A MAGNETIC LOCALIZATION TECHNIQUE DESIGNED FOR USE WITH MAGNETIC LEVITATION SYSTEMS

A THESIS SUBMITTED TO THE GRADUATE DIVISION OF THE
UNIVERSITY OF HAWAI'I AT MĀNOA IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

MECHANICAL ENGINEERING

APRIL 2017

BY

B.J. TIX

THESIS COMMITTEE:

PETER BERKELMAN, CHAIRPERSON

DILMURAT AZIMOV

KYUNGIM BAEK

KEYWORDS: MAGNETIC LEVITATION, MAGNETIC LOCALIZATION, CAPSULE ENDOSCOPY, ROBOTICS, MEDICAL ROBOTICS

DEDICATION

This work is dedicated to my loving wife, Carolyn Tix.

Without her encouragement and support, this work would not have been possible.

ACKNOWLEDGEMENTS

ABSTRACT

Wireless capsule endoscopy (WCE) is a modern medical technology which enables doctors to view and diagnose problems deep within the intestines. Ever since the first uncontrolled WCEs were introduced, researchers have been searching for a way to control the motion of a robotic capsule through the intestines, a task which requires a fusion of mechanical design, electrical hardware, accurate physical models, and software control algorithms. An ongoing roadblock to achieving this goal is the inherent difficulty of merging localization techniques with actuation techniques, as the two often interfere with one another. This research presents a novel approach to merging localization and actuation which is capable of providing precise control over up to six degrees of freedom. The novel contribution of this research centers on a new algorithm which effectively combines previously successful actuation and localization techniques.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

LIST OF ABBREVIATIONS

DOF	Degree of Freedom
GI	Gastrointestinal
GUI	Graphical User Interface
HRIL	Human Robot Interaction Laboratory
IDE	Integrated (Software-)Development Environment
IMU	Inertial Measurement Unit
LSB	Least Significant Byte
maglev	Magnetic levitation system, in particular the maglev system in the HRIL.
MSB	Most Significant Byte
PC	Personal Computer
PD	Proportional / Derivative Controller
PI	Proportional / Integral Controller
PID	Proportional / Integral / Derivative Controller
WCE	Wireless Capsule Endoscope

1 INTRODUCTION

The research presented in this thesis is intended to address a very specific technical concern in the field of magnetic levitation: the need to precisely know the location and orientation of a magnet which we would like to levitate without having a direct visual line of sight to that magnet. This motivation for solving this particular technical problem arises from the field of controlled wireless capsule endoscopy. In order to fully understand the motivation for this work and the particular restrictions which have been adopted, it is important to first understand the current state of wireless capsule endoscopy and the prior attempts which have been made to solve this problem.

Endoscopy and colonoscopy are important medical procedures which allow a doctor to directly view the inside of a patient's digestive tract and are frequently used to diagnose a variety of medical conditions, including colon cancer and ulcers [1]. Traditionally, this is accomplished by inserting a flexible fiber-optic endoscope into the patient's intestine, with various techniques for controlling the movement of the endoscope to traverse the digestive tract [2]–[7]. Wired endoscopes have several significant drawbacks. The endoscope must be inserted into the patient from the outside, typically through the anus in order to view the inside of the lower intestines, and sometimes through the throat in order to view the esophagus. The procedure is notoriously uncomfortable and unpleasant for the patient, and patient discomfort caused by wired capsule endoscopy is a significant limiting factor in the duration of an endoscopy and the depth of insertion of the endoscope, and may even discourage patients from seeking endoscopy for the sake of early-detection of colon cancer [1], [8]–[10]. Furthermore, wired endoscopes can only reach a section of the intestines, limited by the length of the endoscope and the volume of endoscope that the patient can accommodate within their intestines.

In response to these drawbacks, Wireless Capsule Endoscopes (WCE) have been approved for use by the FDA and have been made commercially available within the last few decades [1], [8]–[11]. Commercially available wireless endoscope capsules such as the *PillCam,* shown in Figure 1-1, are uncontrolled, moving

Figure 1-1: The PillCam Wireless Capsule Endoscope [12]

through the digestive tract via peristalsis while recording images, video, or other sensor data to aid medical professionals in patient diagnosis [12], [13]. While the lack of control greatly simplifies the design of the endoscope capsule, it also results in unpredictable movement through the GI tract, resulting in incomplete evaluations roughly 20 percent of the time [14]. Such capsules have the further drawback that they cannot be controlled and therefore cannot be directed to focus on areas of interest, and they cannot be used for any active medical procedure such as drug delivery or cyst removal, which may be possible with a controlled capsule. Preliminary in-vivo feasibility trials of controlled wireless endoscope capsules have already been performed and capsule navigation and control remain active areas of research [15]–[18].

2 PRIOR WORKS

Ultimately, it would be beneficial to have a completely controlled endoscopic capsule, one which a doctor or medical professional could steer through the intestines like a small vehicle: stopping, moving forwards and backwards, and turning under the control of a human driver. If such a device could be built, it would greatly alleviate the need for uncomfortable traditional endoscopies as well as greatly reducing the risk of incomplete evaluation inherent to uncontrolled wireless capsule endoscopy. For the past two decades, researchers have tried numerous techniques for achieving this goal, with varying degrees of success. An overview of previous attempts and techniques is provided in this section.

