Study and Development of Autopilot for space rover and tracking position using IMU

Mr.Tustong Tongnumpen (Grade 11)
[Piboonbumpen Demonstration School, Burapha University, Chonburi, Thailand]

Abstract

In this study, we aim to investigate and evaluate the rover's autopilot performance. Specifically, we want to construct a program and sensor that are inexpensive, readily available, and widely used worldwide in order to build a prototype and test the autopilot program. The outcome showed that the Rover could locate its way to the target and was on its way with Circular Error Probability (CEP) of about 0.654 meters from the target. There are still too many errors in the tracking position using the Inertial measurement unit sensor (IMU) approach alone to determine the rover's true position.

Introduction

Space exploration comes with countless unforeseen challenges, making it critical to study a planet's surface thoroughly. Using a rover is one way to reduce risks, as it allows us to explore without direct human involvement. To ensure the rover can operate effectively, it should be equipped with an autopilot system, enabling it to navigate independently and reach its destination on its own.

Methodology

After printing the body parts of the prototype using a 3D printer, we put together the Arduino Mega processors, GPS position sensor, and ultrasonic range sensor. The prototype and circuit connection show in Fig1. we conducted tests on its autopilot system. The system relies on a magnetic field sensor to orient itself toward the target. A location sensor, such as GPS, continuously updates the distance and trajectory to the destination, ensuring accurate navigation. Additionally, an ultrasonic ranging sensor enables the rover to detect and avoid obstacles in its path, allowing it to reach its target safely without collisions.

For position tracking using an IMU (Inertial measurement unit sensor), we put together BMI160[3] as IMU and ESP32[4] as transmitters to send the wireless data to another ESP32[4] that act as receiver then we calculate the distance traveled over time by applying acceleration and time in a fundamental physics equation. To enhance accuracy, a filter is applied to minimize noise in the data. The processed information is then transmitted to Unity via TCP (Transmission Control Protocol), allowing us to visualize the 3D position in real-time [1].

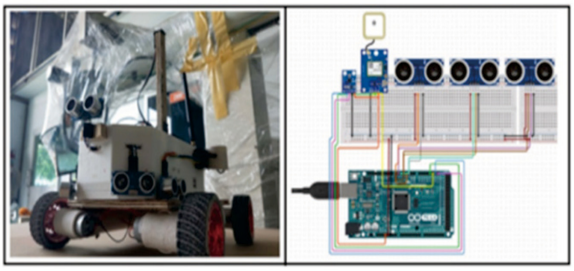
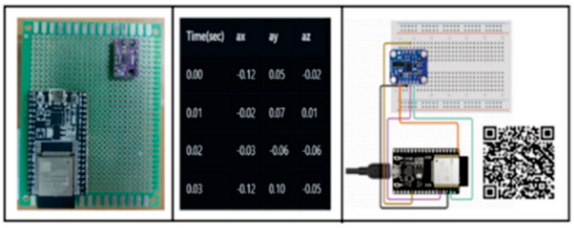


Fig 1: Shows the image of prototype and circuit connection



Time(sec)	ax	ay	az
0.00	-0.12	0.05	-0.02
0.01	-0.02	0.07	0.01
0.02	-0.03	-0.06	-0.06
0.03	-0.12	0.10	-0.05

Fig 2: Shows the image of prototype, Table of result and circuit connection for IMU tracking using BMI160

Result

After testing the autopilot system, the results showed that it could successfully navigate toward the target with a Circular Error Probable (CEP) of 0.654 meters and effectively avoid obstacles along the way. While the IMU-based position tracking could indicate the rover's direction of movement, it was not accurate enough to reliably measure the true distance traveled. The prototype, test results, and circuit connection for IMU tracking using the BMI160[3] as the IMU sensor are shown in Fig 2.

Summary

The autopilot system demonstrates sufficient precision for basic rover operations or as a cost-effective learning tool. However, hardware limitations present challenges. The magnetic field sensor requires frequent calibration, and the Arduino Mega's 16 MHz processor struggles to handle the ranging sensor and GPS simultaneously, leading to occasional crashes or obstacles.

Improving precision is possible by upgrading the GPS, which currently has a 2.5-meter error margin [2] we can use RTK (Real Time Kinematics) or PPS (Precise Positioning Service) modules. Position tracking via the IMU remains error-prone and unreliable for measuring travel distance but can indicate movement direction. Replacing it with a 9DOF (Degree of freedom) sensor may resolve these issues by compensating for external forces such as gravity.

References

[1] Xio Technologies, "Oscillatory Motion Tracking With x-IMU," GitHub Repository, [Online]. Available: https://github.com/xioTechnologies/Oscillatory-Motion-Tracking-With-x-IMU. [14-Mar-2024]
[2] u-blox, "Content Hub," u-blox AG, [Online]. Available: https://content.u-blox.com. [18-Apr-2024]
[3] BMI160 is a sensor for measuring Acceleration and Gyroscope
[4] ESP32 is a Microcontroller which have wifi integrate allow to communicate wirelessly