GEOMAGNETIC COMPENSATION FOR
LOW-COST CRASH AVOIDANCE
PROJECT

A Thesis
presented to
the Faculty of California Polytechnic State University,
San Luis Obispo

In Partial Fulfillment
of the Requirements for the Degree
Master of Science in Mechanical Engineering

by
John C. Torres
April 2011
COMMITTEE MEMBERSHIP

TITLE: Geomagnetic Compensation for Low-Cost Crash Avoidance Project

AUTHOR: John C. Torres

DATE SUBMITTED: April 26, 2011

COMMITTEE CHAIR: Charles Birdsong Ph.D., Associate Professor

COMMITTEE MEMBER: John Ridgely Ph.D., Associate Professor

COMMITTEE MEMBER: Hemanth Porumamilla Ph.D., Assistant Professor
ABSTRACT

Geomagnetic Compensation for Low-Cost Crash Avoidance Project

John C. Torres

The goal of this work was to compensate for the effects of the Earth’s magnetic field in a vector field magnetic sensor. The magnetic sensor is a part of a low-cost crash avoidance system by Stephane Roussel [25] where the magnetic sensor was used to detect cars passing when it was mounted to a test vehicle. However, the magnetic sensor’s output voltage varied when it changed orientation with respect to the Earth’s magnetic field. This limited the previous work to only analyze detection rates when the test vehicle travelled a single heading. Since one of the goals of this system is to be low-cost, the proposed solution for geomagnetic compensation will only use a single magnetic sensor and a consumer-grade GPS. Other solutions exist for geomagnetic compensation but use extra sensors and can become costly.

In order to progress the development of this project into a commercial project, three separate geomagnetic compensation algorithms and a calibration procedure were developed. The calibration procedure compensated for the local magnetic field when the magnetic sensor was mounted to the test vehicle and allowed for consistent magnetic sensor voltage output regardless of the type of test vehicle.

The first algorithm, Compensation Scheme 1 (CS1), characterized the local geomagnetic field with a mathematical function from field calibration data. The GPS heading was used as the input and the output is the voltage level of the Earth’s magnetic field. The second algorithm, Compensation Scheme 1.5, used a mathematical model of the Earth’s magnetic field using the International Geomagnetic Reference Field. An algorithm was developed to take GPS coordinates as an input and output the voltage contributed by the mathematical representation of the Earth’s magnetic field. The output voltages from CS1 and CS1.5 were subtracted from the calibrated magnetic sensor data. The third algorithm, Compensation Scheme 2 (CS2), used a high pass filter to compensate for changes of orientation of the magnetic sensor. All three algorithms were successful in compensating for the geomagnetic field and vehicle detection in multiple car headings was possible.

Since the goal of the magnetic sensor is to detect vehicles, vehicle detection rates were used to evaluate the effectiveness of the algorithms. The individual algorithms had limitations when used to detect passing cars. Through testing, it was found that CS1 and CS1.5 algorithms were suitable to detect vehicles while stopped in traffic while the CS2 algorithm was suitable vehicle detection while the test vehicle is moving.

In order to compensate for the limitations of the individual algorithms, a fused algorithm was developed that used a combination of CS1 and CS2 or CS1.5 and CS2. The vehicle speed was used in order to determine which algorithm to use in order to detect cars. Although the goal of this project is not vehicle detection, the rate of successful vehicle detection was used in order to evaluate the algorithms.
The evaluation of the fused algorithm demonstrated the value of using CS1 and CS1.5 to detect vehicles when stopped in traffic, which CS2 algorithm cannot do. For a study conducted in traffic, using the fused algorithm increased vehicle detection rates by 51%-62% from using the CS2 algorithm alone.

Since this work successfully compensated for geomagnetic effects of the magnetic sensor, the low-cost crash avoidance system can be further developed since it is no longer limited to driving in a single direction. Other projects that experience unwanted geomagnetic effects in their projects can also implement the knowledge and solutions used in this work.
ACKNOWLEDGEMENTS

This work would not have been possible without the support and contributions from my committee members. I would like to thank Dr. Charles Birdsong for giving me an opportunity to work on this interesting and challenging project related to automobile safety. Not only did he provide support and guidance in this project, but has also contributed his time and knowledge in my career search. His patience and encouraging attitude have helped me complete this project and start my career. I would also like to thank Dr. John Ridgely and Dr. Hemanth Porumamilla for providing their expertise through feedback on this thesis work and my presentation defense. My education has certainly been an excellent experience with professors like these.

I would also like to thank Derek Unland for contributing some of his time and guidance in the initial start of this project. I must also thank Chuck Keezer for fixing the power inverter and Jim Gerhardt for providing access to the test vehicle.

Finally, I would like to thank my friends and family for their support. The frozen home-cooked meals have done their part to nourish my body throughout this project.
# TABLE OF CONTENTS

LIST OF TABLES ........................................................................................................................................... X

LIST OF FIGURES ......................................................................................................................................... XI

1. INTRODUCTION ........................................................................................................................................ 1
   1.1 PROJECT BACKGROUND ...................................................................................................................... 1
   1.1.2 Other Magnetic Sensor Applications Affected by the Earth’s Magnetic Field ...................... 6
   1.2 CONCEPT AND INNOVATION ......................................................................................................... 8

2. BACKGROUND INFORMATION .............................................................................................................. 10
   2.1 STATE OF THE ART OF MAGNETIC SENSORS ............................................................................. 10
      2.1.1 Vector Magnetometers ................................................................................................................. 10
      2.1.2 Total Field Magnetometers ........................................................................................................... 14
   2.2 OVERVIEW OF THE EARTH’S MAGNETIC FIELD ........................................................................ 14
      2.2.1 The Magnetic Field as Vectors ..................................................................................................... 15
      2.2.2 Mathematical Model of the Geomagnetic Field ........................................................................... 18
      2.2.3 Variations of the Earth’s Magnetic Field ..................................................................................... 20

3. LITERATURE REVIEW ............................................................................................................................. 22
   3.1 STATIONARY APPLICATIONS OF MAGNETIC SENSORS ............................................................. 22
   3.2 NAVIGATION PURPOSES ................................................................................................................... 26
      3.2.1 Magnetic Sensor Considerations .................................................................................................. 26
      3.3 Compensation for the Geomagnetic Field ....................................................................................... 38

4. PROBLEM STATEMENT AND SOLUTION CONCEPT ........................................................................ 41

5. PRESENTATION OF WORK .................................................................................................................... 48
   5.1 EQUIPMENT ....................................................................................................................................... 48
   5.2 ASSUMPTIONS ................................................................................................................................... 49
5.2 PRELIMINARY INVESTIGATION ................................................................................. 52
  5.2.1 Part I Experiment ................................................................................................. 52
  5.2.2 Part II Experiment ............................................................................................... 57
  5.2.3 Z-Axis Results ..................................................................................................... 60
  5.2.4 Part I and Part II Conclusion .............................................................................. 61
5.3 COMPENSATION SCHEME 1: “GPS COMPENSATION” ......................................... 61
  5.3.1 Sensor Calibration .............................................................................................. 62
  5.3.2 GPS Compensation Algorithm ............................................................................ 75
  5.3.3 Compensation Scheme 1 Limitations ..................................................................... 78
5.4 COMPENSATION SCHEME 1.5: “IGRF COMPENSATION” ...................................... 82
  5.4.1 IGRF Compensation Algorithm .......................................................................... 83
  5.4.2 Compensation Scheme 1.5 Limitations ................................................................. 86
5.5 COMPENSATION SCHEME 2: HIGH-PASS FILTER .............................................. 87
  5.5.1 Preliminary Investigation ..................................................................................... 87
  5.5.2 Hi Pass Filter Algorithm ...................................................................................... 91
  5.5.3 Compensation Scheme 2 Limitations .................................................................... 94
5.6 EVALUATION OF ALGORITHMS .......................................................................... 97
  5.6.1 Proposed Compensation Scheme Fusion ............................................................... 97
  5.6.2 Evaluation Method .............................................................................................. 99
  5.6.3 Evaluation Results .............................................................................................. 104
  5.6.4 Sources of Error .................................................................................................. 109
6. CONCLUSIONS ............................................................................................................ 114
  6.1 GENERAL SUMMARY AND CONCLUSIONS ...................................................... 114
  6.2 FUTURE WORK ...................................................................................................... 117
REFERENCES .................................................................................................................... 119
APPENDIX: MATLAB PROGRAM AND DOCUMENTATION ...................................... 122
CALIBRATION ERROR REFINEMENT ......................................................... 123
GPS COMPENSATION ALGORITHM....................................................... 127
GPS HEADING INTERPOLATION ALGORITHM ....................................... 132
HI-PASS FILTER COMPENSATION ....................................................... 137
IGRF COMPENSATION ........................................................................ 141
MAGNETIC LOW PASS FILTER ............................................................ 148
MAIN PROGRAM ................................................................................. 151
MAGNETIC SENSOR CALIBRATION ....................................................... 158
LIST OF TABLES

Table 1- Merits and demerits of compensation schemes ................................. 97
Table 2- Algorithm evaluation results ................................................................. 105
LIST OF FIGURES

Figure 1- Magnetic voltage fluctuations of the test vehicle turning relative to the Earth’s magnetic field compared to magnetic voltage signatures of cars passing the magnetic sensor .......................................................... 2

Figure 2- Geomagnetic Field Vector Components [1] ........................................... 16

Figure 3- Magnetic sensor behavior during a vehicle pass [5] ............................... 23

Figure 5- X and Y Magnetic Sensor Response to Earth's Magnetic Field .................. 31

Figure 6- Magnetic Circle for a Calibration Run without Magnetic Disturbances ......... 33

Figure 7- Part I: Turning the magnetic sensor 360° about the z-axis ......................... 53

Figure 8- Magnetic Sensor voltages as a function of heading direction from North, CCW. Raw Data ................................................................. 54

Figure 9- Results of Figure 8 with averages shifted to 2.5V ..................................... 56

Figure 10- Turning the magnetic sensor 360° 3ft away from a center point ............... 58

Figure 11- Part II Magnetic Sensor voltages as a function of heading direction from North, CCW. Raw Data ......................................................... 59

Figure 12- Part II Adjusted Data ........................................................................ 60

Figure 13- Calibration movement for test vehicle .................................................. 62

Figure 14- Calibration procedure location. GPS map provided by GPSvisualizer.com .. 63

Figure 15- Uncalibrated magnetic sensor data during calibration procedure .......... 64

Figure 16- Calibrated versus uncalibrated magnetic sensor data ............................ 65

Figure 17- Magnetic circle for calibration procedure ............................................. 67

Figure 18- Magnetic Circle with large y-axis sensitivity compared to the x-axis ...... 68

Figure 19- Spline interpolation of known GPS headings ........................................ 70

Figure 20- GPS heading interpolation of calibration procedure ............................. 72

Figure 21- Calibration data: geomagnetic field as a function of heading degree ...... 74

Figure 22- CS1 Algorithm ................................................................................. 75

Figure 23- Headings and path travelled for experiment on 10/25/10. Maps provided by GPSvisualizer.com ................................................................. 76
Figure 24- Compensation Scheme 1 Results ................................................................. 77
Figure 25- GPS limitation: effects of a sample and hold method for GPS heading ....... 79
Figure 26- GPS limitation- temporary signal loss ...................................................... 81
Figure 27- CS1.5 Algorithm ...................................................................................... 84
Figure 28- CS1.5 Algorithm Results. CS1.5 is in red and CS1 is in blue. .................. 85
Figure 29- Spectral analysis plot for the test vehicle travelling 18-20 degrees per second during calibration procedure ............................................................. 88
Figure 30- Effects of hi-pass cutoff frequency on experimental data ....................... 89
Figure 31- Attenuation of voltage fluctuation of the test vehicle turning at an index of approximately 4.7x10^5 ................................................................. 90
Figure 32- Effects of increasing cutoff frequency attenuating a signal .................... 91
Figure 33- CS2 Algorithm ......................................................................................... 92
Figure 34- Results of CS2 algorithm compared to uncompensated data and CS1 algorithm ........................................................................................................ 93
Figure 35- Comparison of CS2 and CS1 when approaching and stopping next to a truck at a red light ................................................................. 95
Figure 36- Fused compensation scheme algorithm .................................................. 98
Figure 37- Boxplot of voltage ranges when vehicle is stopped using CS1 algorithm .... 100
Figure 38- Boxplot of voltage ranges when a vehicle is present and absent using CS2 algorithm ........................................................................................................ 101
Figure 39- Algorithm signals when a car passes at a close distance ......................... 111
Figure 40- Signals from compensation algorithms from a car passing far away........ 112
Figure 41- MATLAB program flowchart ................................................................ 122
1. INTRODUCTION

1.1 PROJECT BACKGROUND

This project was a continuation of Stephane Roussel’s Master’s Thesis on Sensor Integration for Low-Cost Crash Avoidance Project [25]. In [25], a three-axis magnetoresistive sensor (with an X, Y, and Z axis) and ultrasonic sonars were used to detect the probability of a car present in an adjacent lane of the vehicle with the crash avoidance system mounted on it, which will be referred to as the test vehicle.

While the crash avoidance system developed in [25] showed successful rates of car detection using the two different sensors and a Bayesian filter, the system was limited to predicting the presence of a car when the vehicle travelled in a single heading direction with respect to magnetic North.

When the vehicle turned, the fluctuations of the Earth’s magnetic field caused relatively large voltage fluctuations of a range of about ±0.2 volts per magnetic axis. These large fluctuations of voltage due to the Earth’s magnetic field are large relative to the voltage signals produced when a car passes the magnetic sensor, as in Figure 1. Typical magnetic sensor voltage on one axis (x-axis) of a class 8 truck is 0.04V [25].
Figure 1- Magnetic voltage fluctuations of the test vehicle turning relative to the Earth’s magnetic field compared to magnetic voltage signatures of cars passing the magnetic sensor

The large fluctuations in voltages in Figure 1 are due to the magnetic sensor axis aligning with magnetic North and South when the test vehicle is driven in multiple heading directions. The thesis work presented in this document found that when a magnetic sensor axis aligned with magnetic North, a maximum positive voltage is superimposed on top of the voltage signals produced by other magnetic sources in the environment. When a magnetic sensor axis aligns opposite of magnetic North (magnetic South), a maximum negative voltage (the minimum voltage) would be superimposed on the original signal caused by magnetic anomalies in the environment. When the magnetic sensor axis orients itself in a direction in between magnetic North and South, a voltage signal related to the heading direction is superimposed on the original signal.
In order to compensate for voltage fluctuations caused by the magnetic sensor turning within the Earth’s magnetic field, the previous work in [25] only conducted experiments where the test vehicle drove in one heading so that the Earth’s magnetic field contributed a constant voltage to the magnetic sensor. This constant voltage is taken as the baseline to compare additional voltage signals from detected vehicles.

In order to detect either a car or a truck passing the test vehicle, a formulation of voltage thresholds using the magnitudes of each magnetic sensor axes were created [25]:

\[
|B_x| + |B_y| + |B_z| \tag{1}
\]

\[
B_x^2 + B_y^2 + B_z^2 \tag{2}
\]

Where \(B_x, B_y, \) and \(B_z\) are the raw magnetic sensor axis voltage measured directly from the sensor for the x, y, and z-axes, respectively.

If the sensor was placed in a location free from magnetic sources, the values of (1) and (2) will be larger when a vehicle is present compared to a situation when there are no vehicles present, due to the ferromagnetic material in a vehicle. Therefore, a voltage threshold was used to signify when a vehicle is present when the values for (1) and (2) exceeded the threshold value.
However, because the Earth’s magnetic field changed depending on the magnetic sensor axis orientation, the total voltage magnitude of a passing vehicle changed depending on the test vehicle’s orientation to magnetic North. For example, using Equations (1) and (2) to measure a passing vehicle’s magnetic signature will differ depending on the orientation of magnetic sensor’s axis with magnetic North.

Assume that a car passes the test vehicle from South to North. If the magnetic sensor axis was mounted so that a sensitive axis was aligned with magnetic North, the overall magnitude of (1) and (2) from the vehicle passing will be large. Conversely, the signal will be small if the magnetic sensor was mounted with the sensitive axis aligned with magnetic South during the same vehicle pass.

In the work of Roussel in [25], a change in voltage using Equations (1) and (2) was used to characterize the threshold level. If the change in voltage were greater than a specified threshold, a car would be detected. However, due to the superimposed magnetic field of the Earth, choosing a reference voltage level cannot be done due to the magnetic sensor’s dependence on its orientation within the Earth’s magnetic field. For example, while a magnetic sensor axis is aligned with magnetic North, the reference voltage level would be higher than a situation where the magnetic sensor is aligned with Magnetic South. Thus, a single voltage-change threshold cannot be characterized due to its heading dependence.
In addition, the magnetic sensor readings were not calibrated to compensate for local magnetic field effects of the soft and hard iron components of the test vehicle. It is known that a vehicle body’s magnetization varies with location, time, environment, and operation of onboard electronics \[^{15}\]. The test vehicle’s own magnetization must be compensated for to get a consistent magnetic voltage baseline that could be used to measure a change in voltage due to a passing vehicle’s magnetic signature, regardless of the magnetic material located inside the test vehicle. Because a calibration procedure was not implemented into the crash avoidance system, a certain vehicle detection voltage threshold cannot be defined because it may vary during each operation of the magnetic sensor with the crash avoidance system.

Since the system was developed for heavy-duty semi-trucks, magnetic sensor calibration is crucial because the truck may carry different loads that may create large disturbances in the local magnetic field. The ambient magnetic field sensed by a magnetic sensor will be large if the truck is carrying a load of iron and steel objects compared to a truck carrying tomatoes. Without a calibration procedure, a single detection threshold cannot be defined because it depends on the magnetic signature of the vehicle.
The problems of the Earth’s magnetic field and lack of magnetic sensor calibration in [25] has limited the previous work to travelling in a single direction and having a magnetic sensor voltage detection range that varies during each new trial, which will need to be recoded in the program. Since the project’s ultimate goal is to be a low-cost system for Class 8 trucks, these limitations have hindered the progress of the development of the project. The purpose of this thesis is to develop a system that will address the Earth’s magnetic field and develop a sensor calibration algorithm using only the equipment that was used in the previous project to keep the system low-cost.

1.1.2 Other Magnetic Sensor Applications Affected by the Earth’s Magnetic Field

Other magnetic sensor applications exist where the Earth’s magnetic field, or geomagnetic field, is either used as a reference or is viewed as an unwanted component that must be compensated for.

One application that uses magnetic sensors is compassing. Some specific applications focus on the magnetic sensor itself for compass headings and navigation [15] [4]. In these works, the geomagnetic field is considered as a reference to find the sensor’s orientation with respect to magnetic and geographical North. Therefore, the voltage fluctuations of the geomagnetic field are of interest. Furthermore, a map of ambient magnetic readings to determine the location of magnetic North of a certain regional area in may be of interest to geophysical science organizations such as the Australian Geological Survey Organization [16].
However, some other applications do not desire the geomagnetic field’s influence in their magnetic sensor data and must be removed from a sensor’s readings. For example, an airplane flying over an area is used to record magnetic anomalies in the hills and mountains and does not want the geomagnetic field affecting the magnetic signature of geographical features. In these cases, commercial software may be purchased to remove the geomagnetic field from magnetic sensor readings [11].

Similarly, there are applications of magnetic sensors for compassing and magnetic anomaly detections in unmanned under water vehicles [1][27]. As the vehicle moves underwater, the heading of the vehicle changes with respect to magnetic North. The Earth’s magnetic field will superimpose an additional voltage on the magnetic sensor as a function of heading, which can distort the magnitudes of the voltages the sensor records from ferrous objects or features underwater. In this case, the geomagnetic field contribution should be subtracted from the readings.
In addition, other applications exist where magnetic anomalies, separate from the geomagnetic field, are of interest, such as wearable handwriting input devices using magnetic sensors [12]. In this application, a magnet mounted on a finger generated a magnetic field while the finger wrote letters in a plane. Magnetic sensors mounted at a certain point, such as the wrist, tracked the path the fingers traversed as the magnet mounted on the finger moved towards and away from the magnetic sensor. If the magnetic sensors are mounted on the wrist, the wrist may move in space and the geomagnetic field will superimpose a voltage on the magnetic sensor as a function of the wrist’s orientation to magnetic North. The magnetic sensor will inaccurately sense a path that differs from the true path that the finger traversed which will not generate the correct letter that the user intended to draw. Since only the magnitude of the magnetic field of the magnet is of interest as it moves through space, the geomagnetic field should be subtracted from the voltage readings in order to get accurate tracking measurements.

Although these projects provided solutions for compensation of the geomagnetic field, they are not formulated and optimized for a low-cost vehicle detection system that this thesis work is based upon.

1.2 CONCEPT AND INNOVATION

This thesis will document the background and technical information about the geomagnetic field and magnetic sensor calibration, describe the process of constructing a solution, and present and evaluate the solution.
The solution will be presented as an algorithm that compensated for the geomagnetic field along with a calibration procedure. For the purposes of this thesis, the algorithm was implemented as a MATLAB program for post-processing data that was gathered by a test vehicle. The scope of this project will not cover any real time implementation of this algorithm since it is an ongoing project. However, the algorithm was developed so it can be implemented in real time and the MATLAB source codes are available with this work in the Appendix: MATLAB Program and Documentation.

This solution will differ from other works by only employing a consumer-grade GPS and a single magnetic sensor to keep costs at a minimum. In order to adapt these hardware limitations to vehicle detection applications, the magnetic sensor data was processed using two different algorithms that will be based on vehicle speed. Two different algorithms are used in order to compensate for limitations encountered when one algorithm is used alone.

Implications of the solution in this work can help progress the development of the low-cost crash avoidance system to a commercial product. The system will no longer be limited to detecting cars only when it drives in one direction, but will be able to detect cars when the sensor is oriented in any direction with respect to the geomagnetic field. Furthermore, the results of this work can be applied to other magnetic anomaly detection projects where the geomagnetic field is not a concern, such as an ongoing magnetic anomaly detection project where a magnetic sensor will be used to detect weapons in vehicles.
2. BACKGROUND INFORMATION

2.1 STATE OF THE ART OF MAGNETIC SENSORS

A brief review of the most common types of sensors will be given for reference information to the type of sensor used, in addition to providing additional information on possible alternative magnetic sensors for future development of the low-cost crash avoidance system. The information in this brief review can be found in [18] along with more details and a variety of other sensors not listed. The focus will be on vector magnetometers along with one example of a total field magnetometer.

2.1.1 Vector Magnetometers

Vector magnetometers measure the geomagnetic field with at least one axis that is sensitive in one direction, so that it has a positive and negative magnitude depending on the orientation of the axis with the magnetic field. Disturbances in the localized magnetic field can be characterized with a direction and magnitude. Although vector magnetometers supply more information than total field magnetometers with regards to direction and magnitude, they are susceptible to rotational vibrations that could rotate or move the sensor’s sensitive axes if it is not mounted securely or if exposed to a jarring motion.
• **Search coil magnetometer**

A voltage is generated in a search coil magnetometer when a magnetic flux through a coiled conductor changes. The voltage is proportional to the rate of change of the magnetic flux, which is dependent on the ambient magnetic field. These magnetometers can detect fields as weak as \(2 \times 10^{-5}\) nano-Teslas (nT) and are used in applications such as proximity sensors.