2.1 CONTROL OF THE WIRELESS ENDOSCOPE CAPSULE

2.1.1 MAGNETIC ACTUATION

By placing a permanent magnet within the wireless endoscopic capsule, the position and orientation of the capsule can be roughly controlled with a simple hand-held wand containing a strong permanent magnet or electromagnet. By manually positioning the wand outside of the patient, the endoscope can be pulled into new orientations or positions inside the patient's GI tract [19]. In this technique, a patient swallows a WCE which transmits images in real-time at a short regular intervals, generally multiple images per second to produce a slow but real-time video stream. Once the WCE is inside the patient, a medical professional holding the wand

Figure 2-1: MiroCam Navi HBC (Human Body Controller). By manually positioning this device, the WCE can be communicated with and manipulated in real-time. [22]

containing the external magnet moves the wand outside of the patient's body, using the video stream as feedback about the movement of the WCE. By this method, the movement of the WCE can be roughly controlled. The safety and efficacy of this technique was demonstrated in the esophagus by [19], [20], though with the limitation that the capsule could not be moved up the esophagus due to insufficient magnetic force resisting peristalsis. However, the forces were sufficient to slow the capsule and prolong the transit time through the esophagus, as well as to rotate and reorient the capsule to capture different angles while in transit. More recently, the *MiroCam Navi* (Figure 2-1) is an example of a commercially available system which operates on the same principle and has recently been proven effective in trials by [21], [22].

Though effective, this technique is clearly quite imprecise, relying on a human operator moving an external magnet by hand based on intuition and limited visual feedback from the WCE camera. To address this concern, one possibility is to place the external magnet on a robotic arm and precisely control the motion of that arm to lend a degree of precision to the procedure. Systems utilizing a robotic arm to automatically position the external magnet are described in [23]–[25]. The robotic arm might contain a permanent magnet, as in [23], or might contain an electromagnet which can vary the magnetic forces applied by varying the current to the electro magnet, as presented in [25]. Such systems can even integrate localization algorithms based on measuring the forces generated between the external and internal magnets, as demonstrated in [24]. However, there are several drawbacks to this system as well. The external magnet can only apply forces to the capsule along a single force vector, the line between the capsule and the external magnet. Precise control is only available along this line of force, while moving the capsule in other directions requires slowly dragging the
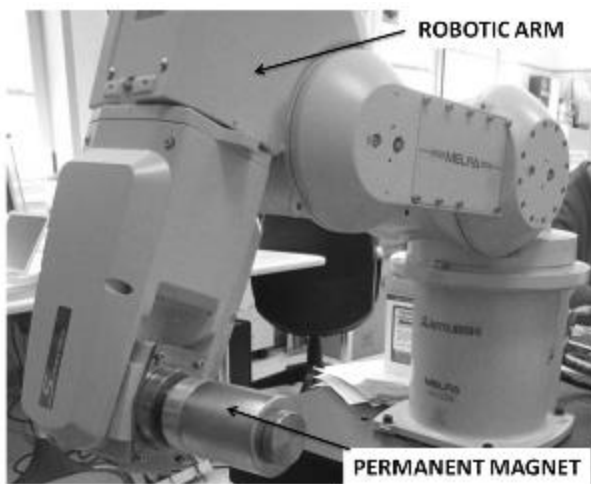
Figure 2-2: A robotic arm used to control the motion of a WCE [23]

capsule, causing it to follow the leading external magnet. A bulky robotic arm is required, dominating the working space above the patient and limiting the movements of any medical professionals in the area, as well as potentially introducing safety concerns with regards to ensuring that the robotic arm does not impact the patient during one of its motions. Since control can only be exerted along a single line of force, reorienting the capsule may require large motions of the robotic arm. While these systems have been demonstrated to work effectively, there are enough drawbacks associated with their use to motivate a search for a superior method of control.

2.1.2 MECHANICAL ACTUATION

An alternative locomotion system which has been extensively researched relies on a small capsule with a helical ridge. The capsule has one or more permanent magnets embedded in its structure. Once the WCE is inside the patient, the patient is exposed to an external rotating magnetic field, causing the capsule to spin. The rotating motion of the helical ridge causes a screw-force lateral motion which can move the capsule forward and backwards through the intestine [26]–[34]. A helical-ridge capsule is shown in Figure 2-3. Other endoscopic capsules make use of leg-like [35] or tail-like [36], [37] appendages to move through the intestines, as shown in Figure 2-4. Both legged capsules and spinning helical ridge capsules suffer from two primary drawbacks:

1) Their motion relies on maintaining traction with the intestine wall, and if traction is lost it is no longer possible to control the motion of the capsule. This can be a particular problem around bends and curves [14].

2) They can only be moved along a single degree of freedom, forward and backwards through the intestines. There is no ability to turn or change orientation, which is a major drawback in comparison to the magnetic actuation techniques which have already been discussed.
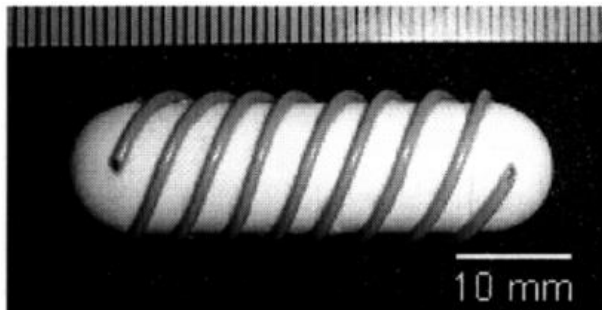
Figure 2-3: A WCE with a helical ridge [32]

Figure 2-4: A WCE with leg-like appendages for locomotion [35]

2.1.3 MAGNETIC LEVITATION

Magnetic levitation has been extensively studied for non-medical purposes, including industrial applications, most notably magnetic bearings [38]–[40], haptic interfaces [41]–[44], transportation and various uses in other fields [39]. Several recent studies explore the possibility of controlling a wireless endoscope through the use of magnetic levitation [45], [46]. The benefit of magnetic levitation over other forms of actuation is that magnetic levitation allows the endoscope to be precisely controlled in five or even six degrees

4

of freedom with a high degree of precision and does not require the use of a bulky, moving robotic arm which may interfere with the movements of medical professionals in the room while the controlled endoscopic procedure is taking place [46]–[48].

