

Analisis Implementasi dan Simulasi JetBot di Webots

1. Simulasi JetBot Basic Motion

Tujuan dari simulasi JetBot Basic Motion adalah untuk menguji gerakan dasar JetBot dalam lingkungan simulasi Webots. Pada saat ini, JetBot diuji melakukan gerakan maju, mundur, belok kiri, dan kanan. Hasil simulasi menunjukkan bahwa API Webots memberikan kontrol robotik yang tepat dengan waktu respons yang tepat, memungkinkan JetBot untuk melakukan manuver dasar dengan mudah. Hasil ini menunjukkan bahwa JetBot berfungsi dengan baik dalam pengendalian dasar, yang menjadikannya dasar yang baik untuk simulasi yang lebih kompleks.

Namun, ada peluang untuk pengembangan. Misalnya, mereka dapat menguji pergerakan di medan yang lebih kompleks atau menambahkan sistem pengendalian yang menggunakan sensor untuk melacak perubahan lingkungan secara real-time.

2. Simulasi jetbot_collect_data

Pada tahap ini, data dikumpulkan untuk melatih model AI penghindaran tabrakan. Data gambar diambil menggunakan kamera JetBot dan diklasifikasikan menjadi dua kategori, yaitu "free" (jalur aman) dan "blocked" (jalur terhalang), dengan masing-masing kategori memiliki 20 gambar. Data ini kemudian disimpan dalam folder terpisah untuk digunakan dalam pelatihan model.

Hasil simulasi ini menunjukkan efisiensi proses pengumpulan data menggunakan API kamera di Webots. Proses manual yang mengharuskan operator untuk mengklasifikasikan gambar memberikan fleksibilitas, tetapi juga dapat menimbulkan bias data. Selain itu, jumlah dataset yang kecil cukup untuk pengujian awal, tetapi kurang ideal untuk skenario yang lebih kompleks atau lingkungan nyata.

Potensi perbaikan meliputi otomatisasi klasifikasi data dengan bantuan sensor tambahan atau meningkatkan jumlah dataset untuk meningkatkan generalisasi model.

3. Simulasi jetbot_collision_avoidance

Simulasi ini mengimplementasikan sistem penghindaran tabrakan menggunakan model deep learning berbasis ResNet18 yang telah dilatih dengan data dari tahap sebelumnya. JetBot memproses gambar dari kamera secara real-time dengan normalisasi dan transformasi tensor sebelum dimasukkan ke model. Prediksi probabilitas jalur aman atau terhalang ditentukan oleh fungsi softmax, dengan aturan navigasi sebagai berikut: jika probabilitas "blocked" < 0.5 , robot bergerak maju; jika ≥ 0.5 , robot berbelok kiri untuk menghindari hambatan.

Hasil simulasi menunjukkan bahwa JetBot dapat beradaptasi dengan kondisi lingkungan untuk navigasi. Sistem ini sederhana namun efektif untuk skenario penghindaran tabrakan dasar. Namun, karena hanya memiliki dua respons navigasi, yaitu maju atau belok kiri, sistem kurang fleksibel untuk situasi yang lebih kompleks. Selain itu, dataset yang lebih kecil dapat membatasi performa model pada berbagai lingkungan. Pengembangan lebih lanjut dapat mencakup penambahan kategori respons seperti belok

kanan atau berhenti, serta meningkatkan dataset untuk lingkungan yang lebih beragam. Selain itu, memanfaatkan GPU selama pelatihan dapat mempercepat proses pengembangan model.

Kesimpulan

Bagaimana sistem robotik berbasis AI dapat digunakan dalam aplikasi nyata ditunjukkan oleh implementasi dan simulasi JetBot di Webots. Sementara `jetbot_collect_data` dan `jetbot_collision_avoidance` menunjukkan kemampuan dasar JetBot untuk bermanuver, `jetbot_collision_avoidance` dan `jetbot_basic_motion` menunjukkan kemampuan AI untuk mengotomatisasi navigasi dengan menghindari tabrakan.

Namun, dataset yang kecil dan navigasi yang mudah adalah beberapa keterbatasan simulasi ini. Untuk meningkatkan kemampuan sistem dalam menghadapi kondisi dunia nyata yang lebih kompleks, adalah penting untuk meningkatkan variasi data, menambah respons navigasi, dan mengoptimalkan proses pelatihan. Metode ini telah terbukti efektif dalam simulasi dan akan memungkinkan aplikasi robotik berbasis AI yang lebih kompleks di masa depan.