

Nama : Eko Putra Nugraha

NIM : 1103213212

Assalamualaikum WR.WB perkenalkan nama saya Eko Putra Nugraha dengan NIM 1103213212, disini saya akan mempresentasikan implementasi line follower pada webots, pertama tama kita memulai dulu dari mengclone project nya dari github <https://github.com/silvery107/vision-line-follower.git> sesuai dengan directory yang diinginkan lalu setelah itu kita perlu menyiapkan aplikasi webots dengan versi 2021a, lalu kita menginstall python dengan versi 3.9 dan setelah install python tersebut kita cari edit enviroment pada search bar di windows setelah itu klik lalu tekan edit enviroment pada bagian path kita masukan path python 3.9 lalu pada command prompt kita masukan file path python 39 dan tambahkan python.exe -m pip install numpy untuk menginstall numpy tersebut setelah itu kita membuka file yang telah kita clone tersebut pada bagian world, untuk mengakses codingannya kita klik robotnya lalu masuk pada tree di sebelah kanan setelah itu tekan pada bagian controller dan disana terdapat 3 macam codingan yang pertama cmd_controller lalu IR_controller dan terakhir vision_controller masing masing dari ketiga codingan tersebut menggunakan sensor berbeda

cmd_controller.py - Optimal jika lingkungannya sederhana dan hanya membutuhkan deteksi dasar untuk menghindari atau mengikuti objek tanpa detail yang kompleks. Pendekatan berbasis sensor jarak ini efektif untuk robot yang bergerak di lintasan tetap atau ruang yang relatif kosong. Kelebihannya adalah lebih efisien dan sederhana, namun kurang akurat di lingkungan kompleks.

🔍 **IR_controller.py** - Lebih optimal untuk pengendalian robot di jalur garis yang membutuhkan akurasi dalam mengikuti lintasan, seperti pada robot line-follower. Dengan pengontrol PD berbasis IR, kode ini dapat mengoreksi arah dengan lebih responsif dan akurat saat mendeteksi penyimpangan. Kekurangannya adalah rentan terhadap kesalahan jika banyak objek yang menyerupai garis lintasan.

🔍 **vision_controller.py** - Paling optimal untuk kondisi lintasan yang kompleks dan dinamis. Menggunakan kamera dan pengolahan gambar memungkinkan deteksi yang lebih canggih untuk pusat garis dan objek di sekitar. Kelebihan utamanya adalah akurasi yang tinggi dan kemampuan menyesuaikan dengan lingkungan yang rumit. Namun, ini memerlukan lebih banyak sumber daya komputasi dan dapat memperlambat performa pada perangkat dengan daya pemrosesan terbatas.

Jika prioritasnya adalah akurasi dalam lingkungan kompleks, **vision_controller.py** dengan kamera dan pengolahan gambar adalah yang paling optimal. Namun, jika fokus pada efisiensi dan lingkungan tidak terlalu kompleks, **IR_controller.py** dengan sensor inframerah adalah pilihan yang lebih baik.

Sekian Terimakasih.