

Nama : Faiz Hibatullah
NIM : 1103210172
Kelas : TK-45-G09

1. Pendahuluan

ROS (Robot Operating System) adalah kerangka kerja yang memfasilitasi pengembangan aplikasi robotika dengan menyediakan berbagai macam alat dan pustaka. Laporan ini menjelaskan cara mengimplementasikan navigasi robot otonom menggunakan ROS1 dengan simulasi TurtleBot3 di Gazebo. Tutorial ini fokus pada dua komponen utama: Simultaneous Localization and Mapping (SLAM) untuk pemetaan lingkungan dan perencanaan navigasi otomatis untuk mengarahkan robot dari satu titik ke titik lainnya. Tujuan utama dari tutorial ini adalah untuk memperkenalkan konsep dasar pengendalian robot, pemetaan lingkungan, dan navigasi menggunakan ROS

2. Persiapan

Persiapan yang dibutuhkan untuk dapat melakukan tutorial ini adalah sebagai berikut:

- **ROS Noetic:** Sistem operasi robot yang akan digunakan untuk mengendalikan robot dan menjalankan algoritma navigasi.
- **Gazebo:** Simulator robot yang akan digunakan untuk mensimulasikan TurtleBot3.
- **TurtleBot3:** Robot virtual yang akan digunakan dalam simulasi, yang telah diprogram untuk berfungsi dengan ROS.
- **RViz:** Alat visualisasi untuk melihat peta, posisi robot, dan jalur yang direncanakan.
- **GMapping:** Algoritma SLAM untuk membuat peta lingkungan.

3. Langkah Implementasi

Pada bagian ini, kita akan menjelaskan langkah-langkah yang dilakukan selama pembuatan tutorial, mulai dari instalasi hingga eksekusi.

1. Instalasi ROS

- Instal ROS Noetic dengan perintah:

```
sudo apt install ros-noetic-desktop-full
```

2. Instalasi Paket TurtleBot3 dan Gazebo

- Instal paket TurtleBot3 dan dependensi untuk menjalankan simulasi:

```
sudo apt install ros-noetic-turtlebot3 ros-noetic-turtlebot3  
simulations
```

- Tambahkan konfigurasi untuk model TurtleBot3 ke file .bashrc:

```
gedit ~/.bashrc  
  
export TURTLEBOT3_MODEL=burger
```

3. Menjalankan Simulasi di Gazebo

- Jalankan simulasi dengan:

```
roslaunch turtlebot3_gazebo turtlebot3_world.launch
```

- Ini akan membuka Gazebo dengan model TurtleBot3 yang siap untuk dijalankan di lingkungan virtual.

4. Pemetaan Lingkungan dengan SLAM

- Jalankan SLAM dengan menggunakan GMapping:

```
roslaunch turtlebot3_slam turtlebot3_slam.launch  
slam_methods:=gmapping
```

- Gunakan *keyboard teleop* untuk menggerakkan robot dan mengeksplorasi lingkungan:

```
roslaunch turtlebot3_teleop turtlebot3_teleop_key.launch
```

- Simpan peta hasil SLAM:

```
roslaunch map_server map_saver -f ~/map
```

5. Navigasi Otomatis dengan Peta

- Jalankan navigasi otomatis dengan perintah:

```
roslaunch turtlebot3_navigation turtlebot3_navigation.launch  
map_file:=$HOME/map.yaml
```

6. Menetapkan Tujuan Navigasi di RViz:

- Di RViz, gunakan alat *2D Nav Goal* untuk menetapkan tujuan navigasi robot.
- Robot akan bergerak otomatis ke lokasi yang ditentukan mengikuti jalur yang telah direncanakan.

4. Hasil

Setelah mengikuti langkah-langkah di atas, robot berhasil melakukan pemetaan lingkungan menggunakan SLAM dan menyimpan peta yang dihasilkan. Robot dapat bergerak secara otomatis menuju tujuan yang ditentukan dengan mengacu pada peta yang disimpan dan menggunakan algoritma perencanaan jalur. Alat visualisasi RViz memungkinkan pengguna untuk melacak pergerakan robot dan memverifikasi apakah robot mencapai tujuannya dengan tepat. Hasil yang diharapkan tercapai, yaitu robot berhasil melakukan navigasi dengan menghindari rintangan, menghitung jalur, dan bergerak secara otonom dari satu titik ke titik lainnya.

5. Kesimpulan

Tutorial ini memberikan pemahaman komprehensif tentang cara mengimplementasikan navigasi robot menggunakan ROS1, TurtleBot3, dan Gazebo. Dari tutorial ini, kita memahami pentingnya integrasi berbagai komponen dalam sistem ROS, seperti teleoperasi, SLAM, dan navigasi otomatis, dalam menciptakan robot yang mampu berfungsi secara otonom. Selain itu, kita juga mendapatkan pemahaman yang lebih mendalam mengenai penggunaan perangkat lunak simulasi seperti Gazebo dan RViz untuk pengujian dan visualisasi dalam pengembangan

robotika. Tutorial ini juga dapat menjadi dasar yang kuat untuk mengeksplorasi topik lebih lanjut dalam robotika, seperti perencanaan jalur, lokalisasi, dan integrasi sensor.