The Pennsylvania State University The Graduate School Department of Mechanical and Nuclear Engineering

ROBUST VEHICLE LOCALIZATION USING GPS, IN-VEHICLE CAMERA, MAGNETIC GUIDANCE AND KALMAN FILTERING

A Thesis in Mechanical Engineering by Anthony J. Mangus

 \bigodot 2013 Anthony J. Mangus

Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Science

December 2013

The thesis of Anthony J. Mangus was reviewed and approved^{*} by the following:

Sean N. Brennan Associate Professor of Mechanical Engineering Thesis Advisor

H. Joseph Sommer Professor of Mechanical Engineering

Karen Thole Professor of Mechanical Engineering Department Head of Mechanical and Nuclear Engineering

*Signatures are on file in the Graduate School.

Abstract

This research focuses on reducing the effect of sensor faults and noise on the lateral vehicle estimation problem. The lateral estimation algorithm aims to localize the vehicle within the confines of a lane, given known and unknown faults. In order to test these algorithms, the Pennsylvania State University Rolling Roadway Simulator (PURRS) was reconfigured and simplified; however, to reduce complexity of testing, the treadmill belt was not used and the vehicle was moved by hand. In addition, a new vehicle was designed and built to provide a more rugged and utilitarian vehicle for use on the PURRS. In this work, these hardware changes are discussed, as well as the development of a Magnetic Guidance Calibration Stand (MGCS). These hardware systems are then used to develop fault reduction algorithms for use with a vehicle equipped with two magnetic sensors, an in-vehicle camera, and simulated GPS sensor. Two algorithms are tested: one to reduce the effect of an unknown fault, such as a sensor failure, and the other for known faults, such as a known change in environment that increases measurement noise. These algorithms were tested offline using data collected using physical hardware on the PURRS. These algorithms are shown to reduce the effect of a fault on the estimation of the vehicle's position.

Table of Contents

List of	Figures	vi
List of	Tables	ix
Ackno	wledgments	x
Chapt	er 1	
Inti	roduction	1
1.1	Thesis Overview	1
1.2	Literature Review	2
	1.2.1 Automated Highway Systems	2
	1.2.2 Magnetic Guidance	3
	1.2.3 In-Vehicle Camera	10
	1.2.4 Global Positioning System (GPS)	12
	1.2.5 Sensor Fusion and Fault Detection	13
Chapt	er 2	
The	e PURRS	16
2.1	Structural Changes to the PURRS	16
2.2	Treadmill Deck Changes to the PURRS	17
2.3	Vehicle Changes to the PURRS	22
2.4	Additional Changes to the PURRS	30
Chapt	er 3	
Sen	sor Characterization	32
3.1	Magnetic Sensor	32
	3.1.1 Magnetic Guidance Calibration Stand	32
	3.1.2 Magnetic Sensor Characterization	38
3.2	In-Vehicle Camera	46
3.3	Simulated GPS	48
3.4	Individual Sensor Testing on the PURRS	51
	3.4.1 Magnetic Sensors	53
	3.4.2 In-Vehicle Camera	53
	3.4.3 Simulated GPS	56

Chapte	er 4	
\mathbf{Esti}	mation Algorithm Development	58
4.1	Averaging Techniques	58
4.2	Kalman Filter	59
4.3	Comparison of Sensor Algorithms	62
Chapte	er 5	
Fau	It Detection and Reduction Algorithm Development	66
5.1	Fault Detection	66
5.2	Position-Referenced Fault Mitigation	69
Chapte	er 6	
Cor	clusions and Future Work	74
6.1	Conclusions	74
	6.1.1 Sensors	74
	6.1.2 Hardware Development	74
	6.1.3 Algorithm Development	75
6.2	Future Work	75
Appen	dix A	
Coc	le	77
Bibliog	graphy	80

List of Figures

1.1	Notation for ground vehicles					. 3
1.2	Shape of inductive wire magnetic field					. 4
1.3	Two wire configuration used by Olson					. 6
1.4	Two wire configuration used by Matsumoto					. 6
1.5	Comparison of guidance methods					. 9
1.6	Magnetic field produced by a magnetic strip					. 10
1.7	Determining road curvature using straightening technique					. 12
1.8	Visualization of two-ellipsoid test		•	•		. 15
2.1	PURRS being used for similitude research					. 17
2.2	Inside of the new PURRS blower box frame					. 18
2.3	Inductive wires on the side of the blower box frame					. 19
2.4	Countersunk slots on the top perforated aluminum sheet					. 20
2.5	First layer of the blower box frame					. 20
2.6	Box frame blower attachment					. 21
2.7	Completed box frame with vehicle and lane lines					. 21
2.8	HPI WR8 Flux with the body (Image courtesy of HPI Racing)					. 23
2.9	HPI WR8 Flux without the body (Image courtesy of HPI Racing)					. 24
2.10	Typical electric remote control car scheme					. 24
2.11	New remote control car scheme					. 25
2.12	New control components installed on PURRS vehicle					. 25
2.13	3D rendering of the roll-pitch-yaw assembly					. 26
2.14	US Digital S2 encoder (Picture courtesy of US Digital)	•				. 26
2.15	Sketch showing sensor positions relative to the vehicle's center of gravity					. 27
2.16	Camera installed on the PURRS vehicle	•				. 28
2.17	Redesigned front bumper on the PURRS vehicle	•				. 28
2.18	Rear MGS1600 magnetic sensor installed on the PURRS vehicle \ldots .	•				. 29
2.19	Fully assembled PURRS vehicle					. 29
2.20	Simulated GPS sensor	•				. 30
2.21	View of the simulated GPS sensor	•	•	•	•	. 31
3.1	RoboteQ MGS1600 magnetic guidance sensor					. 33
3.2	The Magnetic Guidance Calibration Stand (MGCS)					. 33
3.3	Upper half of the MGCS					. 34
3.4	Lower half of the MGCS					. 35
3.5	Example of an encoder disk (Courtesy of US Digital)					. 35
3.6	Example of a low resolution encoder with two rings of slots $\ldots \ldots \ldots$		•			. 36

3.7	Pattern seen from encoder in Figure 3.6 when rotated clockwise					
3.8	Arduino UNO board (Image courtesy of Arduino)					
3.9	MGCS Arduino and daughter board 38					
3.10	Data flow block diagram for the MGCS 38					
3.11	Example of magnetic sensor data from the MGCS using the MGS1600 sensor 39					
3.12	Comparison of magnetic sensor data					
3.13	Discrete marker field shape					
3.14	Measured magnetic field profile 1 inch from Magnetic Strip after re-calibration . 40					
3.15	Visualization of MGCS Data					
3.16	Estimation algorithm performance at 1 inch vertical height from magnet without					
9 1 7	Estimation algorithm performance at 1 inch vertical height from magnet					
3.17	Estimation algorithm performance at 1 inch vertical height from magnet 40					
9.10 9.10	Estimation algorithm error at 1 mich vertical height from magnet					
2.19	Comparison of opposition softings for in valuely approximation in agree					
0.20 2.01	La vehicle comparison of exposure settings for in-vehicle camera					
0.21 2.00	In-venicie camera algorithm interface showing various features					
3.22	View of QDC common charging filesials and DUDDC common sector.					
3.23	View of GPS camera showing fiducials and PURRS components					
3.24	Lateral estimation data from all sensors on the PURRS					
3.25	Lateral estimation data from all sensors on the PURRS (cropped)					
3.20	Estimation algorithm performance of the front magnetic sensor on the PURRS . 54					
3.27	Estimation algorithm performance of the rear magnetic sensor on the PURRS 54					
3.28	Estimation performance of the in-vehicle camera algorithm detecting inside lane					
3.29	Estimation performance of the in-vehicle camera algorithm detecting outside lane					
	edges					
3.30	Estimation performance of the GPS algorithm in the lateral direction 56					
3.31	Estimation performance of the GPS algorithm in the lognitudinal direction 57					
4.1	Relationship between different values calculated and used by the Kalman filter . 65					
4.2	Vehicle center of gravity location measured by the PUURS kinematic boom arm					
	used for estimation algorithm testing					
4.3	Comparison of sensor fusion algorithms 65					
4.4	Comparison of sensor fusion algorithms' error relative to ground truth 63					
4.5	Comparison of sensor fusion algorithms with corrupted GPS measurements 64					
4.6	Comparison of sensor fusion algorithms' error with corrupted GPS measurements 65					
5.1	Performance of estimation algorithms against fault of front magnetic sensor using					
	maximum value of N					
5.2	Innovations from first Kalman filter under fault of front magnetic sensor 69					
5.3	Performance of estimation algorithms against fault of front magnetic sensor with					
	an N value of 1000 \ldots \ldots 70					
5.4	Performance of fault reduction algorithms against a known fault of the front mag-					
	netic sensor (zero mean noise)					
5.5	Fault reduction algorithm error against a known fault of the front magnetic sensor					
	(zero mean noise)					
5.6	Performance of fault reduction algorithms against a known fault of the front mag-					
	netic sensor (non-zero mean noise)					

5.7	Fault reduction algorithm error against a known fault of the front magnetic sensor	
	(non-zero mean noise)	72

List of Tables

2.1	Sensor positions relative to the vehicles center of gravity	27
$3.1 \\ 3.2$	Estimation data correction factors	$\frac{53}{56}$
$4.1 \\ 4.2$	PURRS sensor variances and ranges before GPS corruption	64 64
5.1	Kalman filter combinations with associated sensors	67

Acknowledgments

In most acknowledgments, there are too many people to list, so I will try my best:

- to Clare; you are the best person in the world, and I have no clue why you put up with me. Between the late nights at Building B, proofreading my thesis, and my constant stressing, you have easily earned some sort of Fiancée of the Year award. I can't wait to spend the rest of my life with you and I hope I can help you as much as you helped me while you work on your thesis in a few months.
- to my parents; to whom I owe everything, because without you I would not be here. From being great role models, pushing me to do what is right and not what is easy, to helping me financially go to one of the greatest Universities on the planet.
- to Dr. Brennan for helping me through my time at Penn State and with my thesis work. I never would have gotten to where I am today without your mentoring and leadership.
- to the SMART scholarship program; it makes a world of difference to not need to worry financially when in college. I have been very blessed to have had financial and job security through most of my studies and I am eternally grateful.
- to the Naval Surface Warfare Center, Dahlgren Division for mentoring me during three great internships. I am looking forward to my time helping to improve the safety of this great nation.
- to Dr. Sommer for being my thesis reader on such short notice. I greatly appreciate your help.
- to those who have reminded me that we are all destined to do great things.

Dedication

When I was very young, my father was deployed to serve our country in war. While deployed, he sent a video home describing himself so that my brother and I would know who he was if he never made it home. May this work, and future work, help eliminate the number of loved ones lost in battle so they may never worry about who they leave behind. To my father, my brother, and all men and women who fight for freedom.

l Chapter

Introduction

1.1 Thesis Overview

One of the biggest hurdles faced when automating vehicles is obtaining an accurate estimate of where a vehicle's location relative to its environment. This task becomes more and more critical depending on the environment the vehicle is traveling, an example being: a vehicle traveling in an empty field can allow much higher errors in localization than a vehicle traveling in crowded city streets, because errors in open environments are less likely to result in failures or accidents. The localization problem is unique to each vehicle and its environment, with passenger or cargo vehicles that travel on passenger roadways being an especially difficult instance because they travel at high speeds (60 + mph) and human lives are often at stake.

The goal of this thesis is to evaluate a common algorithm, the Kalman Filter, and its resistance to sensor corruption or failure. The Kalman filter, discussed in depth in Section 1.2.5 and Chapter 4, relies on readings from multiple sensor and models of those sensors to provide an estimate to the states (such as position, velocity, etc.) of a system. These filter systems typically include sensors that are sensitive to their environment, such as GPS sensors requiring line-of-sight to multiple satellites or cameras requiring good lighting conditions, which can result in inaccurate Kalman filter estimates. In this thesis, the Kalman filter will be evaluated with compromised sensor readings, and various solutions will be implemented to retain a high level of accuracy from the filter estimates.

The evaluation of the Kalman filter will occur using sensor readings taken from physical hardware on the Pennsylvania State University Rolling Roadway Simulator (PURRS) which is a scaled vehicle testbed. This data will be collected from two magnetic sensors, a simulated GPS sensor (because line-of-sight cannot be obtained indoors), and a lane line detecting camera, all using the Robot Operating System (ROS). This data will then be post processed in MATLAB for sensor characterization and Kalman filter development. The Kalman filter is a well established technique, however additional layers of processing will be added to improve the robustness of

the system. These layers will include multiple "banks" of Kalman filters and map-based sensor characterization to alert the Kalman filter to known changes in environment.

The remainder of this chapter discusses literature associated with all of the sensors and techniques used throughout the testing of the Kalman filter. Chapter 2 describes the hardware used in the data collection and the development of this hardware for this thesis. The individual sensors and their associated sensing algorithms are described in Chapter 3 and the Kalman filter is described in Chapter 4. Finally, the results from this thesis are described in Chapter 5 along with conclusions and ideas for further development of this work.

1.2 Literature Review

Automated vehicles are a well studied topic in the field of mechatronics, controls, and vehicle dynamics. The design goal for automated vehicles is the same as any other robot, to perform a task so that a human does not have to. To be more precise, these vehicles reduce workload on humans by being able to travel autonomously while carrying dangerous, heavy, or precious cargo. Automated vehicles, specifically automated ground vehicles, are used primarily in two industries: materials handling and passenger driving, both of which are described in more detail below and the focus of this research.

1.2.1 Automated Highway Systems

An Automated Highway System (AHS) is described as any roadway that takes advantage of technologies that increase capacity and safety or decrease environmental impacts or economic and psychological costs of accidents. It is important to note that these goals require solutions where control and estimation are coupled, in that the vehicle will need to accurately know where it is currently and where it is going before it can determine how to get there. Fortunately, the control and estimation problems can be studied individually; the estimation or localization problem is the focus of this research.

The concept of an AHS was first described by General Motors in the 1939 Worlds Fair [1] and since then, many technologies have been studied and applied to highway systems. But very few are currently at a stage of commercial deployment. The main difficulty faced by these technologies is the requirement of the solution to not only be low cost, but also able to operate safely in any roadway environment. These two factors have proved challenging to overcome for many potential technologies: Global Positioning Systems require a relatively clear view of the sky and often fail completely in tunnels. In-vehicle cameras and laser scanning techniques require clear sight of the road or need prior knowledge of the road, which increases computational requirements and sensor cost. Magnetic guidance requires a high upkeep and expensive road infrastructure that would have to be deployed over all roadways to be effective. These technologies have their costs and benefits but individually cannot be used to accomplish an AHS without further improvement.

The realization that one sensor alone cannot be used to create an AHS has caused the develop-

ment of sensor fusion algorithms to combine individual sensors into a single robust measurement. The most robust sensor fusion algorithms use all of the available data to provide the best position estimate, but are also capable of relying on other sensors when one fails. A good example of a robust sensor fusion algorithm is a fusion of GPS and In-Vehicle camera; when the weather is too inclement for the camera to provide usable information the GPS is solely used; conversely, the camera is used when the vehicle enters an area where GPS is unusable, such as a tunnel. If both sensors can be used, the algorithm can utilize both sets of data to provide the best estimation possible. Sensor fusion algorithms appear promising for use in an AHS as well as in an automated vehicle environment and are therefore the focus of this research. In order to develop a sensor fusion algorithm, each individual sensor must be studied. Each of these sensors are outlined in the sections below.

1.2.2 Magnetic Guidance

The fundamental problem of vehicle localization can be divided into two categories: lateral position estimation and longitudinal position estimation. For convention, the lateral direction is shown as the Y-axis in Figure 1.1 while the lognitudinal direction is defined as the X-axis. The lateral position estimation problem is extremely important on passenger roadways because it is responsible for lane-keeping. Many of the sensing methods examined for use with an Automated Highway System are lateral sensing methods, including magnetic guidance.



Figure 1.1: Notation for ground gehicles (Image courtesy of [2])

The earliest research in magnetic guidance was performed by Zworykin et al. in 1958, specifically in the use of inductive wire guidance for lateral vehicle control [3, 1]. The basic concept behind inductive wire guidance is to embed an alternating current carrying wire into the roadway surface. The alternating current creates a magnetic field that can be detected by an inductor coil on the vehicle. Zworykin's work used a single wire to guide each vehicle, but proposed the use of multiple wires to differentiate between lanes in order to aid in lane changing and avoidance maneuvers. One of the primary reasons Zworykin et al. chose to study inductive guidance was that it was one of the inexpensive and feasible options in 1958. An inductive wire system can be created using a function generator, an inductor coil, and basic electronic components; other sensing technologies such as Global Positioning Systems (GPS) and in-vehicle cameras require large, expensive infrastructure and computational power which was not possible in the late 1950's.

Inductive wire guidance was further studied for use in an AHS, notably at The Ohio State University [4, 5, 6, 7]. In the work performed by Olson in 1973 [5], the focus was to determine a way to overcome the magnetic disturbance caused by ferrous and ferromagnetic materials in the structure of the road. This disturbance is significant because most roadways, particularly bridges, have a steel support structure that can significantly affect the shape and strength of a magnetic field. This work reports, later to be reiterated by [7, 1], that the magnetic field produced by the wire is of the shape shown in Figure 1.2. Note that in this Figure, the vertical component of the magnetic field is shown on the Y-axis and the displacement from the wire is shown on the X-axis. The wire is placed at the origin.