• **Fluxgate Magnetometers**

Fluxgate magnetometers are comprised of a ferromagnetic material with a drive and a sense coil. The drive coil magnetically saturates the core, thus increasing its reluctance. The AC current causes an alternating pattern of attraction and lack of attraction to generate magnetic flux to cut the sense coil. The voltage output is then a function of the harmonics of the excitation frequency. When a magnetic field is present, the rate of magnetic attraction changes. Modern fluxgate magnetometers achieve lower power consumption than their predecessors by utilizing a small amount of magnetic hysteresis.
• **Superconductor Magnetometer (SQUID)**

These magnetometers are the most sensitive vector magnetometers that can detect magnetic fields less than 1Hz. Superconducting Quantum Interface Devices (SQUID) uses the principles of superconducting materials to detect magnetic fields and often use a toroid-shaped ring made from lead or niobium. When a magnetic flux is threaded through the ring the induced current would theoretically be conducted without resistance. The current is measured at a small constriction or point contact junction in the ring and would then indicate the ambient magnetic field intensity.

• **Hall Effect Sensors**

When an electric current is passed across a rectangular conductor in the presence of a magnetic field, a voltage difference is observed between the ends of the conductor. This is due to the electron being pushed by a Lorentz force that aligns electrons to one side of the conductor.
• Magnetoresistive Magnetometers

The sensor used in this project, the Honeywell HMC2003, is an anisotropic magnetoresistive sensor (AMR). Magnetoresistive sensors change their resistance when there is an external magnetic field present. AMR sensors are usually made of Ni-Fe alloy material due to its large magnetoresistance and its compatibility with modern solid-state fabrication techniques. The resistance of the sensor depends on the angle between the magnetization of the alloy and the direction of current flow. The magnetization of the alloy will align with the direction of the magnetic field. Resistance of the alloy decreases as the direction of magnetization rotates away from the direction of current flow. The resistance is the lowest when magnetization is perpendicular to the current flow. AMR sensors are well suited to the application of this project because they have a sensitivity of $10^3$ nT to $5 \times 10^6$ nT with a minimum detectable field of 0.1 nT, which can detect the magnetic signature of vehicles. They also have low power consumption of 0.1 to 0.5 mW and can operate in temperatures between -55°C to 200°C, which fits the operating environments of the low-cost crash avoidance system.
2.1.2 Total Field Magnetometers

Unlike vector field magnetometers, total field magnetometers are insensitive to rotational vibrations.

- **Optically Pumped Magnetometers**

  Optically pumped magnetometers are based on the Zeeman effect, which uses an atom’s electron configurations to detect the strength of a magnetic field. Some electron configurations have the same energy levels but can change when a magnetic field is present. These magnetometers usually contain a gas medium, such as cesium or helium in a cylinder, which is exposed to light at a certain wavelength. In cesium’s valence electrons, there are two at a lower energy level and one at a higher energy level. When a specific wavelength of light is shined into the gas, it causes the electrons at the lower energy level to move to a higher energy level. Eventually, the electrons that moved to the higher energy level will go back down to a lower energy state than their original level. Low wavelength RF signals are then used to shift the electrons back to their original energy state. The energy used to generate the frequency required to do this will vary with the ambient magnetic field. The energy required for this operation is dependent on the direction of the ambient magnetic field with respect to the direction of the applied light and RF power, which can cause signal loss if the magnetometer is oriented improperly with respect to a magnetic field [10] [18].

2.2 OVERVIEW OF THE EARTH’S MAGNETIC FIELD

A brief overview of the Earth’s Magnetic field will be given, in order to give background information relevant to this work.
2.2.1 The Magnetic Field as Vectors

The Earth’s magnetic field is based on the magnetic North and South Poles that create a planetary dipole. It has been known that the geographic North Pole is actually a magnetic south pole and that the geographic South Pole is a magnetic north pole [2]. Since opposite charges attract and the fact that the “north” end of a magnet is attracted towards the North Pole, the North Pole is magnetically a south pole. Due to the nature of a dipole’s magnetic field lines, the Earth’s magnetic field is always pointing towards the North. However, the magnetic vector is pointing down towards the Earth in the northern hemisphere, horizontal at the equator, and up away from the Earth in the southern hemisphere [4]. The geomagnetic field can be viewed as a sum of vector components with an x, y, and z component as in Figure 2.
The x-axis of the geomagnetic field will point to geographic North and the y-axis will be pointing to the right towards geographic East. If a location to the West is referenced, the vector will have a minus (-) sign associated with it. Together, the x and y plane will form the horizontal field component, H, and will be parallel to the surface of the earth. The vertical field component, z, is a vector that will be perpendicular with the Earth’s surface. In the northern hemisphere, this component will point downwards. The x, y, and z component together will be a vector referenced as F, the total field as in Equation (3).

\[ F = \sqrt{x^2 + y^2 + z^2} \] (3)
The total field, $F$, will be in the horizontal field plane at the equator, and will have a large vertical magnitude in places near the poles. The angle that $F$ makes with horizontal plane $H$ is the inclination angle, $I$. Because $F$ points towards and away from the center of the earth at the poles, the inclination angle will be large at the poles, and small near the equator [2].

Of particular interest to this project is the angle between magnetic and geographic North, which is the declination angle, $D$. This declination angle varies by location and is about $13^\circ$ in the San Luis Obispo, California area as obtained from GPS data.

The geomagnetic field can be either referenced the X, Y, and Z coordinates (XYZ system) or in the horizontal magnitude towards magnetic North, $H$, magnetic declination, $D$, and vertical component, $Z$ (HDZ system). Geometric relations can be used to relate the two systems together [2] and are displayed in Equations (4) through (7).

\[
H = \sqrt{X^2 + Y^2} \quad (4)
\]

\[
X = H \cos(D) \quad (5)
\]

\[
Y = H \sin(D) \quad (6)
\]

\[
l = \tan^{-1} \frac{Z}{H} \quad (7)
\]
2.2.2 Mathematical Model of the Geomagnetic Field

The geomagnetic field can be modeled mathematically, from the negative gradient of a scalar potential function [2] [7]:

\[ B = -\nabla V \]  \hspace{1cm} (8)

Where:

- **B** = Vector components of the geomagnetic field
- \( V \) = Scalar potential function

A solution for \( V \), the Earth’s magnetic main field, was developed by Gauss in 1838, and is represented in spherical coordinates, as [2] [7]:

\[ V(r, \theta, \phi) = a \sum_{n=1}^{\infty} \left( \frac{a}{r} \right)^{n+1} \sum_{m=0}^{n} \left[ g_n^m \cos(m\phi) + h_n^m \sin(m\phi) \right] P_n^m(\theta) \]  \hspace{1cm} (9)

Where:

- \( r \) = Radius at a certain point in earth (km)
- \( \theta \) = Co-latitude (90° - latitude)
- \( \phi \) = Longitude
- \( n \) = Degree, a value greater than 1
- \( m \) = Order, less than or equal to \( n \)
- \( a = \) Mean Earth Radius, 6371.2 km
- \( P_n^m = \) Legendre Polynomials
- \( g_n^m, h_n^m = \) Gauss Coefficients (constants)
The Legendre Polynomials represent sinusoidal-like oscillations which changes as the colatitude goes from 0° to 360°. Equation (9) is similar to a Fourier series in that it is an infinite summation of sine and cosine functions that make up the geomagnetic field representation. Detailed computations of (9) can be found in [2] [7].

The corresponding X, Y, and Z components of the field are computed by taking the negative divergence of (9) in spherical coordinates as in Equation (8) [2]:

\[
X(\theta, \phi) = \frac{1}{a} \frac{\delta V}{\delta \theta} \\
Y(\theta, \phi) = -\frac{1}{a \sin \theta} \frac{\delta V}{\delta \phi} \\
Z(\theta, \phi) = \frac{\delta V}{\delta r}
\]

(10) (11) (12)

The Gauss coefficients, \(g_n^m\) and \(h_n^m\), are calculated and chosen by the International Association of Geomagnetism and Aeronomy (IAGA). The best Gauss coefficients are chosen to represent every epoch, which is about a five-year time span, and compiled in the International Geomagnetic Reference Field (IGRF). The tabulated data is available for download from the IAGA website [19].
Using this data, the geomagnetic field can be calculated at any point on earth using Equations (9), (10), (11), and (12). When corrections are made to the IGRF for previous years, it is called the Definitive Geomagnetic Reference Field (DGRF). The Gauss coefficients listed in the data are in nano-Teslas and the last column, SV, is the secular variation, which is the rate of annual change from the previous epoch to the latest model. Different coefficients are given from time periods starting from the 1900’s to the 2000’s. It should be noted that in Equation (9), the first summation goes from $n=1$ to $\infty$, however, there is no need to go beyond $n=12$ for practical purposes. In cases where the IGRF needs to be calculated near regions where there is a large magnetic source in the land itself, higher orders may be necessary [2] [19].

Computer programs that calculate the geomagnetic field values are readily available, including website implementations, C++ code, and MATLAB code. Since the goal of this thesis work is to use the results of the IGRF, detailed calculations and explanations of (9), (10), (11), and (12) is omitted, but can be found in [2] [7].

2.2.3 Variations of the Earth’s Magnetic Field

The Earth’s magnetic field as a whole is not constant, but varies with time. For example, the magnetic declination in London changed from 4° West to 23° West to 10° West over the last 300 years [28].
When the magnetic variations during the day are very gradual, it is termed a “quiet geomagnetic day.” For example, during the course of a 24-hour time period, the total horizontal magnetic field, \( H \), for an area can change by as much as 200 nT, which corresponds to a 0.002V change in output with the HMC2003 AMR magnetic sensor. When the magnetic variations are large and erratic within a matter of hours, it is termed an “active geomagnetic day”. For example, the horizontal magnetic field intensity, \( H \), can change by more than 600 nT (or more than 0.006V for the HMC2003) within a 4 to 5 hour time frame. The hourly variations of the magnetic field are termed “diurnal variations” [2].

One of the factors that influence the amount of quiet or active days is solar activity. Solar activity is measured with a sunspot number, which is the number of individual dark, cooler regions of the sun. The quietest days occur on or a year after the minimum sunspot number. Sunspot activity is also related to solar flares and magnetic storms, which facilitates ionization of particles in Earth’s ionosphere, thus affecting the ambient magnetic field and can cause changes up to 1500 nT on the surface of the Earth (0.015V with the HMC2003) [2] [7].

If the contribution of the Earth’s magnetic field is taken to be at a range of ±0.2 V, magnetic variations in the Earth’s magnetic field can range from 3% to 7.5% of the Earth’s magnetic field. For the purposes of this thesis work, time variations in the Earth’s magnetic field will not be considered.
3. LITERATURE REVIEW

To survey the state of the art of the applications of magnetic sensors, a literature review was done. The goals of this literature review are to describe the applications of magnetic sensors in the following situations:

- Stationary applications where the magnetic sensors are not moving
- Navigation Purposes
- Compensate for the geomagnetic field

3.1 STATIONARY APPLICATIONS OF MAGNETIC SENSORS

One of the fields where magnetic sensors have been successfully used is in vehicle detection. In some works, the magnetic sensor is stationary, or statically mounted in one place, to measure any changes in the ambient magnetic field. Examples of vehicle detection applications where the magnetic sensors are stationary are railroad crossings, fast food drive through windows, automatic gates, traffic monitoring, parking lot space detection, and parking meters [13].

Vehicles are ferrous objects that create disturbances in a uniform magnetic field around it. Vehicles are platforms primarily composed of both hard and soft iron materials, which are materials not readily magnetized but retain magnetism, or materials that are easily magnetized and demagnetized, respectively. Thus, vehicles are modeled as a composite of many magnetic dipoles, which can be detected by magnetic sensors [5].
The behavior of a stationary magnetic sensor detecting a vehicle is well defined in [5] [13]. In [5] a 3-axis magnetoresistive sensor was placed 1 foot above ground and the sensor characteristics of different passing vehicles were recorded.

The x-axis was parallel to the direction of travel, the y-axis perpendicular and facing towards the car, and the z-axis pointing up towards the sky. When the vehicle passed the sensor, the largest peak in voltage in the y and z-axis was when the engine passed the sensor, due to the ferrous content of the engine and magnetic flux concentrated on it. The x-axis, however, produced unique voltage patterns for certain vehicles. These magnetic signatures in the x-axis would retain the same shape whether the car travelled forward or backwards from the magnetic sensor, provided that the magnetic sensor was stationary. Furthermore, it was found that the amplitudes of the signal would be smaller when the vehicle was either further away or passed the sensor slowly [5].

![Figure 3- Magnetic sensor behavior during a vehicle pass](image)

23
The behavior of the sensor is due to the disturbance of the magnetic field as it approaches the sensor. Figure 3 demonstrates the behavior of a single axis of a magnetoresistive sensor, and is from [5]. When a vehicle approached the sensor in the opposite direction that the sensitive axis was pointed to, the vehicle pulled the magnetic field towards the car’s engine or axles, and the sensor registered the decreasing magnetic field density in the direction it was pointed to. When the middle of the vehicle (for example, the cabin in between axles) was in front of a sensitive axis, the magnetic sensor would output its nominal value because the magnetic flux concentration is mostly in the engine and axle [5].

Considering these behaviors, it was suggested to calculate the sum of the magnitude of the voltages in the x, y, and z-axis, as in Equation (1) and find a threshold level that would indicate when a vehicle is present. The summation of the voltages magnitudes in the x, y, and z-axis rapidly decreased when a vehicle was more than 5 feet away from the sensor. Some studies have shown that a vehicle 5 feet away created a magnitude voltage reading that is 3.7% the magnitude of the vehicle’s magnetic voltage magnitude when it is 1 foot away from the sensor [5][13].

Since most vehicle detection applications are stationary, the geomagnetic field biasing effect is a minor concern. When a sensor is mounted to its stationary position, the geomagnetic field voltage offset is regarded as the nominal voltage when no vehicles are present [13].
One aspect of static vehicle detection is “falsing”, which is when a vehicle detection algorithm would falsely detect a vehicle present when there is no vehicle. For example, when magnetic sensors are mounted on the road to detect traffic on a two lane street on each lane, a Class 8 truck on one lane may have a large effect on the ambient magnetic field that could trigger the magnetic sensor embedded in the adjacent lane. Proposed solutions include moving the magnetic sensors to the outside edges of each lane so that large magnetic fields from large vehicles are further away [13].

Furthermore, since the geomagnetic field varies with time and environmental conditions, there are possible issues with the null voltage changing so that it exceeds any vehicle detection thresholds. Possible solutions for this is to only consider voltage changes that have large rates of change, as opposed to slowly varying changes that may be induced from the ambient magnetic field changing [13].

A specific project where magnetic sensors were used for static detection applications is in [6], where wireless magnetic sensors were used for traffic surveillance with a 98% detection rate of vehicles. The wireless magnetic sensors were glued to the road at a weigh-in-motion station where it took measurements of trucks passing. The voltage output was observed if it exceeded a threshold detection value, and measured the signal if it lasted more than 10 counts of the sample time. If these requirements were met, the algorithm would register those timestamps with a vehicle present. The same test was repeated in individual parking stalls in a parking lot. It was found that adjacent vehicles parked next to the stall the sensor is in were not detected due to the magnetic field decaying with distance [6].
Additional applications of wireless magnetic sensors are discussed in [23]. Certain magnetic sensor products such as the two-axis fluxgate Groundhog G-8 magnetometer and road-weather sensor are placed into the road by cutting a 6 inch diameter hole into the road and placing the sensors in there. This specific model provides vehicle count, speed, length, lane occupancy, daily and annual average traffic, and other information [23]. Commercial implementation of a magnetic sensor for vehicle detection is an indicator of how well the magnetic sensors work for applications where the sensors are stationary.

3.2 NAVIGATION PURPOSES

Another application of magnetic sensors are for navigational purposes, which primarily uses the horizontal field component, H, and declination angle as a reference point. Before discussing specific works, a brief overview of magnetic sensor considerations is given in the following sub-sections.

3.2.1 Magnetic Sensor Considerations

In order to use magnetic sensors primarily for compassing applications, tilt compensation, temperature effects, and sensor calibration should be considered.

3.2.1.1 Tilt Compensation

For navigation systems, one of the assumptions that is used to first develop a system is that the sensor is level on the Earth’s surface so that the x and y axis of a magnetoresistive sensor are directly parallel with the x and y component of the horizontal field component, H [4].
For further development with magnetic sensors for navigation purposes, additional sensors can be used to compensate for a sensor being tilted with respect to the horizontal plane of the geomagnetic field. If roll is considered to be the rotation around the x-axis and pitch refers to the rotation of the y-axis in Figure 2, the following equations can be used to compensate for roll and pitch [4] [3]:

\[
X_H = X \cos \phi + Y \sin \theta \sin \phi - Z \cos \theta \sin \phi \\
Y_H = Y \cos \theta + Z \sin \theta
\]  

(13) (14)

\[
Azimuth = \tan^{-1} \left( \frac{X_H}{Y_H} \right)
\]

(15)

Where

\(X_H\) = X axis voltage compensated for tilt
\(Y_H\) = Y axis voltage compensated for tilt
\(X, Y, Z\) = Magnetic sensor voltage for respective axis
\(\theta\) = Roll, degrees
\(\phi\) = Pitch, degrees

Azimuth = Heading Degree, clockwise from North

Compass tilt errors are more significant when operating near the magnetic poles due to larger inclination angles than areas near the equator [3].
A sensor that can be used to detect tilt is a gyroscope. A gyroscope’s output signal is directly proportional to the angular rate of turn and are accurate in measuring angular motion. However, the output drifts with time and periodic correction and calibration is necessary. A specific advantage of a gyro is that they are suitable for applications where there vector acceleration is applicable. However, they can cost more than tilt sensors [3].

An alternative solution for tilt determination is the use of an inclinometer or tilt sensor to determine roll and pitch angles. However, some tilt sensors, such as a gimbled tilt sensor, can produce tilt errors from shock and vibrations and can take seconds for its readings to stabilize. Other tilt sensors that can be used are electrolytic or dual axis accelerometers. They do not include any mechanical moving parts and can be used for vehicle applications; however, a vehicle’s acceleration may affect the readings [3].

Despite possible errors due to quick accelerations, it can be justified that an accelerometer’s acceleration errors can cancel out over a period of time due to accelerations and decelerations of a vehicle. Acceleration errors will accumulate if the accelerometer is moved in a circular motion for a long period of time [28]. In addition, it is also assumed that a vehicle that is driving cautiously will not have large amounts of accelerations such as a fighter jet maneuvering in the sky, and so acceleration errors can be assumed to be small or only last a short amount of time [3].
3.2.1.2 Temperature Effects

Another consideration for magnetic sensors for navigation purposes is the temperature effects on the circuitry, primarily the offset drift with temperature and the sensitivity temperature coefficient. When the permalloy thin film in magnetoresistive sensors are heated, the bridge offset changes and causes a change in voltage output, which can be a 20 milliGauss shift with a 3 volt supply voltage [13].

The temperature coefficient is defined as a change in output gain of the sensor over the change in temperature. For magnetoresistive sensors, the temperature coefficient’s are linear so that the X and Y-axis gain will change together. For navigation purposes, temperature effects are canceled when computing the heading because the X voltage is divided by the Y voltage in Equation (15). For temperature offset drift, a set and reset technique for the magnetic sensor is detailed in [3].

For considerations between temperature changes between the initialization to the operating temperature of the sensor, it is found that a magnetic sensor with temperature compensation circuits usually reaches its steady state temperature after 1 minute of operation. The initial and steady state voltages differ by less than 0.5% [8].
3.2.1.3 Magnetic Sensor Calibration

When a magnetic sensor is mounted on a platform with a large amount of ferrous content, such as vehicles, the platform will distort the ambient field from its natural state. This altered magnetic field can affect the heading degree computed from magnetic sensor readings for compassing applications. If the magnetic sensor is securely mounted to a platform on a vehicle, a calibration procedure can be performed in order compensate for the magnetic disturbances that are constantly near the magnetic sensor.

Magnetic distortions in a vehicle come from hard iron and soft iron effects near the magnetic sensor. Hard iron effects are from permanent magnets or magnetized ferrous metal in a vehicle that are constantly superimposing a magnetic field on the magnetic sensor. Soft iron distortions are contributions of magnetically soft materials in a vehicle, which can gain or lose magnetism quickly [3].

When a 3-axis vector magnetometer is rotated on the z-axis on a level plane in a location without magnetic disturbances other than the geomagnetic field as in Figure 4, the output of the x and y-axis should be a sinusoidal response that are phase shifted by 90 ° as depicted in Figure 5 [4].
Figure 5- X and Y Magnetic Sensor Response to Earth’s Magnetic Field

When a sensor axis is aligned with magnetic North, the geomagnetic field will contribute a positive voltage to the magnetic sensor. When a sensor axis is aligned opposite of magnetic North, the geomagnetic field will contribute a negative voltage to the magnetic sensor. Keeping this in mind, Figure 5 is explained as follows.
When the x-axis of the magnetic sensor is aligned with magnetic North, the x-axis will be at its maximum positive value. Because the y-axis is at 90° to the left of the x-axis in Figure 4, it is neither aligned towards magnetic North or towards magnetic South, therefore it will output the null voltage. When the sensor is rotated 90° clockwise so that the y-axis is aligned with magnetic North, it will achieve its maximum positive value. The x-axis, pointing towards the East, will output the null voltage because it is neither pointing towards magnetic North or South. When the x-axis is aligned towards magnetic South, it will output the minimum voltage because it is pointing in the opposite direction of magnetic North. At the same time, the y-axis will be pointing towards the east and will output the null voltage because it is neither aligned with magnetic North or South. This behavior is periodic with respect to a sensor rotating within the geomagnetic field free from other magnetic disturbances because it is only responding to magnetic North [4][5].

The behavior described is depicted in Figure 5. From this figure, it can be observed that the x and y axis should have the same range of voltages as it rotates within the geomagnetic field when it is free from other magnetic disturbances.

The effects of magnetic disturbances can be observed if the x and y magnetic voltages are plotted against each other when the sensor is rotated about the z-axis, using the procedure that was previously described. The plot of the x and y voltages, when free from magnetic disturbances, should look like Figure 6. The shape of the magnetic circle should be a circle because the x and y-axis should have the same range in voltages and are offset by 90°.
However, when the sensor is mounted to a platform, such as a vehicle, and the same rotation procedure is done, the graph in Figure 6 may not look like a perfect, centered circle as in Figure 6. Hard iron effects, which are always constant, will affect the magnetic circle in Figure 6 by shifting the origin of the circle. Soft iron effects will scale the magnetic circle into an eclipse [3] [28]. Therefore, a calibration procedure is required in order to compensate for these magnetic effects.

A calibration procedure is detailed in [3] and [22]. When a vehicle is driven in a circle in an area that is level and free from magnetic disturbances, the data in the x and y-axis are analyzed to obtain scale and offset factors in the x and y axis. The following equations are used in order to calibrate the x and y-axis [3] [19]:

![Figure 6- Magnetic Circle for a Calibration Run without Magnetic Disturbances](image-url)
\[ X_{\text{calibrated}} = X_{sf} \times X_{\text{raw}} + X_{off} \quad (16) \]

\[ Y_{\text{calibrated}} = Y_{sf} \times Y_{\text{raw}} + Y_{off} \quad (17) \]

Where

\( X_{\text{calibrated}} \) and \( Y_{\text{calibrated}} \) = X and Y magnetic sensor voltages calibrated for local hard and soft iron effects

\( X_{\text{raw}} \) and \( Y_{\text{raw}} \) = raw X and Y magnetic sensor voltages

\( X_{sf} \) and \( Y_{sf} \) = X and Y magnetic voltage scaling factors

\( X_{off} \) and \( Y_{off} \) = X and Y magnetic voltage axis offsets

Calculations for the scaling factors and offsets are as follows:

\[ X_{sf} = 1 \text{ or } X_{sf} = \frac{Y_{\max} - Y_{\min}}{X_{\max} - X_{\min}} \quad \text{[choose the greater value]} \quad (18) \]

\[ Y_{sf} = 1 \text{ or } Y_{sf} = \frac{X_{\max} - X_{\min}}{Y_{\max} - Y_{\min}} \quad \text{[choose the greater value]} \quad (19) \]

\[ X_{off} = X_{sf} \left( \frac{X_{\max} - X_{\min}}{2} - X_{\max} \right) \quad (20) \]

\[ Y_{off} = Y_{sf} \left( \frac{Y_{\max} - Y_{\min}}{2} - Y_{\max} \right) \quad (21) \]
Where:

\[ X_{\text{max}} \text{ and } Y_{\text{max}} = \text{The maximum voltage of the x and y axis during a calibration when the vehicle turns} \]

\[ X_{\text{min}} \text{ and } Y_{\text{min}} = \text{The minimum voltage of the x and y axis during a calibration when the vehicle turns} \]

By performing this calibration procedure, the effects of local magnetic disturbances on the platform that the sensor is mounted on will be reduced. However, the best solution is to place the magnetic sensor away from ferrous metal as much as possible. For magnetic sensors used as a compass, it has been found that mounting the magnetic sensor by at least 10 inches above the roof of the vehicle towards the rear center or rear side is an ideal place to mount it [28].