A magnetic levitation system intended for use with a wireless endoscope is presented in [45]. This system is comprised of a single electromagnet coil. The current through this coil is varied, resulting in varying magnetic force, which is used to exert forces over a permanent magnet affixed to an endoscopic capsule. This system has two significant limitations: it can control only a single dimension of travel, in this case the vertical position of the capsule, and it relies on visual localization which would not be possible if the capsule were inside the body of a patient.

The *OctoMag* system is a magnetic levitation system intended for "micromanipulation" of a small capsule within the human eye [46], [49]. This system can achieve impressive precision, achieving positional accuracy with maximum position errors as small as 29.77μm. The OctoMag is intended to drive a very small microrobot only 500μm long and within a workspace defined by a 25mm diameter sphere. The OctoMag uses visual localization for its feedback control, since the microrobot is visible from the outside while operating on the human eye. A similar microrobot control system is presented in [50]. The OctoMag team also postulates that it should be possible to operate the OctoMag with somewhat reduced precision even with no feedback control at all, and present experiments to support this conclusion in [46]. Although the accuracy of the system is

Figure 2-5: The OctoMag System [47]

impressive, the OctoMag's applicability to the field of wireless endoscopy is limited by its extremely constrained working space and the extremely small microrobots which it is designed to control. Furthermore, closed loop control is achieved in this system through the use of a visual motion-tracking camera, which is possible since the controlled capsule is visible through the pupil of the eye being operated on. A capsule inside the human intestines would not be visible to a motion tracking camera, thus necessitating a new means of localization.

A 6 degree of freedom magnetic levitation system comprised of numerous electromagnetic coils is described in [41], [47], [48], [51], [52]. This system relies on a the Northern Digital Inc Optotrak® visual tracking system [53], [54] to provide 6-degree-of-freedom localization data to the magnetic levitation algorithm and relies on a high update rate of over 800hz and is accurate up to 0.01 mm. This system has not

previously been applied to wireless capsule endoscopy. A system utilizing an MRI to guide a permanent magnet through the human stomach is presented in [55]. Other approaches to wireless endoscope control are presented in [56].

Each of these systems can generate magnetic forces on a small permanent magnet, achieving varying degrees of locomotion, control, and levitation. However, they all share a fundamental weakness in terms of their applicability to wireless endoscopy: none of these systems can accurately determine the position and orientation of the magnet without a visual line of sight. In the field of capsule endoscopy, visual line of sight is unavailable when the capsule is within a patient's intestines. *Localization* is the process of determining the magnet's location and orientation, which is necessary for feedback control, and is beneficial for diagnosis even when feedback control is not used [57]. If the ultimate goal of a fully controllable WCE is to be achieved, an effective means of localization must be found which does not rely on visual line-of-sight to the capsule.

2.2 MAGNETIC LOCALIZATION

Commercially available tools including the *ScopeGuide* by Olympus® [58] have been used to localize and visualize the positioning of a colonoscope used during colonoscopy through a technique known as Magnetic Endoscope Imaging (MEI) [59]. Olympus describes the *ScopeGuide* as:

"The world's only technology providing a real-time 3D representation of the shape and position of the colonoscope inside the body" [58]

While not strictly necessary for colonoscopy, Shah et al. [60] found that magnetic imaging systems for colonoscopes significantly improved the performance of both trainees and experienced endoscopists, and was particularly helpful for less experienced trainees. Shah et al. consider this system to be a valuable training tool, enabling quicker training of new colonoscope operators and facilitating discussion between trainee and trainer [61]. However, MEI does not alleviate the need for a bulky colonoscope hose nor does it fundamentally address the physical limitations of wired endoscopy previously discussed. To traverse the less accessible portions of the GI tract, a wireless endoscopic capsule is required.

A wide variety of localization techniques for a wireless endoscopic capsule are summarized in [10]. In addition to the localization techniques that have already been discussed, [10] also covers radio-frequency localization techniques including RFID [57], [62]–[64] and RSSI [65], [66] which both suffer from low accuracy when applied to wireless capsule endoscopy due to the fact that different human tissues absorb radio waves at different rates, as described in [67]. In addition to radio-frequency localization, [10] also discusses X-Ray localization [68], [69], which suffers from unacceptably slow update rates making it unsuitable for use with a real-time control system, as well as various other techniques which have been attempted with limited success. Other methods of magnet localization include measuring the resultant forces between a permanent magnet within an endoscope capsule and external permanent magnets affixed to force sensors [70], using an

ultrasound sensor to detect an internal magnet [71]. Another method is to equip the capsule with a tri-axial sensor capable of detecting the field generated by an external permanent magnet, and using this reading to localize the capsule [72].

The commercially available *Aurora* localization system from Northern Digital is capable of sub-millimeter accuracy and performs its localization by placing a sensing coil on a device intended for internal medical use [73]–[75]. An external magnetic field is applied from outside the patient's body, which induces a current in the sensor coil, and from the induced current and known external field, the position and orientation of the internal sensor can be computed. The commercially available *microBIRD* system operates on similar principles [74], [76].