Figure 1.2: Shape of inductive wire magnetic field shown in [5]

In Olson's work, an array of inductor coils oriented perpendicular to the road surface is used to sense the magnetic field, with an additional coil parallel to the road surface to generate a reference signal. It was determined that the phase of the signal produced on each sensing coil, when compared to the reference coil, was a more robust method of determining the location of the wire. This is because the phase is a discrete property, as the phase changes from one side of the wire to the other. In contrast, the amplitude of the field is dependent on distance from the wire (as seen in Figure 1.2) and environmental conditions. This method was successful on reinforced concrete at 80 mph. These findings are discussed in more detail in Fenton's 1976 work, [6] where a 0.0635 meter maximum tracking error was observed on both straight and curved roads. This level of accuracy shows extreme promise for this technology; however, one major problem does exist with inductive wire guidance: it is expensive to implement in full scale. To create a fully functioning inductive wire guidance system on a roadway, wires would need to be embedded and powered for the entire length of the road. This has the potential to be very high-cost and have very high upkeep, since any break in the wire would result in total failure of the system. Branches are also difficult to implement in inductive guidance systems, because continuous loops are needed for operation.

Another concern for any sensing technology is that it can be implemented easily and safely on a vehicle. The specific concern for inductive guidance is that the inductive coils need to be close to the road surface to get a strong signal from the field producing wire. Olson's work shows that the basic inductive guidance is physically feasible because the array of magnetic sensors is at a height of 8 inches above the road surface [5]. This height is tall enough to easily implement on a passenger vehicle, as the inductive coils could be attached to the vehicle on the bumper or on the suspension system.

In Olson's later work, specifically [7], three different wire and sensing configurations were studied. The first of these configurations is shown in Figure 1.3. In this design, the wire loop is used to create a more robust system by having two magnetic fields for the vehicle to guide to. The length of the roadway could be divided into sections, with one loop in each section per lane, which would reduce the total length of wire per loop. This would have the benefit of reducing the affected length of roadway when a loop fails but would increase cost, because each loop would need its own alternating current generator. The control algorithm would also need to be more robust to handle the transitions between the loops. The other two configurations are the two studied in Olson's earlier work [5], having a single wire and using the magnitude of the measured field or the phase difference between the sensing and reference coil to determine the position of the wire. All three configurations were proved to be viable.

A similar two wire system to the one used by Olson was used to create a communication system that guides the vehicle and could communicate will all of the vehicles using the magnetic fields for guidance [8]. The major difference between the two systems was that the one designed by Matsumoto used a different frequency between the two guide wires to create the communication system, seen in Figure 1.4. These two wire designs do provide benefits over a one wire design, namely a more robust system because two magnetic fields can be sensed and differentiated if two frequencies are used. As previously mentioned, these systems can be designed to reduce the effect of a system failure but at the cost of complexity and more equipment.

An example of an inductive wire guidance system in full size, long-term use is the WesTrack facility. This facility was designed to test the life-cycle of asphalt using vehicles guided by a two wire system[9]. The track had effectively two single wire systems, each with a different frequency of alternating current to give the onboard sensors the ability to identify each wire individually. The trucks also featured two sensor coils on each side of the vehicle to measure both the horizontal and vertical components of the magnetic field. The sensors were mounted to the bumper and only 100 milliamps of current was required in each wire. Since a relatively small amount of current was needed to power the system, even with the sensors mounted to the



Figure 1.3: Two wire configuration used by Olson in [7]



Figure 1.4: Two wire configuration used by Matsumoto in [8]

bumper of a semi-truck, it makes the system more feasible in terms of cost because low current power supplies are inexpensive and safer than higher current power supplies.

In these studies, four driverless tractor trailers drove a 2.9 km oval test track at 65 km/hr for around 15 hours each day to accelerate pavement wear from loading. Their testing resulted in more than 700,000 km of total distance traveled by all of their driverless vehicles. It was reported that the trucks could be guided using the wires to less than 2 mm in lateral variation given the simple control architecture used and that slop in the steering actuators was reported to be significant after the 700,000 km of testing [9]. The WesTrack studies not only show that

inductive wire guidance is possible, but it can be used to guide vehicles on oval and straight sections for long periods of time with high consistency. The inductive guidance system was feasible for these studies because the roadway was relatively short. Maintenance problems were reported with the guidance system, but were attributed to the creep of the pavement causing a stretch and break in the wire. This problem was fixed by installing the wire into a channel in the roadway as opposed to installing the wire inside the pavement during initial construction.

Wire guidance was also used for automated vehicles in working environments starting in the 1960's [10]. These vehicles are known as Automatic Guided Vehicles or AGVs. AGVs can range in size and can perform any job, from materials handling [10, 11, 12] to "patrolling" a power station looking for faults [13] to plowing snow [13, 14]. AGVs were not magnetically guided originally, as they have been reported in use since the 1950's [15]. This technique was and still is very popular and has been well patented [16, 17] and studied [18, 19]. This technique was also expanded to guide and power an AGV through induction [20]. These systems also share the same drawbacks as when they are used for an AHS; they are expensive to implement and upkeep. Inductive guidance systems are more feasible in factory or industrial settings because the possible paths for the vehicles to travel are often fewer in number and much shorter than passenger roadways.

Another magnetic guidance method is to use a passive, discrete magnet buried into the road surface. This method has many advantages including: it is a passive system which requires less maintenance and a lower operating cost, discrete magnets can be installed in drilled holes in the roadway as opposed to cutting a channel for a wire, and failures of a discrete magnet are unlikely and will not cause failure of a large section of roadway. An early study in the use of discrete magnets was in 1971 by Rudolf Mahrt [21] which included the control stability of using magnetic markers and the benefit of using multiple markers at each point instead of using a single marker. This work was continued by Johnston et al. [22], using a Kalman filter and looking at the spacing of the markers as well as the effect of an incorrectly placed marker. This study showed that using markers to guide a vehicle was possible though it took a relatively long distance, 150 ft at 10 ft marker spacing and 20 ft at 1 ft marker spacing, to achieve a desirable response despite the vehicle traveling at 7 mph. Both of these studies were only software simulations because control computations were not possible on board a vehicle due to the large amount of memory required for the Kalman Filter Algorithm.

In the 1990's, the California PATH (Partners for Advanced Transit and Highways) program based at the University of California, Berkley performed a significant amount of research in the area of magnetic guidance. In 1991, Hessburg et al. studied PID control using a scale vehicle and discrete magnetic markers [23]. It was shown that PID control was not suitable in this capacity unless a feedforward loop was used with prior knowledge of upcoming roadway geometry. Hessburg later tested Fuzzy Logic on a full scale vehicle and it was found to be successful [24]. Lateral control of vehicles using discrete magnets was then expanded to include longitudinal control using the known spacing of the markers as well as an accelerometer to estimate velocity and range sensors to avoid collisions [25]. This research culminated in a demonstration of vehicle platooning at highway speeds in 1998. The demonstration featured eight vehicles driving at 60 miles per hour at 6.5 meter spacing. The vehicles used magnetic markers for lateral positioning, and coding of upcoming road information by using alternating polarity of the embedded magnets was also employed. This method is discussed in depth by [26, 27] and an extremely advanced method of imprinting data onto the road surface is shown in [28]. In the 1998 demonstration, the vehicles performed maneuvers such as lane changes, close following, platoon splitting and reforming, and lane keeping. In order to aid in lane changing, additional magnetic markers were used between the lanes which helps the control algorithm during the lane changes. The methodology for how the magnetic markers were placed to achieve these vehicle maneuvers is discussed in [29] and the controller designs are discussed in [30]. The lateral control tracking error was reported to be less than 10 cm throughout the demonstration [31]. This demonstration shows that a guidance system using discrete magnetic markers is possible and further analysis is provided in [32]. The California PATH Program also studied the robustness of using a front and rear magnetic sensor which is discussed further in 1.2.5.

The magnetic guidance method chosen for this research is to use a continuous magnetic strip. This system provides benefits over both wire guidance and discrete magnet systems. The first of these benefits is that there is a continuous magnetic field for the vehicle to track, similar to a wire guidance method. This means there is always a magnetic field to guide to, in contrast to discrete markers where the field is present in some areas and absent in others. Magnetic strips can be passive, like discrete magnets, which eliminates the need for a power system and breaks in the magnetic strips do not cause failure to the system. Branches can be easily created using this system if the vehicle has multiple sensors, as the track appears to widen until it becomes two separate magnetic signatures, which can be sensed and followed. This method was studied by the Macome Corporation in Japan in 1987 for use with AGVs [15]. This research provides a comparison of using a magnetic strip for guidance versus other methods which is shown in Figure 1.5. In this chart, the magnetic strip method is listed as "Magnetic guidance (MACOME)" with inductive wire and optical methods shown in the other columns. Optical methods are often used in manufacturing vehicle capacities and commonly use fiducials and cameras or simple line following techniques. It is unclear what is meant by "Complex branch and joint" as branches and joints are successfully implemented in this paper. This research also shows a simple visualization of the magnetic field given off by a magnetic strip, shown in Figure 1.6. Magnetic tapes were further developed by the 3M company in 1996 and provided an account for how the tapes were created and tested. This work shows an agreement between predicted and observed field strength [33]. It does not discuss the models used to predict the field strength.

A final method for magnetic guidance has been studied for both highway and industrial use which uses discrete magnetic markers that are active or semi-active. An example of one of these systems is shown in [12] which uses a transponder that is activated when a vehicle passes near it. The vehicle has an antenna that creates a magnetic field to which the transponder responds. The antenna can detect this response which includes information about the transponder. This information can be used to locate the vehicle if prior knowledge (such as a map) exists about the

Item		Magnetic guidance (MACOME)	Electro- magnetic guidance	Optical guidance		
Route restriction	Office Factory Open air	fit fit fit	unfit fit fit	fit fit unfit		
Installation	Period Cost	short low	long high	short low		
Route change		easy	difficult	easy		
Complex brand	h & joint	unfit	fit	unfit		
Long route		fit	unfit	fit		
Maintenance	intenance Check easy easy Repair easy difficult		easy difficult	difficult easy		
Communication the route	using	impossible	possible	impossible		

Figure 1.5: Comparison of guidance methods shown in [15]

network of transponders. Radio Frequency Identification (RFID) tags operate in a very similar manner to the transponders and are also mentioned for use in localization [34].

In addition to magnetically guiding passenger and industrial vehicles, magnetic guidance has also been studied for other fields. Naturally occurring magnetic fields have been suggested to guide underwater autonomous vehicles [35, 36] as well as using magnetic signatures given off by underwater mines for mine countermeasure underwater vehicles [37]. A novel use of magnetic guidance is to help people affected by blindness walk on designated sidewalks safely. Researchers designed a walking stick that senses discrete magnets and sounds and alarm based on the strength of the magnetic field [38].

Magnetic guidance has also been used in reverse, to sense vehicles instead of guiding them. This has been shown for use in parking lots to determine occupancy of parking spaces [39] as well as for highways to estimate traffic conditions [40]. This technique has also been studied for use with airports to determine occupancy of runways and gates. An advantage to using magnets to sense occupancy is that it can be done without modification of the vehicle. Gao et al. has even shown that the magnetic detection system can be refined to distinguish between different vehicles based on their magnetic signature [41]. These systems use both the magnetic field created by the vehicle and the disturbance of the natural magnetic field in the area.

In conclusion, it has been shown since the late 1950s that inductive guidance is a viable option for lateral control, but it is an expensive system to implement. Passive systems such as discrete



Figure 1.6: Magnetic field produced by a magnetic strip shown in [15]

markers and magnetic strips are much less expensive to implement and more robust to failure. Magnetic strips have been chosen as the focus of this research because they provide improvements to both inductive and discrete systems. This method will be studied experimentally and results will be discussed in 3.1. Inductive and discrete magnetic guidance methods should be evaluated further for use in smaller scale applications, such as the Larson Transportation Institute Test Track, because these methods have been shown to be very successful. Unfortunately, these methods are not within the scope of this research, but all testing apparatus developed will have the ability to test these methods in the future.

1.2.3 In-Vehicle Camera

One of the most versatile sensors used by a human while driving is their eyes. The eyes provide information for collision avoidance, lane keeping, and as many car states as are shown on the dashboard indicator. Because vision is so central to humans driving vehicles, using a camera as a sensor in a vehicle has been a well-studied topic. Generally, using cameras as sensors falls under two fields: computer vision and photogrammetry. Computer vision is an encompassing field that covers all aspects of using a camera as a sensor: from acquiring the image, processing the image, and applying the processed image data/information [42]. Photogrammetry is a more specific field that is focused on abstracting three dimensional information about an object from multiple two dimensional photographs [43]. This research is within the field of computer vision, but photogrammetry techniques are sometimes used, specifically for camera calibration and distortion reduction [44].

Vehicles experience many different environments, many of which cameras are not suited for, such as night-time driving on poorly lit roads and weather conditions that obscure vision such as fog and snow. Cameras also produce large amounts of data to provide the best detail possible, which is helpful in terms of the capability of vision sensors but comes at a cost of requiring large computations [45]. The need for this much computational power is evident in the 1985 work of Waxman et al. [46] because a significant portion of the research is dedicated to efficient computation methodologies for images which still required 60 to 120 seconds to process. Even with improved efficiency, the maximum estimated speed for the vehicle using their algorithm would be 10 km/hr (6.2 mph). Maximum vehicle speed while using a control algorithm is dictated by processing time and fortunately the continuous improvement in computer computational capacity and efficiency has made vision-based sensors possible on vehicles at highway speeds [45].

Use of vision-based sensors for vehicle guidance is commonly broken into two categories: object/pedestrian detection and lane detection, which is within the focus of this research. Lane detection is a complex problem, particularly because roads are variable in many different ways. Roads vary in the amount of standardization from fairly standardized (in terms of lane width, markings, construction, curve radii, etc.) highways to non-standardized rural roads. Road surfaces also vary considerably from dirt and gravel to asphalt and concrete which are not only different visually, but also determine if markings can be present to help drivers, as well as vision sensors, guide to the roadway. The unstructured road problem, not using markers for road keeping, has been studied using the UNSupervised Clustering Applied to Road Following (UN-SCARF) algorithm [47], in [48], and the 1995 research at the Pohang University on the PRV II vehicle [49].

The structured road problem has been extensively studied and is the problem faced on the Pennsylvania State University Rolling Roadway Simulator (PURRS). Many different solutions exist and serve different purposes, from lane departure warning/intervention research performed at the University of Michigan [50] to fully autonomous vehicle navigation [51, 52]. These algorithms are primarily performed using a single camera; however, stereo cameras have been used [53]. For example, Carnegie Mellon University has taken the single camera approach a step further, by using two cameras, each with a different iris settings, in a system known as Navlab. The different iris settings allow for a greater range of detected colors and effectively forms a six-dimensional color space, as opposed to the standard three-dimensional color space given by a single camera. The two camera, two iris system allows for a greater range because one camera is set to capture details in darker images whereas the other is set for lighter images. The data from both cameras can be used to accurately sense lane lines in scenes with both light and dark sections, such as a road with shadows [54]. Carnegie Mellon has also performed research using inverse perspective mapping, transforming an image taken from a camera to a birds-eve-view of the scene, along with an area of interest for autonomous vehicle navigation. The Rapidly Adapting Lateral Position Handler (RALPH) system uses the transformed images along with a straightening technique to

determine road curvature, shown in Figure 1.7 [52]. Additional examples of inverse perspective lane detection techniques are Borkar et al.'s work using thresholding and Hough transforms [55], Kim's work at the California PATH program [56], and Shu et al. which describes the transforms in depth [57].



Figure 1.7: Determining road curvature using straightening technique used in [52]

1.2.4 Global Positioning System (GPS)

The Global Positioning System (GPS) is a Department of Defense developed system that uses signals from multiple satellites to locate a receiver on earth [58]. Before the system became fully operational, the estimated localization error was 16 m with access to all of the transmitted data (for government and military use) while using the coarse data (for everyone) yielded an error of around 40 m [59] to 100 m [60]. This level of accuracy, although not enough for autonomous highway driving, still led to significant research advances in vehicle localization because GPS could place a vehicle on earth within 100 m of its true position without data intensive maps or high cost infrastructure systems. The Standard Positioning Service (SPS) data available to the public was purposefully corrupted, commonly known as Selective Availability (SA), to prevent users from having the same high-precision localization abilities available to the government Precise Positioning Service (PPS) users. In May of 2000, SA was deactivated, which decreased localization error for SPS users to 5 to 10 m. PPS users have benefited from improvements to the GPS system as well as various estimation techniques that have kept their localization error lower than SPS users. Both PPS and SPS users have used a technique known as Differential GPS (DGPS), the use of a base station at a known location to broadcast and remove some localization errors, to consistently obtain sub 10 m position estimation. If real time position estimation is not required, post processing data with highly accurate data about known GPS errors and satellite positions can obtain millimeter position accuracy [60]. It is important to note that accuracy varies with many factors including the type of GPS receiver. As an example, research performed by Bajikar et al. using DGPS before the 2000 removal of SA reported less than 20 cm nominal

accuracy [61].

Because post-2000 GPS can provide position estimates of less than 10 m, GPS has become widely used in vehicle localization research and application. GPS receivers have become low cost, enabling many drivers to have them at their disposal to help with manned vehicle driving, specifically on passenger roadways. GPS is a line-of-sight sensing technique, which means it is susceptible to loss of signal, specifically in urban environments, and tunnels which makes it unfit for localization in those situations; however, GPS is typically used with other sensors to improve real-time localization estimation, such as inertial sensors [62, 63, 64]. Sensor fusion is typically achieved using Kalman filtering, which is discussed in the next section.