### 3.2.1.4 Changes in Magnetism

Although the calibration procedure in Section 3.2.1.3 Magnetic Sensor Calibration can compensate for the majority of hard and soft iron effects that exist in a platform, it may not compensate for magnetic effects that are due to time or the environment.

For example, the vehicle itself can become magnetized by external fields, such as driving over railroad tracks powered by a DC current. Another example is the magnetic characteristics of a vehicle changing when someone hits the door of a vehicle by an object, or by simply slamming the door [28]. If the magnetic sensor is placed near an electronic component that can be powered on or off, such as a defroster unit on the back window, the magnetic field can also change [15].
Environmental effects include hard and soft iron content of hills, mountains, buildings, underground structures, and bridges that a vehicle may pass through, or the change in a vehicle’s magnetic field when it goes through a carwash [28] [15].

It is also suggested that a magnetic sensor should only be used for vehicles that have a constant magnetic field around it. For example, a magnetic sensor should not be mounted on a tank because the rotating turret or firing the cannon may change the magnetism of the vehicle [28].

3.2.1.5 Navigation Applications

Applications of magnetic sensors as navigation instruments have been done in many works. One of those is described in [17] where a combination of a magnetic compass, GPS, and a gyro were used with a Kalman filter for accurate azimuth (heading) when GPS is not available (which is termed “dead reckoning”) for tracking an individual.

For this application, the magnetic sensor was used to calculate the azimuth based on the magnetic readings and available GPS signals. It was mentioned that calibration procedures were done before mounting it to a person to compensate for hard and soft iron effects on the unit, which included a low pass filter on the data. The work also suggested that magnetic correction maps should be made of places of interest in order to compensate for other magnetic disturbances. In environments that have large external magnetic disturbances, the plotted trajectory from a magnetic sensor and GPS combination using a Kalman filter would be inaccurate, but would perform reasonably well when used in environments without magnetic disturbances [17].
Another example of work that used a magnetic sensor for navigation purposes is detailed in [15]. In this work, two different approaches were developed for an online magnetic compass calibration that will calibrate itself automatically when a magnetic field changes around it. One of the algorithms is termed “parameter adaptive algorithm” which involves comparing the heading from the GPS and magnetic sensor and utilizing the errors to adjust the calibration parameters in Equations (18), through (21). The other approach is a functional learning algorithm that continuously integrates the heading error and utilizes a neural network model. Using the parameter adaptive algorithm when the magnetic declination is small, and the functional learning algorithm when the magnetic declination was large, heading errors were minimized in the experiments conducted in the work [15].

Furthermore, another dead reckoning system using Kalman filters, magnetic sensors, and other aid sensors were used in [19] and was tested on pedestrians and vehicles. In this work, a calibration procedure described in Section 3.2.1.3 Magnetic Sensor Calibration was implemented with calibration coefficients in Equations (16), through (21). The algorithms that used a magnetometer for pedestrian navigation purposes performed well compared to an algorithm that used gyroscopes. However, mounting the magnetometer on a vehicle using the same algorithm performed poorly compared to the pedestrian tests due to the hard and soft iron effects of the vehicle and external magnetic effects were not considered in the scope of [22].
Similarly, a Kalman filter was also used with a magnetic sensor for a vehicle automation system in [21]. A magnetic sensor was mounted on a tractor, and a calibration procedure similar to that described in Section 3.2.1.3 Magnetic Sensor Calibration was done. The resulting magnetic circle (as in Figure 6) resulted in a magnetic circle that reflected hard and soft iron errors. However, the errors were deemed insignificant and the work stated that the offset of the tractor still provided accurate azimuth calculations [21].

From the various applications of magnetic sensor for navigation purposes, sensor calibration is a vital step for correct azimuth calculations.

3.3 Compensation for the Geomagnetic Field

In addition to navigation purposes, magnetic sensors can also be used to compensate for the geomagnetic field.

One of the approaches in compensating for the effect of the geomagnetic field is through the use of multiple sensors. In [18] it is suggested that the change in the geomagnetic field should be removed by using two magnetic sensors. A known distance should separate these magnetic sensors and the corresponding difference between these two sensors should nullify the geomagnetic field.
This technique is used in [1] and [27]. In [1], multiple sensors were used in order to reduce the effects of the noise, including the geomagnetic field, on a total-field gradiometer mounted on an unmanned underwater vehicle. When two sensors were separated by a distance in three orthogonal directions, the difference in output was referred to as a magnetic field gradiometer. In this work, several sensors placed around the underwater vehicle were used by analyzing the noise in the frequency domain of each sensor and subtracting those signals multiplied by a transfer function from the raw data to compensate for geomagnetic field effects.

In addition, [27] details a compensation scheme with a Room Temperature Gradiometer (RTG). The RTG included three three-axis vector magnetometers mounted on a corner of a triangle with another three-axis vector magnetometer mounted in the center of the triangle. The magnetometer in the middle sensed the geomagnetic field and supplied feedback currents that surrounded the corner sensors, which are proportional to the geomagnetic field. The work also suggested using a time domain model to compensate for the earth’s magnetic field by subtracting a DC offset, and weighted magnetic sensor readings from the reference sensors [27].

A similar technique in using multiple sensors is used in [12]. A wearable writing input device is proposed, where the index finger can be used to write or draw letters. The system calculates the magnetic field intensity of a magnet on the index finger by using two magnetic sensors on the opposite wrist. In order to cancel out the effects of geomagnetism, the difference in voltage of the two magnetic sensors are taken, assuming that there is a constant magnetic field around the two sensors and the magnet moves within the same plane [12].
Another method to compensate for the geomagnetic field is to use the results of Equations (9) through (12) to calculate the theoretical geomagnetic field for a certain location. Commercial software, such as Intrepid, by Intrepid Geophysics, is a software package that processes geophysical data and uses the mathematical representation of the geomagnetic field of Equations (9) through (12) [11]. For example, the software can be used for geomagnetic correction for magnetic sensor data obtained by an airplane flying over a region.

Both [11] and [16] include several guidelines for processing magnetic data. It is commonly practiced to use an elevation value with Equation (9) with the average elevation of the experimental route and is stated that the error by this approximation is small. The survey time can also be replaced by the mean survey date because the time variation of the geomagnetic reference field is usually slow. Finally, the variation of the geomagnetic reference field with location is very slow, and needs to be calculated every 100 meters to 500 meters [11] [16].

Although geomagnetic correction techniques exist, additional equipment or software must be used which raises the total cost of a project. The goal of this project is the same, but explored other techniques to achieve it.
4. PROBLEM STATEMENT AND SOLUTION CONCEPT

The main subject of this work is to compensate for the Earth’s magnetic field. Due to the fact that an anisotropic magnetoresistive sensor’s sensitivity range includes the effects of the geomagnetic field, consideration for the Earth’s biasing effects should be done. For the low-cost crash avoidance system continued in this work, the only magnetic signatures of interest are those from passing vehicles. The voltages superimposed by the Earth’s magnetic field limited the work in [25], such that:

- Methods for setting a voltage threshold when the crash avoidance system detects a car by Equations (1) and (2) is dependent on the test vehicle’s orientation with respect to magnetic North. Changes in heading produced large voltage fluctuations with respect to signals produced from vehicles passing, as in Figure 1. Therefore, a single threshold level for detecting a vehicle cannot be done. As a result, the analysis in [25] was limited to driving the car in a constant heading direction with respect to the geomagnetic field, which limits the system’s usefulness as a practical product.
• The response of the magnetic sensor is also dependent on the contents of the test vehicle because a magnetic sensor calibration procedure was not considered. If the test vehicle was carrying ferrous cargo or operating large amounts of onboard electronics, the overall ambient magnetic field of the car calculated by Equations (1) and (2) will be large, as opposed to a situation where the test vehicle is not carrying any ferrous material as cargo. Setting a single vehicle detection threshold using Equations (1) and (2) cannot be done because it depends on the ambient magnetic field in a car, which can change depending on what is present or not present in the test vehicle. Since this system is developed for Class 8 trucks, magnetic sensor calibration is crucial in order to get a consistent baseline voltage because those trucks constantly change the cargo they carry. Having a consistent voltage baseline is necessary for optimum system performance.

This work will address these two issues by:

• Creating an algorithm to subtract the geomagnetic field from the magnetic sensor data, in order to extract a car’s magnetic signature regardless of the test vehicle’s orientation with respect to magnetic North. This algorithm will be a post-processing MATLAB program that can be developed into a real time program for future projects.

• Implementing a calibration procedure that will create a consistent baseline voltage for the magnetic sensor regardless of the cargo a vehicle carries.
• Use only the supplied consumer-grade GPS and a single magnetic sensor in order to keep costs at a minimum. Since the goal of this project is a low-cost crash avoidance system, the work will use the fewest amount of components as possible to keep costs low. Keeping this crash system low-cost will make it a financially attractive system for commercial transportation companies and consumers. Vehicle safety systems such as this project may lead to less vehicle accident rates if more organizations and individuals are able to afford it.

Although magnetic sensors have been used in car detection systems, most of those systems use stationary magnetic sensors. In works such as [6] and [23], magnetic sensors successfully detected cars when they are mounted on the road. In these situations, the geomagnetic field sensed by the magnetic sensor does not change with direction since the magnetic sensor is fixed to one location. The geomagnetic field component experienced by these systems is constant, and is taken as the baseline voltage to which the magnetic signatures of vehicles are added. These works differ from the subject of this thesis in that the magnetic sensor itself is moving with respect to magnetic North, and the baseline voltage changes as a function of the heading degree.
In addition, magnetic sensors have also been used as a compassing device in vehicles. In general, the signals of interest for compassing devices are the Earth’s magnetic field. Therefore, the signals from other magnetic anomalies, such as passing vehicles, are undesirable. The optimal placement of magnetic sensors for compassing applications in vehicles is away from passing cars, such as the roof of a car mentioned in [28]. However, placing a sensor in these areas will degrade the magnitude of the signals caused by passing vehicles (which is the signal of interest in this system) because the magnitude of the signal decreases as it is moved away from the source. The placement of the magnetic sensor in this thesis work is on the side of the test vehicle in order to be as close to passing vehicle as much as possible, which makes this work less applicable as a navigational compass.

Solutions that compensate for the geomagnetic field also exist. Works such as [1], [27], and [12] use the difference between magnetic sensor readings between multiple magnetic sensors to compensate for the geomagnetic field. Although this approach is possible, it is not optimal for this project because it requires multiple sensors that will increase the cost of this low-cost crash avoidance system. For example, [27] uses a total of 4 three-axis vector magnetic sensors to compensate for the geomagnetic field. At the current price of one Honeywell HMC2003 of $220 <https://shop-mps.honeywell.com> implementing this system will cost more than $880, which is considered expensive for the purposes of this project.
Geomagnetic compensation techniques were also presented in the work of [12]. However, it used multiple magnetic sensors and the work assumed that the environment the wearable hand input device is operating in has a constant magnetic field. When operating in a room, sources of magnetism such as computers can be easily moved away or shielded from the magnetic sensors. This assumption is less applicable in this thesis work because the crash avoidance system is exposed to varying magnetic fields from passing cars, bridges, and other structures in the environment as the test vehicle drives through a town. The magnetic noise from these objects cannot be moved away, like it can in small-scale applications in [12].

Other solutions for compensating for the geomagnetic field can be found in commercial software such as the Intrepid Geophysics software package used for post processing data [11]. However, this software has purchase and licensing costs associated with it, which is not ideal for this cost sensitive project. Furthermore, if the software package is purchased, it may not include a source code that can be developed into a real-time running application.

While solutions for geomagnetic compensation exists, they have costs associated with them that this low-cost crash avoidance system project is sensitive to. This thesis work will form a solution to geomagnetic compensation by only using a consumer-grade GPS unit and a single magnetic sensor. Since these units were purchased for the crash avoidance system in the past, there is no additional cost to continue developing this system within the scope of this thesis work.
In addition, implementing the geomagnetic compensation solution presented in this work to the low-cost crash avoidance system developed in [25] is not cost intensive. A magnetic sensor and a GPS is already included in the system. For a possible future-production version of the crash avoidance system, the geomagnetic compensation techniques in this work will also not be cost intensive. Most commercial trucking companies use a GPS system that can be used with the solution present in this work. Therefore, costs will only be retained in the initial purchase of the low-cost crash avoidance system developed in [25].

Implications of this thesis work will allow continued development for the low-cost crash avoidance system. The initial system developed in [25] was limited to successfully detecting cars when the test vehicle only travelled in the same heading direction, which made it impractical for real world applications. However, this solution will now allow the crash avoidance system to function when the test vehicle changes its orientation with respect to the Earth’s magnetic field. Successful real time implementation of this geomagnetic compensation scheme will take the low-cost crash avoidance system one step closer to a commercial product. Having a successful low-cost crash avoidance system will give consumers and commercial transportation companies a low-cost safety system that can be implemented to decrease the risk of vehicle related collisions and accidents without paying a high initial cost.
Furthermore, the solution developed in this work can be applied to other projects that do not require the geomagnetic field’s influence on magnetic sensor readings. Prime examples are in magnetic anomaly detection projects, such as unmanned underwater vehicles or geographical survey planes, where the primary magnetic signals of interest are cars, environmental features, or metallic structures. Developing this solution will help reduce total project and product costs by using only a consumer-grade GPS and a single magnetic sensor, as opposed to using multiple sensors.
5. PRESENTATION OF WORK

5.1 EQUIPMENT

The following hardware was used for this thesis work. All of these components were purchased previously in the work of [25].

- Honeywell 3 Axis Anisotropic Magnetoresistive Sensor HMC2003 [14]
  This is the AMR sensor that was used to record magnetic data when mounted on the side of the test vehicle.
  - 3 Axis, x, y, z
  - 40 μ Gauss Resolution
  - Range: ±2 Gauss
  - 1V/gauss output
  - 2.5V Typical Null Output
  - $220

- Garmin GPS18 [9]
  Consumer grade GPS unit that was used to get heading direction and GPS coordinates.
  - USB
  - Accuracy
    - <15m GPS only
    - <3m WAAS support
  - Updates: 1/s
  - <$90
• NI BNC-2111 DAQ [20]
  o 16 Analog Inputs
  o 2 Analog Outputs
  o 5 Digital IO
  o $349
• Unibrain Fire-I Digital Camera [26]
  The video served as a reference for vehicle presence at certain time intervals.
  This unit will be mounted on the same side of the truck as the magnetic sensor.
  o Firewire
  o 30 fps, 640x480
  o $109

5.2 ASSUMPTIONS
Before starting this analysis, assumptions were made in order to focus on the exploration of solutions for geomagnetic compensation.

It was assumed that the scope of this thesis work will strictly be in a post-processing environment. After data was taken using a test vehicle, the data is processed on a computer with a custom coded MATLAB program. This assumption will allow for spline interpolations of the GPS heading direction to obtain heading information in between the 1 second GPS update intervals.
Due to the fact that all GPS headings travelled are known in a post-processing environment, a spline interpolation between known points can be used to produce heading directions in between known headings. A spline interpolation was chosen because it is an interpolation method that produced reasonable heading directions in between known GPS points. The heading directions generated by the spline interpolation were smooth and continuous, which represents normal driving maneuvers. Real-time spline interpolations may not be possible without a dead-reckoning system.

The next assumption made in this work is that the vehicle traveled on flat land so that the magnetic sensor’s x and y-axis as in Figure 4 remained parallel with the horizontal field component of the geomagnetic field in Figure 2. This assumption is made so that sensor tilt was not considered in this initial solution concept. The assumption is also reasonable since most of the data gathered were in local city streets that have few inclines and declines. For future development of this solution, tilt compensation can be implemented using additional sensors, such as a gyro or accelerometer, and implementing the background information in Section 3.2.1.1 Tilt Compensation. In addition, magnetic sensors with embedded tilt compensation circuitry are available from online electrical retailers such as Sparkfun.com.
Another assumption that was made in this work is that once a sensor is calibrated, the calibration of that sensor will remain true for the rest of the experiment. Active geomagnetic days will not be considered in this work. This assumption is reasonable because the experiments were run immediately after calibration and tests were about 30 minutes in length each, so that the calibration coefficients used in field data are the most up-to-date. Recalibration may be necessary if the tests are longer than 30 minutes and are recommended when initiating a test in a new location. The variations of the calibration data with respect to length of time and different locations are not addressed in this work. Time was limited with the test vehicle so that long test times were not feasible. Also, locations to safely perform the calibration procedure were only officially secured on a particular parking lot at the Cal Poly San Luis Obispo campus, so that the range of the test area were limited to routes near the campus.

In addition, it was assumed that the null voltage output of each axis of the magnetic sensor was the nominal 2.5 V factory specifications. However, when compensation algorithms were compared to each other, the signals were shifted down so that the average of the signals will be 0 V for comparison purposes. This can be justified for this work since this will be a post-processing environment where a complete data set is available. For real time processes, a defined voltage shift is necessary since the instantaneous voltage average changes.
The final assumption that was made is that the test vehicle was driven normally under everyday, dry conditions. The test vehicle turned in a smooth and gradual manner and kept all four tires on the ground. This is done so that the GPS headings are accurate. In addition, only smooth paved roads were driven on so magnetic sensor vibration is minimized. Large vibrations from uneven roads may result in voltage fluctuations from the sensor’s magnetic axes rotating.

5.2 PRELIMINARY INVESTIGATION

5.2.1 Part I Experiment
The first step in this thesis work was to characterize the response of a magnetic sensor and compare it with theoretical results. The goal was to rotate the sensor 360° and characterize the voltage reading as a function of heading.

The aforementioned procedure was done in an area that was free from electronics and any visible sources of magnetism such as vehicles or light poles. The location chosen for this experiment is Dinosaur Cave Park in Pismo Beach, California, with coordinates of approximately (35.15837193482247, -120.64533233642578). This location had open fields, a lack of magnetic sources/electricity, and no cars present within 30 ft. of the test area.

The initial test focused on the simplest case scenario- observing the behavior of the magnetic sensor rotating on its own axis in earth’s magnetic field and comparing the data with theoretical results from Figure 5. The procedures are as follows:
1. Find an area that is not near cars, power generators, train tracks, and other ferrous material and magnetic sources in order to minimize magnetic disturbances. For these tests, the sensor was placed on top of grass in an open field.

2. Locate magnetic north using a compass. Use a ground that is as level as possible.

3. Align the positive x-axis of the magnetic sensor along with magnetic north.

4. Take voltage readings of the x, y, and z-axis.

5. Rotate the magnetic sensor about the z-axis 15° counter-clockwise and measure the output voltage using a multimeter.

6. Repeat until 360° of travel is achieved.

7. Repeat 3 more times at a different location within the test area.

---

**Figure 7- Part I: Turning the magnetic sensor 360° about the z-axis**
The voltages read from the multimeter on the x, y, and z axes were plotted as a function of heading direction from magnetic North, counter clockwise in Figure 8. Three data points were recorded per heading degree because the magnetic sensor was rotated 3 times about the z-axis. Scatter of the voltage data points within each heading is most likely due to the fact that the magnetic sensor axis was not exactly pointing at the same heading degree each time the sensor was rotated by hand. The sensor was rotated 3 revolutions to demonstrate the periodic relationship between voltage and heading degree. The data should have less scatter if the magnetic sensor was rotated at exact degrees. In addition, noise and uncertainty within the magnetic sensor itself could have contributed to the spread of the data.

Figure 8- Magnetic Sensor voltages as a function of heading direction from North, CCW. Raw Data.
The specified null field output of the magnetic sensor is typically 2.5 V [14]. In order to directly compare this data with theoretical results in Figure 5, the x-axis voltage readings were shifted down 0.0627 V and the y-axis readings shifted down 0.0125V. This made the average of the two signals approximately 2.5V, which makes it easier for comparison and interpretation. This adjustment is shown in Figure 9. A possible reason that the x and y axes were not offset at exactly 2.5V were due to the magnetic sensor itself. During the initial testing phase, it was suspected that the first magnetic sensor used had finite changes in resolution and sensitivity during its storage. It is known that the magnetic sensors used in this work can have its resolution and accuracy affected by strong magnetic fields, as discussed in [3].

However, it was evaluated that the magnetic sensor could still be used in this preliminary investigation to observe how it records the geomagnetic field since it produced a graph that is similar to the theoretical results in Figure 5. For experiments with the test vehicle, a new magnetic sensor was purchased from Honeywell with the factory calibration.
Figure 9- Results of Figure 8 with averages shifted to 2.5V

Once the data was shifted for comparison, a 7th degree polynomial trend line was fitted to the data and a trend line was generated and shown in Figure 9.

Compared to the expected results in Figure 5, the experiment generated similar results. When the positive x-axis was facing North, the signal generated the highest voltage reading because the sensitive axis was aligned with the geomagnetic vector. At the same time, the y-axis generated a neutral reading with a voltage of about 2.5V in Figure 9 because it is perpendicular to the direction magnetic North and does not receive a positive or negative contribution from the geomagnetic field.
When the magnetic sensor was turned to the East, the positive x-axis was perpendicular to magnetic North and returned a neutral reading of 2.5V in Figure 9. However, the positive y-axis was aligned in the opposite direction of magnetic North, and thus generated the lowest voltage reading.

The same pattern can be observed when the magnetic sensor was facing south. The x-axis generated its lowest voltage reading because it was facing away from magnetic North, and the y-axis generated a neutral reading of 2.5V.

As the magnetic sensor was returned to its original orientation with the x-axis aligned with magnetic North, the same approximate voltages generated at the start of the experiment at 0° is observed at the end of the experiment at 360°.

5.2.2 Part II Experiment

A similar experiment was done as in Part I, however, instead of rotating the magnetic sensor on its axis, the sensor will be moved in a 3 ft. radius circle to simulate a vehicle traveling in a circle. The positive x-axis will be the nose of the vehicle and the y-axis will always point to the left of the direction of travel. The procedures done are as follows:

1. Find an area that is not near cars, power generators, train tracks, and other ferrous material and magnetic sources in order to minimize magnetic disturbances.
2. Locate magnetic North using a compass. Use a ground that is as level as possible.
3. Locate a center for the circle of travel.
4. Have the magnetic sensor’s positive x-axis pointed towards magnetic north, and place it 3 ft. away from the center of the circle of travel.
5. Record the x, y, and z voltages with a multimeter.

6. From the center of the circle, measure 15° counter clockwise and place the magnetic sensor 3 ft. away from the circle, making sure that the positive x axis points 15° counter clockwise from magnetic north.

7. Repeat until 360° of travel is achieved.

8. Repeat 3 more times at a different location within the test area.

![Diagram of magnetic sensor placement and travel](image)

**Figure 10- Turning the magnetic sensor 360° 3ft away from a center point**

The voltages read from the multimeter on the x, y, and z axes are plotted as a function of heading direction from north, counter clockwise in Figure 6.
Figure 11- Part II Magnetic Sensor voltages as a function of heading direction from North, CCW.