The *GaussSense* board is a commercially available array of Hall-effect sensors and associated circuitry and software which is capable of localizing a small magnet to a position accuracy of within 1 mm and an orientation accuracy of within 1 degree [77]. This system is sold as the *GaussBits* kit intended as a novel interface for app development; by placing an array of sensor behind the screen of a mobile device such as a smartphone or tablet, developers can write apps using the GaussBits API that enable the users of those apps to interact with the software via physical objects with embedded magnets. The user places and moves these physical objects and toys around the screen of the mobile device, and the software reacts to the magnetic localization of the GaussSense for various games and effects [42], [78]–[80]. In my own experiments, I found that the GaussSense board could localize a magnet accurately for certain orientations of the magnet, but struggled with rotational accuracy if the magnet was place with the North or South side of the magnet within roughly 20 degrees of parallel to the plane of the GaussSense board. The most accurate readings could be obtained when the magnet was placed such that both the North and South ends of the magnet were clearly visible to the Hall sensors [42].
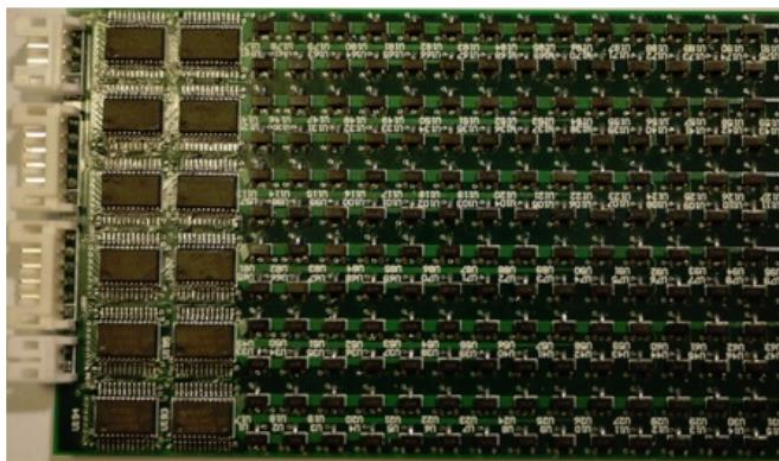
Figure 2-6: The GaussSense Board, an array of hall sensors [77]

I created a simple computer game utilizing the GaussSense board as an input device, enabling the user to move a virtual paddle around a computer screen by physically positioning a simple stylus with a magnetic tip. The game implemented simple 2-Dimensional collision physics to causes a simulated ball to bounce around a virtual 2-D environment in response to the motion of the paddle. This game was integrated with a single-coil electromagnetic actuator, allowing the user to feel the force of the impact of the

virtual paddle to the virtual environment and the virtual bouncing ball. However, once the magnetic actuation was turned on, the GaussSense could no longer accurately determine the location of the stylus, due to the magnetic field created by the electromagnet interfering with the GaussSense's localization algorithm, which was not designed to filter out the interference. This work is presented in [42]. The inability to properly localize the magnet using magnetic flux sensors while in the presence of magnetic fields produced by electromagnetic actuation systems is the major obstacle in combining magnetic localization and actuation for all similar systems and is the major theme of this research.

One successful method of magnet localization is to use an array of sensors to detect the magnetic field generated by a single magnetic dipole, such as a cylindrical magnet, and to use an error-minimization algorithm to estimate the position and orientation of the magnet generating the sensed magnetic field [81]. Hu et al. present an implementation of a similar technique for the localization of a single bar or cylindrical magnet using an array of digital compasses [82]–[85]. Techniques are presented in [82], [83] which can produce 5-degree-of-freedom localization for a symmetrical cylindrical magnet accurate to within approximately 2 mm using a non-linear algorithm. A similar technique is presented in [84], in which an asymmetrical bar magnet to achieve a full six degrees of freedom. Although the localization speeds presented in [82]–[84] are low, on the order of 10hz, I will demonstrate later in this paper that significantly faster speeds are possible with this technique. Concern that the algorithms in [82]–[84] are not suitable for integration with a magnetic control system is expressed in [10], [24]. As I will show in this paper, considerable expansion of the algorithms presented in [82]–[84] is necessary before they can be integrated with a magnetic control system, but such an integration is possible.

Each of the systems presented in this section can localize a magnet with varying degrees of accuracy. However, the techniques thus far presented for localization and actuation have been developed independently of one another, and in general are incompatible with one another. Applying a magnetic field for actuation changes the magnetic flux density which can be detected by magnetic flux density sensors, and therefore disrupts these localization techniques. If a controlled, wireless, magnetic endoscope is to be achieved, magnetic actuation and localization must be seamlessly combined.

2.3 COMBINED LOCALIZATION AND ACTUATION

A patent for the localization of a screw-motion capsule is presented in [26]. This technique solves the problem of interference between the actuation and localization fields by relying on high and low-frequency magnetic signals. The capsule is turned by a low-frequency alternating magnetic field (less than 10 Hz) and localization is accomplished using a small coil within the capsule that detects changes as the capsule moves through a high-frequency alternating magnetic field (1khz-1Mhz). Because the low-frequency and high frequency alternating magnetic fields do not interfere with each other's signals, localization and actuation can

be achieved simultaneously. The limitation of this technique, however, is that the WCE relies on a helical ridge to propel itself forward through screw-motion forces with the side of the intestines, resulting in only a single degree of freedom, the ability to move back and forth, but not to turn or move laterally. The forces produced by this technique are also dependent on a good connection between the helical ridge and the intestine wall, so control may be lost during the motion of the WCE if this connection is compromised for any reason.

The magnetic capsule controlled by an external magnet moved by a robotic arm presented in [23] included a localization technique that was found to be accurate to within roughly 3cm. A more complex localization algorithm relying on a fusion of magnetic field sensing and an inertial measurement unit presented in [24] was able to achieve a positional accuracy within 6.9 mm and orientation accuracy within 7.6 degrees when following a pre-set trajectory at a refresh rate of up to 140hz. However, this system is inconvenient in that it requires the use of a bulky robotic arm which interferes with the workspace of human medical staff and in that the motions of the wireless capsule endoscope are limited by the motion capabilities of this arm.