1.2.5 Sensor Fusion and Fault Detection

Each individual sensor discussed in the sections above yield favorable data for the localization of a vehicle; however, they each suffer from known failure modes: magnetic sensors can be affected by ferrous materials commonly found on roadways [5] or external magnetic fields, cameras used for lane detection are sensitive to environmental factors including lighting and weather conditions, and GPS service can be disrupted when line-of-sight cannot be achieved. Because of these factors, it is common to combine these sensors to achieve more accurate and more robust measurements. The simplest method is to use an averaging technique; in terms of these sensors and this research, the equations would look as follows:

$$X_{lat} = 1/3 * (x_{mag} + x_{IVC} + x_{GPS,lat})$$
(1.1)

$$X_{log} = x_{GPS,log} \tag{1.2}$$

Recalling Figure 1.1 and Section 1.2.2 for lateral vs. lognitudinal notation, Equations 1.1 and 1.2 show the simple averaging algorithm approach for a magnetic sensor, in-vehicle camera, and a GPS sensor. It is important to note that the magnetic sensor and the in-vehicle camera primarily report lateral position estimates, therefore they are only represented in Equation 1.1. This averaging technique is not optimal because it does not represent each sensor adequately in terms of accuracy; the post-2000 GPS position data typically has an accuracy of around 10 m [60], whereas previous research in magnetic sensing technologies has shown an accuracy of 2 mm [9]. Averaging these readings together not only does not take full advantage of the data, but can also make the true accuracy of the reading unclear. In order to solve this multi-sensor problem, the Kalman filter can be used.

The Kalman filter, described by Rudolph Kálmán in 1960, is a study of the data smoothing or interpolation problem with application to data corrupted by noise. The Kalman filter is in state space form, is derived from the Wiener-Hopf integral equation, and is considered optimal if all noise sources are Gaussian. The Kalman filter is described in depth in [65] as well as in Chapter 4. The Kalman filter is a state estimator which means that the filter is used to predict the output of the system the filter is applied to using previous state information and a system model.

The Kalman filter is widely used for estimation of states in noisy systems, which include most real-life dynamic systems, as well as the determination of faults. Since the states are being estimated by the Kalman filter, a very important measure, the innovations, can be defined:

$$v_i = z_i - \hat{z}_{i/i-1} \tag{1.3}$$

where z_i is the output of the system at time i, $\hat{z}_{i/i-1}$ is the unbiased minimum-variance estimate at time i given data up to time i-1, and v_i is the innovations. The innovations term effectively represents is how close the Kalman filter's estimate of the system output was to the actual output. The innovations terms, assuming all system noises are zero mean and Gaussian, will also be zero mean and Gaussian. Knowing this fact, researchers have been able to use tests for whiteness, mean, and covariance to detect faults in the system, because the properties of the innovations term will change with the properties of the system noise [66]. In addition to failure detection, Mehra has shown that the innovations term can be used to verify the optimality of a Kalman filter as well as estimate the process noise and measurement noise matrices required for the Kalman filter [67].

In 1977, Kerr suggested a different approach: to compare the statistics of an error free model to one that includes probable errors, such as known drift rates or biases, as states. Kalman filters generate estimates for both of these models and the confidence regions are analyzed for physical overlap, visually shown in Figure 1.8. If an overlap exists, such as time t_1 and t_2 in the figure, than a failure has not occurred, but when no overlap exists an error has most likely occurred. These uncertainty regions take the form of confidence regions in one dimensional cases or ellipses, such as the ones shown in Figure 1.8, in two dimensional cases [68, 69] as is the case with this research.

Both the work of Kerr and Mehra present failure detection algorithms that have proven successful, but are not designed for failure identification. The use of multiple Kalman filters has been proposed using the techniques designed for single Kalman filters to determine not only when a failure occurred but in which system or sensor the failure occurred. This is done by using a "bank" of Kalman filters to estimate states and faults based on different combinations of sensors [70, 71, 72].

The earliest use of multiple estimators was Magill [73], who estimated stochastic processes. Athans et al. [74] proposed the use of multiple linear-quadratic-Gaussian (LQG) compensators, where Kalman filters are used for state estimation and fault detection, to create the Multiple Model Adaptive Control (MMAC) method. The MMAC method was used to control aircraft systems over different operating conditions and resulted in the development Multiple Model Adaptive Estimator (MMAE) by Maybeck [75] for fault detection and identification in aircraft [70]. The use of multiple Kalman filters for fault detection and identification has been well studied for different estimation problems [70, 72] and will be a focus for this research.





Figure 1.8: Visualization of two-ellipsoid test shown in [68]

Chapter 2

The PURRS

A prominent feature of this research is all of the algorithms are tested on scaled vehicles. Using scaled vehicles offers several advantages over simulations and full scale vehicle testing, namely: lower operating cost, higher safety, and smaller testing areas, while still providing hardware-in-the-loop experiments. Scaled vehicle testing has been used by various research facilities including the University of Illinois [76, 77, 78], the Virginia Polytechnic Institute and State University [79], and the Intelligent Vehicles and Systems Group at the Pennsylvania State University [2] for testing of various algorithms and to determine the similarity between scaled and full size vehicles.

Scaled vehicle testing performed at the University of Illinois and in the Intelligent Vehicles and Systems Group take place on rolling roadway simulators. These testbeds are effectively treadmills with large decks and are capable of higher speeds than treadmills designed for exercise. The treadmill design allows a scale vehicle to travel large lognitudinal distances within a small space but lacks the ability to host tests with large lateral vehicle displacement. The rolling roadway simulator used by the Intelligent Vehicles and Systems group is known as the <u>P</u>ennsylvania State <u>University Rolling Roadway Simulator or PURRS</u>. The PURRS features a 6 ft by 9 ft deck and uses a 2 HP AC motor with Variable Frequency Drive (VFD) to drive the belt and rollers at various speeds. A boom arm with two high-resolution encoders is used to measure the true position of the vehicle. Various improvements and changes have been made to the PURRS since it was used in [2], which will be described in the sections below.

2.1 Structural Changes to the PURRS

One of the unique features of the PURRS was the capability to angle the treadmill deck seen in Figure 2.1. This capability was originally performed by four linear actuators and string potentiometers for feedback and these actuators were strong enough to move the PURRS treadmill frame in its original design, but could not lift the frame with the improvements discussed later in this chapter. Fortunately, the lab which houses the PURRS also has a 6 degree of freedom

motion base which is used for immerse vehicle simulation studies. Because the motion base typically moves the cab for a commercial vehicle, either a semi-truck or farm tractor, it has the capability to move the treadmill frame if needed. The PURRS frame would need to be modified in the future to attach to the motion base, but would give the ability for the PURRS to be used in research involving sloped roadways. Until this modification is completed, the exterior frame for the PURRS, the frame that contained the actuators and potentiometers, was removed to save space. Castor wheels and leveling, vibration damping feet were added to the interior frame, which contains the motor, rollers, and deck for the treadmill, to increase the mobility of the frame.



Figure 2.1: PURRS being used for similitude research (Image courtesy of [2])

2.2 Treadmill Deck Changes to the PURRS

Previous research and testing using the PURRS has shown that the friction between the belt and the supporting deck needed to be reduced. The original design used particle board to support the treadmill belt to form the roadway surface, but the surface is approximately 6 ft by 9 ft which caused the friction to become too large for high speed testing or testing with large, heavy vehicles; other concerns such as the particle board being uneven contributed to the high friction as well.

A new deck was designed to reduce friction using a large area air bearing. The basic design

is very similar to an air hockey table: a blower pumps air into a box with small holes in the top where the air escapes. The escaping air applies a force on the treadmill belt, which reduces friction between the belt and the underlying surface. This new air bearing deck design is unique because of its size as well as the need for the box to be less than 5 inches tall to not interfere with the existing frame or treadmill belt. The surface of the box frame was chosen to be aluminum due to cost, weight, and conductivity to eliminate the possibility of static buildup on the surface due to the belt sliding over the supporting deck surface. The the majority of the blower box frame was constructed with wood, to achieve a balance between low cost, strength, and weight. The frame can be seen in Figure 2.2: Figure 2.2 shows some of the features of the new air



Figure 2.2: Inside of the new PURRS blower box frame

bearing box for the PURRS, particularly the design for future magnetic guidance research: there are three magnetic strips in the new deck, the two shown in white are magnetic north side up where the strip in the middle is magnetic south side up, like the magnetic strip on the MGCS. There are also wires pre-laid for future inductive guidance research every half foot to allow for multiple types of systems to be tested. These wires are neatly routed down the ends of the box frame and labeled, as seen in Figure 2.3.

A unique feature of the blower box frame is the ability to change the size of the holes that the air escapes through. This is done by having two perforated aluminum plates that are able to slide relative to each other, with these two 1/8 inch thick plates having 1/4 inch holes on 3/8inch centers. The first of the 4' by 8' foot sheets is attached to the rest of the box frame and



Figure 2.3: Inductive wires on the side of the blower box frame

is not intended to move. The second sheet is mounted to the box frame through the top sheet, but the screws sit in countersunk slots, as seen in Figure 2.4, to allow for the top sheet to slide relative to the bottom sheet. Because two sheets are used, the top layer of the box frame is actually two 1/8 inch thick layers, one for each perforated sheet. And since the treadmill deck is 6' by 9', and the perforated sheets are 4' by 8', there are non-perforated sheets surrounding the perforated sheets to make up for this extra distance. The bottom layer is shown in Figure 2.5, which consists of the perforated sheet in the middle, and hardboard surrounding it to make the surface 6' by 9'.

Figure 2.5 also shows the blower in the top of the figure, which is typically designed to dry floors in buildings and was chosen because of its ability to blow large volumes of air, and was extremely low cost in comparison to an industrial three phase blower. The blower is attached to a ducting system which routs the air to three points in the box frame through the connections shown in Figure 2.6. These attachments are spaced relatively equally along the length of the blower box and allow for a quick disconnect between the blower and the air bearing box. The completed blower box, mounted in the frame is shown in Figure 2.7 with the new PURRS vehicle and lane lines.



Figure 2.4: Countersunk slots on the top perforated aluminum sheet



Figure 2.5: First layer of the blower box frame



Figure 2.6: Box frame blower attachment



Figure 2.7: Completed box frame with vehicle and lane lines

2.3 Vehicle Changes to the PURRS

Before this research, very specialized vehicles were designed for use on the PURRS. The vehicle used by Lapapong [2] did not have a motor, had a very accurate rack and pinion steering system, and was weighted in a way to be as close to a full-scale vehicle as possible. This vehicle can be seen in Figure 2.1. These characteristics created a vehicle that was very fragile and difficult to use with non-related research. This created the need for a new vehicle that is more rugged and more applicable to a wide range of research. Unfortunately, this comes at the cost of the scale vehicle not behaving exactly like a full size vehicle, but for this research, the focus is much more on the sensing technologies instead of the vehicle dynamics.

An off-the-shelf remote controlled (R/C) car was used to create an all-purpose treadmill vehicle. These vehicles are designed to be rugged and provide a cheaper platform for research, in both financial cost and time. Using a pre-designed vehicle also reduces the cost for replacement parts in the event of a crash and many upgrade parts are already available to change the handling characteristics, performance, and durability of the vehicle. Using standard motors and the factory drivetrain allows for more flexibility in research as motor models could be tested without potentially complex drivetrain development for a custom-made vehicle.

Scale vehicles come in many sizes and designs; an 1/8th scale rally-car design was chosen because 1/8th scale vehicles typically have ample room to mount additional electronics but do not come with the high price point associated with larger scale vehicles. The rally car design was chosen because it is the most like a passenger vehicle in both size and suspension characteristics. The vehicle chosen is the HPI Racing WR8 Flux because it is a rugged, 1/8th scale rally car design vehicle from a well-known and well-respected company. This vehicle is shaft-driven four wheel drive and comes with motor, steering servo, electronic speed control, and a 2.4 GHz transmitter/receiver combo which will all be used, with additional components, for computer control. Another important feature for this vehicle is that it is very close to being a 1/8th scale vehicle in size when compared to the vehicle it is styled after, the 2012 Ford Fiesta. Often Remote Controlled vehicles will widely vary in size within the same scale, but the HPI WR8 Flux is fairly close in width, the primary dimension of interest because the lateral position is being estimates, to actually being a 1/8th scale vehicle. This is shown by the following calculation

$$Scale_{actual} = \frac{Dim_{real}}{Dim_{scale}} = \frac{67.8}{8.9} = 7.6 \tag{2.1}$$

where the real and scale dimensions are provided by Edmunds and HPI respectively. Please note that these dimensions are given in inches. The vehicle in its factory configuration is pictured in Figures 2.8 and 2.9.

Using a new, stock remote controlled vehicle on the PURRS posed an interesting challenge in how to control the servos and motors through ROS and still have the ability for manual control; controlling the vehicle through ROS gives the ability to test control algorithms on the PURRS, while having manual control is useful for troubleshooting and emergency situations.



Figure 2.8: HPI WR8 Flux with the body (Image courtesy of HPI Racing)

For reference, the original control architecture, and the standard architecture for most electric remote controlled vehicles, is given as Figure 2.10. The vehicles previously used on the PURRS used a motor to control the steering, but did not have a motor for propulsion, and there was no easy way of controlling the steering manually.

Two options were considered for the control of the new PURRS vehicle: The first option is to use the trainer port on a remote controlled vehicle transmitter. In normal usage, the trainer port gives the ability to connect another transmitter to the original transmitter and to switch which transmitter is controlling the vehicle. This is very similar to having two sets of steering wheels and pedals in a car for drivers education classes. It gives the trainer the ability to control the vehicle when they want to, and the trainee control when the trainer deems it safe. In terms of this research, the trainer would be the manual control system and the trainee would be an Arduino connected to the trainer port sending computer control commands for the vehicle. This option has two major problems, the first being that all commands, both manual and computer, must be sent through the transmitter, over the air, and then to the receiver. This creates delays which could affect the models used when controlling the vehicle via computer. The second problem is that the Arduino would need to interface and communicate with the trainer port, which can be difficult.

The second option, and the option used on this vehicle, is to use an Arduino to control which signals are being sent to the servo and motor. The transmitter and receiver have an additional channel which is the state of a switch on the transmitter, normally used to control flaps or landing gear on a model airplane or lights on a model car. This switch will be used by the Arduino to determine if manual or computer control signals should be sent. The overall architecture is shown



Figure 2.9: HPI WR8 Flux without the body (Image courtesy of HPI Racing)



Figure 2.10: Typical electric remote control car scheme

in block diagram format in Figure 2.11. The ROS computer will send control commands to the Arduino which will convert and send them to the electronic speed control and servo which both operate using the standard servo control pulse width modulation timing. The Arduino responds to commands it receives from ROS by sending current control data to the servo and electronic speed control. This is done for two reasons; the first is for ROS to know what control data is being sent to the actuators for use in vehicle models and the other is to maintain a high level of robustness in the serial communication between the two devices. In addition to back and forth communication between the Arduino and ROS, each message is coded with a stop and start character to verify the fidelity of each message.

The newly designed control system for the vehicle was installed on the PURRS vehicle and can be seen in Figure 2.12. The shield seen in this figure provides a permanent wiring board for the control Arduino without modifying the existing electrical cables on the vehicle as well as providing two LEDs to distinguish which signal, either manual or computer, is in control of the



Figure 2.11: New remote control car scheme

vehicle. The receiver and its waterproof housing were relocated next to the Arduino for ease of wiring.



Figure 2.12: New control components installed on PURRS vehicle

An important part of this hardware-focused research is getting an accurate description of as many vehicle states as possible, such as position, velocity, and pose, which will allow for the best comparison when estimation of these states is performed. The PURRS was already equipped with a boom arm, shown attached to the vehicle in Figure 2.1, which measures the lateral and longitudinal position of the vehicle using encoders. The boom arm was damaged when the
treadmill was moved to its new location and minor repairs were performed. Roll, pitch, and yaw of the vehicle are also important parameters in defining the pose of the vehicle and are measured using encoders mounted on an assembly with freedom in each of these rotational directions. The assembly used in previous research on the PURRS was not easily adaptable to different vehicles and was overly complicated, therefore a new assembly was designed and built.

Keeping adaptability in mind and using lessons learned from the previous design, a new design was developed and built. A 3D rendering of the new roll-pitch-yaw assembly is shown as Figure 2.13. This assembly has three US Digital S2 Encoders, similar to the ones shown in Figure 2.14. These encoders require a flexible shaft coupling to prevent stresses on the shaft of the encoder; these stresses could cause inaccurate encoder readings or mechanical failure of the encoder.



Figure 2.13: 3D rendering of the roll-pitch-yaw assembly



Figure 2.14: US Digital S2 encoder (Picture courtesy of US Digital)

The design of the roll-pitch-yaw assembly is quite simple; it is based on using U-Brackets as frames to hold shafts using bearings which allows for simple manufacturing since the parts are almost identical for each frame. These U-Brackets are also solid aluminum which creates a stronger structure, extremely necessary in the event of a crash, and considerably reduces the number of parts in the assembly. The bottom two frames are connected using a block, giving it a similar look and design to a universal joint. The bottom frame is attached to the vehicle at the upper frame plate shown in Figure 2.9 whereas the top frame is connected to the boom arm on the PURRS. The frames are also deigned such that they will work without the flex shafts if the encoders are upgraded in the future to ones with built-in flexible mounting plates such as the encoders used on the Magnetic Guidance Calibration Stand (MGCS).

