TESTING – Odometer and Correction

Project: Design an Autonomous Robot

Task: To design an autonomous robot that is capable of navigating to a predetermined position while avoiding obstacles and firing objects at two targets. This is to be done in the shortest time possible.

Document Version Number: 1.0

Date: March 16th

Tester: James Fuh, Yutian Zheng

Software Version: 1.1

Hardware Version: 2.0

Goal: The goal of this test is to know the accuracy of the odometer and its correction.

Table of Contents

[PURPOSE 3](#_Toc413919896)

[OBJECTIVES 3](#_Toc413919897)

[PROCEDURE 3](#_Toc413919898)

[EXPECTED RESULTS 3](#_Toc413919899)

[FORMAT OF OUTPUT REQUIRED 3](#_Toc413919900)

[SAMPLE CALCULATIONS 4](#_Toc413919901)

[TEST REPORT 4](#_Toc413919902)

[CONCLUSION 5](#_Toc413919903)

[ACTION 5](#_Toc413919904)

[DISTRIBUTION 5](#_Toc413919905)

# 

# PURPOSE

The purpose of the test is to know how accurate the odometer reading is comparing to the measured coordinates while the odometer correction is on. This is the most essential part as there are many other codes such as Navigation and Obstacle Avoidance will use odometer as the base.

This test will be done with the hardware version 2.0 and a testing code that can be found in the “Odometer & Correction v1.1” folder.

# OBJECTIVES

The objective of the test is to know if the odometer can work properly and display the right information on X/Y coordinates and the angle theta. The objective of the test on odometer correction is to investigate on how well the correction code can reduce the error when the correction mode is turned on.

# PROCEDURE

1. Place the robot at a position (0, 0) (i.e.: at an intersection of the grids).
2. Run the odometer code, and the robot will run in a square path of 60 x 60 cm.
3. Measure change in the X and Y components after the robot stops.
4. Repeat step 2) and 3) 10 times.
5. After that, turn on the correction mode, and repeat 1), 2), 3), and 4).

# EXPECTED RESULTS

The expected result is that the odometer reading will read about (0, 0) once it goes around a 60x60cm square. We want to make sure that the odometer does not display numbers that deviate too much from x & y coordinates measured. The best case of the test is that the odometer will go back to the origin and display (0 , 0)after going around the square. The worst case scenario of the test is that the odometer reading is way off from the actual x & y reading (ex: read (0.25,0.25) but the actual coordinates are (-7,-7)).

# FORMAT OF OUTPUT REQUIRED

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Odometry Correction Disabled | | | | |  |
|  | Odometer Values (cm) | | Actual Values (cm) | | Error (cm) | |
| Observation | X | Y | X | Y | X | Y |
| 1 | 0.26 | 0.26 | 6.3 | -6.2 | 6.04 | -6.46 |
| 2 | 0.25 | 0.24 | 7.0 | -6.3 | 6.75 | -6.54 |
| 3 | 0.24 | 0.25 | 4.9 | -5.5 | 4.66 | -5.75 |
| 4 | 0.24 | 0.26 | 8.5 | -5.5 | 8.26 | -5.76 |
| 5 | 0.25 | 0.26 | 7.2 | -5.7 | 6.95 | -5.96 |
| 6 | 0.24 | 0.24 | 6.6 | -5.3 | 6.36 | -5.54 |
| 7 | 0.26 | 0.24 | 7.1 | -6.2 | 6.84 | -6.44 |
| 8 | 0.26 | 0.26 | 6.4 | -6.7 | 6.14 | -6.96 |
| 9 | 0.26 | 0.25 | 6.4 | -5.9 | 4.14 | -6.15 |
| 10 | 0.25 | 0.26 | 5.7 | -6.2 | 5.45 | -6.46 |
| Mean | 0.251 | 0.252 | 6.61 | -5.46 | 6.159 | -6.202 |
| Standard Deviation | 0.00876 | 0.00919 | 0.95737 | 1.67857 | 1.18949 | 0.44444 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Odometry Correction Enabled | | | | |  |
|  | Odometer Values (cm) | | Actual Values (cm) | | Error (cm) | |
| Observation | X | Y | X | Y | X | Y |
| 1 | 0.26 | 0.09 | 6.2 | -4.1 | 5.96 | -4.19 |
| 2 | 0.25 | 0.20 | 4.4 | -4.2 | 4.15 | -4.0 |
| 3 | 0.17 | 0.26 | 5.2 | -4.4 | 5.03 | -4.7 |
| 4 | 0.24 | 0.34 | 6.2 | -4.9 | 5.96 | -5.24 |
| 5 | 0.26 | 0.26 | 6.3 | -3.9 | 6.04 | -3.66 |
| 6 | 0.23 | 0.34 | 5.4 | -4.7 | 5.17 | -5.04 |
| 7 | 0.26 | 0.25 | 6.3 | -4.3 | 6.04 | -4.55 |
| 8 | 0.23 | 0.27 | 4.7 | -5.2 | 4.47 | -5.47 |
| 9 | 0.19 | 0.34 | 5.7 | -4.0 | 5.51 | -4.34 |
| 10 | 0.26 | 0.27 | 6.3 | -4.7 | 6.04 | -4.97 |
| Mean | 0.235 | 0.262 | 5.67 | -4.44 | 5.437 | -4.616 |
| Standard Deviation | 0.03171 | 0.07598 | 0.715 | 0.42216 | 0.70487 | 0.57602 |

# SAMPLE CALCULATIONS

The mean value can be obtained by the following formula:

For the mean of error of X with odometer correction, cm

In this test, the mean values for error of x and y are 5.437 and -4.616 cm, respectively.

The standard deviation can be obtained as

In this test, the standard deviation for x-axis and y-axis are respectively 0.70487 and 0.57602 cm**.**

# TEST REPORT

The collected results are acceptable because the error of x and y is within 4 to 6 cm off, and the standard deviation values are low when the correction mode is turned on, 0.70487 and 0.57602 cm for each axis respectively.

As the testing goes along, we have noticed that sometimes the light sensor does not detect some of the gridlines accurately. It means that we need to adjust the threshold constant for the gridline in the code accordingly.

# CONCLUSION

This testing is considered “failed” as there are still some rooms of improvement to improve the accuracy of the odometer and the light sensor. We should reduce the error of the odometer to at least within 3 cm off, and change the light constant for the gridline accordingly so that the correction mode can function properly as soon as the light sensor detects a gridline.

Also the correction mode should be able to deal with the case when the light sensor detects the intersection of gridline and adjust the odometer reading accordingly.

# ACTION

The codes for the odometer correction need to be revised in order to reduce the errors and to deal with the case when the robot is on the intersections of the gridlines.

Gridline constant need to be changed in order for the correction mode to adjust the reading.

# DISTRIBUTION

This testing belongs to the software development.