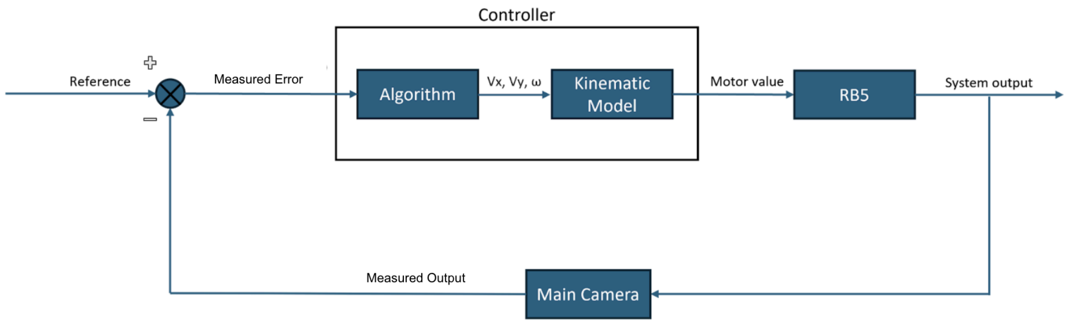
Homework 2

A69032537 Ming-Kai Liu
A69033395 Yu-An Chen

*we downscale the waypoints range from 1m x 2m to 0.5m x 1m due to spatial limitation
Demo Link: https://youtu.be/N5bIZA7h7wo?si=oz--Psqucw02C-VZ

(a) Close-Loop Controller



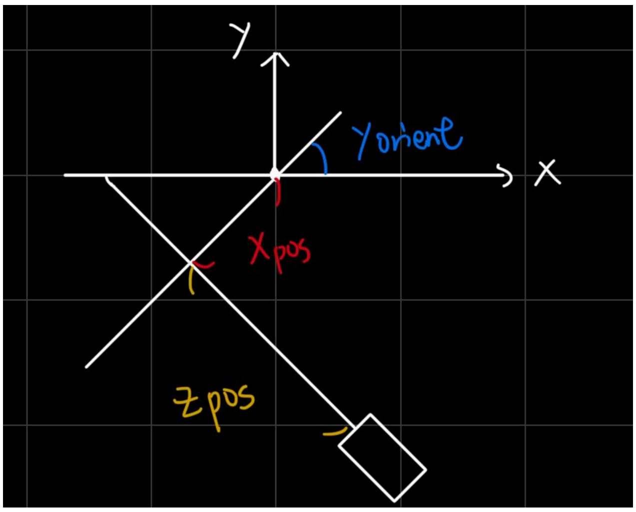
Node **Algorithm** takes apriltag detection as input and computes the V_x , V_y and the angle needed to rotate.
Node **Kinematic Model** takes V_x , V_y and the angle needed to rotate as input and output motor value of each wheel.
Node **RB5** performs the actual movement.
Node **Main Camera** performs the apriltag detection and pose estimation.

(b) Landmarks

We used the AprilTag as our landmark. For each waypoint, we used two apriltag. One is for aligning the θ in the waypoint (x, y, θ), one is for the robot to move toward the next waypoint.

(c) Car pose

(1) Estimating the Car Pose



Given:
C_a^W : the coordinate of the apriltag in the world coordinate system
x_{pos}: estimated position x return by apriltag detection algorithm
z_{pos}: estimated position z return by apriltag detection algorithm
y_{orient}: estimated rotation y return by apriltag detection algorithm

We assign each apriltag their **own** coordinate system where the y-axis is point inward the apriltag and x-axis is point to the left of the april tag. We compute the coordinate of the robot in the apriltag coordinate system C_r^a:

$$C_r^a = \left(\frac{-x_{pos}}{\cos\left(y_{orient}\right)} + \left(z_{pos} + cameraOffset + x_{pos} \cdot \tan\left(y_{orient}\right)\right) \cdot \sin\left(y_{orient}\right),\ -\left(z_{pos} + cameraOffset\right) \cdot \cos\left(y_{orient}\right) \right)$$

where *cameraOffset* = 7 (cm)

Since we have C_a^W for each apriltag, we can compute the coordinate of the robot in the world coordinate system $C_r^W = R^{-1}\left(C_r^a - t\right)$, where R is the rotation matrix and t is the translation vector. We only consider the x-direction (in robot coordinate) distance between the camera and robot itself by adding the *cameraOffset*, and neglect the y-direction distance.

(2) Noise Handling
We take 10 detection at a time and average them to get the less noisy estimated pose.

(d) Handling Failure of Detecting Landmarks

During motion, when there is no camera detection, we will stop after the movement to wait for the camera to update the robot's status. Based on the actual status, we can accurately determine the robot's position and make necessary adjustments

(f) Car Movement Analysis

Our robot didn't run very smoothly but can run accurately. This is because the april detection isn't fast enough to keep up the algorithm speed. For example, at t = 10, we might receive april detection captured at t = 9.5. Therefore, we have to sleep for 1 second before doing april detection to get the real-time detection.

(g) Total Moving Distance

(0, 0, 0) to (0.5, 0, 0) → 1.067 (unit = 50cm)
(0.5, 0, 0) to (0.5, 1, π) → 0.9196 (unit = 100cm)
(0.5, 1, π) to (0, 0, 0) → 1.171 (unit = 111.8cm)
Total moving distance = 1.059 (unit = 261.8cm)

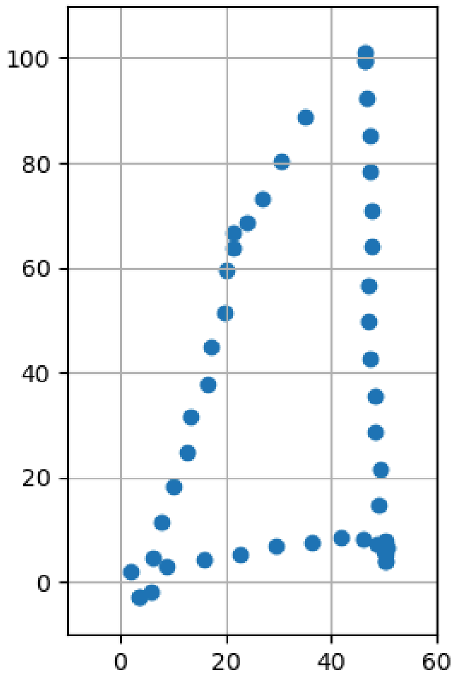
(h) Location Error

(0.5, 0, 0): robot location = (0.532, 0.058, 0.0344)
rotation error = -0.0344
translation error = $\sqrt{0.032^2 + 0.058^2} = 0.0662\,m$

(0.5, 1, π): robot location = (0.464, 0.994, 3.104)
rotation error = 0.0376
translation error = $\sqrt{0.036^2 + 0.006^2} = 0.0365\,m$

(0, 0, 0): robot location = (0.0368, -0.0267, -0.0339)
rotation error = 0.0339
translation error = $\sqrt{0.0368^2 + 0.0267^2} = 0.0455\,m$

(i) Trajectory



(j) Reflection

The overall performance is acceptable since the robot did travel to each waypoint accurately. However, there are two small problems. First, we didn't consider the y-direction (in robot coordinate) distance between the camera and the robot, thus the estimated robot coordinate would be slightly leftward to its actual coordinate (see the trajectory between (0, 0, 0) and (0.5, 0, 0), the point is slightly leftward to the actual robot's position). Secondly, apriltag detection sometimes gives wrong estimation, which seems to be derived from the failure of the algorithm to correctly capture the apriltag ("**Error, more than one new minima found**").