The camera used for lane detection and the two MGS1600 magnetic sensors were installed on the vehicle and can be seen in Figures 2.16, 2.17, and 2.18 respectively. Their locations relative to the center of gravity of the vehicle as shown in Figure 2.15 and Table 2.1. The camera and rear magnetic sensors were installed using metal brackets and laser cut acrylic. The front magnetic sensor required a redesign of the bumper assembly on the front of the car. The new design moved the bumper further away from the front tires to allow room to install the front magnetic sensor. The new PURRS vehicle in its entirety can be seen in Figure 2.19.



Figure 2.15: Sketch showing sensor positions relative to the vehicle's center of gravity

Variable	Distance (in)
a	6
b	6
С	6
f	9.5
r	9.25

Table 2.1: Sensor positions relative to the vehicles center of gravity



Figure 2.16: Camera installed on the PURRS vehicle



Figure 2.17: Redesigned front bumper on the PURRS vehicle



Figure 2.18: Rear MGS1600 magnetic sensor installed on the PURRS vehicle



Figure 2.19: Fully assembled PURRS vehicle

2.4 Additional Changes to the PURRS

Mentioned earlier in this chapter, the PURRS features a boom arm with encoders that is used as a ground truth for the vehicles position. This boom arm is fixed to the PURRS frame and the roll, pitch, yaw encoder assembly. The method to read the encoders on the boom arm and the pose assembly was changed to an Arduino DUE, a more powerful version of the Arduino UNO. Other than the processor speed, the DUE nominally operates at 3.3V whereas the UNO operates at 5V, which is the suggested voltage applied to the encoders used on the PURRS. The encoders were operated at 3.3V and they worked as expected. The DUE is also larger and has more digital pins that are interrupt enabled, therefore a shield was designed to take seven encoders. The DUE is programmed using the same methods described in Section 3.1 for the encoders on the Magnetic Guidance Calibration Stand. In addition to changing the hardware used to read the encoders, the boom arm was also reconfigured to travel over the back of the car to avoid blocking the camera used for lane detection.

For this research, GPS is used for localization of the vehicle, but GPS is a sensor that requires line-of-sight typically does not work indoors. A simulated GPS system was created by hanging a camera, a Logitech c615 webcam, from the ceiling and using fiducal tracking to emulate a GPS signal. This technique is discussed further in Section 3.3 and the camera is shown in the ceiling in Figure 2.20.



Figure 2.20: Simulated GPS sensor

The GPS camera was hung in the ceiling roughly in the center of the PURRS to obtain

the least distorted images of the entire treadmill surface. The camera is mounted on piece of Unistrut, which is attached to the I-beams in the ceiling, to allow for variation in the mounting location of the camera. This allows the camera's position to be fine tuned to be directly in the center of the PURRS, as well as gives the option to move the camera if a different view is needed. Figure 2.21 shows what the simulated GPS camera sees when a picture is taken.



Figure 2.21: View of the simulated GPS sensor

The final change to the PURRS was the addition of lane lines. Discussed in Section 2.3 and Table ??, the vehicle selected for use on the PURRS is physically $1/8^{\text{th}}$ scale in size. According to the Pennsylvania Department of Transportation (PennDOT), lane lines range from 4 to 6 inches in width [80] and the Federal Highway Administration reports lanes are 9 to 12 feet across [81]. In scale size this results in lane lines being 1/2 to 3/4 of an inch in width and lanes being 13.5 to 18 inches across. 3/4 inch thick lane lines were chosen because it is the same width as standard electrical tape, which will reduces the cost of putting lane lines on the treadmill considerably. The lane lines were placed 16 inches apart because it is roughly the mean of the scale widths calculated. These lane lines can be seen in Figure 2.21.

Chapter

Sensor Characterization

A critical step before sensor fusion algorithms can be developed is to fully characterize each sensor. This is very specific to each sensor, therefore each process and its results will be outlined below.

3.1 Magnetic Sensor

Discussed in Section 1.2.2, magnetic guidance is a well studied method for localizing autonomous vehicles in both passenger highway and materials handling capacities. This technique has been used with various magnetic field producing objects such as: inductive wires, discrete permanent magnets, and magnetic strips. Magnetic strips have been chosen for this study because they provide a constant magnetic field for the sensor to measure and they have a low infrastructure cost compared to an inductive wire system.

RoboteQ MGS1600 magnetic sensors were acquired for this research and feature 16 individual sensors at 10 mm spacings. The MGS1600 is shown in Figure 3.1. This sensor was chosen because it is designed to read magnetic strips and is commercially available which reduces development time and troubleshooting compared to developing a sensor specifically for this research. This sensor can report magnetic strip location with 1 mm of accuracy using the factory programmed algorithm; however the raw sensor readings are used for this research to provide as much transparency as possible. To test this sensor for accuracy and initial algorithm development, a calibration stand was designed and built.

3.1.1 Magnetic Guidance Calibration Stand

The Magnetic Guidance Calibration Stand (MGCS) was designed to meet a very specific set of requirements:

• hold various magnetic sensors at a fixed angle relative to the magnetic field producing object



Figure 3.1: RoboteQ MGS1600 magnetic guidance sensor

- traverse the sensor laterally across the magnetic field at a fixed height
- easily change the vertical height of the sensor relative to the magnetic object
- accept any magnetic object(s) with little modification
- have no effect on the magnetic field

To meet these requirements, a pendulum was built with a pulley system to keep the sensor at a fixed angle. The pendulum was constructed with non-ferrous materials to avoid disturbing the magnetic field. An image of the pendulum is shown in Figure 3.2.



Figure 3.2: The Magnetic Guidance Calibration Stand (MGCS)

The Magnetic Guidance Calibration Stand (MGCS) features two high resolution encoders to accurately measure the position of the sensor on the end of the pendulum arm relative to a set zero point. Using two encoders is redundant; however, the second encoder allows for more robust position measurements. If the pendulum is subject to small slippage of components, the second encoder will be able to measure it to keep the pendulum as accurate as possible. The belt can also be removed to allow the sensor to move freely, which allows for more in-depth studies for magnetic or other sensors that are attached to the end of the pendulum. For the purposes of the study, the belt will be used to keep the sensor at a fixed angle relative to the magnetic object and the second encoder will be used to verify the sensor angle to ensure the validity of each test. The encoders, pulleys and belts can be seen in Figures 3.3 and 3.4.

Magnetic sensors are a relatively new field within our research group, therefore the MGCS was designed with a level of flexibility to allow for future research including different sensors, measurement ranges and magnetic field producing objects. The MGCS uses an adjustable framing system to allow for changes in size or the addition of new hardware. The magnetic sensor is attached to this adjustable framing to easily vary the height of the sensor from the magnetic objects. The pendulum can move relative to the wall because it is made with the adjustable framing system which allows the MGCS measurement range to be variable relative to the magnetic field. A wire management system is installed to keep data wires organized and to prevent interference with the magnetic field from the data wires which would occur if they were not properly shielded.



Figure 3.3: Upper half of the MGCS

One challenging aspect to the MGCS, and later in the rebuilding of the Pennsylvania State University Rolling Roadway Simulator (PURRS), is the need for a low-cost solution to read multiple encoders. One method used frequently is using counter and timer microchips, but the cost and number of microchips are proportional to the number of encoders. These costs make this method prohibitive when reading a large number of encoders; therefore, encoders were studied



Figure 3.4: Lower half of the MGCS

in depth to develop a lower cost and less complex solution.

Encoders use light shone through a disk to measure rotation; an example of an encoder disk is shown in Figure 3.5. These disks have slots that allow the light to shine onto a light sensor with the number of slots corresponding to the angular resolution of the encoder. Simple encoders will only have one ring of slots. These encoders are lower in cost but cannot differentiate between clockwise and counterclockwise rotation. Conversely, encoders with multiple rings of slots can differentiate between different directions of rotation but frequently have a higher cost. An example of an encoder disk with two rings of slots is shown in Figure 3.6, with the open slots shown in black.



Figure 3.5: Example of an encoder disk (Courtesy of US Digital)

When the encoder disk shown in Figure 3.6 is rotated clockwise, the two light sensors, labeled as "A" and "B", will emit similar signals to Figure 3.7. The unique encoder positions labeled as 1 through 4 are shown on both figures, and because the signals from the two internal light

sensors repeat every four encoder positions, encoders with slot configurations like the one shown in Figure 3.6 are called quadrature encoders. If the encoder disk is rotated counterclockwise, the signal will be the reverse of what is shown in Figure 3.7, which allows the encoder to differentiate between rotational directions. Quadrature encoders also give a higher resolution for a given number of slots which allows the encoders on the MGCS with 1250 slots per ring to have 5000 measurable positions and a resolution of about 0.07 degrees. For scale, the encoder disk shown in Figure 3.6 has 4 slots per ring and 12 measurable positions.



Figure 3.6: Example of a low resolution encoder with two rings of slots



Figure 3.7: Pattern seen from encoder in Figure 3.6 when rotated clockwise

The simplest method of reading an encoder is to constantly read the A and B channels to reproduce a plot like the one shown in Figure 3.7 which is computationally inefficient and slows the rate at which the encoder channels can be sampled. Efficiency can be increased by only sampling the encoder channels when it is known that the signal has changed, which can be done using interrupts. Interrupts are designed to "interrupt" the current software process when a certain event happens, with the event, in this case, being an encoder channel signal changing. Using interrupts makes the system computationally efficient because the sampling process is only used when needed, and the system will not miss encoder counts because the interrupt process takes precedence over any other process.

The MGCS and PURRS used the Arduino platform, which is very effective at using interrupts, to read encoder signals. The chipsets used on the various Arduino platforms have built-in interrupt functionality which makes interrupts a low level and efficient process. The Arduino UNO board, shown as Figure 3.8, is a low-cost board (about 30 USD) and is used for the MGCS to read the two encoders. The Arduino code used on the MGCS is given in Appendix A.



Figure 3.8: Arduino UNO board (Image courtesy of Arduino)

The MGCS was designed to run on a system called the Robot Operating System (ROS). ROS is a cross platform software system that allows for real time or pseudo-real time operation. For this research, it is run in Ubuntu, a Linux-based operating system, and is used for data acquisition for the MGCS. The MGCS has an Arduino with daughter board, shown in Figure 3.9, which reads the two encoder signals and processes them to give encoder counts. These counts are transmitted over a serial USB connection to the computer running ROS, which converts them to MGCS arm position data before recording them. The MGS1600 magnetic sensor also transmits its sixteen raw sensor readings over USB to the ROS computer. The ROS computer is capable of recording both sets of data simultaneously and time stamping each recording. The time stamp is extremely accurate and allows for matching of encoder and magnetometer readings, even if they are sampled at different rates. In this case the sensors do sample at different rates: the magnetic sensor at 100 Hz, the Arduino at about 200 Hz. The data flow and structure of the MGCS is shown in Figure 3.10. ROS saves all of the data files which are then parsed and finally processed using MATLAB.



Figure 3.9: MGCS Arduino and daughter board



Figure 3.10: Data flow block diagram for the MGCS

3.1.2 Magnetic Sensor Characterization

After the construction of the MGCS was completed, the next step was the characterization of the magnetic sensor. An example of the magnetic sensor data taken with the MGCS is shown in Figure 3.11. Each color is representative of one of the 16 sensors on the MGS1600, although not all of the sensors are shown. This figure is a plot of the raw magnetic reading versus the lateral distance from the magnet with the sensor 1 inch above the magnet. This figure provides some insight to magnetic strips and the magnetic sensor, particularly that the readings from the sensor are similar to previously reported results: Figure 3.12 shows the data from the MGCS has a similar shape to the results found by the Macome Corporation, but a different sign convention for the magnetic field. The example data also corresponds well with findings of passive, discrete magnets such as the field shape shown in Figure 3.13. Research in these types of magnets [82, 83, 84, 85] are still relevant to magnetic strips because strips take the same field shape as discrete magnetic markers placed in a row.



Figure 3.11: Example of magnetic sensor data from the MGCS using the MGS1600 sensor



Figure 3.12: Comparison of magnetic sensor data

Figure 3.11 also shows some important features of the magnetic field as well as some pitfalls associated with the magnetic sensor. The most important feature of the field shape is how parabolic the shape is immediately around the sensor. This was instrumental in choosing the algorithm used to estimate the position of the sensor relative to the magnet, which is discussed more in depth below. The pitfall of the magnetic sensor is how variant the readings are relative to each other, specifically far away from the magnet. The variance is shown in Figure 3.11. This shows that the zero value of each individual sensor is different and that any algorithm developed for this sensor needs to be insensitive to this variation. The sensor was re-calibrated using a sensor interface program provided by RoboteQ to reduce the variation between sensors, and another scan was taken, shown in Figure 3.14. In this figure, all 16 of the sensors are plotted



Figure 3.13: Discrete marker field shape shown in [82]

and shown in the same color and it can be seen that the readings far away from the sensors have less variation from sensor to sensor.



Figure 3.14: Measured magnetic field profile 1 inch from Magnetic Strip after re-calibration

The scans are shown in Figures 3.11 and 3.14 were instrumental in determining the estimation algorithm used with the MGS1600 magnetic sensor. Since the shape of the magnetic field is parabolic in the area immediately around the magnet, a thresholding and curve fitting algorithm using linear least-squares was chosen. The first step in the development of this algorithm was to create a visualization method for all of the data being recorded from the Magnetic Guidance Calibration Stand (MGCS). An example of how the data was visualized is shown below in Figure 3.15. In the first subplot, the 16 individual data points, one for each sensor on the MGS1600, are plotted versus the distance along the magnetic sensor in blue, with zero being the center of the sensor. Discussed in more depth below, the values that are below the threshold are shown as red circles and the parabola fit to those points is shown in green. The second subplot is a visualization of the pose of each part of the MGCS. The blue line is the adjustable framing arm to which both encoders are attached and the red line is the connection between the lower encoder and the magnetic sensor itself. The encoders are shown as black circles and the sensor is the green circle. The final subplot shows the performance of the estimation algorithm. When an estimated position is generated, it is plotted as a blue circle and the desired position is on the red line representing perfect estimation. The green lines are the x and y axes and are only used to aid in visualization of the data.

A thresholding technique is then applied to the data at each time step to eliminate an incorrect zero value on a sensor from affecting the algorithm. The technique is simply to ignore any individual sensor readings above the threshold value (since the peak reading from the magnetic field is a negative number) and to keep any values below the threshold. The threshold value was chosen to be a reading of -50 from an individual sensor because it is outside of the range of incorrect zero readings observed.

After the data is thresholded, a linear least-squares algorithm is used to fit the thresholded data point with a 2^{nd} order polynomial curve. The equation for a second order polynomial is given as:

$$y = P_1 x^2 + P_2 x + P_3 \tag{3.1}$$

The estimated position of the sensor is the minimum value of the polynomial, which is easily found by taking the derivative and setting it equal to zero. The derivative of Equation 3.1 is:

$$y = P_1 2x + P_2 \tag{3.2}$$

Setting y = 0 and solving for x gives

$$x = -P_2/(2P_1) \tag{3.3}$$

which corresponds to the minimum value of the polynomial. A curve will not fit all of the data points perfectly, therefore least-squares is used to determine the curve that best fits all of the data points below the threshold. Taking the data at a given time and stacking it to form a matrix gives The least-squares equation is shown as:

$$Xm = Y \tag{3.4}$$



Figure 3.15: Visualization of MGCS Data

where

$$X = \begin{bmatrix} x_1^2 & x_1 & 1 \\ x_2^2 & x_2 & 1 \\ \vdots & \vdots & \vdots \\ x_n^2 & x_n & 1 \end{bmatrix}$$
(3.5)

$$m = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \end{bmatrix}$$
(3.6)

$$Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$
(3.7)

and n is equal to the number of sensor values within the threshold. The variables x and y correspond to the individual sensor position relative to the center of the MGS1600 magnetic sensor and the individual sensor reading respectively. The values in the m vector correspond to the polynomial coefficients P_1 through P_3 in Equation 3.1. To solve for the 2nd order polynomial values, Equation 3.4 is rearranged to become

$$m = (X^T X)^{-1} X^T Y (3.8)$$

which correspond to the coefficients of the 2^{nd} order polynomial that best fits the data. These coefficients are then used in Equation 3.3 to find the estimated sensor position relative to the magnet.

A few assumptions were used to reduce incorrect estimates, such as the parabola defined by the least-squares fit would take a shape similar to the ones seen in Figures 3.11 and 3.14, implying that the point of zero slope of the curve fit occurs below the threshold value. The other assumption used was that the algorithm could not estimate a sensor position that was outside of the length of the sensor, which limits the algorithm to having at least one of the individual sensors above the magnet (a range of about three inches from the center of the sensor). Figure 3.16 shows the estimation algorithm performance without the 3 inch range restriction, and the estimation performance is significantly reduced at distances outside this range.

The magnetic sensor algorithm has drawbacks, namely that three sensors must have values within the threshold to perform the least-squares analysis. This is intuitive because three points are required to form a unique 2nd order polynomial. Another drawback to this algorithm is it is sensitive to the threshold value; if the thresholded value is set too close to zero, incorrect zero points can affect the estimation and if it is set too far from zero, the effective vertical measurement range of the sensor will be reduced.

