Name: Rana Talal Ahmad Khan

Reg No: 450436

Submitted to: Dr. Yasar Ayaz

Report on Wheeled Mobile Robot Pose and Motion Estimation

Abstract

This study investigates four steps for estimating the pose and motion of a wheeled mobile robot using a TurtleBot3 in a differential drive setup. The research includes visualizing the controlled movement of the robot through four phases: feed forward control, circular motion control, closed-loop advancement, and specific pose motion. The experiments begin with simulations on RVIZ (a visual simulator) and proceed with real-time testing. The final results demonstrate high efficiency and accuracy in the TurtleBot's real-time operations.

Introduction

Intelligent control systems are increasingly vital in enhancing daily human life, with applications in collision prevention, traffic control, motor control, and other areas. This paper focuses on differential drive mobile robots, specifically the TurtleBot, which has numerous applications such as room cleaning, goods delivery, and food services. The study aims to investigate pose estimation and motion control for these robots.

Methodology

- The motion and pose of the mobile robot are controlled by its velocity, with the rotation of the robot's wheels altering its position.
- A relationship exists between the angular velocity of the wheels and the linear velocity of the robot, which is described using kinematics equations.
- Equations:
- Linear velocity equation: $v=\frac{v_r+v_l}{2}=r\left(\frac{w_r+w_l}{2}\right)$ $v=\frac{v_r+v_l}{2}=r\left(\frac{w_r+w_l}{2}\right)$

- Angular velocity equation: $w = \frac{v_r - v_l}{T} = r\left(\frac{w_r - w_l}{T}\right)$
- Where:
- v: Linear velocity of the robot.
- w: Angular velocity of the robot.
- vr: Linear velocity of the right wheel.
- wr: Angular velocity of the right wheel.
- vl: Linear velocity of the left wheel.
- wl: Angular velocity of the left wheel.
- r: Radius of the robot wheels.
- T: Distance between the two wheels.
- Pose and orientation of the mobile robot are calculated by integrating the robot's velocity components, a technique known as odometry.
- Odometry can face challenges due to slip between the ground and the robot's wheels.
- Equations for odometry:
- x-coordinate: $x = \int_{t_0}^{t} v \cos\theta \, dt + x_0$
- y-coordinate: $y = \int_{t_0}^{t} v \sin\theta \, dt + y_0$
- Orientation: $\theta = \int_{t_0}^{t} w \, dt + \theta_0$

Experimental Setup

- The experimental setup includes a TurtleBot3 differential drive mobile robot, which is small, lightweight, and 3D printed.
- A laptop with a core i7-7th generation processor, 16GB RAM, and 256GB SSD interfaces with the robot using the ROS (Robot Operating System) software.
- Communication between the TurtleBot3 and the laptop is through a wireless LAN connection.
- A Python-based platform with necessary libraries such as TensorFlow and ROS packages is used for the experiment.
- A virtual environment on RVIZ software is created, representing a map with boundaries and obstacles, and a real-time environment resembling the virtual one is also set up.

Results

- **Simulation on RVIZ Software**:
  - The experiment utilizes Gmapping to create a map in RVIZ, with the TurtleBot starting from a specific point and navigating towards an endpoint while avoiding obstacles.
  - Figures from the experiment illustrate the TurtleBot's movement from the starting point to the endpoint, with red arrows indicating the bot's direction.
- **Trajectory and Path Following:**
  - In the experiment, the TurtleBot successfully follows a desired trajectory in a real-time environment.
  - The starting point and successful completion of the desired trajectory without obstacles are depicted.
  - The actual trajectory followed by the bot (in red) aligns closely with the desired trajectory (in blue). Gray points represent obstacles in the environment.

Conclusion

- The kinematics-based motion control and pose estimation approach is useful for specific applications in controlled environments.
- The experiment involves a small control environment and utilizes a differential drive TurtleBot for pose estimation and movement control.
- The study demonstrates that the methodology yields positive results in a controlled environment, with the TurtleBot's real-time movements closely matching the simulated results.
- Overall, the approach offers good performance for the tasks outlined in the study.