2.4   NOVEL CONTRIBUTIONS

Ultimately, discovering an effective means of combining magnetic localization and magnetic levitation is an open research question with no definitive solution. In fact, [10] even states that:

"...the question of how to remove the interference between magnetic localization and magnetic actuation is still to be answered." [10], pp.4

None of the prior work in this area has achieved a combined magnetic localization and actuation that:

1) Combines localization and actuation for a wireless capsule endoscope within the human body.
2) Provides precise control over at least 5 degrees of freedom.
3) Does not require a visual line of sight, which is unavailable for any capsule within the human body.

The system presented in this thesis satisfies all three of these requirements. The novel contributions of this research are:

1) Investigates the possibility of a combined magnetic localization and levitation system utilizing an array of electromagnetic coils [47], [48] and an array of 3-dimensional digital compasses [82], [83] combined via a novel algorithm to achieve simultaneous levitation and localization. This technique enables a magnet to be controlled wirelessly and precisely in 5 or 6 degrees of freedom without the need for a visual line-of-sight to the magnet, a previously unprecedented result.

9

2) Demonstrates simulated results showing that fusion of these two techniques is possible in principle and demonstrating the efficacy of the algorithm itself.

3) Presents experimental results in the combination of magnetic localization and magnetic levitation along with recommendations for how these results could be improved upon in future work.

4) Demonstrates that the localization algorithms previously presented in [82]–[85] can be run in real-time, at speeds hundreds of times faster than has previously been reported, without sacrificing the accuracy of the localization algorithm.

3 SYSTEM DESIGN

3.1 MOTIVATION

A novel system for combined levitation and localization has been designed and built in the Human Robot Interaction Laboratory (HRIL) at UH Mānoa [86]. The motivation for this project was to demonstrate that a small capsule equipped with a permanent magnet can be controlled in at least 5 degrees of freedom without the need for a visual line of sight. This achievement has clear applications to the field of wireless capsule endoscopy, where a swallowed wireless endoscope capsule cannot be tracked by a camera or visual motion-tracking system. As previously discussed, previous methods of localization without line-of-sight provide fewer degrees of freedom and require bulky and expensive robotic arms which interfere with the clinical workspace. The newly designed system presented in this thesis has neither of these drawbacks.

3.2 SYSTEM ARCHITECTURE

The new system is composed of four main components:

1. The *maglev*, which provides magnetic levitation and actuation. The maglev consists of an array of electromagnet coils, power supplies, and current amplifiers which interface with a PC to precisely control the current and voltage to each coil

2. The *sensor array*, which provides magnetic localization without the need for a visual line of sight. In addition to the sensor array hardware, a simulated version of the array which runs purely in software was also created and used in several experiments.

3. The *Optotrak* motion tracking system [53], [54], which provides precise visual localization used for validation of the new localization algorithms. The localization algorithms themselves do not rely on data from the Optotrak, which is only used for experiment validation.

4. The *software*, running on an Ubuntu desktop PC [87] which runs the algorithms required for localization and levitation as well as containing functions for various tests and procedures, and a graphical user interface which displays the state of the maglev, sensor array, and localization data from both the Optotrak and the sensor array.