This algorithm was tested with scans of the magnetic field taken at various heights. The



Figure 3.16: Estimation algorithm performance at 1 inch vertical height from magnet without estimation range restriction

performance of the estimation algorithm at a height of 1 inch above the magnetic strip is shown in Figure 3.17. This figure shares the same layout as the third subplot in Figure 3.15 in that the red and green lines correspond to perfect estimation and the axes respectively. Figure 3.17 shows that the estimation algorithm performs extremely well as most of the estimated position points, shown in blue, are near or on the red line. This is also seen in Figure 3.18 which shows the error between the estimated and true position for the same data set. The standard deviation and mean are shown in red and green respectively. The mean error is non-zero for this data set, which is most likely due to an imprecise zero value for the pendulum arm which adds a bias to the true position data.

As mentioned in Section 1.2.2, it is critically important to ensure that any sensing methods be deployable on commercial vehicles. The main concern for magnetic sensing technologies is the ability to reliably detect the magnetic field from a significant distance from the magnet. This is because, specifically in terms of an Automated Highway System application, the vehicles typically have a significant amount of suspension travel to provide ride comfort. If the magnetic sensor were mounted on the vehicle close to the ground, it could come in contact with the roadway when the suspension is compressed; therefore, the requirement for sensing distance exists for magnetic sensing technologies. Other applications, such as Automatic Guided Vehicles (AGVs), often do not have this constraint because the terrain that the vehicles traverse is often very flat and suspension systems are not typically used. As shown in Figure 3.12b, and shown larger in Figure 1.6, the strength and shape of the magnetic field are a function of the height of the sensor relative to the magnetic strip. To understand the limits of the magnetic sensor and the estimation algorithm, the magnetic field was scanned at various heights using the MGCS. The relationship



Figure 3.17: Estimation algorithm performance at 1 inch vertical height from magnet



Figure 3.18: Estimation algorithm error at 1 inch vertical height from magnet

between the standard deviation and vertical distance to the sensor are shown in Figure 3.19. The Pennsylvania State University Rolling Roadway Simulator (PURRS) features a 1/4" thick aluminum plate between the magnet and the roadway surface, discussed in Section 2.2, therefore the standard deviation at various heights with a 1/4" aluminum plate between the magnet and the sensor was also studied. The standard deviation up to 4 inches is shown in this figure because the estimation algorithm could not estimate the sensor position relative to the magnet at any height above 4 inches. This vertical measurement range can be expanded by moving the

threshold value closer to zero, but the algorithm could be affected by a drifted sensor zero value. Figure 3.19 shows the expected conclusion that the closer vertically to the magnet the sensor is, the better the estimation that can be made of the sensor position.



Figure 3.19: Estimation algorithm error at varying vertical height from magnet

Figure 3.19 also shows that the combination of the magnetic sensor, magnetic tape, and estimation algorithm are not suitable for an Automated Highway System environment. This is because the algorithm is constrained to the sensor within 4 inches vertically of the magnetic strip which is not enough to be practically and safely deployed on passenger vehicles, especially if the magnetic strip is embedded into the road surface. However, this technique can still be used with the PURRS, AGVs, and could be improved using a stronger magnetic strip, a more sensitive magnetic sensor, or an algorithm optimized for smaller magnetic signatures. This algorithm and equipment, although not immediately applicable to an AHS environment, will still be used for the rest of this research because the primary focus is to study sensor fusion techniques using magnetic sensing, not to develop a magnetic sensing system that is immediately deployable on an AHS vehicle.

3.2 In-Vehicle Camera

Mentioned in Section 1.2.3, the use of in-vehicle cameras to detect lane boundaries is a wellstudied field. Within this field, extensive research as gone into guidance on structured roads, or roads with consistent and often standardized road markings such as lane lines, which is the environment simulated in these studies. Fortunately, the simulated driving environment created on the PURRS is controlled to reduce the complexity often faced by lane detection algorithms; a vehicle on the PURRS is facing a wall that is painted flat black with relatively few features and



(a) Automatic exposure

(b) Manual exposure

Figure 3.20: Comparison of exposure settings for in-vehicle camera

road markings are close to perfect in terms of placement, width, and consistency. This controlled environment allows for non-complex algorithms that reduce computation loads and development time that would not be possible in a real-life driving environment.

The lane detection algorithm takes full advantage of the controlled environment of the PURRS to be as simple as possible. Each image is converted into the Hue, Saturation, Value (HSV) color space and thresholded such that only the yellow lane lines remain using the <u>Open</u> Source <u>Computer V</u> ision Library (OpenCV). After the image is thresholded, the image becomes a matrix with zeros in the pixel locations that have values outside the threshold, and a set value for pixels inside the threshold. The HSV color space is used because it is less sensitive to changes in light than the RGB (or BGR in OpenCV) color space. Unfortunately, the camera used, a Logitech c615 webcam, used digital exposure settings that automatically adjusted based on current lighting conditions. This caused the yellow lines in the camera images to appear similar in color to the background, reducing the thresholding algorithm's accuracy and robustness. Fortunately, a program was found to manually override the exposure settings can be seen in Figure 3.20.

The top half of the image is ignored and assumed to be at or above the horizon line; however, the horizon can be calculated by using the intersection of the detected lanes [86]. During testing, the detected lanes intersected at or near the middle of the image; therefore, ignoring the top half of the image was justified and creates a simpler and faster, in terms of computation time, algorithm.

One of the most difficult problems faced in lane detection is finding the edges of the lane in a robust manner. This is often accomplished with Sobel masks [57], Hough transforms [87, 55, 86], and Canny filtering [88, 89, 86]. Because the PURRS is a controlled environment with little background interference to the thresholded data, much simpler algorithms could be implemented than the ones that are typically used.

Two lane detection algorithms with the same basic structure were tested on the PURRS, both of which take advantage of the accurate thresholded data. These algorithms both assume that both lane lines can be seen in the image and that nothing else appears in the thresholded data. The first algorithm assumes that only one lane line will appear in each half of the image, if the image is split vertically. Each matrix value is then evaluated to determine if it is within the threshold, starting with middle of each row of pixels and moving outwards. This is then repeated for every row from the bottom of the image to the set horizon line. If the edges of the lane lines are found, the algorithm then fits lines to them using least-squares analysis. The center of the lane is then determined by averaging the location of the edges of the lanes and a line is fit to these values, also using least-squares analysis.

On the PURRS, the lane width is known, therefore a pixel-to-inch ratio can be found for each row of pixels. The combination of the pixel-to-inch conversion with the intersection of the estimated center line of the lane with the bottom of the image is used to estimate the vehicle's location relative to the lane. The second algorithm uses the same structure; however, instead of starting from the middle of the image, it starts from the sides of the image and works inwards. This detects the outside edges, preventing the failure mode seen by the other algorithm that occurs when a lane line crosses the middle of the image.

These algorithms have known flaws, which are:

- the vehicle must be close to and nearly parallel to the center of the lane
- the estimate of the pixel-to-inch conversion is dependent on camera calibration, number of lane edge points found, and lane width
- the algorithm assumes straight, continuous lane lines
- the algorithm performance is not robust to poorly recognized lane features

Both algorithms produce similar visual interfaces, mainly for troubleshooting during testing. Figure 3.21 shows the various features used in the in-vehicle camera algorithm, including the horizon line, the lane lines found by thresholding, the detected edges of those lines, and the estimated center line. Figure 3.22 shows the difference between the two estimation algorithms, namely the difference in the position of the detected lane line edges.

These algorithms were tested on the PURRS, and their results are discussed in Section 3.4.

3.3 Simulated GPS

Discussed in Section 1.2.4, Global Positioning Systems (GPS) are a relatively accurate method for localization that requires line of sight for proper operation. Because line of sight cannot be achieved on the Pennsylvania State University Rolling Roadway Simulator (PURRS) due to the system being indoors, a simulated GPS environment was created and used. These systems can use either multiple beacons broadcasting position data, effectively a small-scale version of GPS, or an overhead camera to extrapolate the position of the vehicle, which is the method used for this research.

The algorithm used to simulate GPS is a color detection algorithm within ROS, using the OpenCV library. These algorithms require a colored fiducial to track and are often affected by



Figure 3.21: In-vehicle camera algorithm interface showing various features



Figure 3.22: Comparison of in-vehicle camera algorithms

background coloring that is similar to the fiducial in color. The PURRS provides an excellent background for this algorithm because most of the background will be black, gold, and brown; the colors of the PURRS, the motor used on the PURRS, and the floor respectively. Other colors are present however, such as the red printed circuit board Arduino shield on the vehicle and the yellow AC motor wires. The simulated GPS algorithm requires two fiducial colors, one for the vehicle and the other for the PURRS frame. The algorithm uses the HSV color space and two thresholds, one for each fiducial color, to eliminate all colors except for the fiducial colors. Once the other colors are removed, all of the remaining pixels (each distinct cluster of pixels is often called a blob) are recorded and their areas are calculated. The vehicle fiducial algorithm only keeps the blob with the largest area, whereas the frame fiducial algorithm keeps the four largest areas, as there are four fiducials on the frame. The centroid of each blob is then calculated using an OpenCV algorithm and their location, in pixels, is given. The fiducials for the simulated GPS algorithm can be seen in Figure 3.23.



Figure 3.23: View of GPS camera showing fiducials and PURRS components

The four fiducials on the PURRS frame, although not necessary for the GPS algorithm, significantly improve the pixel-to-inch calculation required to localize the vehicle fiducial. The frame algorithm takes the four blobs with the largest area and assumes that they are the corners of the frame. The pixel location of each blob is evaluated to determine which fiducial corresponds to which corner of the frame. This information is used to calculate the pixel distance between the four blob centroids, and since the physical distance is known between the fiducials on the frame, the conversion from pixel to physical distance measurement can be calculated. This could be done without the frame fiducials by hand calibration, but using them makes the GPS algorithm insensitive to poor camera calibrations and would not require recalculation of the pixel to physical distance conversion if the PURRS frame moves. Once the pixel to physical distance conversion is found, it is applied to the vehicle fiducial to calculate the location of the vehicle. This measurement is then reported for use in the various filtering techniques described in Chapter 4. The performance of this algorithm, in both the lateral and lognitudinal directions, is discussed in Section 3.4.

3.4 Individual Sensor Testing on the PURRS

Each sensor was tested on the PURRS in order to verify results from previous testing of the magnetic sensor and establish performance of each sensor. All of the sensors were mounted in their proper place during this testing, as described by Chapter 2 to ensure consistency between sensor characterization and final algorithm testing. The vehicle was moved by hand over the treadmill deck surface slowly (around 1/2 inch per second) from one side of the PURRS to the other, both in the lateral and longitudinal directions. The magnetic sensors, the in-vehicle camera and the lateral component of the GPS were characterized using the data from moving the car laterally, while the lognitudinal component of the GPS was characterized when the car was moved in the lognitudinal direction. The encoder arm was calibrated and used as a ground truth measurement to define the error of the estimated measurements.

The sensor measurements are shown in Figure 3.24. This data is used for all of the subsequent analysis in the remainder of this section. In this figure, a few characteristics of the algorithms



Figure 3.24: Lateral estimation data from all sensors on the PURRS

become obvious; namely the in-vehicle camera algorithm that detects the outside of the lane makers does not perform as well as the other in-vehicle camera algorithm. The second characteristic is that majority of the sensors only work within a short range of the center of the line, with the GPS sensor being an exception. This is expected, because the magnetic sensors were only found to have a lateral range of 3 inches from the center of the sensor, and the lane detection algorithm needs both lanes in view to estimate the vehicle's position. Figure 3.24 was adjusted to show the range of the sensors more effectively and is shown as Figure 3.25. This figure shows that the effective range of the in-vehicle camera algorithm that will be used (detecting the inside of the lanes) is ± 8 inches, while the magnetic sensors have a range of about ± 3 inches, which is the same range found on the MGCS. Figure 3.25 also confirms the in-vehicle camera algorithm using the inside of the lane lines out-performs the algorithm using the outside of the lane lines.



Figure 3.25: Lateral estimation data from all sensors on the PURRS (cropped)

In each test the error, calculated by

$$e = x_{est} - x_{truth} \tag{3.9}$$

was evaluated over the whole sensing range to determine the mean. The true reference is obtained using the encoder-equipped boom arm on the PURRS.

When calculating the estimation algorithm performance, constant biases and offsets were prevalent in the data sets. These biases are most likely due to imprecision in: the origin value of the true position of the vehicle, the placement of the magnetic strip, etc. These biases are remedied by using a least-squares curve fit to compensate for the errors. This is very similar to the curve fit described in Section 3.1, but a first order fit is used, described by

$$y_{meas} = mx_{est} + b \tag{3.10}$$

where y_{est} and x_{true} are the estimated and true positions of the sensor respectively. Ideally, the relationship should be

$$y_{ideal} = x_{est} \tag{3.11}$$

which represents perfect estimation of the true position of the sensor. In order to correct for the biases, Equation 3.11 is substituted in to Equation 3.10

$$y_{meas} = my_{ideal} + b \tag{3.12}$$

and solving for y_{ideal} yields:

$$y_{ideal} = \frac{y_{meas} - b}{m} \tag{3.13}$$

which is used to correct the estimated data to give it a slope of one and an offset (y intercept) of zero. The m and b values for each sensor were found, using the data shown in Figures 3.24 and 3.25, and are listed in Table 3.1.

Sensor	Slope Correction (m) []	Intercept Correction (b) [in]	
GPS Lateral	1.0742	0.5874	
Front Magnetic Sensor	0.9479	0.2884	
Rear Magnetic Sensor	0.9589	0.0271	
In-Vehicle Camera	0.9589	0.6786	

Table 3.1: Estimation data correction factors

3.4.1 Magnetic Sensors

During the testing of the PURRS, the estimation algorithm was retested to ensure consistency between the MGCS and the PURRS. The purpose of this test was to verify that differences between the systems, particularly the difference in height of the sensors relative to the magnet and the different MGS1600 sensors used on the PURRS, did not effect the accuracy of the curve fitting algorithm. There are two MGS1600 magnetic sensor on the PURRS vehicle, one mounted to the front of the vehicle and another mounted to the rear of the vehicle. The position and specifics for how the sensors are attached are described in Section 2.3. The performance of the magnetic sensors is shown in Figure 3.26 and Figure 3.27. In these figures, the estimation algorithm does not appear to perform as well as it did when used on the MGCS (shown in Figure 3.17). This is most likely due to a difference in the magnetic sensor height from the magnetic strip. Even with this difference, the standard deviation was found to be 0.58 inches and 0.47 inches for the front and rear magnetic sensors respectively. The difference between these two standard deviations is also due to the sensor height, because the housing that holds the front sensor is intentionally higher than the rear to prevent the front of the vehicle from touching the ground due to the increased weight on the front suspension. It is important to note that the range for the magnetic sensors is intentionally restricted to ± 2.5 inches in the lateral direction because the error becomes too large outside this range.

3.4.2 In-Vehicle Camera

It was extremely important to characterize the in-vehicle camera because, in contrast to the magnetic guidance sensors, the algorithms were not tested in any other manner before. Mentioned in



Figure 3.26: Estimation algorithm performance of the front magnetic sensor on the PURRS



Figure 3.27: Estimation algorithm performance of the rear magnetic sensor on the PURRS

the beginning of this section, one of the major goals was to determine the better of the two algorithms developed for the camera, one detecting the outside of the lanes, and the other detecting the inside, and to decide which would be used for later development. From Figures 3.24 and 3.25, it became obvious that the inside detection algorithm provided better estimation results, therefore it will be used for the remainder of this research. These two algorithms were tested on the new PURRS deck surface with two 3/4" wide lane lines spaced at 16 inches apart, as discussed in Section 2.4, and the results are shown in Figures 3.28 and 3.29. Using these figures, a range



Figure 3.28: Estimation performance of the in-vehicle camera algorithm detecting inside lane edges



Figure 3.29: Estimation performance of the in-vehicle camera algorithm detecting outside lane edges

of ± 6 inches was determined to be optimal for these sensors and a standard deviation of 0.33 inches was observed by the algorithm tracking the inside edge of the lane lines.

3.4.3 Simulated GPS

The simulated GPS sensor, like the in-vehicle camera, was not tested in any other manner before it was tested on the PURRS, and it provides both a lateral and lognitudinal position estimate. Figure 3.30 shows the lateral estimation accuracy of the simulated GPS algorithm.



Figure 3.30: Estimation performance of the GPS algorithm in the lateral direction

Figure 3.30 shows extremely accurate estimation by the simulated GPS algorithm which resulted in a standard deviation of the error of about 0.12 inches. This level of accuracy is consistent over the whole lateral range of the PURRS, which is roughly ± 30 inches. The GPS algorithm was also tested in the lognitudinal direction, and the results are shown in Figure 3.31. In this figure, the GPS algorithm also estimates the position of the vehicle with very high accuracy and the error had a standard deviation of 0.19 inches.

In summary, the variances and ranges for each sensor found during testing are shown in Table 3.2.

Sensor	Scaled Variance (σ^2) [in ²]	Variance (σ^2) [ft ²]	Range [in]
GPS (Lognitudinal)	0.0339	2.1696	± 60
GPS (Lateral)	0.0141	0.9024	± 30
Front Magnetic Sensor	0.3311	0.3311	± 2.5
Rear Magnetic Sensor	0.2214	0.2214	± 2.5
In-Vehicle Camera	0.1067	6.8288	± 6

Table 3.2: PURRS sensor variances and ranges

In this table, the scaled variances, which were observed during testing, and the equivalent variance for a full size vehicle. It is important to note that the magnetic sensor does not scale with vehicle size, unless a bigger sensor is used in addition to a larger magnetic strip.