Raw Data.

The data was shifted so that the average of the data for each axis was at the null field output of 2.5V for ease of interpretation. The adjusted data is plotted in Figure 12.
Figure 12- Part II Adjusted Data

Trend lines of 7th degree polynomials were also used to fit the data. Similar results are obtained from this experiment as in Part I.

5.2.3 Z-Axis Results

For these two experiments, the results obtained from the z-axis magnetic sensor were not a concern. The voltage of this axis was consistent, and only varied between a peak-to-peak range of 0.1 V. This is expected because the z-axis is not changing its orientation in reference to magnetic North, as long as it remains level with the ground. However, since the z-axis voltage did vary within a limited range, the sensor may have been tilted while it rested on top of the uneven ground at a small degree with respect to the geomagnetic field’s horizontal plane.
5.2.4 Part I and Part II Conclusion

Since the data in Part I and Part II are almost exactly the same, it is concluded that the major effect of the earth’s magnetic field in a magnetic sensor is its current orientation with magnetic North. Part I demonstrated this when the magnetic sensor was rotating about a single stationary axis. Part II verified that even if a magnetic sensor travels a certain distance, the voltage readings only vary with orientation to the magnetic North at its current location.

However, the x-axis has a higher DC offset than the y-axis in Figure 8 and Figure 11. It is expected that the DC offset of both of the signals should be at 2.5 V according to manufacturer specifications [14]. This shift in the average of the x-axis signals could be attributed to hard iron effects around the testing area or sensor errors in the x-axis as previously discussed in 5.2.1 Part I Experiment.

5.3 COMPENSATION SCHEME 1: “GPS COMPENSATION”

With regards to a solution to geomagnetic field compensation, it was initially thought that the equation of the fitted trend (Section 5.2 Preliminary Investigation) could be used to subtract that magnitude of the Earth’s magnetic field, under the assumption that the voltages that correspond to headings measure only the geomagnetic field. The same experimental procedures from Section 5.2 Preliminary Investigation is repeated with the test vehicle, which was a truck owned by Cal Poly San Luis Obispo. The magnetic sensor was mounted on the test vehicle as in Figure 13 with the GPS unit.
5.3.1 Sensor Calibration

The same turning procedure described in Section 5.2.2 Part II Experiment was replicated with a test vehicle as depicted in Figure 13. The location for this procedure was done in an area with the least amount of magnetic disturbances and was large enough so that the test vehicle may make several turning maneuvers without compromising anyone’s safety. The magnetic sensors were placed on the rear driver’s side with the x-axis pointing to the left of the vehicle, the y-axis pointing behind the vehicle, and the z-axis pointing up.

![Figure 13- Calibration movement for test vehicle](image)

As the vehicle traveled in a circle at 7-10MPH (or approximately 18° per second), the magnetic sensor recorded the response of the geomagnetic field while the GPS recorded heading data. The goal was to separate each voltage reading into their respective heading direction.
The turning calibration procedure was done in an empty parking lot at California Polytechnic San Luis Obispo. Special arrangements were made with the campus police department to ensure that this was a safe environment to conduct the turning procedure in. Maps of the site with plotted GPS coordinates and a picture of the test vehicle is displayed in Figure 14. The red circle in the map is the GPS position of the test vehicle during the turning/calibration procedure.

![Figure 14- Calibration procedure location. GPS map provided by GPSvisualizer.com](image)

### 5.3.1.1 Results of Calibration procedure

The results described by the turning procedure in the prior section are displayed in Figure 15. Data was only recorded once the vehicle was moving in a circular path. The magnetic voltages of the x and y-axis should be sinusoidal and periodic if it is only measuring the geomagnetic field, which the results in Figure 15 reflect. The trailing ends at the beginning and end of the calibration run are due to the vehicle starting from rest and slowing down to a stop once the truck is driving in a circular path.
From the discussion in Section 3.2.1.3 Magnetic Sensor Calibration and Section 5.2 Preliminary Investigation, the x and y-axis should have the same DC offset and the same peak-to-peak voltage. However, the uncalibrated signals in Figure 15 have an average signal that is higher than 2.5V, which is the expected average of the signal. In addition, the y-axis has a 4% larger peak-to-peak amplitude than the x-axis, when it is expected that they should have the same peak-to-peak amplitude.
These differences in the actual field data and theoretical results are likely due to the hard and soft iron magnetic effects in a vehicle that offset the null voltage or increase sensitivity in one axis of a magnetic sensor as described in [3]. For vehicles, concentrations of disturbances in the magnetic field are usually in areas near wheel axles and the engine. In order to align the periodic signals of the x and y-axis to have the same DC offset and peak-to-peak amplitude, calibration of the magnetic sensor was done.

Equations (16) through (21) are applied to the raw signals of the data obtained from the calibration procedure through MATLAB. Once the calibration procedure was done, calibration coefficients \(x_{sf}, x_{off}, y_{sf}, y_{off}\), were stored as variables in MATLAB for future use to calibrate magnetic sensor readings for test field data. The results of the calibration are shown in Figure 16.

![Figure 16- Calibrated versus uncalibrated magnetic sensor data](image_url)
The offset variable $x_{off}$ and $y_{off}$ in Equations (20) and (21) had the effect of setting the DC offset of the x and y-axis to the same level in order to compensate for the hard iron effects of mounting it to the vehicle. The scaling factor variables $x_{sf}$ and $y_{sf}$ in Equations (18) and (19) scale each axis equivalently to compensate for unequal axis sensitivity. Using the calibration procedure successfully calibrated the raw signal (dashed lines) into the calibrated signal (solid lines) in Figure 16. Regardless of the type of test vehicle used, or the amount of ferrous cargo in the test vehicle, the calibration procedure should result in similar calibrated signals in Figure 16.

In order to use a commercial version of this system, universal compatibility of the magnetic sensor voltage output is important. The same calibrated signals should result regardless if the test procedure was done by a Class 8 truck or a small compact car. Ignoring this calibration step would produce x and y-axis signals with varying DC offsets or sensitivity that is dependent on the vehicle and its contents. The lack of consistency in the signals will make a universal geomagnetic compensation algorithm difficult to implement between different vehicles.

Furthermore, the effects of the hard and soft iron effects of the vehicle can be seen when the magnetic circle (refer to Section 3.2.1.3 Magnetic Sensor Calibration) is plotted in Figure 17.
Figure 17- Magnetic circle for calibration procedure

Figure 17 is a plot of corresponding x and y-axis voltage pairs at a specific point in time for calibrated and uncalibrated signals. The positive DC offset of the uncalibrated signal (red) is demonstrated by a shift of the center of the magnetic circle to the left (positive x-axis) and up (positive y-axis). The slightly larger peak-to-peak value of the y-axis slightly elongates the uncalibrated, red magnetic circle in the y-axis direction. The calibration procedure eliminates the DC offset of the magnetic sensor and shifts the magnetic circle to the (2.5V, 2.5V) coordinate (which is the null offset value) and scales the y-axis down in order to keep the x and y-axis ranges equal, thus producing a more circular plot as shown in the green.
A more exaggerated example of an uncalibrated magnetic circle with a y-axis noticeably more sensitive to the x-axis is shown in Figure 18. This data was obtained from the first magnetic sensor used, which was used in a mobile cart to perform the calibration-circling maneuver in the initial phase. Since a vehicle was not used, hard iron effects were not present; thus, uncalibrated data did not have a large DC offset. The data shown in Figure 18 was centered to (0V, 0V) point instead of the usual (2.5V, 2.5V) null field output point for comparison purposes.

Figure 18- Magnetic Circle with large y-axis sensitivity compared to the x-axis
The y-axis peak-to-peak voltage during a calibration procedure was greater than the x-axis peak-to-peak voltage range, which created an ellipse with the y-axis scaled larger than the x-axis for the uncalibrated data. The calibrated data scaled the ellipse into the calibrated circle, which is the blue data in Figure 18.

5.3.1.2 Heading-Voltage Function

The calibration procedure detailed in the previous section assumed that the calibrated sinusoidal signals in the x and y-axis of the magnetic sensor in Figure 16 were only due to the magnetic sensor axis changing its orientation with respect to magnetic North and geomagnetic field. If measurements of the geomagnetic field could be arranged in terms of heading directions, it is possible to characterize the geomagnetic field with a mathematical function that will output the voltage from the geomagnetic field as a function of heading degree.

During the calibration-turning procedure, the GPS collected heading data while the magnetic sensor characterized the geomagnetic field.

5.3.1.3 GPS Interpolation

One of the limitations of the consumer-grade GPS unit that was used in this work is that it was limited to 1 update per second [9]. While this update rate is typical for consumer grade GPS systems, commercial and military based GPS rates with faster update rates can be more costly. For the post-processing algorithm, the limitation of the consumer-grade GPS used in this work was compensated by an interpolation scheme that generated reasonable headings in between known positions.
Once a test has been conducted with the test vehicle, a data file with the known GPS headings at certain times is available. In between these known GPS headings, a spline interpolation in MATLAB can be used with the `pchip` command, which is a Piecewise Cubic Hermite Interpolation Polynomial. The interpolation function interpolates a cubic polynomial in between two points in such a way that the first derivative of the polynomial interpolation is always continuous. The slope at each point is chosen so that the interpolation polynomial contains each known point and preserves local extremums. The result is presented in Figure 19.

![Figure 19 - Spline interpolation of known GPS headings](image)
Over the course of a single trial, there are many known GPS heading points that can be used for a smooth spline interpolation as shown in Figure 19. The interpolated heading directions of the spline fit represent reasonable heading values in between known GPS heading points that approximate realistic steering maneuvers. It was assumed that the interpolated heading directions represent the true path driven by the test vehicle.

Interpolation in between the GPS refresh rates was necessary to assign heading degrees to each magnetic sensor data point that was recorded in between the GPS update intervals.

Considering the calibration procedure, the GPS headings reflect the test vehicle travelling in a circular path, with headings linearly varying between 0° and 360°. An example of the GPS headings during the calibration procedure is displayed in Figure 20. At first glance, the plot may seem discontinuous, but the test vehicle is travelling between 0° and 360°, which demonstrates a wrap-around effect.
5.3.1.4 Results of Heading-Voltage Function

The calibration data is plotted for the x, y, and z-axis in Figure 21. The x and y axis are periodic, and an 8\textsuperscript{th} degree polynomial fit characterizes the voltage signals. An 8\textsuperscript{th} degree polynomial was chosen through empirical testing since it best represents the shape of the relationship between heading degree and voltage, as in Figure 21. It is also possible to assign a sinusoidal fit, but a polynomial fit was chosen since it can be characterized and implemented in code conveniently during this design phase.

The z-axis data varied slightly, but not as much as the ranges of the x and y-axis. The sum of squared residuals (SSR) value for the calibration data in Figure 21 are 0.11, 0.19, and 0.15 for the x, y, and z, axes respectively. These SSR values can be considered small and indicated that the polynomial fit can represent the data well.
The variation of the z-axis signal could have been attributed to the sensor or parking lot being tilted or contributions from noise in the sensor. However, it will be assumed that the variations in the z-axis will be insignificant.

It was assumed that the data represented by the voltages in Figure 21 are only due to the geomagnetic field. After the calibration procedure, a set of 8th degree polynomial coefficients in MATLAB were calculated from the polynomial fit of the data.

The polynomial fit, which characterizes only the geomagnetic field, can be used to subtract the effects of the geomagnetic field on each of the sensor axes if the GPS heading is known. This equation will be referred to as the heading-voltage function and will output the geomagnetic field’s voltage when the heading degree is the input. For example, if the input heading degree is 0°, the output of the heading voltage calibration function should be a value that is approximately -0.01 V for the x-axis and -0.2 V for the y-axis, according to Figure 21.
Figure 21- Calibration data: geomagnetic field as a function of heading
5.3.2 GPS Compensation Algorithm

Once the polynomial fit of the calibration data and the GPS headings were available, an algorithm to compensate for the Earth’s magnetic field was developed. It is graphically displayed in Figure 22, which is named Compensation Scheme 1 (CS1) or “GPS compensation”.
Figure 22 - CS1 Algorithm
For this compensation scheme, the GPS heading was matched with the corresponding magnetic sensor data for a specific point in time. The GPS heading was used as an input to the heading-voltage function generated in the calibration procedure. The output voltage of the geomagnetic field from the heading-voltage function as in Figure 21, were subtracted from the corresponding calibrated magnetic sensor values. The output was a compensated magnetic sensor voltage in the x, y, and z axes that did not include the effects of the geomagnetic field.

For one of the studies, the test vehicle was driven through San Luis Obispo, CA. The course headings and GPS map are displayed in Figure 23.

The results of Compensation Scheme 1 (CS1) for the x-axis is displayed in Figure 24. The uncompensated data is red and the compensated data is in blue.
Figure 24- Compensation Scheme 1 Results

From Figure 24, the uncompensated red data reflects large voltage fluctuations due to changes in heading of the test vehicle, such as the large change in voltage at the index before $5 \times 10^5$. When the CS1 algorithm is used, the geomagnetic component of the earth is subtracted and brings the data to a base level that only displays voltage fluctuations of magnetic anomalies around the magnetic sensor. Therefore, the blue compensated data in Figure 24 can be interpreted as magnetic fluctuations due to magnetic anomalies in the environment, such as passing vehicles. It should be noted that when certain patterns of voltage fluctuations are observed in the uncompensated (red) data, the same voltage fluctuation pattern is preserved in the compensated (blue) data in Figure 24.
Since the CS1 algorithm only subtracted the measured contributions of the geomagnetic field, the results of the algorithm as a geomagnetic field compensation technique was considered a success since the output voltage only measured the magnetic signatures of ferrous objects in the environment, such as vehicles, and not the geomagnetic field.

5.3.3 Compensation Scheme 1 Limitations
Although the CS1 algorithm successfully compensated for the geomagnetic field as displayed in Figure 24, the algorithm worked only when each magnetic voltage reading had a corresponding accurate GPS heading. However, because the GPS unit only updates once per second, only 1 in every 1000 magnetic sensor data points have a corresponding GPS heading.

The effect of this limitation is shown in a driving simulation in Figure 25 where the CS1 algorithm is used in a real-time program without a dead reckoning system. A sample and hold method for the GPS heading is used to handle the GPS updates. The plot on the top of Figure 25 is the uncompensated and compensated magnetic sensor x-axis data and the plot on the bottom is the corresponding heading direction of the magnetic sensor.
If the GPS headings were processed with a sample and hold scheme, the CS1 algorithm will only process a GPS heading once every second due to the limitations of the GPS hardware. Thus, the algorithm will continuously use the last known GPS heading position as the input to the heading-voltage function until it gets a new GPS heading position one second later. This sample and hold method is reflected in the simulated GPS headings on the bottom plot of Figure 25, where the flat tops of the heading direction symbolize the algorithm retaining the last known GPS heading. The jumps in headings are due to the algorithm receiving an updated GPS heading coordinate every second. The jagged lines of the heading directions do not reflect the smooth and continuous driving lines of a vehicle.
The GPS heading directions obtained from a sample and hold scheme were used as the input heading for the heading-voltage function of the geomagnetic field (Section 5.3.1.4 Results of Heading-Voltage Function), and the compensated signal is displayed in red on the top plot of Figure 25. Because the GPS headings were discontinuous, the corresponding voltages generated from the heading-voltage function of the geomagnetic field are also discontinuous. Thus, when the discontinuous voltages are subtracted from the calibrated signal, the resulting compensated signal will be an unreasonable discontinuous signal.

Thus, the one update per second limitation of the GPS does not make CS1 a suitable algorithm for a real-time program without a dead reckoning system or a GPS unit that updates faster at once per second, which can become costly.

Another limitation of GPS is that sometimes it may lose a valid GPS signal. The effect of this limitation is displayed in Figure 26. At approximately the $1.4 \times 10^5$ index, the GPS signal was lost during a slow, 3-point turn, as displayed by the discontinuous jump in GPS heading and interpolated GPS heading.
Due to the fact that the GPS signal was lost, the “incorrect” GPS headings were used in the heading-voltage function of the geomagnetic field. The incorrect GPS signals generated an incorrect voltage that was subtracted from the continuous compensated readings, as shown by the discontinuous compensated voltage signal (blue line) on the top plot of Figure 26. Therefore, the CS1 algorithm will not give the proper compensated readings if the GPS signal is lost.
5.4 COMPENSATION SCHEME 1.5: “IGRF COMPENSATION”

Compensation Scheme 1.5 (CS1.5), or “International Geomagnetic Reference Field (IGRF) compensation”, is similar to CS1 in that it subtracts the components of the geomagnetic field according to a reference value. However, instead of subtracting geomagnetic field values based on field testing data, CS1.5 uses the mathematical IGRF model of the geomagnetic field according to Equations (9) through (12) as discussed in Section
2.2.2 Mathematical Model of the Geomagnetic Field.

The advantage to using the IGRF model is that it is less dependent on the field calibration model. In CS1, the characterization of the geomagnetic field through the heading-voltage function relies on field calibration data. If the field calibration data included effects of magnetic anomalies, such as an underground pipe, the characterized geomagnetic field will contain errors. In some cases, it is not possible to remove these anomalies or find a particular location absolutely free from magnetic disturbances. In contrast, CS1.5 will use theoretical values to characterize the geomagnetic field, which do not contain influences from magnetic anomalies.

One important note is that there is an addendum to the scaling factor equation in Equations (18) and (19). The following is the modified scaling factors for the x and y-axis, which was originally proposed in [22].

\[
X_{igrf} = \frac{2|H|}{X_{\text{max}} - X_{\text{min}}} X_{sf} 
\]  
(22)

\[
Y_{igrf} = \frac{2|H|}{Y_{\text{max}} - Y_{\text{min}}} Y_{sf} 
\]  
(23)

Similarly, the magnetic sensor z-axis will also be scaled to the magnetic sensor reading.

\[
Z_{igrf} = \frac{2|Z_{igrf}|}{Z_{\text{max}} - Z_{\text{min}}} Z_{sf} 
\]  
(24)

Where
\( X_{\text{IGRFsf}}, Y_{\text{IGRFsf}}, \text{and } Z_{\text{IGRFsf}} = \) scaling factors for magnetic sensor readings for IGRF calculation

\( Z_{\text{IGRF}} = Z \) component of IGRF field

The modified IGRF scaling factors scale the magnetic readings to those computed by the IGRF model. The rest of the calibration procedure and calibration equations are unchanged.

5.4.1 IGRF Compensation Algorithm

The CS1.5 compensation algorithm is similar to the algorithm presented in Figure 22. However, instead of fitting a polynomial trend line to the calibration data as in Figure 21, Equations (9) through (12) were used to compute the mathematical geomagnetic field at a certain location using the latest IGRF tabulated values for the Gauss coefficients.

A MATLAB program originally written by Charles Rino [24] was used to calculate the IGRF values for a certain GPS coordinate. The algorithm for CS1.5 is displayed in Figure 27.
Compensation Scheme 1.5: IGRF Compensation

GPS Coordinates

IGRF Calculation

IGRF Components in X, Y, Z axis

[Calibrated Magnetic Sensor Data] – [Polynomial Fit Function Output]

Magnetic Sensor Data: X, Y, Z axes

IGRF Sensor Calibration

Calibrated Magnetic Sensor Data: X, Y, Z axes

Compensated Magnetic Sensor Data in X, Y, Z axes

Figure 27 - CS1.5 Algorithm
The GPS coordinates are processed in the IGRF calculations to produce the mathematical representation of the geomagnetic field in the x, y, and z-axes for that coordinate point. The calculation algorithm then converts these magnetic fields into equivalent voltages. At the same time, the corresponding magnetic sensor data is calibrated using the scale factors in Equations (22), through (24). The magnetic field voltages generated from the IGRF calculations are then subtracted from the calibrated magnetic sensor voltages. The result will be a compensated voltage signal in the three orthogonal axes that does not include the effects of the geomagnetic field.

Results of using the CS1.5 algorithm are displayed in Figure 28 for the x-axis during one of the trial studies.

Figure 28- CS1.5 Algorithm Results. CS1.5 is in red and CS1 is in blue.
The compensated CS1.5 (red) algorithm is compared to the compensated CS1 algorithm (blue). The overall signal of the CS1.5 algorithm matches closely with the CS1 algorithm, however the CS1.5 signal is scaled due to the different scaling coefficients in Equations (22) and (23). But in general, the voltage fluctuations of both algorithms do not include effects of the vehicle turning, and the patterns of fluctuations due to magnetic anomalies are present in both compensated signals. Therefore, the CS1.5 algorithm is another alternative to compensating for the geomagnetic field.

5.4.2 Compensation Scheme 1.5 Limitations

Limitations of this algorithm are similar to that of the CS1 algorithm, such as GPS dependency. Although the geomagnetic field varies slowly within the same area, the 1-Hz update rate of the GPS signal will still result in a slightly discontinuous compensated signal, as in Figure 25, since the heading of the vehicle needs to be known at all times. In addition, this method is still susceptible to GPS signal loss.

Although the CS1.5 algorithm is still limited by the same issues as CS1, this method can offer an alternative method that relies less on field calibration data where errors may exist. For example, the values calculated for the local magnetic field will not be susceptible to data that was calibrated during an active day where the natural magnetic field changes a finite amount throughout the day.
5.5 COMPENSATION SCHEME 2: HIGH-PASS FILTER

Since CS1 and CS1.5 depend on GPS, another algorithm was developed that does not rely on GPS, which was Compensation Scheme 2 (CS2). The variations of magnetic voltages when a vehicle turns within the geomagnetic field can be viewed as slowly varying changes in voltage that are at a lower frequency than the higher frequency signals of cars passing by. Therefore, it is possible to attenuate these low frequencies with a high pass filter. The high pass filter that is used will be a second order Butterworth filter, which is the same filter type that was used in the previous work in [25].

5.5.1 Preliminary Investigation

Before applying a high pass filter, a cutoff frequency was needed to characterize the slowly varying changes in voltage due to the geomagnetic field.

First, a spectral analysis was done on the magnetic sensor data of the test vehicle turning in a circle during the calibration procedure. During the calibration procedure, the test vehicle turned at 18° to 20° per second and it was assumed that the changes in voltage during this time are only due to the magnetic sensor measuring its change in orientation within the geomagnetic field.

A spectral analysis was done to reveal the dominant low frequency changes due to the variation of the sensor changing its orientation within the geomagnetic field for that turning speed. The spectral analysis plot is displayed in Figure 29 with a sampling frequency of 1000 Hz.
The dominant low frequency in this 18° to 20° per second circular turning procedure is less than 0.1Hz. However, a vehicle can experience much faster turning rates during normal driving, such as a 90° right turn onto an adjacent street that lasts for less than 2 seconds. Therefore, the plot in Figure 29 serves as a starting point for the cutoff frequency. Through empirical testing and simulations, it was found that the cutoff frequency of 0.488 Hz is a suitable cutoff frequency that suppresses most of the voltage fluctuations of a vehicle turning within the geomagnetic field. Figure 30 shows the effect of using different frequencies as a cutoff frequency for a second order Butterworth filter.
The original uncompensated signal is displayed in blue. As the cutoff frequency is increased, the low frequency signals are attenuated to the point where only higher frequencies existed in the signal.

Figure 30- Effects of hi-pass cutoff frequency on experimental data

Analysis of a voltage fluctuation due to the test vehicle changing directions within the geomagnetic field can be seen at an index of approximately $4.7 \times 10^5$ in the uncompensated signal (blue) of Figure 30. This voltage fluctuation is attenuated, and is shown in detail in Figure 31.
As the cutoff frequency is increased, the voltage fluctuation due to the test vehicle turning is attenuated.

