Brief Report

Project Name:

Supervisor: Yun Wu

Student: Jamie Smith

Date: 25/06/2025

Plan: 20/06/2025-

1. Calibrate the sensor, especially the camera (hint: how to save the calibration setting in ROS):

[JetRacer\_ROS\_AI\_Kit\_Tutorial\_VIII:\_Start\_the\_Camera\_Node](https://www.waveshare.com/wiki/JetRacer_ROS_AI_Kit_Tutorial_VIII:_Start_the_Camera_Node)

[JetRacer\_ROS\_AI\_Kit\_Tutorial\_VII:\_Robot\_Odometer\_Calibration](https://www.waveshare.com/wiki/JetRacer_ROS_AI_Kit_Tutorial_VII:_Robot_Odometer_Calibration)

1. Obstacle avoidance (hint: how is the state machine pipeline working):

[Using camera only](https://gitlab.eeecs.qub.ac.uk/3048777/csc-3002-efficient-autonomous-obstacles-avoidance-on-jetracer-1-2025):

[Using fused camera & LiDar](https://gitlab.eeecs.qub.ac.uk/3048777/csc-3002-efficient-autonomous-obstacle-avoidance-on-jet-racer-2025)

1. Identify the place where the PID code is implemented and think about how it can be replaced by MPC (hint: under stand the input/output of the controller)

Brief Report (any progress, thoughts, and development achievements):

1. **Line-following achieved:** The robot now reliably tracks the path at slow speeds via the camera module. Image-processing latency is still apparent, so the plan to integrate an MPC controller to sustain higher speeds is clear.
2. **GitLab:** I next need to setup a GitLab repository to document progress and store any files I modify.
3. **Obstacle-avoidance refinement:** With the repository in place, I’ll revisit the two prior avoidance strategies.

Questions (any concerns related to development, progress, or supports):