Figure 3.31: Estimation performance of the GPS algorithm in the lognitudinal direction

Chapter

Estimation Algorithm Development

One of the difficult problems faced when using multiple sensors to measure the same quantity is to determine the most effective and efficient way to combine these measurements into a single estimate. The major trade-offs when deciding which data-fusion technique is best for a particular system are computation time versus the effective use of each measurement, and its associated statistics. In the following sections, data fusion techniques of varying complexity are examined to determine which will be used on the PURRS to determine the lateral position of the vehicle.

4.1 Averaging Techniques

The simplest technique possible for sensor fusion is a simple average of the measurement given from each sensor at a given time. This is defined as:

$$x_{est} = \frac{1}{N} \sum_{i=1}^{N} x_{i,meas} \tag{4.1}$$

where N is the number of sensor measurements, which would be four in the case of the PURRS. This method is extremely simple to implement, especially if the number of measurements is constant and known, but this sensor fusion algorithm is extremely poor at estimating system states in noisy systems. This algorithm also ignores the variance of each measurement, which means that sensors with more precise measurements are weighted equally in the average as measurements with low precision. Therefore all sensor readings are trusted equally, regardless of whether one sensor is better than another. This can be remedied by using a weighted average:

$$x_{est} = \left(\sum_{i=1}^{N} \frac{1}{\sigma_i^2}\right)^{-1} \sum_{i=1}^{N} \frac{x_{i,meas}}{\sigma_i^2}$$
(4.2)

where N is the number of measurements and σ_i^2 is the variance associated with the measurement $x_{i,meas}$. This technique is very similar computationally to an non-weighted average, especially

if the variance of each sensor is known. This technique uses knowledge of the variance of each sensor to give different weights to each value in the average, with the sensor with the lowest variance (highest precision) gaining the most weight.

In addition to the two averaging techniques, a median filter was used, which simply uses the median of the sensor readings at a given time and reports this value as the estimate. The median filter is a simple filter to implement and it is adept at removing outliers; however, it does not use the precision of the sensors when determining the estimate. This technique, as well as the two averaging sensor-fusion algorithms, does not benefit from a system model, which helps to constrict the estimate to the dynamics of the system to reduce noise and improve the estimation. Kalman filters do include system models, and are often used in sensor fusion problems with associated dynamic systems.

4.2Kalman Filter

The Kalman filter, originally introduced by Rudolph Kálmán in 1960 [65], is the cornerstone of the sensor fusion algorithm used in this research. The Kalman filter uses the state space approach to combine a system model with noisy measurements to estimate the states of system. In this work, a discrete implementation is assumed, with a sample rate equal to the true measurment of the vehicle's position. This is given by the encoders and boom arm geometry and is equal to 200 Hz. The measurements that are aquired at lower frequencies are converted to the higher frequency using a zero-order hold.

The following symbols will be used to introduce the Kalman filter: . . .

. .

A_k	=	state matrix at time k
B_k	=	input matrix at time k
C_k	=	output matrix at time k
D_k	=	feedforward matrix at time k
Q_k	=	system error matrix at time k
R_k	=	measurement error matrix at time k
$\hat{x}_{k k-1}$	_1=	state estimate at time k given information up to $k-1$
$P_{k k}$	-1=	covariance of the state estimate $x_{k k-1}$
z_k	=	innovations at time k
S_k	=	covariance of the innovations z_k
K_k	=	optimal Kalman gain

The Kalman filter equations assume a system of the following form:

$$x_k = A_k x_{k-1} + B_k u_{k-1} + w_k \tag{4.3}$$

$$y_k = C_k x_k + D_k u_k + v_k \tag{4.4}$$

where w_k and v_k are zero-mean, white, Gaussian noises with covariance Q_k and R_k respectively. The Kalman filter is often separated into two steps, the prediction and update steps. The prediction step uses the state estimate and covariance calculated from the previous time to predict what the state and covariance will be at the current time. The state estimate is calculated by

$$\hat{x}_{k|k-1} = A_k \hat{x}_{k-1|k-1} + B_k u_{k-1} \tag{4.5}$$

where $\hat{x}_{k|k-1}$ is the predicted state estimate at time k given information up to k-1. The covariance of $\hat{x}_{k|k-1}$ is shown as

$$P_{k|k-1} = A_k P_{k-1|k-1} A_k^T + Q_k \tag{4.6}$$

These values are then compared to the actual measurement to determine the innovations with associated covariance in the update step.

$$z_k = y_k - C_k \hat{x}_{k|k-1} \tag{4.7}$$

The innovations, z_k , are an extremely important metric because their statistical properties, such as covariance, whiteness, and mean, are used to detect sensor failures. The covariance of the innovations term is defined as

$$S_k = C_k P_{k|k-1} C_k^T + R_k \tag{4.8}$$

The optimal Kalman gain, the new state estimate and covariance are calculated using the innovations, also during the update step. The Kalman gain is shown as

$$K_k = P_{k|k-1} C_k S_k^- 1 (4.9)$$

whereas the new state estimate and associated covariance are defined as

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k z_k \tag{4.10}$$

$$P_{k|k} = (I - K_k C_k) P_{k|k-1} \tag{4.11}$$

It is important to note that both the state estimate and its associated covariance are used during the calculations performed in the prediction step at the next time step. This relationship is shown in Figure 4.1 by the curved black arrows. The values calculated during the prediction and update steps are shown in green and red respectively.

The vehicle dynamics required for the Kalman filter can be approximated in many different ways, with the bicycle model being very common [1, 13, 50, 90, 91]. Because initial testing involved moving the vehicle by hand, only a kinematic Kalman filter is considered. Therefore the dynamic system used to represent the lateral position of the vehicle is given by

$$x_k = x_{k-1} + w_k \tag{4.12}$$



Figure 4.1: Relationship between different values calculated and used by the Kalman filter

which means that the position of the vehicle at k is the position at the previous time, k-1, with an uncertainty equal to the covariance of w_k , which will be denoted as σ^2_{model} .

For the system being tested in this research, the following are used for the Kalman Filter:

$$A_{k} = A = 1 \quad B_{k} = B = 0 \quad C_{k} = C = \begin{vmatrix} 1 \\ 1 \\ 1 \\ 1 \end{vmatrix} \qquad D_{k} = D = 0 \tag{4.13}$$

$$Q_{k} = Q = \sigma_{model}^{2} \qquad R_{k} = R = \begin{bmatrix} \sigma_{gps}^{2} & 0 & 0 & 0 \\ 0 & \sigma_{mag,f}^{2} & 0 & 0 \\ 0 & 0 & \sigma_{mag,r}^{2} & 0 \\ 0 & 0 & 0 & \sigma_{cam}^{2} \end{bmatrix}$$
(4.14)

where all of the matrices are considered time-invariant for the basic Kalman filter; however, this assumption will be relaxed later to make the Kalman filter more robust, and will be discussed in Chapter 5. The R_k matrix is populated on its main diagonal with the covariances of the simulated GPS, front and rear magnetic sensors, and the in-vehicle camera sensor readings. The cross-correlation of the noise is assumed to be zero, therefore the remainder of the matrix is sparce. This assumption is made because all of the sensors are separated from each other and use shielded cabling; however, this assumption might become invalid when the car is moving due to magnetic fields that may be present only during motion; for example, due to high currents in
electric drive motors.

4.3 Comparison of Sensor Algorithms

Each estimation algorithm was implemented and tested on the PURRS to form a comparison of the accuracy of each algorithm. The PURRS vehicle was moved, by hand, back and forth within the sensing range of all of the sensors, which is limited to roughly ± 2.5 inches, due to the magnetic sensors. The true lateral position of the center of gravity of the vehicle is shown in Figure 4.2.



Figure 4.2: Vehicle center of gravity location measured by the PUURS kinematic boom arm used for estimation algorithm testing

The estimated positions and error relative to the ground truth, obtained using the boom arm on the PURRS, are shown in Figure 4.3 and Figure 4.4.

Figures 4.3 and 4.4 show that all of the sensor fusion algorithms perform well at estimating the position of the vehicle. This is to be expected, because the data acquired from the sensors has a low variance, shown in Table 4.1. These values were obtained during testing described in Chapter 3 and was initially reported as Table 3.2, however it is reiterated here for convenience.

To further test the accuracy of these algorithms, the GPS sensor measurements were corrupted with a zero-mean, Gaussian noise with a standard deviation of 3 1/3 feet (a variance of 16 2/3 feet²) in terms of a full size vehicle, which corresponds to 5 inches on the scale vehicle. Using the corrupted GPS position estimates, Figures 4.5 and 4.6 show the estimation accuracy of the sensor fusion algorithms. Please note that the averaging technique experienced much higher errors in estimation than the other algorithms, therefore is not shown on the plot to show the remaining algorithms performance in more detail.



Figure 4.3: Comparison of sensor fusion algorithms



Figure 4.4: Comparison of sensor fusion algorithms' error relative to ground truth

Figures 4.5 and 4.6 show that the data fusion algorithms, excluding the averaging algorithm, still perform well given the corrupted GPS data. Figure 4.6 shows that the weighted average and Kalman filtering algorithms performed better than the median filter and the Kalman filter and weighted average performed identically. This is due to the kinematic model chosen for the Kalman filter. The kinematic Kalman filter model utilized a large value for model uncertainty, $\sigma^2_{model} = 10$ inches², to allow the Kalman filter to measure the vehicle's changing lateral position. This uncertainty is much greater than the sensor uncertainties, which means the model was used

Sensor	Scaled Variance (σ^2) [in ²]	Variance (σ^2) [ft ²]	Range [in]
GPS (Lognitudinal)	0.0339	2.1696	± 60
GPS (Lateral)	0.0141	0.9024	± 30
Front Magnetic Sensor	0.3311	0.3311	± 2.5
Rear Magnetic Sensor	0.2214	0.2214	± 2.5
In-Vehicle Camera	0.1067	6.8288	± 6

Table 4.1: PURRS sensor variances and ranges before GPS corruption



Figure 4.5: Comparison of sensor fusion algorithms with corrupted GPS measurements

significantly less than the sensor measurements by the Kalman filter when calculating the estimate of the vehicle's position. In addition to the low contribution of the model to the Kalman filter estimate, the sensor measurements are combined in the filter using a weighted average, which explains the similar performance of the Kalman filter and the least-squares algorithm. The mean and variance of the error between the estimated and true position of the vehicle are given in Table 4.2.

Algorithm	Variance $(\sigma^2)[in^2]$	Mean [in]
Weighted Average	0.2000	0.0023
Median	2.4992	-0.0444
Kalman	0.2000	0.0023
Average	89.1936	0.0076

Table 4.2: Estimation algorithm error statistics with corrupted GPS measurement

The identical performance of the Kalman filter and the least squares algorithm makes it difficult to justify using the more complicated Kalman filter as the base of the fault detection



Figure 4.6: Comparison of sensor fusion algorithms' error with corrupted GPS measurements

algorithm. It is important to keep in mind that the Kalman filter model used for this stage of testing has reduced the benefit of the filter significantly, by nearly eliminating the model's effect on the filter by assuming it has a high covariance noise or uncertainty. This was done to allow the Kalman filter to track the car in the lane when it was being pushed by hand. If the vehicle was moving under its own power, the model uncertainty could be significantly reduced, therefore giving the Kalman filter more information. This information, assuming the model and its uncertainty are accurate, would give the Kalman filter a better estimation performance over the least squares method. Due to the fact that the fault detection algorithm is intended to be used with a vehicle moving under its own power, the fault detection algorithm will be developed using a Kalman filter during this research to give it instant applicability to a self-powered vehicle.

Chapter

Fault Detection and Reduction Algorithm Development

5.1 Fault Detection

As mentioned in Section 1.2.5, various Kalman filter methods have been developed for the detection of faults for sensors. These methods use the innovations calculated by the Kalman filter and compare them to the typical values seen by a Kalman filter implemented on a system without faults; the innovations are typically zero mean, white, and have a consistent covariance [66]. If the innovations deviate from the expected statistics, then a fault has most likely occurred and the Kalman filter estimate is no longer valid.

Testing for the statistics of the innovations term is made easier using the standardized innovation sequence, which is calculated by

$$\eta_k = S_k^{-1/2} z_k \tag{5.1}$$

or, by substituting for S_k :

$$\eta_k = (C_k P_{k|k-1} C_k^T + R_k)^{-1/2} z_k \tag{5.2}$$

The mean of the standardized innovation, $\hat{\bar{\eta}}$ is given by the following equation:

$$\hat{\eta} = \frac{1}{N} \sum_{i=1}^{N} \eta_i \tag{5.3}$$

where N samples are used to define the statistic, where N is proportional to the sensitivity of the statistic. This means that the lower the value of N, the more each measurement affects the statistic and the faster a faulty sensor will be detected. But a low N value will also increase the chance of detecting a fault incorrectly.

The whiteness, defined as \hat{c}_j , is calculated using the following equation:

$$\hat{c}_{j} = \frac{1}{N} \sum_{i=j}^{N} (\eta_{i} - \hat{\eta}) (\eta_{i-j} - \hat{\eta})^{T}$$
(5.4)

where j is the lag used in the autocorrelation function and N is proportional to the sensitivity, similar to the calculation of the mean. The final statistic is the variance, \hat{c}_o , which is compared to a predetermined variance, determined through testing with a system not experiencing faults:

$$\hat{c}_o = \frac{1}{N} \sum_{i=1}^{N} (\eta_i - \hat{\bar{\eta}}) (\eta_i - \hat{\bar{\eta}})^T$$
(5.5)

These techniques are used to detect faults, but they do not correct the estimate when a fault is detected. This correction step is often done by using a "bank" of Kalman filters, which is a series of Kalman filters where each individual Kalman filter uses a different combination of sensors found on the system. Because the PURRS vehicle is equipped with four sensors, there are 15 different combinations of sensors, representing the full factorial combination of sensor possibilities. Therefore, 15 Kalman filters are used and are all run simultaneously to allow the quickest transitions between filters. The sensor combinations and their associated Kalman filters are outlined in Table 5.1.

Filter Number	# of Sensors	GPS	Front Mag.	Rear Mag.	Camera
1	4	Yes	Yes	Yes	Yes
2	3	Yes	Yes	Yes	No
3	3	Yes	Yes	No	Yes
4	3	Yes	No	Yes	Yes
5	3	No	Yes	Yes	Yes
6	2	Yes	Yes	No	No
7	2	Yes	No	Yes	No
8	2	Yes	No	No	Yes
9	2	No	Yes	Yes	No
10	2	No	Yes	No	Yes
11	2	No	No	Yes	Yes
12	1	Yes	No	No	No
13	1	No	Yes	No	No
14	1	No	No	Yes	No
15	1	No	No	No	Yes

Table 5.1: Kalman filter combinations with associated sensors

If a fault is detected, by using the statistics of the innovations, then the innovations of the remaining Kalman filters in the bank are checked for faults. The Kalman filter not in a fault condition that uses the highest number of sensors is then used. Any sensor that is not used by this filter is assumed to be the faulty sensor. This method is a very simple, top-down search

algorithm that assumes that a non-faulted filter will be chosen immediately after the fault occurs, and the filter with the most sensors provides the best estimate. This assumption leads to a simple search algorithm that performs quite well, as seen later in Figure 5.1; however, other search algorithms, such as a bottom-up filter that reverts to a single sensor and builds up, could be easily applied to the fault detection algorithm. The combination of the innovation statistics and Kalman filter bank create the fault detection algorithm used in this research.

To test this idea, a fault was introduced into the same data used in Chapter 4 to determine if the fault detection algorithm could recognize the fault and switch to a non-faulty Kalman filter. The induced fault was a complete failure of the front magnetic sensor by having it report a constant value after a set time. The output of the 1st Kalman filter and the fault reduction algorithm utilizing the bank of Kalman filters is shown in Figure 5.1.



Figure 5.1: Performance of estimation algorithms against fault of front magnetic sensor using maximum value of ${\cal N}$

In this figure, the fault detection algorithm is shown as "Adaptive Kalman" filter, and although it momentarily performed worse than first Kalman filter, which utilizes all four sensors, immediately after the fault occurred, it regained desired estimation performance after about 5 seconds. This delay is a function of the statistic sensitivity defined by N, which in this case was equal to the total number of samples taken up to the current measurement, roughly 5000 when the failure begins. The innovations calculated by the first Kalman filter are shown in Figure 5.2, which show how the mean values of the innovations change after the fault occurs; before the fault, the innovations have small variance and zero mean, but after the fault the mean and variance change significantly. The effect of reducing the value of N to 1000 is shown in Figure 5.3. When the N value is reduced to this level, the sensitivity of the algorithm to change in the innovations is greater, causing the algorithm to detect the failure quickly. However, this sensitivity also causes the algorithm to switch between filters in the Kalman filter bank rapidly and often chooses a filter that is affected by the fault, thus reducing performance.



Figure 5.2: Innovations from first Kalman filter under fault of front magnetic sensor

5.2 Position-Referenced Fault Mitigation

In addition to a fault detection algorithm, a new fault mitigation algorithm has been implemented successfully. The driving idea behind this algorithm is to use known information about a vehicle's environment to help improve or retain the accuracy of the estimation algorithm. This would occur by having the vehicle use a map or some other form to communicate information about a road to the estimation algorithm, based on the vehicle's current position. An example of where this is beneficial would be to warn the estimation algorithm of an upcoming tunnel, which often cause GPS sensors to lose signal, so that it knows to either reduce the weight given to the GPS measurement in the Kalman filter, or to switch temporarily to a filter that does not include the GPS measurement at all.