From Figure 30 and Figure 31, if a low cut off frequency is used to filter the signal, then the voltage fluctuations due to the vehicle turning within the geomagnetic field will not be attenuated. If a cut off frequency larger than 0.488 Hz is used, then there is the undesirable possibility that higher frequency signals, such as cars passing, will also be attenuated as shown in Figure 32. As the cutoff frequency of the high pass filter is increased, larger amplitude signals are attenuated. Thus, it is suggested that the low cut off frequency that should be used is approximately 0.488 Hz.
5.5.2 Hi Pass Filter Algorithm

The algorithm for CS2 is shown in Figure 33. In this algorithm, the magnetic sensor data are calibrated and then directly processed by the high pass filter. The advantage to this algorithm is that it is free from GPS dependence other than the initial calibration procedure used to calibrate the data. Therefore, this algorithm will not have the limitations of CS1 and CS1.5, which is the limited update rate of the GPS and possible signal loss. In addition, the algorithm should process quickly in a real-time program since it is only a second order filter.
Results of the CS2 algorithm in comparison to the uncompensated signals and the CS1 algorithm are displayed in Figure 34.
Figure 34- Results of CS2 algorithm compared to uncompensated data and CS1 algorithm

Compared to the uncompensated signal (green), the compensated voltage signals generated by CS2 (red) do not contain large fluctuations in voltage due to the test vehicle turning. The higher frequency signals contained in the uncompensated data are also present in the CS2 compensated data. It can then be concluded that this algorithm is a successful solution to compensate for the Earth’s magnetic field bias.
If the compensated signals of CS1 and CS2 are compared, the CS2 signal removes information about the total ambient magnetic field magnitude and only keeps the higher frequency signals. Thus, the CS2 algorithm can be used to detect passing cars because those scenarios produce higher frequency voltage changes, which the CS2-processed outputs retain. Furthermore, the patterns of the voltage fluctuations are slightly shifted in the CS2 signal due to lower frequency signals that are filtered out. Therefore, if the CS2 algorithm is used as the signal to formulate a threshold voltage level to detect a passing vehicle’s presence, as in Equations (1) and (2), a different voltage threshold voltage level would be used with CS2 than with CS1.

5.5.3 Compensation Scheme 2 Limitations

Due to the fact that the high pass filter removes low frequency signals, information about the magnitude of the ambient magnetic field is missing from the signal. For example, the signal from nearby vehicles would be quickly attenuated by the filter and lost if the test vehicle is surrounded by other vehicles while remaining still in traffic. This scenario was encountered during testing and Figure 35 displays the sum of the magnitudes of each sensor axis (Equation (1)) produced by CS1 and CS2 as the test vehicle approaches a truck waiting at a stoplight.
Figure 35 - Comparison of CS2 and CS1 when approaching and stopping next to a truck at a red light
As the test vehicle approached the truck waiting at a stoplight, both CS1 and CS2 algorithms produced voltage signatures of magnetic anomalies in the environment, including the truck. Eventually, the speed of the test vehicle slowed down as it approached next to the stopped truck. Because the speed of the test vehicle relative to the truck was very slow, the CS2 algorithm (blue signal in Figure 35) attenuated the slowly changing voltage and the resulting signal magnitude decreased and died away. However, because the CS1 algorithm (red signal in Figure 35) is capable of measuring the magnitude of the ambient field, approaching the rear axle of the truck produced a decrease in the magnitude in voltage, which signified that the origin of the magnetic sensor axes were near the center of the axle. When the test vehicle finally stopped in the middle of length of the truck, the CS2 algorithm attenuated the constant voltage signal (which is considered as a low frequency signal) while the CS1 algorithm measured the increase in the ambient magnetic field due to the presence of the truck.

The example in Figure 35 demonstrates that when the speed of the test vehicle relative to another vehicle on the road is slow or close to zero, the CS2 algorithm will filter out the voltage signals.
5.6 EVALUATION OF ALGORITHMS

5.6.1 Proposed Fusion Compensation Scheme

A summary of the merits and demerits of each of the algorithm is displayed in Table 1.

<table>
<thead>
<tr>
<th>Method</th>
<th>Merit</th>
<th>Demerit</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS1: GPS</td>
<td>• Can detect cars present in most conditions</td>
<td>• GPS limited to 1 update per second</td>
</tr>
<tr>
<td>CS1.5: IGRF</td>
<td>• Can detect ambient magnetic field</td>
<td>• GPS signal can be lost which will give inaccurate compensated signals</td>
</tr>
<tr>
<td>CS2: Hi-Pass</td>
<td>• Independent of GPS limitations</td>
<td>• Cannot detect cars that pass slowly or when stopped</td>
</tr>
<tr>
<td></td>
<td>• Can process data quickly</td>
<td></td>
</tr>
</tbody>
</table>

Considering the compensation schemes presented, the advantages and limitations of each of the compensation schemes complement each other. While CS1/1.5 can detect the ambient magnetic field, CS2 cannot. However, CS1/1.5 is limited by the update rates and signal dependency of the GPS hardware, while CS2 is independent of a GPS. Thus, a proposed compensation scheme will combine these methods to support each other’s strengths and weaknesses. The algorithm is graphically displayed in Figure 36.
The GPS speed will be used in order to determine which compensation scheme to use based on vehicle speed. If the GPS outputs a vehicle speed equal to or greater than 1 MPH, the vehicle will be considered as moving. If the GPS speed is less than 1 MPH, it will be considered stationary and will include scenarios where a vehicle slowly creeping in traffic at speeds between 1 MPH and 0 MPH.

Figure 36- Fused compensation scheme algorithm
If the vehicle’s speed is greater than 1 MPH, the limitations of the GPS update rate are not desired and only cars passing closely are of interest. Therefore, the CS2 algorithm will be used. In contrast, if the test vehicle is stationary, then detecting other stationary cars in the adjacent lanes becomes a goal. Thus, the CS1 or CS1.5 algorithm can be used since it can measure the total ambient field. In addition, since the test vehicle is not moving quickly, the sample and hold method of GPS heading direction should not impose any limitations on the signal. This fused algorithm combines the strengths and weaknesses of the compensation schemes and will provide a basis for future vehicle detection algorithms using the magnetic sensor.

5.6.2 Evaluation Method

In order to evaluate the effectiveness of the proposed compensation schemes, the rate of vehicle detection was used as a benchmark. It should be noted that the goal of this thesis work was not vehicle detection, since this was the work explored in [25] and an ultrasonic sensor is needed in order to detect vehicles effectively. The goal of this thesis work was to compensate for the geomagnetic field, which was successfully done through the proposed compensation schemes. Vehicle detection rates will be used as a measurement tool to evaluate geomagnetic compensation algorithms that can be further developed in the future. The relatively low rates of success in this work should not be a cause for alarm, but should be interpreted as a comparison tool to evaluate the proposed compensation schemes. Explanations for the specific results obtained are explored in Section 5.6.4 Sources of Error.

A binary form of vehicle detection based on the magnitude of (1) was used. If the value of (1) exceeds a certain voltage, then a simulation will evaluate that a vehicle is present.
In order to choose a voltage threshold, a statistical analysis was done based on the magnitudes of the voltage ranges (using Equation [1]) when a car is present and absent. The box plots of the analysis are shown in Figure 38 and Figure 37. It should also be noted that the signals generated by CS1/1.5 were shifted down to the 0 Volts axis by the average of the signal for comparison purposes. Subtracting the average of the signals was done in this work because all of the data is known for a trial due to the post-processing environment. If a future detection algorithm is used based on the fused compensation scheme in Figure 36, voltage detection thresholds will need to be redefined since the signals cannot be shifted by the average of constantly changing signal.

Figure 37 - Boxplot of voltage ranges when vehicle is stopped using CS1 algorithm
From Figure 37, 0.03V was approximately the 25th percentile of voltage ranges when a car is present for the scenarios when CS1 was used. This will be the minimum voltage level when the algorithm detects a vehicle is present. This voltage level was chosen in order to provide some level of protection against accepting a voltage magnitude that will falsely detect a vehicle. For example, setting a voltage detection threshold level that is too low may cause the evaluation algorithm to detect a false positive, such as a signal caused by an underground steel pipe that generates a magnetic field.
Similarly, the detection voltage range for CS2 is chosen to be 0.003V, which is between the 25th and 35th percentile voltage of the “vehicle present” category in Figure 38. The detection voltage range for CS2 will be limited to a maximum voltage of 0.07V to provide some level of protection from large magnetic anomalies encountered during freeway driving that is not typical of a vehicle. For example, driving underneath an overpass that uses a steel or iron truss may generate large voltage magnitudes outside the normal range of a vehicle passing.

The aforementioned voltage thresholds for CS1/CS1.5 and CS2 were used in order to detect a vehicle passing the magnetic sensor. When the resulting magnitude from Equation (1) was within the ranges specified for vehicle presence, the algorithm used to evaluate the compensation schemes will consider that a vehicle was present. Figure 36 displays the detection algorithm with the threshold voltages.

In order to compare the results that the evaluation algorithm makes with what actually happened, a video camera with a frame-by-frame timestamp was mounted beneath the magnetic sensor. When the front of a passing vehicle was seen at the edge of a frame, the start time was recorded up to the point where the rear of the vehicle was at the opposite edge of the frame. The timestamps of this vehicle pass were compared to the timestamps the evaluation algorithm sensed a car present.
Quantitative variables to evaluate the compensation schemes are presented in Equations (25), (26), and (27).

\[
\text{Vehicle present} = \frac{\# \text{ of points algorithm successfully detected a vehicle}}{\text{Total data points of a vehicle present from video}} \times 100\% \quad (25)
\]

\[
\text{False alarm} = \frac{\# \text{ of points algorithm falsely detected a vehicle}}{\text{Total data points when vehicle is NOT present}} \times 100\% \quad (26)
\]

\[
\text{Absent} = \frac{\# \text{ of points algorithm successfully did not detect a vehicle}}{\text{Total data points when vehicle is NOT present}} \times 100\% \quad (27)
\]

“Vehicle present” measured the percent success rate the evaluation algorithm detected a vehicle compared to the actual times a vehicle is present from the video. “False alarm” measured the data points the evaluation algorithm falsely detected a vehicle when a vehicle was not present in video. “Absent” measured the data points that the evaluation algorithm did not detect a vehicle compared to the total number of data points that a video recorded a vehicle was not present.

The fused compensation scheme and CS2 was evaluated with the detection criteria in Figure 36 and evaluation parameters in Equations (25), (26), and (27).
5.6.3 Evaluation Results

In order to thoroughly evaluate the compensation schemes, three driving scenarios were used in the observational study.

The first scenario is a heavy traffic scenario. The test vehicle was driven through city streets near Cal Poly San Luis Obispo, CA, primarily on Santa Rosa Road during rush hour traffic between 5PM-6PM. During this scenario, the test vehicle was constantly stopped at intersection stoplights along with other vehicles.

The second scenario is a highway-driving scenario on Highway 1 near Cal Poly San Luis Obispo, CA. The speed of the test vehicle was kept between 55 MPH to 65 MPH and traffic was light. However, several vehicles passed the test vehicle during the time.

The final evaluation scenario is a combination of highway driving and city driving. The route was from Cal Poly San Luis Obispo, to Madonna Road, Los Osos Valley Road, and back to Cal Poly. The highway-driving portion was through Highway 101 with light traffic and several vehicles passing the test vehicle. The city-driving portion included medium traffic.

The results of the evaluations are in Table 2. The Fusion algorithm with the CS1 and CS1.5 variant are compared to CS2. CS2 is evaluated independently because it can properly function without the limitations of GPS.
Table 2- Algorithm evaluation results

<table>
<thead>
<tr>
<th>Compensation Scheme</th>
<th>Street driving, heavy stop and go traffic</th>
<th>False Alarm (%)</th>
<th>Absent Success (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fusion w/ CS1</td>
<td>79.22</td>
<td>66.00</td>
<td>34.00</td>
</tr>
<tr>
<td>Fusion w/ CS1.5</td>
<td>68.42</td>
<td>75.02</td>
<td>24.98</td>
</tr>
<tr>
<td>CS2 Only</td>
<td>17.05</td>
<td>46.81</td>
<td>61.83</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Compensation Scheme</th>
<th>Freeway driving, light traffic</th>
<th>False Alarm (%)</th>
<th>Absent Success (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fusion w/CS1</td>
<td>81.63</td>
<td>59.39</td>
<td>40.62</td>
</tr>
<tr>
<td>Fusion w/ CS1.5</td>
<td>81.63</td>
<td>64.42</td>
<td>35.59</td>
</tr>
<tr>
<td>CS2 Only</td>
<td>81.63</td>
<td>59.37</td>
<td>40.64</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Compensation Scheme</th>
<th>Combined freeway and street driving, medium traffic</th>
<th>False Alarm (%)</th>
<th>Absent Success (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fusion w/CS1</td>
<td>63.33</td>
<td>50.14</td>
<td>49.86</td>
</tr>
<tr>
<td>Fusion w/ CS1.5</td>
<td>68.54</td>
<td>56.28</td>
<td>43.72</td>
</tr>
<tr>
<td>CS2 Only</td>
<td>63.55</td>
<td>49.57</td>
<td>50.43</td>
</tr>
</tbody>
</table>

For the “street driving, heavy stop and go traffic” scenario, CS2 yielded a 17.05% vehicle detection rate. When the Fusion algorithm with CS1 or CS1.5 was included to detect nearby vehicles when the test vehicle was stopped, the success detection rate increased to 79.22% with the Fusion-CS1 algorithm and to 68.42% with the Fusion-CS1.5 algorithm.
During traffic scenarios where the test vehicle remains stationary along with other vehicles at a stoplight, the CS2 algorithm does not detect the ambient magnetic field and produces a signal that was highly attenuated. However, using a compensation scheme that can detect ambient magnetic fields such as CS1 or CS1.5 increases the vehicle detection rates because it included magnetic field information contributed by the presences of other vehicles. The absent success rates were lower than the detect success rates due to the environment of the street driving scenario. Often times, the test vehicle was near other magnetic anomalies during the experiment, such as metallic poles, buildings, and possibly other magnetic sensors at stop light intersections. However, the high detection rates of the Fusion algorithm show the advantage of using either CS1 or CS1.5 to detect vehicles when the test vehicle is stationary, as opposed to only using CS2.
For the freeway driving scenario, the Fusion schemes and the CS2 algorithms had similar detection rates of 81.6%. Since this driving scenario involved the test vehicle driving on the freeway with very few stops, the CS2 algorithm was employed during most of trial. Thus, the CS1 and CS1.5 algorithms were not used as often to evaluate vehicles passing. The absent success rates were lower than the successful detection rates due to other magnetic anomalies in the driving route. For example, when the test vehicle was driving adjacent to a concrete divider, large, periodic voltage fluctuations were present in the data. It is possible that these signals could have been from steel concrete rebar, which created a large voltage magnitude within the range of vehicle detection. Although the evaluation results between Fusion-CS1 and CS2-only did not differ for this freeway driving scenario, it demonstrated that including a Fusion algorithm in a scenario where the test vehicle is constantly moving did not significantly impact the success rates. It is suggested that the Fusion algorithm should be used in the crash avoidance system since it does not affect the performance of vehicle detection even if the driving scenario consists mainly with freeway driving.
The last traffic scenario explored was a combined freeway and street-driving scenario under medium traffic. For the algorithm evaluation, the Fusion-CS1 algorithm and CS2 only algorithms performed similarly. However, Fusion-CS1.5 had a 6% greater detection rate with a 6% decrease in absent success rate most likely due to different scaling factors of the IGRF calibration. Regardless, the Fusion and CS2 algorithms performed similarly when compared to each other due to traffic configurations and the type of driving done.

For example, most of the passing vehicles were present when the test vehicle was moving, so the CS2 algorithm was frequently used. Since this study involved car passes in the freeway, interpretation of the results is similar to the freeway-driving scenario, and the Fusion algorithm is still recommended for the crash avoidance system since the traffic conditions can change into a heavy traffic scenario where the Fusion algorithm can detect vehicles that the CS2 algorithm cannot.

From these results, it can be concluded that there are scenarios where the Fusion algorithm can benefit the detection system, such as the heavy traffic scenario where the CS1 and CS1.5 algorithms detected the ambient magnetic field from nearby cars while stuck in traffic that the CS2 algorithm did not detect. The Fusion compensation scheme can also be useful for Class 8 trucks during city driving. For example, the truck driver should be alerted when a vehicle is stopped in a blind spot in an adjacent lane, in case they wanted to switch lanes once the traffic starts to move.
A discrete definition for a “good” success or absent rate cannot be done in this work since vehicle detection rates are only based on the magnetic sensor. Direct comparison with Roussel’s work in [25] cannot be done because that work fused magnetic sensors and ultrasonic sensors to detect vehicles. However, the results in Table 2 can be improved with the addition of an ultrasonic sensor and a Bayesian filter that was used in [25]. The success and absent rates in Table 2 can be interpreted as base rates that can be improved when these compensation schemes are implemented into the crash avoidance system.

5.6.4 Sources of Error

This section explores explanations for underlying variables that contributed to the evaluation results in Table 2.

One of the limitations of this evaluation is that confirmation of a vehicle’s presence is only based on one camera that pointed in a single direction. During the trial studies, the test vehicle was driven in a way so that the only vehicle in proximity of the magnetic sensor was captured by the camera. However, due to the dynamic conditions of real-world traffic, this ideal situation did not happen all the time. It was possible that another vehicle was in the proximity of the magnetic sensor during a car pass that was captured by the camera and could have increased the “false alarm” percentage in Equation (26) and decrease the “absent success” percentage in Equation (27).
Similarly, there were other magnetic anomalies that existed in the environment that produced voltage magnitudes similar to passing vehicles. Concrete rebar, light poles, railroad tracks, and building frames are some objects in the environment that could have produced similar voltage magnitudes as passing vehicles. Since the video did not capture any passing vehicles during these times, “false alarms” will increase and “absent success” will decrease. It should be noted that Roussel’s work in [25] addressed this issue by combining magnetic with ultrasonic sensors, which yielded much better detection rates.

Furthermore, the distance to the passing vehicle and the magnetic sensor was not considered. The closer a vehicle is the magnetic sensor, the greater the magnitude of the signal. For example, a car that passed the test vehicle at a close distance with a relatively fast speed generated a large signal (Equation (1)) as in Figure 39. As the car approached the sensor at an estimated distance of less than 5 feet away, the signals generated from CS1, CS1.5, and CS2 are relatively large in comparison to a vehicle that is further away from the magnetic sensor in Figure 40.
Figure 39- Algorithm signals when a car passes at a close distance

When a vehicle passed the test vehicle with a greater distance from the magnetic sensor, the signal generated from CS2 was relatively small as in Figure 40.
It is estimated that the distance of the passing car in Figure 40 was approximately greater than or equal to 4 feet away. According to research data, the signal magnitude decreases as the distance from the magnetic sensor to the passing vehicle increases. Specific works conclude that the magnitude of the signal decreases 96.3% when it is 5 feet away from the vehicle [5] [13].
The example of the car pass in Figure 40 may be an example of a small magnitude signal due to a passing car being far away from the magnetic sensor. In these scenarios, the magnetic sensor will not detect the passing vehicle while the camera does. These scenarios will decrease the “vehicle present” percentage in Equation (25) because the camera detects a car’s presence while the magnetic sensor does not due to a weak signal magnitude that does not exceed the voltage detection threshold. However, this issue was addressed in Roussel’s work in [25] through the combination of a magnetic and ultrasonic sensor. Once this solution for geomagnetic compensation is incorporated into a real-time crash avoidance system, the detection rates should improve.
6. CONCLUSIONS

6.1 GENERAL SUMMARY AND CONCLUSIONS

Several solutions for geomagnetic compensation have been shown in this work while incorporating a magnetic sensor calibration procedure into the system developed in [25]. The calibration procedure compensated for hard and soft iron magnetic effects when mounting the sensor to a test vehicle and allows for consistent magnetic readings regardless of the test vehicle type or cargo carried.

The first solution, CS1, used field calibration values to characterize the geomagnetic field. When it is assumed that the voltage data gathered at certain heading directions are only due to the geomagnetic field, a polynomial fit can be made from the data to form a heading-voltage function. The heading-voltage function used GPS headings as the input and output a voltage that corresponds to the observed geomagnetic field. This value was subtracted from the calibrated sensor data to produce a signal free from geomagnetic effects, as in Section 5.3.1.4 Results of Heading-Voltage Function.

The second solution, CS1.5, used the mathematical model of the geomagnetic field with IGRF (International Geomagnetic Reference Field) coefficients to subtract the effects of the geomagnetic field. GPS coordinates are used to calculate the x, y, and z-axis components of the geomagnetic field and are then subtracted from the calibrated magnetic sensor data to produce a compensated voltage free from magnetic effects.
However, CS1 and CS1.5’s dependence on GPS limited its usefulness with the hardware used in this work. The GPS system only updated at 1Hz, but the aforementioned algorithms need a GPS coordinate or heading direction with each magnetic sensor data point. The magnetic sensor sampled at 1000Hz, thus magnetic sensor data points without corresponding GPS data produced discontinuous signals from the CS1 and CS1.5 algorithms. There were also instances during testing where the GPS lost its signal and produced inaccurate compensated signals. These limitations demonstrated the necessity of another type of compensation algorithm.

The CS2 algorithm used a high-pass, second order Butterworth filter to compensate for the geomagnetic field. The rate at which the magnetic sensor changes its orientation within the geomagnetic field is viewed as a signal that varies slowly, and can be attenuated with a high-pass filter. Results of this algorithm showed success in eliminating voltage fluctuations due to the geomagnetic field while being independent of the limitations of GPS. However, this algorithm also lost information about the local ambient magnetic field and will output a small signal in scenarios where it is stopped near other vehicles in stop-and-go traffic.

The strengths and weaknesses of these algorithms complemented each other and were combined in a fused compensation algorithm displayed in Figure 36. The CS2 algorithm was used in instances where the vehicle is moving and only passing vehicles are of interest. The CS1 or CS1.5 algorithms was used when the vehicle is not moving in order to detect vehicles in proximity of the test vehicle.
An evaluation method was proposed that used the rate of vehicle detection to judge the fused algorithm’s effectiveness as opposed to only using the CS2 algorithm. Although the goal of this thesis work was not vehicle detection, the evaluation demonstrated the benefits of the fused algorithm to compensate for the unique limitations of each algorithm. The fused algorithm showed a high detection rate in scenarios where the vehicle was used in heavy traffic scenarios compared to the CS2 algorithm. It is suggested that a form of this algorithm should be implemented with the crash avoidance system.

Since these algorithms were developed in this work, the crash avoidance system started in [25] is no longer limited successful vehicle detection to driving in a single direction. With the implementation of the geomagnetic compensation algorithms presented in this work, the crash avoidance system can detect cars while the vehicle is navigating through streets and freeways in changing directions. Further development with the crash avoidance system is now possible.

The unique aspects of this work is that it only uses a single magnetic sensor and a consumer grade GPS to keep the crash avoidance system at a low cost for the user. Other solutions to geomagnetic compensation exist, but use multiple magnetic sensors or other software that add to the cost of the system. The work in this thesis can also assist other work with magnetic sensors that only desire signals from magnetic anomalies from the environment with a limited budget.
6.2 FUTURE WORK

In order to refine and implement the compensation algorithms presented in this work, implementation of tilt compensation and a dead reckoning system with the crash avoidance system is suggested.

One of the major assumptions in this work is that the magnetic sensor’s x and y-axes are constantly parallel with the horizontal plane of the geomagnetic field, H. While this assumption works well in leveled city streets, tilt-compensated hardware and software can allow the CS1 and CS1.5 algorithms to be used on roads with steep inclines and declines. When the test vehicle is driving up a hill, the magnetic sensor’s x and y-axes are no longer parallel to the horizontal component of the geomagnetic field and the signals generated by CS1 and CS1.5 will not be accurate. Section 3.2.1.1 Tilt Compensation details tilt compensation techniques using additional sensors such as gyroscopes and includes tilt compensated equations for the x and y-axes (Equations (13) and (14)) that can be included in the future crash avoidance system.