Each of these components will be explored in greater detail throughout this chapter.
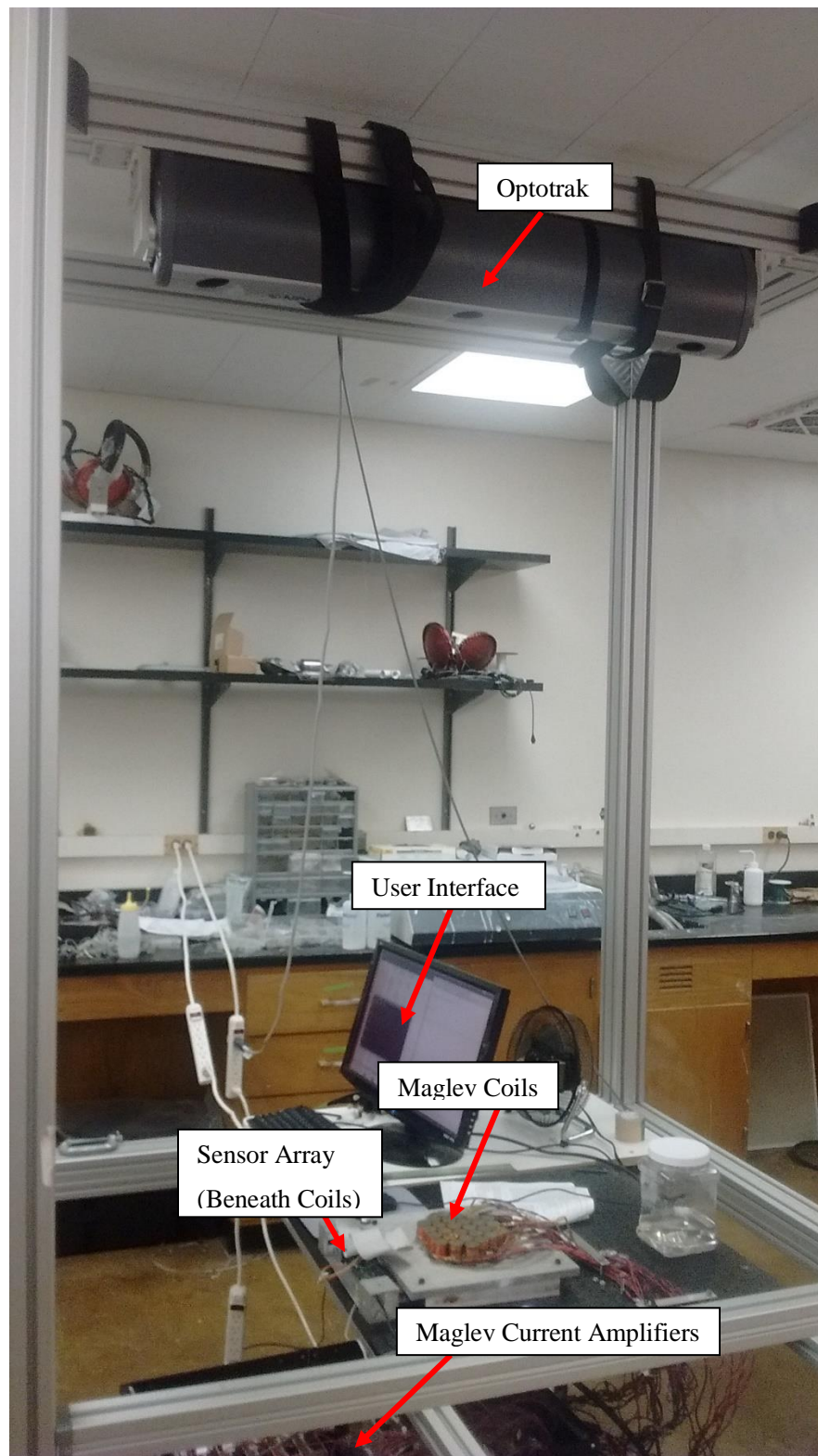
Figure 3-1: System Design Photo
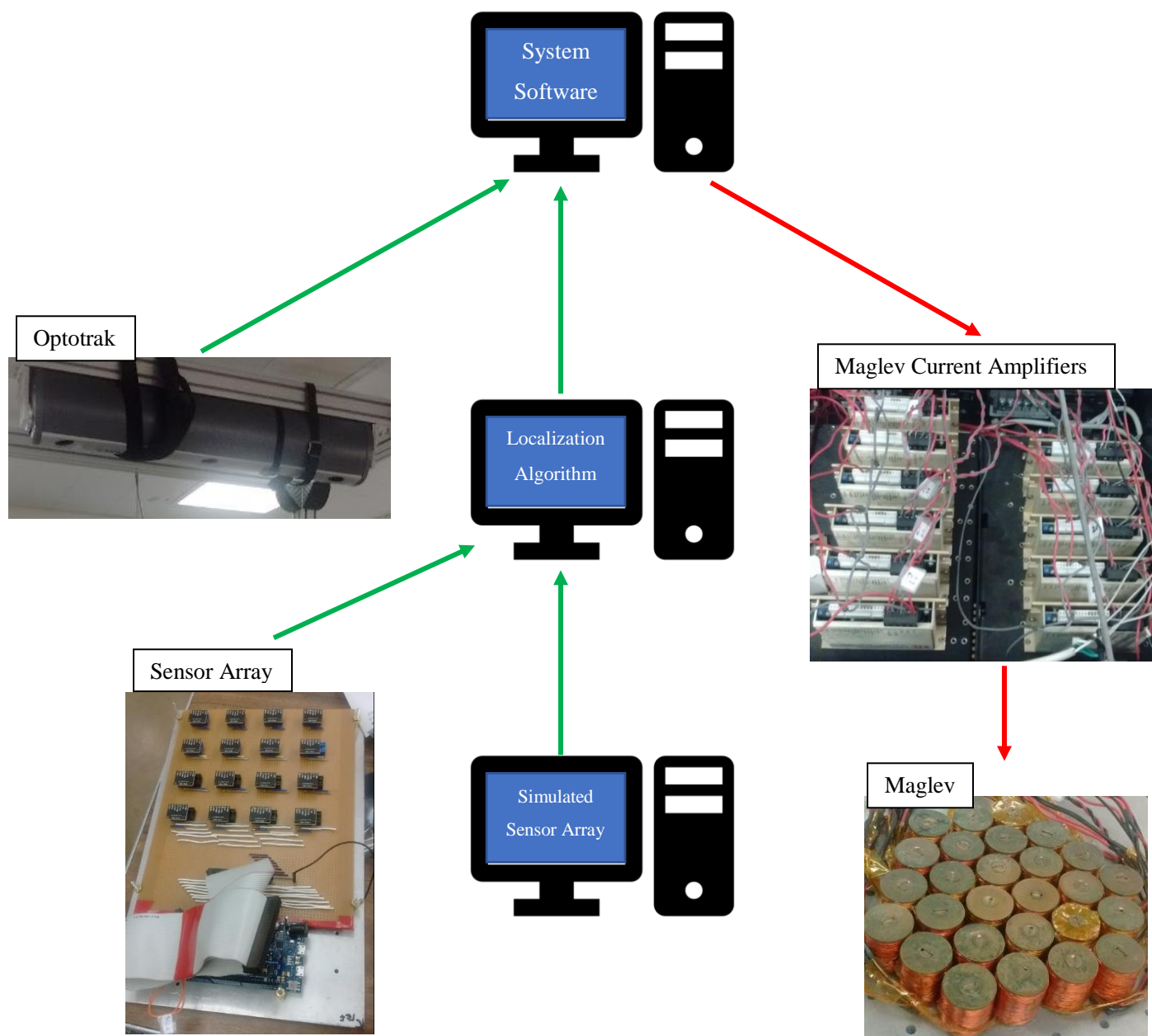
Figure 3-2: System Architecture Diagram

3.3 COMPUTER SPECIFICATIONS

Every component of this system interfaces with a desktop computer tower with the following specifications:

Operating System	Ubuntu 14.04 LTS (Low latency) [87]
Processor	Intel® Core™ i7-2600 CPU @ 3.40GHz × 8 [88, p. 7]
RAM	7.7 GiB
Graphics	Gallium 0.4 on NVC0 [89]
PCI Board	United Electronic Industries PD2-AO-32-16 [90]

Table 3-1: Computer Specifications

Unless otherwise specified, this computer, located in the HRIL, was used to run the software for every simulation and the control software for every physical experiment presented in this thesis.