In addition to the GPS sensor, the in-vehicle camera algorithm is also sensitive to environmental conditions such as: quality/existence of lane lines, environmental conditions, and lighting conditions, which could be quantified and given to the estimation algorithm with varying levels of difficulty and complexity. This information could be something as simple as: "the lane lines are, on average, x inches wide and y inches apart on this road due to the standards set out by



Figure 5.3: Performance of estimation algorithms against fault of front magnetic sensor with an N value of 1000

the transportation authority" to something as complex as: "the lane lines from position x to position y are degraded or non-existent" or "a variance of z for the in-vehicle camera algorithm has been shown to give optimal performance on this stretch of road".

In order to test this fault reduction algorithm, the same bank of Kalman filters is used from the fault detection algorithm, as well as the same data set. However, instead of causing a total failure in a sensor, an increase in variance was introduced to simulate a position-referenced expectation that one sensor may contain more noise than normal. In a real-life implementation of this algorithm, this information could be conveyed by an on-board or actively broadcast map containing information about possible sensor faults and their physical locations on a roadway. In this research, a sensor fault, again in the front magnetic sensor, was introduced during a certain interval by increasing the covariance of the sensor measurements. The fault mitigation algorithm was then told to increase the covariance associated with the front magnetic sensor during the same interval used when corrupting the measurements. The results of this test are shown in Figure 5.4 and Figure 5.5.

This figure shows that the fault reduction algorithm maintained tracking performance during the fault experienced by the front magnetic sensor. In addition to adding a zero mean noise, a non-zero mean Gaussian noise was added to the signal over the same failure period. The tracking performance of the fault reduction algorithm, shown in Figure 5.6, are nearly identical to the ones shown in Figure 5.4.

The fault mitigation algorithm developed in this thesis has several advantages over a Kalman



Figure 5.4: Performance of fault reduction algorithms against a known fault of the front magnetic sensor (zero mean noise)



Figure 5.5: Fault reduction algorithm error against a known fault of the front magnetic sensor (zero mean noise)



Figure 5.6: Performance of fault reduction algorithms against a known fault of the front magnetic sensor (non-zero mean noise)



Figure 5.7: Fault reduction algorithm error against a known fault of the front magnetic sensor (non-zero mean noise)

filter not using any of these techniques. The primary advantage is the reduction of a fault's impact on the state estimate, if the fault is known to exist. This information can come via a map or broadcast to the vehicles on a road, and would most likely be a function of the longitudinal position of the vehicle on the road; however, this research related the fault occurrence at certain instances in time because the fault was introduced in post processing.

Another advantage of this fault mitigation algorithm is that it can be used liberally in a safe manner; for instance, if the position of the start and end of a tunnel is uncertain, the map can assume the fault is over a range that safely accounts for this uncertainty, without seriously affecting the estimation of the vehicle's position. This does come with a major assumption, which also applies to the algorithm as a whole, which is that the vehicle's position must be known a-priori. These are analogous because the bank of Kalman filters represents the worst case scenarios experienced by the fault mitigation algorithm, specifically the known total failure of one or more of the sensors.

Chapter

Conclusions and Future Work

6.1 Conclusions

Conclusions drawn from this work are organized in subsections below

6.1.1 Sensors

One of the most important contributions from this work is the use of magnetic sensors for vehicle estimation, an unstudied sensor application field for the Intelligent Vehicles and Systems Group. This work has shown the benefits for magnetic guidance, and has shown that although permanent magnet strips are concluded to be the most viable option, they do not provide enough range with the sensor used to be practical for use with a full size vehicle. Although magnetic strips provide enough sensing range to be used with the scale vehicle, another magnetic sensing technology with a larger effective range, such as inductive guidance, should be used with full-sized vehicles.

This research has also created some very simple algorithms for indoor simulated GPS using a camera hung from the ceiling, and for lane detection using thresholding and blob tracking. Both of these algorithms could be further developed and refined to create greater accuracy or could be applied to other systems facing similar challenges.

6.1.2 Hardware Development

Another important contribution provided by this research is the development of multiple hardware platforms. The first platform, the Pennsylvania State University Rolling Roadway Simulator, was redesigned to simplify the design and to add an air-bearing to reduce friction. Not only was this a very unique engineering exercise covering multiple disciplines, a combination of fluid dynamics and mechanical design, but it also provided the capability for the PURRS to be used for high speed testing. In addition, sensor capabilities were added to the PURRS such as the hanging of a camera to simulate GPS, lane lines were added for lane detection, and magnetic strips and inductive wires were added for magneitc guidance.

The other sensing platform developed was the Magnetic Guidance Calibration Stand. This apparatus is a pendulum which gives the capability to calibrate magnetic sensors accurately by varying the horizontal distance from the magnet, with the ability to vary the height of the sensor from the magnet. This calibration stand provided critical information about the magnetic strips used in this research, particularly that the magnetic strips do not produce a strong enough magnetic field for use with a full size vehicle. This stand also provides an easy way to test magnetic sensor algorithms without having to use the PURRS.

6.1.3 Algorithm Development

The main focus of this research was the development of algorithms to reduce the effect of known and unknown faults on the lateral estimation algorithm. The unknown faults were handled by using a bank of Kalman filters comprised of every combination of Kalman filters that can be formed with a some or all of the sensors on the vehicle. This algorithm effectively uses the best Kalman filter estimate by looking at the statistics of the innovations term produced by each Kalman filter. This algorithm, although successful at reducing the effect of the fault, requires a large number of computations, which increases significantly with the number of sensors on the vehicle.

The known fault mitigation algorithm uses a more simple algorithm compared to the unknown faults algorithm, but would require potentially complex infrastructure if implemented on passenger roadways. The mitigation algorithm uses known information about the environment to change a single Kalman filter's parameters to gain a better estimate than a Kalman filter not using these changing parameters. A basic example of the information that could be used is the start and end locations of a tunnel, which would indicate where the GPS measurement would not be useful because satellite reception would be limited to non-existent. This algorithm, like the fault reduction algorithm used to detect and correct for unknown faults, was also shown to reduce the effect of a fault on the estimation of the vehicle's position.

6.2 Future Work

The next step for this research is implementation on a moving full scale or small scale vehicle. The testing of this algorithm was made using readings from physical sensors, as opposed to simulated sensor readings, but typical vehicle models, such as the bicycle model, could not be used in the Kalman filters because the vehicle was not moving. This reduced the need for the Kalman filter; however, Kalman filters were still used to maintain the applicability of the fault detection and mitigation algorithms to a vehicle moving under its own power.

An additional factor to consider when the vehicle is moving is the need for real time operation. Kalman filters are a more computationally intensive algorithm than averaging algorithms. The computational load would be increased further, and could be come problematic, if a bank of Kalman filters is used.

In addition, various algorithms using the statistics of the innovations terms could be developed to determine which one or combination is most effective at determining faults. This testing could also include changing the interval of data used for the approximation of these statistics, shown as N in Equations 5.3 through 5.5. Using a smaller interval makes the statistics more sensitive to faults, whereas a lager interval reduces the number of faults being detected incorrectly which could be used to fine-tune the accuracy of the fault detection algorithm. A universal metric to define the filter with the lowest probability of fault should be developed. This metric needs to be independent of the number of innovations terms, as each filter in the Kalman filter bank has a different number of filters.

Briefly mentioned in Chapter 5, the search algorithm used with the bank of Kalman filters was very basic. This algorithm could become more effective if a universal metric was determined for the probability of fault, the use of adaptive thresholding to determine when a fault has occured, as well as testing other search algorithms. The reaction time of the fault detection algorithm was around 5 seconds with a fault in a single sensor, which although excellent for many dynamic systems, is insufficient for passenger vehicles. This reaction time could be improved with a more optimized search algorithm.

This research has also pointed out a few areas of improvement on the various hardware test beds. The simulated GPS system used on the PURRS could be improved to become a more robust ground-truth measurement than the boom arm, which is prone to bias and requires re-calibration every time the system is used. This improvement can come in the form of camera calibration and improvement to the fiducial system implemented on the vehicle and frame. One of the sources for error, although insignificant in this research because the GPS signal was purposefully corrupted to make it more scale-realistic, was that the fiducials were mounted at different heights relative to the treadmill surface. This created more error toward the edges of the treadmill due to the camera's perspective. The elimination of the difference in height between the two sets of fiducials will reduce this error significantly.

Appendix A

Code

```
#!/usr/bin/env python
import roslib
import sys
import rospy
import numpy
import math
import serial
from std_msgs.msg import Int32MultiArray, Float32MultiArray
from math import sin, cos, pi
import time
```

class arduinoencoder:

```
def __init__(self):
#ROS publishers
self.data_pub = rospy.Publisher("raw_purs_encoder_data",
Int32MultiArray)
self.posdata_pub = rospy.Publisher("vehicle_position_data",
Float32MultiArray)
self.RPYdata_pub = rospy.Publisher("vehicle_RPY_data",
Float32MultiArray)
self.ser = serial.Serial()
self.ser = serial.Serial()
self.ser.port = '/dev/PURRS_ENCODERS'
self.ser.timeout = 1
```

```
self.ser.open()
    self.msg_length = 2
def main(args):
  ic = arduinoencoder()
  rospy.init_node('PURRS_encoders', anonymous=True)
  ic.ser.write('R1Sr\n')
  ic.ser.flushInput()
  while not rospy.is_shutdown():
    data_str = ic.ser.readline()
    data = data_str.strip().split(':')
    #print len(data)
    \# print data
    if len(data) = 9 and data[0] = 'A_{-}' and data[8] = '_S':
      data = data [1: len(data) - 1]
      \# print data
      idata = numpy.array(data,dtype=int)
      fdata = numpy.array(data,dtype=float)
      #print data
      msg = Int32MultiArray()
      msg.data.append(int(data[0]))
      msg.data.append(int(data[1]))
      msg.data.append(int(data[2]))
      msg.data.append(int(data[3]))
      msg.data.append(int(data[4]))
      msg.data.append(int(data[5]))
      msg.data.append(int(data[6]))
      ic.data_pub.publish(msg)
      convangle=numpy.array ([[fdata[0]*2*pi/(4*2048.0)],[fdata[1]*2*
         pi/(4*2048.0)], [fdata[2]*2*pi/(4*2048.0)], [fdata[3]*2*pi
```

```
/(4*2048.0)],[fdata[4]*2*pi/(4*2048.0)],[fdata[5]*2*pi/(4*2048.0)],[fdata[5]*2*pi/(4*2048.0)]])
```

```
\#print\ convangle
```

```
posmsg=Float32MultiArray()
posdata = numpy.array([[0.0],[0.0],[0.0],[0.0]])
#posdata[0,0]=(-36.5*sin(float(convangle[0,0])) + 4.5 - 2.5)
-17.25 # x-axis position of encoder 2 in inches
posdata[0,0]=36.625*sin(float(convangle[0,0])) + 4.5
posdata[1,0]=36.625*cos(float(convangle[0,0])) - 37 # y-axis
position of encoder 2 in inches
```

```
# -44.75 -32
```

```
posdata [2,0] = float (posdata [0,0]) + 37*cos (float (convangle [1,0]) + float (convangle [0,0])) # x-axis position of vehicle CG in inches
```

posdata [3,0] = float (posdata [1,0]) - 37*sin (float (convangle [1,0]) + float (convangle [0,0])) # y-axis position of vehicle CG in inches

print posdata

```
posmsg.data.append(float(posdata[0,0]))
posmsg.data.append(float(posdata[1,0]))
posmsg.data.append(float(posdata[2,0]))
```

```
posmsg.data.append(float(posdata[3,0]))
```

ic.posdata_pub.publish(posmsg)

```
RPYmsg = Float32MultiArray()
RPYmsg.data.append(float(convangle[3,0]))
RPYmsg.data.append(float(convangle[4,0]))
RPYmsg.data.append(float(convangle[2,0]))
ic.RPYdata_pub.publish(RPYmsg)
```

```
if __name__ '__main__':
    main(sys.argv)
```

Bibliography

- FENTON, R. E. and R. J. MAYHAN (1991) "Automated Highway Studies at the Ohio State University - An Overview," *IEEE Transactions on Vehicular Technology*, 40(1), pp. 100–113.
- [2] LAPAPONG, S. (2007) Vehicle Similitude Modeling and Validation of the Pennsylvania State University Rolling Roadway Simulator, Ph.D. thesis, The Pennsylvania State University.
- [3] ZWORYKIN, V. K., L. E. FLORY, and W. S. PIKE (1958), "Automatic Control System for Vehicles," .
- [4] FENTON, R. E. (1970) "Automatic Vehicle Guidance and Control- A State of the Art Survey," IEEE Transactions on Vehicular Technology, VT-19(1), pp. 153–161.
- [5] OLSON, K. W. (1973) "Reference System for Vehicular Lateral Guidance," Proceedings of the IEEE, 61(12), pp. 1764–1765.
- [6] FENTON, R. E., G. C. MELOCIK, and K. W. OLSON (1976) "On the Steering of Automated Vehicles: Theory and Experiment," *IEEE Transactions on Automatic Control*, **12**(3), pp. 306–315.
- [7] OLSON, K. W. (1977) "Wire-Reference Configurations in Vehicle Lateral Control," IEEE Transactions on Vehicular Technology, VT-26(2), pp. 161–172.
- [8] MATSUMOTO, S. (1971) "Highway Communication System using Guidance-Cable," Vehicular Technology, IEEE Transactions on, 20(1), pp. 12–16.
- [9] ASHMORE, C. and T. M. MITCHELL (1997), "WesTrack: Putting ITS to Work,".
- [10] HELPS, F. G. (1962) "Driverless Tractors for Materials Handling," in *Recent Developments in Industrial Electronics*, March, London, pp. 273–275.
- [11] ARKIN, R. C. and R. R. MURPHY (1990) "Autonomous Navigation in a Manufacturing Environment," *IEEE Transactions on Robotics and Automation*, 6(4), pp. 445–454.
- [12] MAKELA, H. and T. V. NUMERS (2001) "Development of a navigation and control system for an autonomous outdoor vehicle in a steel plant," *Control Engineering Practice*, 9(5), pp. 573–583.
- [13] TAN, H.-S. and B. BOULGER (1999) "A Snowplow Steering Guidance System Using Roadway Markers - Problem Formulation and Solution Concept," Vehicle System Dynamics, 32(1), pp. 3–22.

- [14] STEINFELD, A. and H.-S. TAN (2000) "Development of a Driver Assist Interface for Snowplows Using Iterative Design," *Transport Human Factors*, 2(3), pp. 247–264.
- [15] KAMEWAKA, S. and S. UEMURA (1987) "A Magnetic Guidance Method for Automated Guided Vehicles," *IEEE Transactions on Magnetics*, 23(5), pp. 2416–2418.
- [16] HOUSKAMP, R. W. (1987), "Electric Field Guidance System for Automated Vehicles,".
- [17] ALOFS, C. W. and R. R. DRENTH (1995), "Method and Apparatus for Guiding a Driverless Vehicle Using a Senor Tracking a Cable Emitting an Electromagnetic Field,".
- [18] NAKANO, Y., M. MINAMI, and T. ASAKURA (1998) "Development of Magnetic Nail Guidance System for Mobile Manipulators," in *IEEE Conference on Intelligent Transportation* System, Boston, MA, pp. 791–796.
- [19] MELLADO, M., E. VENDRELL, A. CRESPO, P. LOPEZ, J. GARBAJOSA, C. LOMBA, K. SCHILLING, H. STUTZLE, and R. MAYERHOFER (1999) "Application of a real time expert system platform for flexible autonomous transport in industrial production," *Computers in Industry*, 38(2), pp. 187–200.
- [20] DUNLOP, G., M. JUFER, and M. PERROTTET (1997) "A Phase Sensitive Guidance System for an Inductively Powered Automatic Guided Vehicle," in 1997 8th International Conference on Advanced Robotics. Proceedings. ICAR'97, IEEE, Monterey, CA, pp. 181–186.
- [21] MAHRT, R. (1971) "Principles of Automatic Guidance of Vehicles on a Lane by Means of Permanent Magnetic Nails and Board Computer Control," in *Vehiclular Technology Conference*, pp. 250–252.
- [22] JOHNSTON, A. R., A. TOURAJ, and J. Y. LAI (1979) "Automated Vehicle Guidance Using Discrete Reference Markers," Vehicular Technology, IEEE Transactions on, 28(1), pp. 95– 106.
- [23] HESSBURG, T., H. PENG, M. TOMIZUKA, W.-B. ZHANG, and E. KAMEI (1991) "An Experimental Study on Lateral Control of a Vehicle," in *American Control Conference*, 1991, Boston, MA, pp. 3048–3089.
- [24] HESSBURG, T. and M. TOMIZUKA (1993) "Fuzzy Logic Control for Lateral Vehicle Guidance," in *IEEE Conference on Control Applications*, August, Vancouver, B.C., Canada, pp. 55–63.
- [25] LEE, H., D. W. LOVE, and M. TOMIZUKA (1995) "Longitudinal Maneuvering Control for Automated Highway Systems Based on a Magnetic Reference/Sensing System," in *American Control Conference, Proceedings of the 1995*, Seattle, WA, pp. 150–154.
- [26] IENG, S.-S. and P. BRIAND (2006) "Infrastructure and Vehicle Communication for Speed Limitation Based on Magnetic Markers," in *Intelligent Vehicles Symposium*, Tokyo, Japan, pp. 144–149.
- [27] PANG, Y., A. LOPEZ DE LA CRUZ, and G. LODEWIJKS (2008) "Bipolar magnetic positioning system for automated guided vehicles," in 2008 IEEE Intelligent Vehicles Symposium, IEEE, Eindhoven, The Netherlands, pp. 883–888.
- [28] LU, Z., X. LIU, Q. ZHANG, and R. HE (2013) "Research on Co-operative Vehicle-Infrastructure Systems Method of Digital Magnetized Asphalt Pavement," 2013 Third International Conference on Intelligent System Design and Engineering Applications, pp. 1159– 1162.

