

Laporan UAS Robotika

Nama Lengkap: Muhammad Nugraha Sadewa

Mata Kuliah: Robotika

Bab 1: Pendahuluan

Pada laporan ini, kami akan menjelaskan langkah-langkah dan analisis yang dilakukan untuk mensimulasikan Seven Degree of Freedom Arm menggunakan Robot Operating System (ROS). Laporan ini mencakup command yang dimasukkan ke terminal Ubuntu, serta analisis hasil simulasi. Simulasi dilakukan dengan tujuan mengontrol robot secara virtual sebelum implementasi fisik.

Bab 2: Tujuan

1. Menjalankan simulasi robot lengan 7 DOF di dalam ROS.
2. Mengimplementasikan fungsionalitas pick and place menggunakan ROS.
3. Menambahkan objek kolisi untuk menguji kemampuan robot dalam mendeteksi dan menghindari hambatan.
4. Mengontrol posisi sendi robot menggunakan ROS dan memverifikasi statusnya.
5. Mengimplementasikan komunikasi antar node menggunakan Publisher, Subscriber, dan Service di ROS.

Bab 3: Alat dan Bahan

Hardware:

- Komputer dengan Ubuntu 20.04 atau lebih baru.
- RAM minimal 4 GB dan prosesor Intel i5 atau lebih tinggi.

Software:

- Robot Operating System (ROS) Noetic.
- Rviz untuk visualisasi.
- Gazebo untuk simulasi robot.
- MoveIt! untuk perencanaan gerakan.
- CoppeliaSim untuk simulasi lanjutan.

Bab 4: Langkah-Langkah Pengerjaan

Chapter 6

A. Menjalankan Antarmuka MoveIt!

Command:

```
roslaunch seven_dof_arm_config demo.launch
```

Penjelasan:

Perintah ini digunakan untuk membuka antarmuka MoveIt!, yang memungkinkan konfigurasi dan simulasi visual lengan robot.

Hasil dan Analisis:

- Antarmuka MoveIt! berhasil dibuka dengan model lengan robot terlihat di layar.
- Pengguna dapat mengontrol dan menguji pergerakan lengan robot dengan efisien.

B. Menjalankan Simulasi di Gazebo

Command:

```
roslaunch seven_dof_arm_gazebo  
seven_dof_arm_bringup_moveit.launch
```

Penjelasan:

Command ini membawa simulasi ke dalam lingkungan Gazebo sekaligus mengintegrasikan kontrol MoveIt! untuk lengan robot.

Hasil dan Analisis:

- Simulasi berjalan baik di Gazebo.
- Model lengan robot bergerak sesuai perintah, menunjukkan integrasi antara MoveIt! dan Gazebo berhasil.

Chapter 7

A. Menambahkan Objek Kolisi

Command:

```
roslaunch seven_dof_arm_test add_collision_object
```

Penjelasan:

Command ini menjalankan script yang menambahkan objek, seperti kotak atau silinder, dalam ruang simulasi. Objek ini menjadi hambatan yang harus dihindari robot.

Hasil dan Analisis:

- Objek kolisi berhasil ditambahkan dan terdeteksi oleh robot.
- Robot menghindari objek tanpa masalah.

B. Melakukan Tugas Pick and Place

Command:

```
roslaunch seven_dof_arm_test pick_place
```

Penjelasan:

Command ini menjalankan script untuk mengambil objek dan memindahkannya ke lokasi target menggunakan MoveIt!.

Hasil dan Analisis:

- Robot berhasil melaksanakan tugas pick and place dengan akurat.
- Tidak ada kesalahan atau tabrakan saat pergerakan berlangsung.

Chapter 5

A. Mengontrol Sendi Robot di CoppeliaSim

Command untuk Menggerakkan Sendi:

```
rostopic pub /csim_demo/seven_dof_arm/elbow_pitch/cmd
std_msgs/Float32 "data: 1.0"
```

Command untuk Verifikasi Status:

```
rostopic echo /csim_demo/seven_dof_arm/elbow_pitch/state
```

Penjelasan:

- Perintah pertama digunakan untuk mengontrol posisi sendi elbow_pitch.
- Perintah kedua memverifikasi status posisi sendi.

Hasil dan Analisis:

- Sendi elbow_pitch bergerak sesuai perintah.
- Status posisi berhasil diverifikasi menggunakan rostopic echo.

Chapter 2

A. Implementasi Komunikasi Antar Node

Publisher dan Subscriber untuk Topik

Command Publisher:

```
roslaunch masterling_ros_demo_pkg demo_topic_publisher
```

Command Subscriber:

```
roslaunch masterling_ros_demo_pkg demo_topic_subscriber
```

Penjelasan:

- Publisher menerbitkan data ke topik tertentu, sementara Subscriber mendengarkan data dari topik tersebut.
- Keduanya digunakan untuk mengimplementasikan komunikasi data antar node di ROS.

Hasil dan Analisis:

- Data berhasil diterbitkan dan diterima tanpa kendala.

Service untuk Komunikasi Sinkron**Command Service Server:**

```
roslaunch masterling_ros_demo_pkg demo_service_server
```

Command Service Client:

```
roslaunch masterling_ros_demo_pkg demo_service_client
```

Penjelasan:

- Service memungkinkan komunikasi sinkron antar node, di mana client mengirimkan permintaan dan server memberikan respons.

Hasil dan Analisis:

- Respons dari server diterima dengan baik oleh client, menunjukkan implementasi berhasil.

Bab 5: Analisis Hasil**1. Keberhasilan Simulasi:**

- Simulasi berjalan lancar dengan konfigurasi MoveIt!, Gazebo, dan CoppeliaSim.

2. Pengujian Objek Kolisi:

- Robot mendeteksi dan menghindari objek kolisi tanpa kendala.

3. Tugas Pick and Place:

- Pergerakan robot akurat dan mulus.

4. Kontrol Sendi:

- Robot dapat dikontrol melalui ROS dengan posisi yang sesuai.
- Status sendi berhasil diverifikasi.

5. Implementasi Komunikasi:

- Publisher dan Subscriber memungkinkan pengiriman data secara asinkron.
- Service memberikan solusi komunikasi sinkron antar node.

Bab 6: Kesimpulan

Dari hasil simulasi dan implementasi, dapat disimpulkan bahwa ROS, MoveIt!, Gazebo, dan CoppeliaSim memungkinkan pengendalian robot yang kompleks dengan efisien. Semua perintah dan kode yang digunakan menunjukkan hasil yang diharapkan, seperti:

1. Pengendalian gerakan robot dengan akurasi tinggi.
2. Deteksi objek kolisi dan penghindaran yang optimal.
3. Validasi gerakan dalam simulasi virtual sebelum implementasi fisik.
4. Komunikasi antar node dengan Publisher-Subscriber dan Service berhasil diimplementasikan dengan baik, mendukung pengembangan sistem robotika yang lebih kompleks.