3.4 MAGLEV

The maglev system has been previously demonstrated in other experiments performed within the HRIL and has been shown to be capable of sub-millimeter precision when paired with the Optotrak motion sensor [42], [47], [51], [52]. In its present configuration, the maglev consists of 27 electromagnet coils controlled by individual current amplifiers.

3.4.1 COIL DESIGN

As described in [48], electromagnet coils within this system are built around hollow copper spools with an inner diameter of 12.5 mm, an outer diameter of 25 mm, and a height of 27 mm. The copper core provides for rapid heat dissipation, and the absence of an iron core simplifies the equations which govern the relationship between current provided and magnetic flux produced. Each spool is wound by hand with 1000

Spool Base

Finished
Electromagnet
Coil
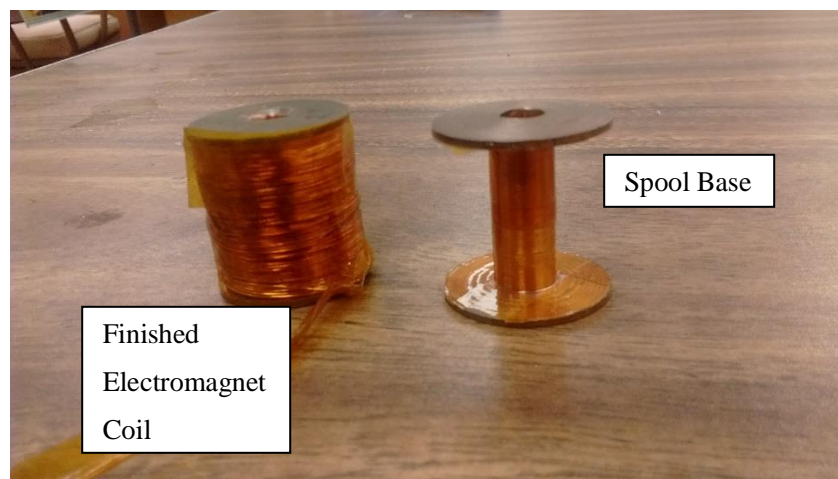
Figure 3-3: Electromagnet Coil and Spool

windings of copper wire with an insulating coating. The final resistance of the completed coil is approximately 8.8 Ω. Overheating may cause the insulated coating to melt, leading to a short in the coil and requiring a new coil to be wound. To extend the life of the coils and avoid the tedious process of winding new coils, we have taken to wrapping the spool base in insulating tape to prevent shorts at the edges of the spool and to somewhat soften the same edges, which have been found to be points where particular strain is put on the thin wire and shorts are likely to occur. Thus far, no coils wrapped in insulating tape have experienced shorts, so I recommend this practice to future researchers seeking to replicate this work using similar materials.

3.4.2 COIL LAYOUT

The maglev consists of 27 individual coils laid out in a hexagonal pattern comprised of 6 rows containing between 3 and 6 coils per row. The center-to-center distance between any two adjacent coils is 35 mm. Adjacent rows are offset from one another laterally by a distance of 17.5 mm, resulting in an alternating pattern of coils as shown in figure 3-4.

Figure 3-4: Maglev Coil Layout

Each coil is individually controlled by a Copley Controls [91] pulse-width modulation current amplifier. The amplifiers interface with the PC via a PCI board and thus the currents output to each coil can be independently controlled by the control software within a range of +/- 4 Amps.

3.4.3 THEORY OF OPERATION

The electromagnet coils described above will produce a magnetic field proportional to the current which is run through the coil. Thus, it is possible to exert an exact force and torque on a magnet positioned near the coils by running precisely calculated currents through each of the coils. Through previous work presented in

14

[48], [92], it was found that an accurate estimation of the forces generated by a single coil on a given magnet could be modelled numerically using the *Mathematica* [93] software with the *Radia* [94] software package, which was developed for the design of static magnet assemblies. By calculating the possible forces and torques generated at 1A current for different orientations and positions of the magnet relative to a coil, interpolation can be used to calculate a very accurate estimate of force and torque for any given position and orientation of the magnet. Furthermore, since the forces produced by the coils are proportional to the applied current, an accurate estimate of forces generated can be obtained for any possible level of current application within the operating range of the coils.

Given an array of N coils, [47], [92] present the equation for the forces and torques applied to a magnet positioned over the maglev as:

$$F = A \times I \tag{3-1}$$

Where F is a 6x1 force/torque vector, A is a 6xN transformation matrix, and I is the Nx1 vector containing the currents applied to each coil, as shown in equation 3.2 [47], [92].

$$
\begin{bmatrix} F_x \\ F_y \\ F_z \\ T_x \\ T_y \\ T_z \end{bmatrix}
=
\begin{bmatrix}
a_{fx}(r_1, z, \theta_1, \phi, \psi_1) & \cdots & a_{fx}(r_N, z, \theta_N, \phi, \psi_N) \\
a_{fy}(r_1, z, \theta_1, \phi, \psi_1) & \cdots & a_{fy}(r_N, z, \theta_N, \phi, \psi_N) \\
a_{fz}(r_1, z, \theta_1, \phi, \psi_1) & \cdots & a_{fz}(r_N, z, \theta_N, \phi, \psi_N) \\
a_{tx}(r_1, z, \theta_1, \phi, \psi_1) & \cdots & a_{tx}(r_N, z, \theta_N, \phi, \psi_N) \\
a_{ty}(r_1, z, \theta_1, \phi, \psi_1) & \cdots & a_{ty}(r_N, z, \theta_N, \phi, \psi_N) \\
a_{tz}(r_1, z, \theta_1, \phi, \psi_1) & \cdots & a_{tz}(r_N, z, \theta_N, \phi, \psi_N)
\end{bmatrix}
\begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ \vdots \\ I_N \end{bmatrix}
\tag{3-2}
$$

Where F_x, F_y, and F_z are the forces felt by the magnet in the x, y, and z directions and T_x, T_y, and T_z are the torques in the R_x, R_y, and R_z directions as shown in Figure 11. In matrix A, z is the vertical distance between the magnet and the top of the coil array. r_i is the planar distance between the vertical axes of the magnet and the ith coil. θ_i is the angle made between the line from the magnet to the ith coil and the horizontal plane. ϕ represents the tilt of the magnet. ψ represents the horizontal angle of

Figure 3-5: Maglev force and torque reference axes [90]

the line between the magnet and the ith coil [92].