- [29] GULDNER, J., S. PATWARDHAN, H.-S. TAN, and W.-B. ZHANG (1999) "Coding of Road Information for Automated Highways," *Intelligent Transportation Systems*, 4(3-4), pp. 187– 207.
- [30] RAJAMANI, R., H.-S. TAN, B. K. LAW, and W.-B. ZHANG (2000) "Demonstration of integrated longitudinal and lateral control for the operation of automated vehicles in platoons," *IEEE Transactions on Control Systems Technology*, 8(4), pp. 695–708.
- [31] TAN, H.-S., R. RAJAMANI, and W.-B. ZLHANG (1998) "Demonstration of an Automated Highway Platoon System," in *Proceedings of the American Control Conference*, June, Philadelphia, Pennsylvania, pp. 1823–1827.
- [32] TAN, H.-S., J. GULDNER, S. PATWARDHAN, and B. BOUGLER (1999) "Development of an Automated Steering Vehicle Based on Roadway Magnets - A Case Study of Mechatronic System Design," *IEEE/ASME Transactions on Mechatronics*, 4(3), pp. 258–272.
- [33] HOPSTOCK, D. M. and L. D. WALD (1996) "Verification of Field Model for Magnetic Pavement Marking Tape," *IEEE Transactions on Magnetics*, **32**(5), pp. 32–34.
- [34] PARENT, M., P. BODU, M. VIARENGO, M. MIGLIETTA, and J. CANOU (2011) Navigation Technologies for Cybercars and Advanced City Vehicles, Tech. Rep. May 2006.
- [35] POLVANI, D. G. (1986) "Magnetic Guidance of Autonomous Vehicles," in Oceans '86, Washington D.C., pp. 1407–1412.
- [36] (1987) "Magnetic Guidance of Autonomous Vehicles Part 2," in Unmanned Untethered Submersible Technology, Proceedings of the 1987 5th International Symposium on, pp. 257–264.
- [37] WIEGERT, R. (2003) "Magnetic Anomaly Guidance System for Mine Countermeasures Using Autonomous Underwater Vehicles," in Oceans 2003. Celebrating the Past ... Teaming Toward the Future (IEEE Cat. No.03CH37492), IEEE, San Diego, CA, pp. 2002–2010 Vol.4.
- [38] HU, L., W.-Z. LOU, R. SONG, C. GAO, and X. LI (2009) "A Novel Design of Micromagnetic Sensor Guidance," in *Proceedings of the 2009 4th IEEE International Conference* on Nano/Micro Engineered and Molecular Systems, Shenzhen, China, pp. 235–237.
- [39] LEE, S., D. YOON, and A. GHOSH (2008) "Intelligent parking lot application using wireless sensor networks," 2008 International Symposium on Collaborative Technologies and Systems, pp. 48–57.
- [40] NISHIBE, Y., N. OHTA, K. TSUKADA, H. YAMADERA, Y. NONOMURA, K. MOHRI, and T. UCHIYAMA (2004) "Sensing of Passing Vehicles Using a Lane Marker on a Road With Built-In Thin-Film MI Sensor and Power Source," *IEEE Transactions on Vehicular Technology*, **53**(6), pp. 1827–1834.
- [41] GAO, H., T. HEUER, K. DIMITROPOULOS, N. GRAMMALIDIS, M. WEINMANN, M. HUH-NOLD, T. ASTHEIMER, P. KIRRANE, C. STOCKHAMMER, A. KURZ, J. PFISTER, and U. HARTMANN (2009) "Safe airport operation based on innovative magnetic detector system," *Intelligent Transport Systems, IET*, 3(2), pp. 236–244.
- [42] MILLER, G., S. FELS, and S. OLDRIDGE (2011) "A Conceptual Structure for Computer Vision," in 2011 Canadian Conference on Computer and Robot Vision, IEEE, St. Johns, NL, pp. 168–174.

- [43] LEIFER, J. (2003) "A Close-Range Photogrammetry Laboratory Activity for Mechanical Engineering Undergraduates," in *FIE 2003 33rd Annual Frontiers in Education*, Boulder, CO, pp. 7–12.
- [44] WANG, H., S. SHEN, and X. LU (2012) "Comparison of the Camera Calibration between Photogrammetry and Computer Vision," in 2012 International Conference on System Science and Engineering (ICSSE), IEEE, pp. 358–362.
- [45] BERTOZZI, M., A. BROGGI, M. CELLARIO, A. FASCIOLI, P. LOMBARDI, and M. PORTA (2002) "Artificial Vision in Road Vehicles," in *Proceedings of the IEEE*, vol. 90, pp. 1258– 1271.
- [46] WAXMAN, A. M., J. L. MOIGNE, and B. SRLNIVASAN (1985) "Visual Navigation of Roadways," in *IEEE Transactions on Robotics and Automation*, pp. 862–867.
- [47] CRISMAN, J. D. and C. E. THORPE (1991) "UNSCARF, A Color Vision System for the Detection of Unstructured Roads," in *IEEE International Conference on Robotics and Au*tomation, Sacramento, CA, pp. 2496–2501.
- [48] WALLACE, R., K. MATSUZAKI, Y. GOTO, J. CRISMAN, J. WEBB, and T. KANADE (1986) "Progress in Robot Road-Following," in *Proceedings 1986 IEEE International Conference* on Robotics and Automation, vol. 3, Institute of Electrical and Electronics Engineers, pp. 1615–1621.
- [49] KIM, K., S. OH, S. KIM, H. JEONG, J. HAN, C. LEE, B. KIM, and C. KIM (1995) "An autonomous land vehicle PRV II: progresses and performance enhancement," in *Proceedings* of the Intelligent Vehicles '95. Symposium, Ieee, Detroit, Michigan, pp. 264–269.
- [50] LEBLANC, D. J., G. E. JOHNSON, P. J. T. VENHOVENS, G. GERBER, R. DESONIA, R. D. ERVIN, C.-F. LIN, A. G. ULSOY, and T. E. PILUTTI (1996) "CAPC : A Road-Departure Prevention System," *IEEE Control Systems*, (December), pp. 61–71.
- [51] SCHNEIDERMAN, H. and M. NASHMAN (1992) "Visual Processing for Autonomous Driving," in *IEEE Workshop on Applications of Computer Vision*, pp. 164–171.
- [52] POMERLEAU, D. (1995) "RALPH: Rapidly Adapting Lateral Position Handler," in Proceedings of the Intelligent Vehicles Symposium, pp. 506–511.
- [53] LUONG, Q.-T., J. WEBER, D. KOLLER, and J. MALIK (1995) "An integrated stereo-based approach to automatic vehicle guidance," in *Computer Vision*, 1995. Proceedings., Fifth International Conference on, c, Cambridge, MA, pp. 52–57.
- [54] CRISMAN, J. D. and C. E. THORPE (1988) "Color Vision for Road Following," in Proceedings of SPIE Conference on Mobile Robots, Cambridge, MA, pp. 1–10.
- [55] BORKAR, A., M. HAYES, and M. T. SMITH (2009) "Robust Lane Detection and Tracking with RANSAC and Kalman Filter," in *IEEE International Conference on Image Processing* (*ICIP*), Cairo, pp. 3261–3264.
- [56] KIM, Z. (2008) "Robust Lane Detection and Tracking in Challenging Scenarios," IEEE Transactions on Intelligent Transportation Systems, 9(1), pp. 16–26.
- [57] SHU, Y. and Z. TAN (2004) "Vision Based Lane Detection in Autonomous Vehicle," in Fifth World Congress on Intelligent Control and Automation (IEEE Cat. No.04EX788), vol. 6, IEEE, pp. 5258–5260.

- [58] NOE, P. S. and K. A. MYERS (1976) "A Position Fixing Algorithm for the Low-Cost GPS Receiver," *IEEE Transactions on Aerospace and Electronic Systems*, AES-12(3), pp. 295–297.
- [59] DENARO, R. (1981) "Navstar: the all-purpose satellite," IEEE Spectrum, (May), pp. 35–40.
- [60] MISRA, P. and P. ENGE (2011) Global Positioning System Signals, Measurements, and Performance, 2nd ed., Ganga-Jumuna Press, Lincoln, Massachusetts.
- [61] BAJIKAR, S., A. GORJESTANI, P. SIMPKINS, and M. DONATH (1997) "Evaluation of In-Vehicle GPS-Based Lane Position Sensing for Preventing Road Departure," in *IEEE Transactions on Intelligent Transportation Systems*, Boston, MA, pp. 397–402.
- [62] LEAVITT, L. E. and B. L. PETERS (1988) "Low Cost High Accuracy Integrated GPS-Inertial Navigator for Reconnaissance Missions," in *Position Location and Navigation Symposium*, Orlando, FL, pp. 381–388.
- [63] FARRELL, J. A., M. BARTH, R. GALIJAN, and J. SINKO (1998) GPS/INS Based Lateral and Longitudinal Control Demonstration: Final Report, Tech. rep.
- [64] PANZIERI, S., A. MEMBER, F. PASCUCCI, and G. ULIVI (2002) "An Outdoor Navigation System Using GPS and Inertial Platform," *IEEE/ASME TRANSACTIONS ON MECHA-TRONICS*, 7(2), pp. 134–142.
- [65] KALMAN, R. E. (1960) "A New Approach to Linear Filtering and Prediction Problems," ASME Journal of Basic Engineering, 82(Series D), pp. 35–45.
- [66] MEHRA, R. K. and J. PESCHON (1971) "An Innovations Approach to Fault Detection and Diagnosis in Dynamic Systems," *Automatica*, 7(5), pp. 637–640.
- [67] MEHRA, R. (1970) "On the Identification of Variances and Adaptive Kalman Filtering," IEEE Transactions on Automatic Control, 15(2), pp. 175–184.
- [68] KERR, T. H. (1977) "Real-Time Failure Detection : A Nonlinear Optimization Problem That Yields a Two-Ellipsoid Overlap Test," *Journal of Optimization Theory and Applications*, 22(4), pp. 509–536.
- [69] (1980) "Statistical Analysis of a Two-Ellipsoid Overlap Test for Real-Time Failure Detection," *IEEE Transactions on Automatic Control*, 25(4), pp. 762–773.
- [70] GOEL, P., G. DEDEOGLU, S. I. ROUMELIOTIS, and G. S. SUKHATME, "Fault Detection and Identication in a Mobile Robot Using Multiple Model Estimation and Neural Network,"
- [71] BRUMBACK, B. D. and M. D. SRINATH (1987) "A Fault-Tolerant Multisensor Navigation System Design," *IEEE Transactions on Aerospace and Electronic Systems*, AES-23(6), pp. 738–756.
- [72] MIRABADI, A., N. MORT, and F. SCHMID (1998) "A Fault Tolerant Train Navigation System Using Multisensor, Multifilter Integration Techniques," in *FUSION '98 International Conference*, pp. 340–347.
- [73] MAGILL, D. T. (1965) "Optimal Adaptive Estimation of Sampled Stochastic Processes," IEEE Transactions on Automatic Control, AC-10(4), p. 1965.

- [74] ATHANS, M., D. CASTAGON, K.-P. DUNN, C. S. GREENE, W. H. LEE, N. R. SANDELL, and A. WILLSKY (1977) "of the F-8C Aircraft Using a Multiple Model Adaptive Control (MMAC) Method- Part I : Equilibrium Flight," *IEEE Transactions on Automatic Control*, AC-22(5), pp. 768–780.
- [75] MAYBECK, P. S. and P. D. HANLON (1993) "Performance Enhancement of a Multiple Model Adaptive Estimator," in *IEEE Conference on Decision and Control*, San Antonio, TX, pp. 462–468 Vol. 1.
- [76] BRENNAN, S., A. ALLEYNE, and M. DEPOORTER (1998) "The Illinois Roadway Simulator
 A Hardware-in-the-Loop Testbed for Vehicle Dynamics and Control," in *Proceedings of the American Control Conference*, June, Philadelphia, Pennsylvania, pp. 493–497.
- [77] BRENNAN, S. and A. ALLEYNE (1999) "A Scaled Testbed for Vehicle Control: the IRS," in Proceedings of the 1999 IEEE International Conference on Control Applications (Cat. No.99CH36328), vol. 1, IEEE, Kohala Coast-Island of Hawaii, Hawaii, USA, pp. 327–332.
- [78] (2000) "The Illinois Roadway Simulator: A Mechatronic Testbed for Vehicle Dynamics and Control," *IEEE/ASME Transactions on Mechatronics*, 1(4), pp. 349–359.
- [79] KACHROO, P., K. OZBAY, R. G. LEONARD, and C. UNSAL (1995) "Flexible low-cost automated scaled highway (FLASH) laboratory for studies on automated highway systems," in 1995 IEEE International Conference on Systems, Man and Cybernetics, Intelligent Systems for the 21st Century, vol. 1, pp. 771–776.
- [80] (2013), "PennDOT Beginning Traffic Line Painting for Motorists Safety," .
- [81] (2007), "Mitigation Strategies for Design Exceptions July 2007 (Lane Width),".
- [82] CHAN, C.-Y., B. BOUGLER, D. NELSON, P. KRETZ, H.-S. TAN, and W.-B. ZHANG (2000) "Characterization of Magnetic Tape and Magnetic Markers as a Position Sensing System of Vehicle Guidance and Control," in *Proceeding of the American Control Conference*, June, Chicago, Illinois, pp. 95–99.
- [83] XU, H.-G., M. YANG, C.-X. WANG, and R.-Q. YANG (2010) "Magnetic Sensing System Design for Intelligent Vehicle Guidance," *IEEE/ASME Transactions on Mechatronics*, 15(4), pp. 652–656.
- [84] RYOO, Y.-J. and J.-H. PARK (2012) "Design and Development of Magnetic Position Sensor for Magnetic Guidance System of Automated Ground Vehicle," in 12th International Conference on Control, Automation and Systems, Jeju Island, Korea, pp. 988–991.
- [85] LEE, S.-Y. and H.-W. YANG (2012) "Navigation of automated guided vehicles using magnet spot guidance method," *Robotics and Computer-Integrated Manufacturing*, 28(3), pp. 425– 436.
- [86] ASSIDIQ, A. A. M., O. O. KHALIFA, R. ISLAM, and S. KHAN (2008) "Real Time Lane Detection for Autonomous Vehicles," in *International Conference on Computer and Communication Engineering*, Kuala Lumpur, pp. 82–88.
- [87] ZHANG, H.-B., K. YUAN, S.-Q. MEI, and Q.-R. ZHOU (2004) "Visual Navigation of an Automated Vehicle Based on Path Recognition," in *Third International Conference on Machine Learning and Cyberneetics*, August, Shanghai, pp. 26–29.
- [88] CANNY, J. (1986) "A Computational Approach to Edge Detection." IEEE Transactions on Pattern Analysis and Machine Intelligence, 8(6), pp. 679–98.

- [89] MACEK, K., B. WILLIAMS, S. KOLSKI, and R. SIEGWART (2004) "A lane detection vision module for driver assistance," in *Proceedings of the IEEE/APS Conference on Mechatronics* and Robotics, pp. 77–82.
- [90] STOMBAUGH, T. S., E. R. BENSON, and J. W. HUMMEL (1998) "Automatic guidance of Agricultural Vehicles at High Field Speeds," in 1998 ASAE Annual International Meeting, Orlando, FL.
- [91] FARRELL, J. A., H.-S. TAN, and Y. YANG (2003) "Carrier Phase GPS-Aided INS-Based Vehicle Lateral Control," *Journal of Dynamic Systems, Measurement, and Control*, **125**(3), pp. 339–353.

Vita

Anthony J. Mangus

ajm5386@psu.edu

Education:

M.S. in Mechanical Engineering, 2013 The Pennsylvania State University, University Park, PA

B.S. in Mechanical Engineering, 2013 The Pennsylvania State University, University Park, PA Honors in Mechanical Engineering Minor in Engineering Mechanics

Professional Experience:

Mechanical Engineering Intern for U.S. Navy 2010-2012 Assisted on various engineering projects including design and field implementation Will be hired as a Mechanical Engineer after graduation

Awards:

The SMART Scholarship, 2009-2013 Louis A. Harding Memorial Scholarship, 2010-2013 James W. Dunlop Memorial Scholarship, 2011-2013 Eagle Scout, 2008

Additional Experience

Member, The Pennsylvania State University Marching Blue Band 2008-2013 Manager, The Pennsylvania State University Marching Blue Band 2011-2013 Guide, The Pennsylvania State University Marching Blue Band 2011-2013 Member, Penn State Model Railroad Club 2012-2013 Member, Penn State R/C Race Club 2009-2012 President, Penn State R/C Race Club 2010-2012