This work was also constrained by consumer-grade GPS hardware that is limited to only 1 update per second. This affects the usefulness of the CS1 and CS1.5 algorithms since the update rate can result in discontinuous compensated signals if a sample and hold method was used for the heading directions, such as Figure 25. In addition, temporary GPS signal loss can also cause a discontinuous compensated signal, as in Figure 26. By implementing a dead reckoning system, such as the works mentioned in Section 3.2 Navigation Purposes, each voltage data point can have a corresponding heading or coordinate point. This will yield compensated signals from CS1 and CS1.5 that are less discontinuous even if GPS signal is lost.
Furthermore, the crash avoidance system can benefit from set/reset hardware for the magnetic sensor. Some of the initial challenges with this work were working with a magnetic sensor that was suspected to lose its factory sensitivity specifications. Some of the magnetic axes were inverted 180° and there was a loss of sensitivity in the sensor. Implementing magnetic reset hardware, as discussed in [3] can maintain magnetic sensors properly.

The algorithms for geomagnetic compensation presented in this work should be considered as a starting point for an improved vehicle detection algorithm with the magnetic sensor. Distinguishing magnetic signatures from vehicles and other magnetic anomalies, such as bridges and railroad tracks, will increase the successful detection rate of the crash avoidance system. A suggestion would be an analysis of the separate x, y, and z-axis voltage components of a car passing compared to other magnetic anomalies such as a bridge. It is possible that driving underneath an overpass will have a large z-axis voltage compared to a vehicle passing the magnetic sensor in the same horizontal plane. Therefore, a separate voltage threshold for the x, y, and z-axis or an analysis of the rate of change of the signal in each of the axis can help increase the success of the crash avoidance system.

Implementing these suggestions in the low-cost crash avoidance system can help progress the project as a commercial product to help decrease the amount of vehicle related accidents. Suggestions noted in this section should be viewed as interesting challenges and it is encouraged to further develop this project to increase roadway safety and save lives.
REFERENCES


APPENDIX: MATLAB PROGRAM AND DOCUMENTATION

In order to process the data, a MATLAB program was created that calibrated the magnetic sensor, and compensated the data according to the algorithms described in the previous sections. The program flowchart is displayed in Figure 41. The program main.m contains a text-based user interface that asks the user what action to take with a given data set: calibration, compensation, or both. The following pages are the MATLAB source codes and documentation.

Figure 41- MATLAB program flowchart
CALIBRATION ERROR REFINEMENT
John Torres, 2011 jctorres@calpoly.edu Cal Poly SLO ME Master's Thesis

Contents

- Function Call
- Description
- READ ME Notes
- Input Variables
- Output Variables
- Program Code Begin
- Initial Error Refinement
- Second Error Refinement

Function Call

\[ \text{function} \]
\[ [\text{xfit\_loop}, \text{xfit\_loop\_stat}, \text{yfit\_loop}, \text{yfit\_loop\_stat}] = \text{calibration\_error\_refinement}(\text{shiftx\_calib, shifty\_calib, fitted\_headings}) \]

Description

If the calibration data contain any outside anomalies such as a car passing by near the calibration point, it will cause errors in the calibration procedure by superimposing voltage on the characteristic geomagnetic field. This different voltage in a certain degree heading may cause the polynomial fit to drift away from the true calibration value. If the correlation factor for the model fit is less than .99, this program will calculate the difference between the data and the fitted polynomial model and eliminate any data points in the x and y axis that are greater than 0.01V. This will be called an error. A new polynomial fit will then be constructed from the remaining data.

READ ME Notes

- Please note that this error refinement and conditioning algorithm will systematically remove some data in the fitting procedure. If absolutely bad data is used, a polynomial fit may not result in a sinusoidal wave. Please consider this before using this program. This should be used as a last resort just in case if a calibration cannot be done.
- If refinement is done more than twice, this program will give an error for the values of xfit and yfit. This is done to discourage the user from using bad calibration data.
- This program only assumes that the vehicle calibration was done on a flat surface that is parallel to the Earth's surface. Therefore, only the x and y data are analyzed.

Input Variables
• **shiftx_calib** - The calibrated readings of the x axis shifted to the 0 axis
• **shifty_calib** - The calibrated readings of the y axis shifted to the 0 axis
• **shiftz_calib** - The calibrated readings of the z axis shifted to the 0 axis

**Output Variables**

• **x_fit_loop** - An 8th degree polynomial fit that has been refined in the x axis.
• **x_fit_loop_stat** - The statistics, such as R^2 value, of the 8th degree polynomial in the x axis.
• **y_fit_loop** - An 8th degree polynomial fit that has been refined in the y axis.
• **y_fit_loop_stat** - The statistics, such as R^2 value, of the 8th degree polynomial in the y axis.

**Program Code Begin**

**Initial Error Refinement**

This error refinement will remove any data points in the calibration data that are larger than 0.01V from the x and y calibration data.

```matlab
% Preallocate space for x_calib_error and y_calib_error, which will hold
% the difference between the calibrated reading and the fitted model interpolation.
x_calib_error=NaN(size(shiftx_calib,1),1);
y_calib_error=NaN(size(shiftx_calib,1),1);

% Calculate the error based on the difference between the calibrated reading and the fitted model interpolation.
for u=1:size(fitted_headings,1)
    x_calib_error(u,1)=shiftx_calib(u)-xfit(fitted_headings(u));
    y_calib_error(u,1)=shifty_calib(u)-yfit(fitted_headings(u));
end

% echart1 is a matrix used to organize calibrated readings corresponding
% with the related errors.
echart1=[shiftx_calib shifty_calib x_calib_error y_calib_error fitted_headings];

% This loop will keep readings that have errors less than 0.01 volts from
% the modeled polynomial fit and move it to a new matrix titled echart2.
p=1; % p is used as a separate counter for the loop to create echart 2.
for u=1:size(echart1,1)
    if abs(echart1(u,3))<=.01 && abs(echart1(u,4))<=.01  % If the x and y errors are less than 0.01 volts
        echart2(p,:)=echart1(u,:); % move it to echart2.
        p=p+1;
    end
end
```

126
% Create polynomial fits for the x and y calibration data in echart2.
[xfit_loop,xfit_loop_stat]=fit(fitted_headings,echart2(:,1),'poly8');
[yfit_loop,yfit_loop_stat]=fit(fitted_headings,echart2(:,2),'poly8');

echart_loop=echart2; % Copy the matrix to another variable

**Second Error Refinement**

If the polynomial model still does not fit the data, one more refinement phase will be taken to remove any voltage readings that are larger than 0.01V from the x and y calibration data.

% Check to see if the R^2 value of the x and y axis is still less than .99
if xfit_loop_stat.rsquare<.99 || yfit_loop_stat.rsquare<.99
    % Display a message to the user to ask if they would like to refine the data one more time.
    conditioning=input('The fitted model still does not fit well with the data. Would you like to refine the calibration data again? This will result in more data loss. [y/n]');

    % If the user wants to refine the data, the same refinement procedure will be taken.
    if conditioning=='y'
        % Calculate the error based on the difference between the calibrated reading and the fitted model interpolation.
        for u=1:size(echart_loop,1)
            x_calib_error_loop(u,1)=echart_loop(u,1)-xfit_loop(echart_loop(u,5));
            y_calib_error_loop(u,1)=echart_loop(u,2)-yfit_loop(echart_loop(u,5));
        end

        % echart_loop is a matrix used to organize calibrated readings corresponding with the related errors.
        echart_loop(:,3:4)=[x_calib_error_loop y_calib_error_loop];

        % This loop will keep readings that have errors less than 0.01 volts from the modeled polynomial fit and move it to a new matrix titled echart_loop2.
        p=1;
        for u=1:size(echart1,1)
            if abs(echart_loop(u,3))<=.01 && abs(echart_loop(u,4))<=.01 % If the x and y errors are less than 0.01 volts
                echart_loop2(p,:)=echart_loop(u,:);
                p=p+1;
            end
        end

    end
end
p=p+1;
end
end

% Create polynomial fits for the x and y calibration data in
% echart_loop2.

[xfit_loop,xfit_loop_stat]=fit(echart_loop2(:,5),echart_loop2(:,1),'poly8');

[yfit_loop,yfit_loop_stat]=fit(echart_loop2(:,5),echart_loop2(:,2),'poly8');

% If the rsquared value is still less than .99, consider the
% calibration data faulty and ask the user to do a new
% calibration.
if xfit_loop_stat.rsquare<.99 || yfit_loop_stat.rsquare<.99
    sprintf('The fitted model does not fit well with data. Further refinement should be avoided. Please use new calibration data.');
end

% Mark the data to state that it is Invalid Data.
xfit_loop='INVALID DATA';
xfit_loop_stat='INVALID DATA';
yfit_loop='INVALID DATA';
yfit_loop_stat='INVALID DATA';
end
end

Published with MATLAB® 7.10
GPS COMPENSATION ALGORITHM
John Torres, 2011 jctorres@calpoly.edu Cal Poly SLO ME Master's Thesis

Contents

• Function Call
• Description
• READ ME Notes
• Input Variables
• Output Variables
• Program Code Begin
• Data calibration
• Subtracting the Geomagnetic Field
• Program Code End
• Graph of the Results

Function Call

function [shiftx_pure_gps shifty_pure_gps shiftz_pure_gps]=gps_fix(fd,
fitted_headings, xsf, xoff, xfit, ysf, yoff, yfit, zsf, zoff, zfit)

Description

This program will first calibrate the low pass filtered data of each axis of the magnetic sensor. The sensor data will be calibrated using the variables listed in the "Input Variables" section according to the following equations:

\[ x_{\text{calib\_gps\_rated}} = x_{\text{raw}} \times sf + x_{\text{off}} \]

\[ y_{\text{calib\_gps\_rated}} = y_{\text{raw}} \times sf + y_{\text{off}} \]

\[ z_{\text{calib\_gps\_rated}} = z_{\text{raw}} \times sf + z_{\text{off}} \]

Then, the algorithm will subtract the geomagnetic field from the magnetic sensor readings based on the magnetic sensor field calibration polynomial fit. For each magnetic sensor axis, the components of the geomagnetic field characterized by the polynomial fit will be subtracted from the low pass filtered data at a certain heading degree from magnetic North.

\[ \text{compensated data} = \text{low pass filtered data}(\Theta) - \text{magnetic sensor calibration polynomial fit}(\Theta) \]

Where \( \Theta \) is the heading degree from magnetic North, according to GPS.

The program then shifts those readings to the y=0 axis to use for algorithm processing.
READ ME Notes

- If the magnetic sensor data contains many points, the subtraction of the geomagnetic field will take a long time. On a Macbook Pro running a 2.33 Ghz Intel Core 2 Duo Processor, 30 minutes worth of driving data takes somewhere between 3-4 hours to process. Therefore it is normal for this program to take some time to run. However, a real-time implementation of this algorithm will be much quicker, since it will do the calculation as it receives GPS data and magnetic sensor data, not all at once.

Input Variables

- \( fD \) = A matrix of magnetic sensor data after a low-pass Butterworth filter has been applied. The format for each of the columns of the matrix are as follows:

\[
[\text{time} \ \text{filtered}_x \ \text{filtered}_y \ \text{filtered}_z \ \text{fitted}_\text{headings}]
\]

- \( \text{fitted}_\text{headings} \) - a vector of heading directions from magnetic north for each magnetic sensor reading.
- \( xsf \) - Magnetic sensor x scaling factor
- \( ysf \) - Magnetic sensor y scaling factor
- \( zsf \) - Magnetic sensor z scaling factor
- \( xoff \) - Magnetic sensor x offset
- \( yoff \) - Magnetic sensor y offset
- \( zoff \) - Magnetic sensor z offset
- \( xfit \) - polynomial fit of the 8th degree that represent the x voltage from the geomagnetic field as a function of heading degree
- \( yfit \) - polynomial fit of the 8th degree that represent the y voltage from the geomagnetic field as a function of heading degree
- \( zfit \) - polynomial fit of the 8th degree that represent the z voltage from the geomagnetic field as a function of heading degree

Output Variables

*\( \text{shift}_x\_\text{pure}\_\text{gps} \) - a vector of the magnetic sensor x axis readings without the effects of the geomagnetic field using the GPS subtraction algorithm. These data are also compensated for the magnetic effects of the vehicle.

*\( \text{shift}_y\_\text{pure}\_\text{gps} \) - a vector of the magnetic sensor y axis readings without the effects of the geomagnetic field using the GPS subtraction algorithm. These data are also compensated for the magnetic effects of the vehicle.

*\( \text{shift}_z\_\text{pure}\_\text{gps} \) - a vector of the magnetic sensor z axis readings without the effects of the geomagnetic field using the GPS subtraction algorithm. These data are also compensated for the magnetic effects of the vehicle.

Program Code Begin
Data calibration

The following section will calibrate the data to compensate for local magnetic effects found in the vehicle.

```matlab
% The first step is to shift the magnetic sensor data down to the y=0 axis
% for calibration.
shiftx=fD(:,2)-mean(fD(:,2));
shifty=fD(:,3)-mean(fD(:,3));
shiftz=fD(:,4)-mean(fD(:,4));

% The following loop will calibrate each data to compensate for magnetic
% effects of the vehicle according to equations in the description.
for z=1:size(shiftx,1)
    shiftx_calib_gps(z,1)=xsf*shiftx(z,1)+xoff;
    shifty_calib_gps(z,1)=ysf*shifty(z,1)+yoff;
    shiftz_calib_gps(z,1)=zsf*shiftz(z,1)+zoff;
end
```

Subtracting the Geomagnetic Field

For each magnetic sensor reading, the polynomial generated by the calibration algorithm will produce a corresponding voltage reading for a magnetic sensor axis for the current heading degree from magnetic North. The following loop will go through each magnetic sensor reading and subtract the geomagnetic field to produce magnetic sensor data that is compensated for the geomagnetic field using field calibration values.

```matlab
for z=1:size(shiftx,1)
    shiftx_pure_gps(z,1)=shiftx_calib_gps(z,1)-xfit(fitted_headings(z,1));
    shifty_pure_gps(z,1)=shifty_calib_gps(z,1)-yfit(fitted_headings(z,1));
    shiftz_pure_gps(z,1)=shiftz_calib_gps(z,1)-zfit(fitted_headings(z,1));
end

% This next line of code will reshift the magnetic sensor readings back
to % the y=0 axis for use in a driving simulation algorithm.
shiftx_pure_gps=shiftx_pure_gps-mean(shiftx_pure_gps);
shifty_pure_gps=shifty_pure_gps-mean(shifty_pure_gps);
shiftz_pure_gps=shiftz_pure_gps-mean(shiftz_pure_gps);
```

Program Code End
Graph of the Results

The following commands will graph the results of the compensation. If GPS headings are accurate and not lost, the changes in voltages present in the raw readings will be brought towards the \(y=0\) axis in the GPS compensated readings (i.e. the peak to peak values of \(shiftx\_calib\_gps\), which is the uncompensated values, should be larger than the peak to peak values of \(shiftx\_pure\_gps\) which does not include the added voltages of the geomagnetic field.

```matlab
% The following code will graph the uncompensated versus the GPS
% compensated x axis data along with a subplot of the heading degree
figure()
subplot(2,1,1);
hold on
plot(shiftx_calib_gps,'b')
plot(shiftx_pure_gps,'r')
xlabel('Index')
ylabel('Voltage (V)')
title('Magnetic Sensor X Voltage: Uncompensated vs. GPS Comepnsated')
legend('Uncompensated','GPS Compensated')
hold off
subplot(2,1,2);
hold on
plot(fitted_headings,'b')
xlabel('Index')
ylabel('Heading Degree CW from Magnetic North')
title('Heading Degree CW from Magnetic North During Driving')
legend('Vehicle Heading Degree')
axis ([0 size(shiftx_calib_gps,1) 0 360])
set(gca,'YTick',[0 45 90 135 180 225 270 315 360])
hold off

% The following code will graph the uncompensated versus the GPS
% compensated y axis data along with a subplot of the heading degree
figure()
subplot(2,1,1);
hold on
plot(shifty_calib_gps,'b')
plot(shifty_pure_gps,'r')
xlabel('Index')
ylabel('Voltage (V)')
title('Magnetic Sensor Y Voltage: Uncompensated vs. GPS Comepnsated')
legend('Uncompensated','GPS Compensated')
hold off
subplot(2,1,2);
hold on
plot(fitted_headings,'b')
xlabel('Index')
ylabel('Heading Degree CW from Magnetic North')
title('Heading Degree CW from Magnetic North During Driving')
legend('Vehicle Heading Degree')
axis ([0 size(shifty_calib_gps,1) 0 360])
set(gca,'YTick',[0 45 90 135 180 225 270 315 360])
hold off

% The following code will graph the uncompensated versus the GPS
```
% compensated z axis data along with a subplot of the heading degree
figure()
subplot(2,1,1);
hold on
plot(shiftz_calib_gps,'b')
plot(shiftz_pure_gps,'r')
xlabel('Index')
ylabel('Voltage (V)')
title('Magnetic Sensor Z Voltage: Uncompensated vs. GPS Compensated')
legend('Uncompensated','GPS Compensated')
hold off
subplot(2,1,2);
hold on
plot(fitted_headings,'b')
xlabel('Index')
ylabel('Heading Degree CW from Magnetic North')
title('Heading Degree CW from Magnetic North During Driving')
legend('Vehicle Heading Degree')
axis ([0 size(shiftz_calib_gps,1) 0 360])
set(gca,'YTick',[0 45 90 135 180 225 270 315 360])
hold off

Published with MATLAB® 7.10
GPS HEADING INTERPOLATION ALGORITHM
John Torres, 2011 jctorres@calpoly.edu Cal Poly SLO ME Master's Thesis

Contents

• Function Call
• Description
• READ ME Notes
• Input Variables
• Output Variables
• Program Code Begin
• Initialize GPS data and sensor matrices
• Reference magnetic north instead of geographic north
• Match GPS timestamps with sensor readings
• Heading Groupings
• Spline Interpolations
• Heading Corrections
• Program Code End
• Graph Results

Function Call

function [fitted_headings]=headfit(sensor, GPSdata)

Description

This program will interpolate GPS heading direction data according to a spline interpolation. The continuous direction of a vehicle is best approximated by a smooth linear interpolation of a spline.

READ ME Notes

BEFORE USING THIS PROGRAM Please read these notes about the format of the GPS data and the sensor output data:

* The output of the GPS textfile should be labeled "GPSdata". The format for this text file should have formatted columns as follows:
[time latitude longitude speed
heading_direction_from_geographic_north
magnetic_declination_angle]

The output of the magnetic sensor data text file should be labeled "sensor". The format for this text file should have formatted columns as follows:
If the original LABVIEW program for crash avoidance from Stephane Roussel is used, please be aware that the program can insert blank lines (return spaces) in the data. Use a text editor or Excel to get rid of these blank lines. This program may give errors if these blank lines are left in the imported data.

**NOTE ABOUT TIME FORMAT** The time should be represented as decimal representations of the day. If the time is 12 noon, the decimal representation should be 0.5. If the time is 11:56:15.8281 AM the decimal representation should be 0.4974864352.

If the native time format of the data is in hour:minutes:seconds (using 24 hour military time format) the decimal representation can be achieved quickly through Excel. When time stamps are imported into excel, use the "Format Cells" command > Catagory: Number > Decimal Places> 10, for example.

**NOTE ABOUT TIME SYNC BETWEEN GPS AND SENSOR FILES** When importing the GPS and sensor file data, both files will need to start at about the same time. Use a text editor or excel to delete sensor data without a corresponding GPS time data.

**Input Variables**

- **sensor** - Magnetic input data file. See READ ME Notes section for format.
- **GPSdata** - GPS data file. See READ ME Notes section for format.

**Output Variables**

- **fitted_headings** - a vector of heading directions from magnetic north for each magnetic sensor reading.

**Program Code Begin**

**Initialize GPS data and sensor matrices**

```matlab
heading=GPSdata(:,5);  % Make a copy of the GPS heading data into "heading"
mag_offset=GPSdata(:,6);% Make a copy of the magnetic declination of the GPS data into "mag_offset"
GPSsensor=sensor;  % Create a new matrix that will include the magnetic sensor data and GPS data.
```

**Reference magnetic north instead of geographic north**

Create heading degree data that references headings from magnetic north rather than geographic north.
for z=1:size(heading)
    delta_head=heading(z,1)-mag_offset(z,1);  % Subtract the declination angle from the GPS heading data
    if delta_head>0
        shifthead(z,1)=delta_head;          % If GPS heading declination is a positive number, leave it alone
    elseif delta_head<0
        shifthead(z,1)=360+delta_head;      % If GPS heading declination is a negative angle, convert that into a positive angle between 0 and 360 degrees
    end
end

Match GPS timestamps with sensor readings

Create a count of how many magnetic sensor data points belong to each heading count in the GPS data.

time_count=0;

for i=2:size(GPSdata,1)  % The first timestamp from the GPS at the time of the heading position
    time1=GPSdata(i-1,1);
    time2=GPSdata(i,1);   % The second timestamp from the GPS at the time of the heading position
    for f=1:size(sensor,1)
        check=sensor(f,1);  % This is the timestamp of the magnetic sensor reading for a certain point of time
        if (check>=time1) && (check<time2)  % Check if the current magnetic sensor reading "check" is between the first GPS timestamp and second GPS timestamp
            time_count=time_count+1;  % If the above statement is true, the current magnetic sensor reading "check" will belong to the first timestamp
        end
    end
    tab(i,1)=time_count;  % "tab" will keep an index of how many magnetic sensor data points belong to each GPS time stamp with heading information
end

Heading Groupings

This algorithm will group heading degrees together that constitute 1 full revolution between 0 and 360 degrees. When calibrating a vehicle by driving in circles, the magnetic sensor readings will be grouped according to their headings.

head_count=0;  % Initialize heading spline interpolation variables
head_size=size(shifthead,1);
head_start=2;
i=2;

section(1,1)=1;  % The "section" vector will count how many GPS headings belong in 1 full revolution between 0 and 360 degrees.
for z=head_start:head_size
    head1=shifthead(z-1,1);
    head2=shifthead(z,1);
    if (abs(head1-head2)<=299) && z<=head_size-1  % If the difference between heading 1 and heading 2 is less than 299, then the data point will belong in the same heading group
        head_count=head_count+1;
    elseif abs(head1-head2)>299 % If the difference between heading 1 and heading 2 is greater than 299, a new group that counts another revolution will start
        head_count=head_count+1;
        head_start=head_count+1;
        section(i,1)=head_start;
        i=i+1;
    elseif z==head_size % At the last comparison, group the last heading with the last known GPS heading.
        head_count=head_count+1;
        section(i,1)=head_count+1;
        i=i+1;
    end
end

Spline Interpolations

This section of code will do a spline interpolation between the known GPS headings, based upon how many magnetic sensor data points are in between known GPS headings.

for z=2:size(section,1)
    limit1=section(z-1,1);  % limit1 and limit2 will keep a count of how many GPS headings will need to be interpolated for magnetic sensor readings
    limit2=section(z,1);    % taken between known GPS coordinates.
    The if statements allow a for loop to run through and match each heading and sensor reading pair.
    if z==2;
        i1=tab(limit1)+1;
        i2=tab(limit2)-1;
    elseif z==size(section,1)
        i1=tab(limit1);
        i2=tab(limit2);
    else
        i1=tab(limit1);
        i2=tab(limit2)-1;
    end

    pl=section(z-1,1);
    p2=section(z,1)-1;
    if z==size(section,1) % This section of code is the actual spline interpolation of the readings with the pchip command
        i2=size(sensor,1);
    end
end
fitted_headings(i1:i2,1)=pchip(tab(p1:p2+1,1),shifthead(p1:p2+1,1),i1:i2);
else
fitted_headings(i1:i2,1)=pchip(tab(p1:p2,1),shifthead(p1:p2,1),i1:i2);
end
end

**Heading Corrections**

This section of the code will fix heading interpolations if it goes out of range between 0 and 360 degrees.

```matlab
for z=1:size(fitted_headings,1)
    if fitted_headings(z,1)<0           % If the heading direction angle is negative, shift that angle to be between 0 and 360 degrees
        fitted_headings(z,1)=360+fitted_headings(z,1);
    elseif fitted_headings(z,1)>360    % If the heading direction angle greater than 360 degrees, shift that angle to be between 0 and 360 degrees
        fitted_headings(z,1)=360-fitted_headings(z,1);
    end
end
```

**Program Code End**

**Graph Results**

This section plots the result of this algorithm. All Interpolated GPS headings should only vary between 0 and 360 degrees, and should include the known GPS headings. If it doesn't, then there's something wrong with the input files or a bug exists in this code.

```matlab
plot(fitted_headings,'g.', 'MarkerSize',4)
axis ([0 size(sensor,1) 0 360])
set(gca,'YTick',[0 45 90 135 180 225 270 315 360])
hold on
plot(tab,shifthead,'ro')
xlabel('Index')
ylabel('Degrees CW from Magnetic North')
title('Known GPS Headings and Interpolated GPS Headings')
legend('Interpolated Headings', 'Known GPS Headings')
```
HI PASS FILTER COMPENSATION
John Torres, 2011 jctorres@calpoly.edu Cal Poly SLO ME Master's Thesis

Contents

- Function Call
- DESCRIPTION
- Input Variables
- Output Variables
- Program Code Begin
- Butterworth Filter Setup
- Filter Data
- Program Code End
- Graph Results

Function Call

function 
[x_hi_fixed,y_hi_fixed,z_hi_fixed]=hi_pass_fix(fD,fitted_headings,xsf, 
ysf,zsf)

DESCRIPTION

This is a modified version of Stephane Roussel's Butterworth Filter setup. This will treat the movement of a vehicle within the geomagnetic field as a slowly varying change in voltage that can be filtered out with a hi-pass Butterworth filter. However, this will only leave the data with measurements of the changes in the magnetic field and information about the total magnitude of the magnetic field will not be accounted for.

