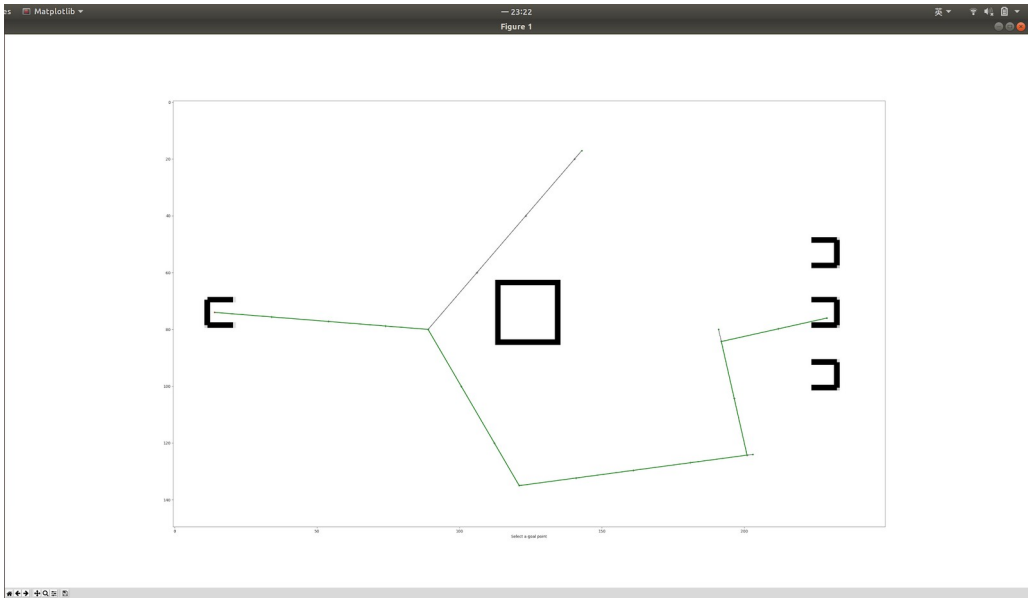
  2 more points

Interface

Astar



RRT



Task 4: Place to Box

- Scenario: Now LoCoBot has picked a object and arrive in destination. It needs to use apriltag to localize itself and place object to box.
- Used module: AprilTag, MoveIt
- Input: Image (ROS message)
- Task: Place the object inside box
- Score: Detect the target box (12), placing (12)
- Baseline: Lab 4 and 8

Demo Round

Task1 & 2: Team 1 ~ Team 5

Task3 & 4: Team 6 ~ Team 9

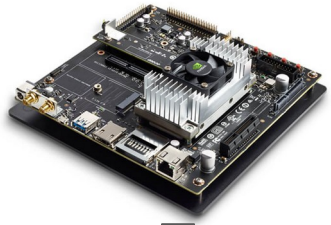
Come to demo if you are ready

Each team have two 8 minutes chances to demo in first round…

Each team will have extra chance in second round...

You can get a chance to earn extra point if you accomplish all the tasks

Submission Format

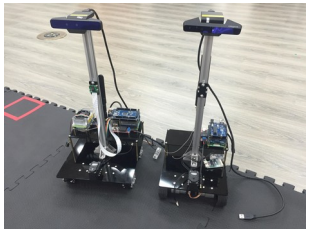


Develop and build docker image in Tx2

Docker Hub

docker

Push image into Docker Hub



Implement (Docker run)