PROBLEM 3

Table of Contents

..... 1
Biases 1
Local gravitational acceleration 1
Accelerometers 2
Magnetometers 2
Remove bias 2
Pitch and Roll 2
Heading 3


```
clear variables; close all; clc

data_table_acc = readtable('Accelerometer_pitch_roll_head.csv');
data_table_mag = readtable('Magnetometer_pitch_roll_head.csv');
```

```
Warning: Column headers from the file were
modified to make them valid MATLAB
identifiers before creating variable names
for the table. The original column headers
are saved in the VariableDescriptions
property.
Set 'PreserveVariableNames' to true to use
the original column headers as table
variable names.
Warning: Column headers from the file were
modified to make them valid MATLAB
identifiers before creating variable names
for the table. The original column headers
are saved in the VariableDescriptions
property.
Set 'PreserveVariableNames' to true to use
the original column headers as table
variable names.
```

Biases

```
bias_mag = [-12.11214937;
            -19.67616054;
            22.73696197];
bias_acc = [1.948244375;
            1.926002907;
            -3.760826138];
```

Local gravitational acceleration

```
g = 9.80333; % m/s/s
```

Accelerometers

```
acc_x = data_table_acc{:, 2};
acc_y = data_table_acc{:, 3};
acc_z = data_table_acc{:, 4};
```

Magnetometers

```
mag_xb = data_table_mag{:, 2};
mag_yb = data_table_mag{:, 3};
mag_zb = data_table_mag{:, 4};
```

Remove bias

```
acc_x_wo_bias = acc_x - bias_acc(1);
acc_y_wo_bias = acc_y - bias_acc(2);
acc_z_wo_bias = acc_z - bias_acc(3);

mag_xb_wo_bias = mag_xb - bias_mag(1);
mag_yb_wo_bias = mag_yb - bias_mag(2);
mag_zb_wo_bias = mag_zb - bias_mag(3);
```

Pitch and Roll

```
roll_data = atan( acc_y_wo_bias ./ acc_z_wo_bias );
pitch_data = asin( acc_x_wo_bias / g );

roll_mu  = mean( roll_data )
disp('rad')
pitch_mu = mean( pitch_data )
disp('rad')

roll_var = var(roll_data)
disp('rad')
pitch_var = var(pitch_data)
disp('rad')

fprintf('The roll angle is %f deg +/- %f deg\n', ...
 roll_mu*180/pi, 3*sqrt(roll_var)*180/pi);


fprintf('The pitch angle is %f deg +/- %f deg\n', ...
 pitch_mu*180/pi, 3*sqrt(pitch_var)*180/pi);


roll_mu =

    0.0210

rad
```

```
pitch_mu =

   -0.2017

rad

roll_var =

   2.6802e-07

rad

pitch_var =

   3.2389e-07

rad
The roll angle is 1.203306 deg +/- 0.088988 deg
The pitch angle is -11.556843 deg +/- 0.097824 deg
```

Heading

```
tmp1 = [...
 cos(pitch_mu) sin(pitch_mu)*sin(roll_mu)
 sin(pitch_mu)*cos(roll_mu); ...
 0 cos(roll_mu) -sin(roll_mu); ...
 -sin(pitch_mu) cos(pitch_mu)*sin(roll_mu) cos(pitch_mu)*cos(roll_mu)]
 * ...
 [mag_xb_wo_bias'; mag_yb_wo_bias'; mag_zb_wo_bias'];

mag_x_wo_bias = tmp1(1,:)';
mag_y_wo_bias = tmp1(2,:)';

magnetic_heading_data = -atan2( mag_y_wo_bias, mag_x_wo_bias );
declination    = -14.07*pi/180;
% Declination  for Worcester, MA found using World Magnetic Model
% https://www.ngdc.noaa.gov/geomag/calculators/
magcalc.shtml#declination

true_heading_data = declination + magnetic_heading_data;
true_heading_mu  = mean(true_heading_data)
disp('rad')
true_heading_var = var(true_heading_data)
disp('rad')

fprintf('The heading is %f deg +/- %f deg\n', ...
 true_heading_mu*180/pi, 3*sqrt(true_heading_var)*180/pi);

disp('This is a valid result as the device orientation matched the
 calculated values')


true_heading_mu =
```

```
   -1.2314

rad

true_heading_var =

   1.1804e-04

rad
The heading is -70.552140 deg +/- 1.867522 deg
This is a valid result as the device orientation matched the
 calculated values
```

Published with MATLAB® R2020a