Input Variables

fD= A matrix of magnetic sensor data after a low-pass Butterworth filter has been applied. The format for each of the columns of the matrix are as follows:  
[time filtered_magnetic_x filtered_magnetic_y filtered_magnetic_z fitted_headings]

fitted_headings - a vector of heading directions from magnetic north for each magnetic sensor reading.  
xsf- Magnetic sensor x scaling factor  
ysf- Magnetic sensor y scaling factor  
zsf- Magnetic sensor z scaling factor

Output Variables
x_hi_fixed- Magnetic sensor x-axis data after a hi pass filter has been applied to it.
y_hi_fixed- Magnetic sensor y-axis data after a hi pass filter has been applied to it.
z_hi_fixed- Magnetic sensor z-axis data after a hi pass filter has been applied to it.

Program Code Begin

Butterworth Filter Setup

Set up the Butterworth filter of 2\textsuperscript{nd} order and a cutoff frequency of \textit{Wn}

\begin{verbatim}
N = 2; % Specifies a 2nd order filter
Wn = .4888/(500); % If a sampling rate of 1kHz is used, the cutoff
% frequency is 0.4888 Hz.
[B,A] = butter(N,Wn,'high'); % Set the Butterworth filter as a hi pass

% Compute the initial conditions for Vx
ic1 = fD(1,2);
z2 = B(3)*ic1-A(3)*ic1;
z1 = B(2)*ic1+z2-A(2)*ic1;
z1 = [z1;z2];

% Compute the initial conditions for Vy
ic2 = fD(1,3);
z4 = B(3)*ic2-A(3)*ic2;
z3 = B(2)*ic2+z4-A(2)*ic2;
z2 = [z3;z4];

% Compute the initial conditions for Vz
ic3 = fD(1,4);
z6 = B(3)*ic3-A(3)*ic3;
z5 = B(2)*ic3+z6-A(2)*ic3;
z3 = [z5;z6];
\end{verbatim}

Filter Data

Apply the high pass filter fix to the low pass filtered data. The filtered data will be multiplied by the scaling factors

\begin{verbatim}
x_hi_fixed=xsf*filter(B,A,fD(:,2),zi1);
y_hi_fixed=ysf*filter(B,A,fD(:,3),zi2);
z_hi_fixed=zsf*filter(B,A,fD(:,4),zi3);
\end{verbatim}

Program Code End

Graph Results

This section will compare the uncompensated data and hi-pass compensated data. The Hi-Pass Compensated data should have the variations of the vehicle moving within the geomagnetic field taken out.
% Plot uncompensated x voltage versus hi-pass compensated x Voltage
figure()
subplot(2,1,1);
hold on
plot(fd(:,2)-mean(fd(:,2)), 'b')
plot(x_hi_fixed, 'r')
axis ([0 size(x_hi_fixed,1) min(fd(:,2)-mean(fd(:,2))) max(fd(:,2)-mean(fd(:,2)))])
xlabel('Index')
ylabel('Voltage (v)')
title('Magnetic sensor X Voltage: Uncompensated vs. Hi-Pass Compensated Data')
legend('Raw', 'Hi-Pass Compensation')
hold off
subplot(2,1,2);
hold on
plot(fitted_headings, 'b')
xlabel('Index')
ylabel('Heading Degree CW from Magnetic North')
title('Heading Degree CW from Magnetic North During Driving')
legend('Vehicle Heading Degree')
axis ([0 size(x_hi_fixed,1) 0 360])
set(gca, 'YTick', [0 45 90 135 180 225 270 315 360])
hold off

% Plot uncompensated y voltage versus hi-pass compensated y Voltage
figure()
subplot(2,1,1);
hold on
plot(fd(:,3)-mean(fd(:,3)), 'b')
plot(y_hi_fixed, 'r')
axis ([0 size(y_hi_fixed,1) min(fd(:,3)-mean(fd(:,3))) max(fd(:,3)-mean(fd(:,3)))])
xlabel('Index')
ylabel('Voltage (v)')
title('Magnetic sensor Y Voltage: Uncompensated vs. Hi-Pass Compensated Data')
legend('Raw', 'Hi-Pass Compensation')
hold off
subplot(2,1,2);
hold on
plot(fitted_headings, 'b')
xlabel('Index')
ylabel('Heading Degree CW from Magnetic North')
title('Heading Degree CW from Magnetic North During Driving')
legend('Vehicle Heading Degree')
axis ([0 size(y_hi_fixed,1) 0 360])
set(gca, 'YTick', [0 45 90 135 180 225 270 315 360])
hold off

% Plot uncompensated z voltage versus hi-pass compensated z Voltage
figure()
subplot(2,1,1);
hold on
plot(fd(:,4)-mean(fd(:,4)), 'b')
plot(z_hi_fixed, 'r')
axis ([0 size(z_hi_fixed,1) min(fd(:,4)-mean(fd(:,4))) max(fd(:,4)-mean(fd(:,4)))])
xlabel('Index')
ylabel('Voltage (v)')
title('Magnetic sensor Z Voltage: Uncompensated vs. Hi-Pass Compensated Data')
legend('Raw', 'Hi-Pass Compensation')
hold off
subplot(2,1,2);
hold on
plot(fitted_headings,'b')
xlabel('Index')
ylabel('Heading Degree CW from Magnetic North')
title('Heading Degree CW from Magnetic North During Driving')
legend('Vehicle Heading Degree')
axis ([0 size(z_hi_fixed,1) 0 360])
set(gca,'YTick',[0 45 90 135 180 225 270 315 360])
hold off

Published with MATLAB® 7.10
IGRF COMPENSATION
John Torres, 2011 jctorres@calpoly.edu Cal Poly SLO ME Master's Thesis

Contents

• Function Call
• Description
• READ ME Notes
• Input Variables
• Output Variables
• Program Code Begin
• Check if IGRF File Path is Loaded
• IGRF Calculator Initialization
• Input parameters for the IGRF calculations
• Variable initializations
• Calculation of IGRF
• Program Code End
• Graph of the Results

Function Call

function
[shiftX_pure_IGRF,shiftY_pure_IGRF,shiftZ_pure_IGRF]=IGRF_fix(fD,GPSdata,fitted_headings,xsf,ysf,zsf,xoff,yoff,zoff,xmax,xmin,ymax,ymin)

Description

This program will compensate for the geomagnetic field by making use of the IGRF's tabulated values for the horizontal and vertical component of the earth's magnetic field. This algorithm uses the IGRF program converted by Charles Rino (2010). This assumes that the vehicle remains on a parallel plane with the horizontal component of the geomagnetic reference field.

READ ME Notes

BEFORE USING THIS PROGRAM please add the path to /IGRF_calculator/. The way this is done varies based on the operating system. However, more recent versions of MATLAB will enable this to be done graphically.
The following copyright notice from Charles Rino is included in order to use the following program:
Copyright (c) 2010, Charles Rino All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above
copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

**Input Variables**

\( fD = \) A matrix of magnetic sensor data after a low-pass Butterworth filter has been applied. The format for each of the columns of the matrix are as follows:

\[
\text{[time filtered\textunderscore magnetic\textunderscore x filtered\textunderscore magnetic\textunderscore y filtered\textunderscore magnetic\textunderscore z fitted\textunderscore headings]}
\]

**GPSdata** - GPS data file.

**fitted\textunderscore headings** = a vector of heading directions from magnetic north for each magnetic sensor reading. This is obtained by first running the headfit function.

**xsf** - Magnetic sensor x scaling factor

**ysf** - Magnetic sensor y scaling factor

**zsf** - Magnetic sensor z scaling factor

**xoff** - Magnetic sensor x offset

**yoff** - Magnetic sensor y offset

**zoff** - Magnetic sensor z offset

**xmax** - Maximum voltage reading in the x direction

**xmin** - Minimum voltage reading in the x direction

**ymax** - Maximum voltage reading in the y direction

**ymin** - Minimum voltage reading in the y direction

**Output Variables**

**shiftx\_pure\_IGRF** - Compensated x axis magnetic sensor compensated for the geomagnetic reference field using the IGRF method.
shifty_pure_IGRF - Compensated y axis magnetic sensor compensated for the geomagnetic reference field using the IGRF method.
shiftz_pure_IGRF - Compensated z axis magnetic sensor compensated for the geomagnetic reference field using the IGRF method.

Program Code Begin

Check if IGRF File Path is Loaded

The following code will display a message to remind the user to set the proper path to the IGRF calculator program.

```matlab
sprintf('Please add to the working directory. Once IGRF_calculator folder is added to the working file path directory type return and press enter or type dbquit to quit. ') keyboard
```

IGRF Calculator Initialization

The following commands initialize the IGRF calculator and were taken from Charles Rino's sample file.

```matlab
global gh % Define a global variable gh that will store the IGRF coefficients. if exist('GHcoefficients','file')==2 % Load the IGRF Coefficient data.
    load('GHcoefficients')
else
    gh=GetIGRF11_Coefficients(1);
end
```

Input parameters for the IGRF calculations

In order to calculate the IGRF the altitude and fractional year is needed to use the proper coefficients. The following commands will ask for user inputs of the altitude and the year.

```matlab
alt = input('What was the mean altitude in kilometers of the location of the calibration? => '); % User input for altitude in km fyear= input('What was the date of the calibration? Input in decimal form i.e. June 1, 2010 is 2010.5=> '); % User input for fractional years
```

Variable initializations

The following variables are pre allocated vectors of NaN's (Not a Number) for program execution speed purposes. These variables will be defined later on.

```matlab
shiftx_pure_IGRF=NaN(size(fD,1),1); shifty_pure_IGRF=NaN(size(fD,1),1); shiftz_pure_IGRF=NaN(size(fD,1),1); 
```
shiftx_calib_IGRF=NaN(size(fD,1),1);
shifty_calib_IGRF=NaN(size(fD,1),1);
shiftz_calib_IGRF=NaN(size(fD,1),1);

% Hhistory will keep a record of the x and y components of the calculated
% IGRF field. This will only be used for reference purposes
Hhistory=NaN(size(fD,1),4);

Calculation of IGRF

The following code will compute the IGRF components for each GPS heading, and scale each magnetic sensor data to the IGRF values.

% Calculate the average longitude and Latitude of the trial run
nlat_avg=mean(GPSdata(:,2));
elong_avg=mean(GPSdata(:,3));

% Calculate the average magnetic field values
H_avg=igrf11syn(fyear,alt,nlat,elong);

% Calculate the magnetic field scaling factor for the x and y axis.
H_sf=10^-5*(H_avg(1,1)^2+H_avg(2,1)^2)^(1/2);

% Calculate the Z scaling factor scaled to IGRF values
zsf=2*10^-5*H_avg(3,1)/(zmax - zmin);

% The average of the signal will be subtracted from the readings
% in order to bring the readings to the y=0 axis for the x, y, and z
% magnetic sensor axis.
shiftx_IGRF=fD(:,2)-mean(fD(:,2));
shifty_IGRF=fD(:,3)-mean(fD(:,3));
shiftz_IGRF=fD(:,4)-mean(fD(:,4));

% The shifted data will be calibrated according to scaling factors and
% offsets calculated from the calibration program. The equation is
% (scaling factor)*(raw data)+(offset). More details about this
equation
% can be found in sensor_calibration.m
for z=1:size(shiftx_IGRF,1) % This for loop will
    shiftx_calib_IGRF(z,1)=2*H_sf*xsf*shiftx_IGRF(z,1)/(xmax-xmin)+xoff;
    shifty_calib_IGRF(z,1)=2*H_sf*ysf*shifty_IGRF(z,1)/(ymax-ymin)+yoff;
    shiftz_calib_IGRF(z,1)=zsf*shiftz_IGRF(z,1)+zoff;
end

% The following loop will subtract the IGRF component of the voltage
from each data entry.
for z=1:size(shiftx_IGRF,1)
    time=fD(z,1); % Store the current time for the data entry
    for zz=2:size(GPSdata,1) % This loop will get the corresponding time of the data entry
        if time>=GPSdata(zz-1,1) && time<=GPSdata(zz,1) % This will match the magnetic sensor entry with a corresponding GPS data based on time.
            nlat=GPSdata(zz,2); % nlat is the latitude at the current time
            elong=GPSdata(zz,3); % elong is the longitude at the current time
            declination=GPSdata(zz,6); % declination is the current magnetic declination angle.
        end
    end
    % This will calculate the geomagnetic reference field component and store it as a column of the x, y, and z components of the geomagnetic reference field and store it as the vector called "magneticfieldcomponents". The units are for these components are nT.
    magneticfieldcomponents=igrf11syn(fyear,alt,nlat,elong);
    % H is the magnitude of the horizontal component of the geomagnetic reference field. It is defined as (x^2+y^2)^(1/2) and it is multiplied by 10^-5 in order to convert nT to volts.
    H=10^-5*(magneticfieldcomponents(1,1)^2+magneticfieldcomponents(2,1)^2)^(1/2);
    % Hx is the voltage that the magnetic sensor's x axis will experience due to the magnetic reference field.
    Hx=cos(pi()*(fitted_headings(z,1)-90)/180)*H;
    % Hy is the voltage that the magnetic sensor's y axis will experience due to the magnetic reference field.
    Hy=cos(pi()*(fitted_headings(z,1)-180)/180)*H;
    % The following commands will store the current values of H, Hx, Hy, and the heading direction in the history matrix for reference purposes.
% The following commands will subtract the horizontal and vertical components of the geomagnetic reference field from the calibrated magnetic sensor values.
shiftx_pure_IFRG(z,1)=shiftx_calib_IFRG(z,1)-Hx;
shifty_pure_IFRG(z,1)=shifty_calib_IFRG(z,1)-Hy;
shiftz_pure_IFRG(z,1)=shiftz_calib_IFRG(z,1)-10^-5*(magneticfieldcomponents(3,1));

end

% Reshift the signals back to the y=0 axis.
shiftx_pure_IFRG= shiftx_pure_IFRG-mean(shiftx_pure_IFRG);
shifty_pure_IFRG= shifty_pure_IFRG-mean(shifty_pure_IFRG);
shiftz_pure_IFRG= shiftz_pure_IFRG-mean(shiftz_pure_IFRG);

Program Code End

Graph of the Results

The following commands will graph the results of the compensation. If GPS headings are accurate and not lost, the changes in voltages present in the raw readings will be brought towards the y=0 axis in the IGRF compensated readings (i.e. the peak to peak values of shiftx_calib_IFRG, which is the uncompensated values, should be larger than the peak to peak values of shiftx_pure_IFRG which does not include the added voltages of the geomagnetic field.

% Shift the uncompensated readings tot he y=0 axis.
shiftx_calib_IFRG=shiftx_calib_IFRG-mean(shiftx_calib_IFRG);
shifty_calib_IFRG=shifty_calib_IFRG-mean(shifty_calib_IFRG);
shiftz_calib_IFRG=shiftz_calib_IFRG-mean(shiftz_calib_IFRG);

% The following code will graph the uncompensated versus the IGRF compensated x axis data along with a subplot of the heading degree
gfigure()
subplot(2,1,1);
hold on
plot(shiftx_calib_IFRG,'b')
plot(shiftx_pure_IFRG,'r')
xlabel('Index')
ylabel('Voltage (V)')
title('Magnetic Sensor X Voltage: Uncompensated vs. IGRF Compensated')
legend('Uncompensated','IGRF Compensated')
hold off
subplot(2,1,2);
hold on
plot(fitted_headings,'b')
xlabel('Index')
ylabel('Heading Degree CW from Magnetic North')
title('Heading Degree CW from Magnetic North During Driving')
legend('Vehicle Heading Degree')
axis ([0 size(shiftx_calib_IGRF,1) 0 360])
set(gca,'YTick',[0 45 90 135 180 225 270 315 360])
hold off

% The following code will graph the uncompensated versus the IGRF
% compensated y axis data along with a subplot of the heading degree
figure()
subplot(2,1,1);
hold on
plot(shifty_calib_IGRF,'b')
plot(shifty_pure_IGRF,'r')
xlabel('Index')
ylabel('Voltage (V)')
title('Magnetic Sensor Y Voltage: Uncompensated vs. IGRF Compensated')
legend('Uncompensated','IGRF Compensated')
hold off
subplot(2,1,2);
hold on
plot(fitted_headings,'b')
xlabel('Index')
ylabel('Heading Degree CW from Magnetic North')
title('Heading Degree CW from Magnetic North During Driving')
legend('Vehicle Heading Degree')
axis ([0 size(shifty_calib_IGRF,1) 0 360])
set(gca,'YTick',[0 45 90 135 180 225 270 315 360])
hold off

% The following code will graph the uncompensated versus the IGRF
% compensated z axis data along with a subplot of the heading degree
figure()
subplot(2,1,1);
hold on
plot(shiftz_calib_IGRF,'b')
plot(shiftz_pure_IGRF,'r')
xlabel('Index')
ylabel('Voltage (V)')
title('Magnetic Sensor Z Voltage: Uncompensated vs. IGRF Compensated')
legend('Uncompensated','GPS Compensated')
hold off
subplot(2,1,2);
hold on
plot(fitted_headings,'b')
xlabel('Index')
ylabel('Heading Degree CW from Magnetic North')
title('Heading Degree CW from Magnetic North During Driving')
legend('Vehicle Heading Degree')
axis ([0 size(shiftz_calib_IGRF,1) 0 360])
set(gca,'YTick',[0 45 90 135 180 225 270 315 360])
hold off
MAGNETIC LOW PASS FILTER
John Torres, 2011  jctorres@calpoly.edu  Cal Poly SLO ME Master's Thesis

Contents

• Function Call
• DESCRIPTION
• READ ME NOTES
• INPUT VARIABLES
• OUTPUT VARIABLES
• Program Code Begin
• Butterworth Filter Setup
• Computing Initial Conditions
• Filter Data
• fD- Low Pass Filter Data Matrix
• Program Code End
• Graph Results

Function Call

function  [fD]=lowpass_magfilter(sensor, fitted_headings)

DESCRIPTION

This is the original low pass Butterworth filter Stephane Roussel created for the magnetic sensor data. This will filter out high frequency noise from magnetic sensor data.

READ ME NOTES

PLEASE RUN THE HEADFIT function first!! i.e.

[fitted_headings]=headfit(sensor, GPSdata)
The output of the magnetic sensor data textfile should be labeled "sensor". The format for this text file should have formatted columns as follows:

[time  magnetic_sensor_x_voltage  magnetic_sensor_y_voltage  magnetic_sensor_z_voltage]

If the original LABVIEW program for crash avoidance from Stephane Roussel is used, please be aware that the program can insert blank lines (return spaces) in the data. Use a text editor or Excel to get rid of these blank lines. This program may give errors if these blank lines are left in the imported data.

NOTE ABOUT TIME FORMAT : The time should be represented as decimal representations of the day. If the time is 12 noon, the decimal representation should be 0.5. If the time is 11:56:15.8281 AM the decimal representation
should be 0.4974864352.

If the native time format of the data is in hour:minutes:seconds (using 24 hour military time format) the decimal representation can be achieved quickly through Excel. When time stamps are imported into Excel, use the "Format Cells" command > Category: Number > Decimal Places> 10, for example

**INPUT VARIABLES**

sensor = magnetic sensor data file
fitted_headings = a vector of heading directions from magnetic north for each magnetic sensor reading. This is obtained by first running the headfit function.

**OUTPUT VARIABLES**

fD = A matrix of magnetic sensor data after a low-pass Butterworth filter has been applied. The format for each of the columns of the matrix are as follows:

\[
\begin{bmatrix}
\text{time} & \text{filtered}_{\text{magnetic_x}} & \text{filtered}_{\text{magnetic_y}} & \text{filtered}_{\text{magnetic_z}} & \text{fitted}_{\text{headings}}
\end{bmatrix}
\]

**Program Code Begin**

\[
D = \text{sensor}; \quad \% \text{Make a copy of the sensor matrix}
\]

**Butterworth Filter Setup**

Set up the Butterworth filter of Nth order and a cutoff frequency of Wn

\[
N = 2; \quad \% \text{Specifies a 2nd order filter}
\]

\[
Wn = .004; \quad \% \text{If a sampling rate of 1kHz is used, the cutoff frequency is 2Hz}
\]

\[
[B,A] = \text{butter}(N,Wn,'low'); \quad \% \text{Set the Butterworth filter as a low pass}
\]

**Computing Initial Conditions**

\[
\begin{align*}
\text{ic1} &= D(1,2); \quad \% \text{Compute the initial conditions for Vx} \\
\text{z2} &= B(3)\text{ic1} - A(3)\text{ic1}; \\
\text{z1} &= B(2)\text{ic1} + z2 - A(2)\text{ic1}; \\
\text{zi1} &= \begin{bmatrix} z1; z2 \end{bmatrix}; \\
\text{ic2} &= D(1,3); \quad \% \text{Compute the initial conditions for Vy} \\
\text{z4} &= B(3)\text{ic2} - A(3)\text{ic2}; \\
\text{z3} &= B(2)\text{ic2} + z4 - A(2)\text{ic2}; \\
\text{zi2} &= \begin{bmatrix} z3; z4 \end{bmatrix}; \\
\text{ic3} &= D(1,4); \quad \% \text{Compute the initial conditions for Vz} \\
\text{z6} &= B(3)\text{ic3} - A(3)\text{ic3};
\end{align*}
\]
\[
z_5 = B(2) \cdot ic3 + z_6 - A(2) \cdot ic3; \\
z_i3 = [z_5; z_6]; \\
\]

**Filtered Data**

Filtered the data that was passed to the function with filter

\[
f_{Vx1} = \text{filter}(B, A, D(:,2), zi1); \\
f_{Vy1} = \text{filter}(B, A, D(:,3), zi2); \\
f_{Vz1} = \text{filter}(B, A, D(:,4), zi3); \\
\]

**fD- Low Pass Filter Data Matrix**

\[
fD = [D(:,1), f_{Vx1}, f_{Vy1}, f_{Vz1}, \text{fitted\_headings}]; \\
\]

**Program Code End**

**Graph Results**

This section will compare the filtered and unfiltered data.

%Plot filtered X voltage versus Unfiltered X Voltage
figure()
plot(sensor(:, 2), 'g')
hold on
plot(fD(:,2), 'r', 'LineWidth', 1)
xlabel('Index')
ylabel('Voltage (v)')
title('Magnetic sensor X Voltage: Raw Data versus Low Pass Filtered Data')
legend('Raw', 'Low Pass Filtered')

%Plot filtered Y voltage versus Unfiltered Y Voltage
figure()
plot(sensor(:, 3), 'g')
hold on
plot(fD(:,3), 'r', 'LineWidth', 1)
xlabel('Index')
ylabel('Voltage (v)')
title('Magnetic sensor Y Voltage: Raw Data versus Low Pass Filtered Data')
legend('Raw', 'Low Pass Filtered')

%Plot filtered Z voltage versus Unfiltered Z Voltage
figure()
plot(sensor(:, 4), 'g')
hold on
plot(fD(:,4), 'r', 'LineWidth', 1)
xlabel('Index')
ylabel('Voltage (v)')
title('Magnetic sensor Z Voltage: Raw Data versus Low Pass Filtered Data')
legend('Raw', 'Low Pass Filtered')

*Published with MATLAB® 7.10*
Contents

- Description
- READ ME Notes
- Program Code Begin
- Program Info
- Choosing what to do
- Data Check
- Sensor Calibration
- Calibration: Heading Interpolation
- Sensor Correction
- Field Test Correction: Heading Interpolation
- Program Code End

Description

This program will calibrate a 3 axis magnetic sensor as well as correct data for the geomagnetic field. Further information about this algorithm will be in the Master's Thesis document.

READ ME Notes

- If working with a large data file, the correction phase using the GPS or IGRF algorithm will take a while to process. For a 30 minute driving file, the GPS/IGRF correction will take 2-4 hours (2.3 Ghz Intel Core Duo on a Mac using MATLAB 2009). However, real time implementation of this code should be almost instantaneous since data is processed as it is received.
- Refer to the separate M files for more descriptions of the functions.

Program Code Begin

Program Info

This code will print a welcome message to introduce the user to the program.

```matlab
fprintf('Magnetic Sensor Calibration and Compensation Program \r\nJohn Torres 2011\nCal Poly SLO\nMS Thesis \r')
fprintf('This program will calibrate a set of magnetic sensor readings
\nmoving within the geomagnetic reference field. \r')
fprintf('Before using this program, you will need to load the magnetic
\nsensor and GPS data of the calibration run and/or test run. \r')

% If the user want to review the formatting of the data, they can enter 'y' in the help_data_format variable.
help_data_format=input('Before starting, would you like to review the
```
format of the input data (y/n)? ', 's');

% The following help messages will be displayed if the user enter 'y' to the previous question.
if help_data_format=='y'
    help_gps_format=sprintf('
The GPS data should be formatted as a matrix with the following columns: [time latitude longitude speed heading_direction_from_geographic_north magnetic_declination_angle] 
');
    help_sensor_format=sprintf('
The magnetic sensor data should be formatted as a matrix with the following columns: [time magnetic_sensor_x_voltage magnetic_sensor_y_voltage magnetic_sensor_z_voltage] 
');
    help_time_format=sprintf('
The time should be represented as decimal representations of the day. If the time is 11:56:15.8281 AM the decimal representation should be 0.4974864352. 
');
    help_additional_notes=sprintf('
The GPS and magnetic sensor data file should start at the same time. Make sure there are \n no blank lines in the sensor data file. 
');
    help_further_nfo=sprintf('
For further information, please refer to the comments in the documentation and .m files of the programs. 
');
    disp(help_gps_format);
    disp(help_sensor_format);
    disp(help_time_format);
    disp(help_additional_notes);
    disp(help_further_nfo);
end

Choosing what to do

This section of the program will prompt the user what they want to do: calibrate the sensor, correct test field data, or both.

% The following message will instruct the user to load data.
fprintf('Load the GPS and magnetic sensor data. For calibration, the GPS data should be named calibration_GPSdata and the magnetic sensor data should be calibration_sensor. 
');
fprintf('For the actual testing data, the GPS data should be named fieldtest_GPSdata and the magnetic sensor data should be fieldtest_sensor. 
');

% what_to_do will store the option 1, 2, or 3 according to what they want to do. The choice will then execute the proper action within an if loop.
what_to_do=input('Did you want to (1) calibrate magnetic sensors and correct field data or (2) Calibrate magnetic sensors only or (3) correct test field data only [1,2,3]? : ');

Data Check

The following will check to see if the proper data exist in order to execute the algorithm.
load_flag=0; % load_flag will be used to signal that the user is done loading the data. % It will be initialized as 0 to signify as no data % loaded.

% The following question will ask the user to load the data, and set % the load_flag=1 to signal that they are done loading their % variables.

sprintf('Please load calibration_GPSdata & calibration_sensor and/or fieldtest_GPSdata & fieldtest_sensor. When finished type return and press enter or to quit type dbquit and press enter. ') keyboard
load_flag=1; % load_flag will indicate that the user has loaded the variables.

% This loop will check to make sure that the proper data is loaded. % The while loops within load_flag will execute until the proper % variable names % exist or until the user presses ctrl+c or command+. (Mac)

if load_flag==1 % If the user states that the data is loaded, double check all variables exist
    % If both calibration and correction or just the calibration is to be
    % done, the following checks are made.
    if what_to_do==1 || what_to_do==2
        while exist('calibration_GPSdata','var')==0
            sprintf('Please load/rename calibration_GPSdata. When finished type return and press enter or to quit type dbquit and press enter. ') keyboard
        end
        while exist('calibration_sensor','var')==0
            sprintf('Please load/rename calibration_sensor. When finished type return and press enter or to quit type dbquit and press enter. ') keyboard
        end
    end
    % If both calibration and correction or just correction is to done, the % following checks are made.
    if what_to_do==1 || what_to_do==3
        while exist('fieldtest_GPSdata','var')==0
            sprintf('Please load/rename fieldtest_GPSdata. When finished type return and press enter or to quit type dbquit and press enter. ') keyboard
        end
        while exist('fieldtest_sensor','var')==0
            sprintf('Please load/rename fieldtest_sensor. When finished type return and press enter or to quit type dbquit and press enter. ') keyboard
        end
    end
end
keyboard
end
end

% If only the correction needs to be done, the following additional
% checks are made.
if what_to_do==3
    while exist('xsf','var')==0
        sprintf('Please load/rename xsf. When finished type return
and press enter or to quit type dbquit and press enter. ')
        keyboard
    end
    while exist('ysf','var')==0
        sprintf('Please load/rename ysf. When finished type return
and press enter or to quit type dbquit and press enter. ')
        keyboard
    end
    while exist('zsf','var')==0
        sprintf('Please load/rename zsf. When finished type return
and press enter or to quit type dbquit and press enter. ')
        keyboard
    end
    while exist('xoff','var')==0
        sprintf('Please load/rename xoff. When finished type return
and press enter or to quit type dbquit and press enter. ')
        keyboard
    end
    while exist('yoff','var')==0
        sprintf('Please load/rename yoff. When finished type return
and press enter or to quit type dbquit and press enter. ')
        keyboard
    end
    while exist('zoff','var')==0
        sprintf('Please load/rename zoff. When finished type return
and press enter or to quit type dbquit and press enter. ')
    end
    while exist('xfit','var')==0
        sprintf('Please load/rename xfit. When finished type return
and press enter or to quit type dbquit and press enter. ')
        keyboard
    end
    while exist('yfit','var')==0
        sprintf('Please load/rename yfit. When finished type return
and press enter or to quit type dbquit and press enter. ')
        keyboard
    end
    while exist('zfit','var')==0
        sprintf('Please load/rename zfit. When finished type return
and press enter or to quit type dbquit and press enter. ')
        keyboard
    end
end
end

% Print a message to the user indicating loading was a success
sprintf('\r\r Variables successfully loaded.')
Sensor Calibration

The following commands will calibrate the magnetic sensor. These commands will execute when both calibration and correction (choice 1) are necessary or just calibration (choice 2).

```matlab
if what_to_do==1 || what_to_do==2
```

Calibration: Heading Interpolation

The following code will interpolate the GPS heading angles based on a spline interpolation. `calibration_fitted_headings` will contain these interpolations.

```matlab
[calibration_fitted_headings]=headfit(calibration_sensor, calibration_GPSdata);
```

%%% Calibration: Low Pass Filter
% The following code will apply a low pass filter to the raw magnetic
% sensor data to clean up the noise. `calibration_fD` will contain the
% calibration magnetic sensor data after the low pass filter has been
% applied.
% ```matlab
% [calibration_fD]=lowpass_magfilter(calibration_sensor, calibration_fitted_headings);
% ```

%%% Sensor Calibration
% The following command will export calibration coefficients for the
% magnetic sensor.
% ```matlab
[xsf, xoff, xfit, ysf, yoff, yfit, zsf, zoff, zfit,xmax,xmin,ymax,ymin]=sensor_calibration(calibration_fD, calibration_fitted_headings);
% ```

% Print a message to the user to indicate calibration complete
sprint('Sensor calibration is complete. Calibration coefficients are ready to use.')
end

Sensor Correction

The following commands will correct the magnetic sensor data. These commands will execute when both calibration and correction (choice 1) are necessary or just correction (choice 3).
if what_to_do==1 || what_to_do==3

**Field Test Correction: Heading Interpolation**

The following code will interpolate the GPS heading angles based on a spline interpolation. fieldtest_fitted_headings will contain these interpolations.

```
[fieldtest_fitted_headings]=headfit(fieldtest_sensor, fieldtest_GPSdata);

%%% Field Test Correction: Low Pass Filter
% The following code will apply a low pass filter to the raw magnetic
% sensor data to clean up the noise. fieldtest_fD will contain the
% fieldtest magnetic sensor data after the low pass filter has
% been applied.
[fieldtest_fD]=lowpass_magfilter(fieldtest_sensor, fieldtest_fitted_headings);

%%% Field Test Correction (Dynamic): Hi Pass Filter Correction
% The following code will apply a hi pass filter to the low pass
% filtered data to correct for the geomagnetic reference field.
[x_hi_fixed,y_hi_fixed,z_hi_fixed]=hi_pass_fix(fieldtest_fD,fieldtest_fitted_headings,xsf,ysf,zsf);

% Print a message to the user indicating the hi pass filter correction is complete
sprintf(['\r \r Hi pass filter correction is complete.'])

%%% Field Test Correction (Static)
% The user will have the choice of using either the GPS correction,
% IGRF model correction, or both. The following input will ask for the
% user's choice in the correction method.

gps_or_IGRF=input('Would you like to correct the data using the (1) GPS method, (2) IGRF method, or (3) both [1,2,or 3]? : ');

if gps_or_IGRF==1 || gps_or_IGRF==3 % If the user chooses option 1 or 3, the following correction method will be used
[shiftx_pure_gps shifty_pure_gps shiftz_pure_gps]=gps_fix(fieldtest_fD, fieldtest_fitted_headings, xsf, xoff, xfit, ysf, yoff, yfit, zsf, zoff, zfit);

% Print a message to the user indicating the GPS correction is complete.
```
sprintf(‘\r\r GPS correction is complete.’)
end

if gps_or_IGRF==2 || gps_or_IGRF==3 % If the user choses option 2 or 3, the following correction method will be used.
[shiftx_pure_IGRF,shifty_pure_IGRF,shiftz_pure_IGRF]=IGRF_fix(fieldtes\nt_fd,fieldtest_GPSdata,fieldtest_fitted_headings,xsf,ysf,zsf,xoff,yoff
,zoff,xmax,xmin,ymax,ymin);

% Print a message to the user indicating the IGRF correction is
% complete.
sprintf(‘\r\r IGRF correction is complete.’)
end
end

Program Code End

Published with MATLAB® 7.10
MAGNETIC SENSOR CALIBRATION
John Torres, 2011  jctorres@calpoly.edu  Cal Poly SLO ME Master's Thesis

Contents

•  Function Call
•  Description
•  READ ME Notes
•  Input Variables
•  Output Variables
•  Program Code Begin
•  Calculation of Scaling Factors and Offsets
•  Calibration of Magnetic Sensor readings
•  Characterizing the Geomagnetic Field Using a Polynomial Fit
•  Error Checking
•  Plots of Polynomial Fits
•  Refinement Comparison
•  Program Code End

Function Call

function  [xsf, xoff, xfit, ysf, yoff, yfit, zsf, zoff, zfit,xmax,xmin,ymax,ymin]=sensor_calibration(fD, fitted_headings)

Description

This program will calibrate the magnetic sensor based on field calibration values. These values are not scaled to the values predicted by the IGRF.

READ ME Notes

•  Due to hard iron effects of mounting the magnetic sensor to a vehicle, the magnetic sensors must be calibrated in an open space with the least amount of magnetic disturbance. Scaling and offset factors are calculated when the vehicle is driven in a circle. The maximum and minimum voltages of the calibration run will be used for the x, y, and z axis. The scaling factors xsf, ysf, and zsf and are calculated according to the following equations:

\[
xsf = \max\left[(ymax - ymin)/(xmax - xmin)\right]\|1] \\
ysf = \max\left[(xmax - xmin)/(ymax - ymin)\right]\|1] \\
zs = 1
\]

xsf and ysf components will be either the calculated value or 1, which ever is numerically larger. These equations are formed under the assumption that the vehicle will be parallel to the horizontal plane of the geomagnetic field (i.e. not on
a steeply inclined surface).

The offset values are calculated according to the following equations:

\[
\begin{align*}
  x_{off} &= ((x_{max} - x_{min})/2 - x_{max}) \times xsf \\
  y_{off} &= ((y_{max} - y_{min})/2 - y_{max}) \times ysf \\
  z_{off} &= ((z_{max} - z_{min})/2 - z_{max}) \times zsf
\end{align*}
\]

One of the developed algorithms to subtract the geomagnetic field of the earth is to use the field calibration values. Assuming that the calibration was done in an area with the least amount of external magnetic bias, the voltages experienced by each axis of the sensor at a certain heading degree (from the GPS data) can be assumed to be contributions from the Earth's geomagnetic field. These data sets can be used to make an equation that can output the voltage from the geomagnetic field. This will be done by using a polynomial fit on the x, y, and z voltages as a function of heading degree. A polynomial fit was chosen because it is the quickest and easiest fit that can output usable results.

If MATLAB warns about a badly conditioned polynomial due to multiple voltages for each heading (i.e. multiple y values per x value), disregard this message. This happens due to the nature of the data collection, and the polynomial fit is trying to "average" multiple voltages per known heading.

**NOTE ABOUT TIME FORMAT** The time should be represented as decimal representations of the day. If the time is 12 noon, the decimal representation should be 0.5. If the time is 11:56:15.8281 AM the decimal representation should be 0.4974864352.

If the native time format of the data is in hour:minutes:seconds (using 24 hour military time format) the decimal representation can be achieved quickly through Excel. When time stamps are imported into excel, use the "Format Cells" command > Catagory: Number > Decimal Places> 10, for example.

**Input Variables**

\( fD = \) A matrix of magnetic sensor data after a low-pass Butterworth filter has been applied. The format for each fo the columns of the matrix are as follows:

\[
\begin{bmatrix}
  \text{time} & \text{filtered\_magnetic\_x} & \text{filtered\_magnetic\_y} & \text{filtered\_magnetic\_z} & \text{fitted\_headings}
\end{bmatrix}
\]

\( \text{fitted\_headings} \) - a vector of heading directions from magnetic north for each magnetic sensor reading.

**Output Variables**

These output variables will be used to scale low-pass filtered magnetic sensor
voltages according to the equation:

\[
\begin{align*}
    x_{calibrated} &= x_{raw} \times xsf + xoff \\
    y_{calibrated} &= y_{raw} \times ysf + yoff \\
    z_{calibrated} &= z_{raw} \times zsf + zoff
\end{align*}
\]

\textit{xsf} - Magnetic sensor x scaling factor  \\
\textit{ysf} - Magnetic sensor y scaling factor  \\
\textit{zsf} - Magnetic sensor z scaling factor  \\
\textit{xoff} - Magnetic sensor x offset  \\
\textit{yoff} - Magnetic sensor y offset  \\
\textit{zoff} - Magnetic sensor z offset  \\
\textit{xfit} - polynomial coefficients of the 8th degree that represent the x voltage from the geomagnetic field as a function of heading degree  \\
\textit{yfit} - polynomial coefficients of the 8th degree that represent the y voltage from the geomagnetic field as a function of heading degree  \\
\textit{zfit} - polynomial coefficients of the 8th degree that represent the z voltage from the geomagnetic field as a function of heading degree  \\
\textit{xmax} - Maximum voltage reading in the x direction  \\
\textit{xmin} - Minimum voltage reading in the x direction  \\
\textit{ymax} - Maximum voltage reading in the y direction  \\
\textit{ymin} - Minimum voltage reading in the y direction

\textbf{Program Code Begin}

\textbf{Calculation of Scaling Factors and Offsets}

\begin{verbatim}
% Shift the low-pass filtered magnetic data down to the 0 axis for calibration based on the average of the signal
shiftx=fD(:,2)-mean(fD(:,2));
shifty=fD(:,3)-mean(fD(:,3));
shiftz=fD(:,4)-mean(fD(:,4));

% Find the minimum and maximum value of each magnetic axis
xmax=max(shiftx); xmin=min(shiftx);
ymax=max(shifty); ymin=min(shifty);
zmax=max(shiftz); zmin=min(shiftz);

xtemp=(ymax-ymin)/(xmax-xmin);  \textit{xsf} (refer to READ ME Notes)
ytemp=(xmax-xmin)/(ymax-ymin);  \textit{ysf} (refer to READ ME Notes)
zsf=1;  \textit{zsf} (refer to READ ME Notes)

% xsf will take on the value of the largest of the two: 1 or xtemp
if xtemp>1
\end{verbatim}
xsf=xtemp;
elseif xtemp<1
xsf=1;
end

% ysf will take on the value of the largest of the two: 1 or ytemp
if ytemp>1
  ysf=ytemp;
elseif ytemp<1
  ysf=1;
end

% Calculate the offset values for each axis (refer to READ ME Notes)
xoff=((xmax-xmin)/2-xmax)*xsf;
yoff=((ymax-ymin)/2-ymax)*ysf;
zoff=((zmax-zmin)/2-zmax)*zsf;

**Calibration of Magnetic Sensor readings**

Calibrate each magnetic sensor reading for each axis using the equations in the section "Output Variables"

% This for loop will go through each reading in the x, y, and z direction
% and calibrate them.
for z=1:size(shiftx,1)
  shifth_calib(z,1)=xsf*shiftx(z,1)+xoff;
  shifty_calib(z,1)=ysf*shifty(z,1)+yoff;
  shiftz_calib(z,1)=zsf*shiftz(z,1)+zoff;
end

% Reshift the calibrated values to their original position. This is only
% used for reference and may not be of interest.
calibx=shifth_calib+mean(fd(:,2));
caliby=shifty_calib+mean(fd(:,3));
calibz=shiftz_calib+mean(fd(:,4));

**Characterizing the Geomagnetic Field Using a Polynomial Fit**

An 8th order polynomial fit will be constructed from the voltages of the x, y, and z axis as a function of the heading degree it was recorded at.

% Use the polyfit command to generate polynomials that represent the x, y, % and z voltages as a function of GPS heading degree.

warning off % Turn off warnings during fitting process. Refer to
"READ ME Notes"

[xfit,xfit_stat]=fit(fitted_headings,shifth_calib,'poly8');
[yfit,yfit_stat]=fit(fitted_headings,shifty_calib,'poly8');
[zfit,zfit_stat]=fit(fitted_headings,shiftz_calib,'poly8');

warning on % Turns warnings back on. Refer to "READ ME Notes"
Error Checking

This will check if the polynomial fits the data well by checking the R^2 value. If the R^2 value of the X and Y axis (which is the basis of the analysis) is greater than .99, the polynomial fit will be considered good. If it is not greater than .99, the user will be asked to start and input new calibration data or condition the current calibration data. Please refer to calibration_error_refinement.m for more info.

```matlab
if xfit_stat.rsquare<.99 || yfit_stat.rsquare<.99
    sprintf('The fitted model does not fit well with the data. If there are other calibration files that can be used please type dbquit and press enter to restart the program.')
    sprintf('If you would like to refine the current calibration data press type return and press enter. Please note that there may be data loss in refining the data.')
    keyboard
    conditioning=input('Are you sure you want to refine the calibration data? This may result in data loss [y/n]: ');
    if conditioning=='y'
        [xfit_loop,xfit_loop_stat,yfit_loop,yfit_loop_stat]=calibration_error_refinement(shiftx_calib,shifty_calib,fitted_headings);
    end
end
```

Plots of Polynomial Fits

This section will plot the polynomial fits over the magnetic sensor axis voltage as a function of interpolated heading. The plots of the polynomial fits should match closely with the calibrated data. If calibration refinement was done, then the variable xfit_loop and yfit_loop will also be plotted.

```matlab
figure();
plot(fitted_headings,shiftx_calib,'g.','MarkerSize',5);
axis ([0 360 min(shiftx_calib) max(shiftx_calib)])
set(gca,'XTick',[0 45 90 135 180 225 270 315 360])
hold on
plot(xfit,'r');
if exist('xfit_loop','var')==1  % This will plot xfit_loop if it exists
    plot(xfit_loop,'b')
end
xlabel('Degree Heading CW from Magnetic North')
ylabel('Voltage')
title('X Voltage as a Function of GPS Heading')
legend('Calibrated data','Original Polynomial Fit','Refined Polynomial Fit')
```
hold off

figure();
plot(fitted_headings,shifty_calib,'g.','MarkerSize',5);
axis ([0 360 min(shifty_calib) max(shifty_calib)])
set(gca,'XTick',[0 45 90 135 180 225 270 315 360])
hold on
plot(yfit,'r');
if exist('yfit_loop','var')==1  % This will plot yfit_loop if it exists
  plot(yfit_loop,'b');
end
xlabel('Degree Heading CW from Magnetic North')
ylabel('Voltage')
title('Y Voltage as a Function of GPS Heading')
legend('Calibrated data','Polynomial Fit','Refined Polynomial Fit')
hold off

figure();
plot(fitted_headings,shiftz_calib,'g.','MarkerSize',5);
axis ([0 360 min(shiftz_calib) max(shiftz_calib)])
set(gca,'XTick',[0 45 90 135 180 225 270 315 360])
hold on
plot(zfit,'r');
xlabel('Degree Heading CW from Magnetic North')
ylabel('Voltage')
title('Z Voltage as a Function of GPS Heading')
legend('Calibrated data','Polynomial Fit')
hold off

**Refinement Comparison**

If refinement was done in the calibration data, this code will ask the user to make a judgment whether to use the original polynomial fits or use the refined one.

```matlab
if exist('xfit_loop','var')==1 && exist('yfit_loop','var')==1
  replace_fit=input('Which polynomial fit matches the calibration data, the refined (r) fits or the original (o) [r/o]? : ');
  if replace_fit=='r'
    xfit=xfit_loop;
    yfit=yfit_loop;
  end
end
```

**Program Code End**

*Published with MATLAB® 7.10*