A single radially symmetrical magnet, in this case a cylindrical magnet, can be controlled by this system with five degrees of freedom since the radial symmetry of the magnet prevents control over the magnet's roll angle. As the maglev contains 27 separate, individually controlled inputs to this 5-DOF system, the system as a whole is over-actuated and there are potentially infinite solutions for appropriate coil currents which will satisfy equation 3.2. The control software for the maglev thus finds the minimum "semi-inverse" function using the *dgels* function provided by the LAPACK linear algebra library [95], [96]. This numerical method finds the minimum current vector which will satisfy equation 3.2, and this set of currents will be sent to the current amplifiers for each coil. When this is done at speeds of 800 Hz with closed-loop PD feedback control on the position and orientation of the magnet, stable levitation is possible and has been presented in [47], [48], [51]. However, equation 3.2 and by extension the levitation control algorithm as a whole requires an accurate reckoning of the magnet's position and orientation in order to determine the necessary forces and torques required to move the magnet to the desired position or hold it in its current position. Previously, this has been done using the Optotrak motion tracking system [53], [54].

3.5 OPTOTRAK

In previous research performed using this maglev system, the Optotrak motion tracking system has been used to track the position and orientation of the magnet or multi-magnet structures being levitated [47], [48], [51], [92]. The Optotrak is a visual position and orientation tracking system that precisely tracks the motion of
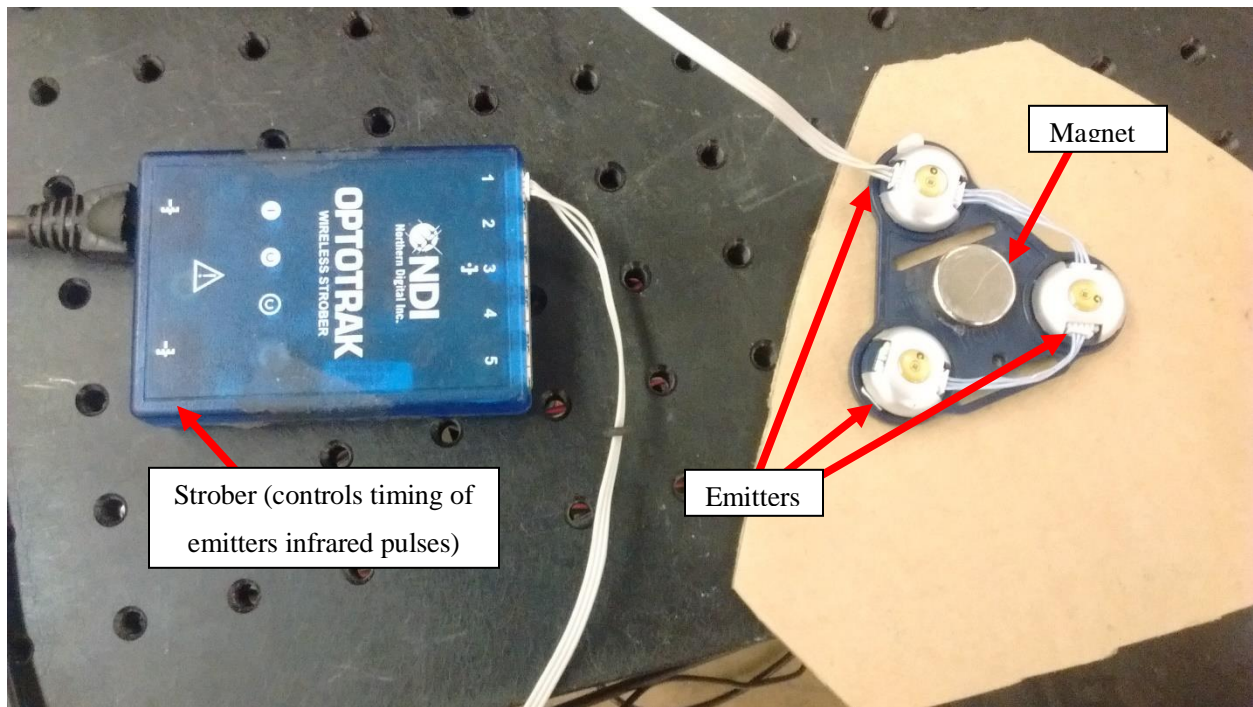
Figure 3-6: Cylindrical magnet affixed to three Optotrak infrared emitters

16

small infrared emitters, which emit pulses of infrared light that enable an update rate of over 800 Hz when used with three emitters, the minimum number required for 5- or 6-DOF tracking information. Figure 3-6 shows a cylindrical magnet affixed to three Optotrak infrared emitters.

The Optotrak has been shown to be accurate to within 0.01 mm and operates at an update rate sufficient for controlled levitation, over 800 Hz as shown in [92]. However, as a visual tracking system the Optotrak relies on having a direct line of sight to each emitter in order to operate correctly. If the visual line of sight between the Optotrak and even one of the three emitters is blocked, for example by a human hand placed above the emitters, the Optotrak is no longer able to estimate the position or orientation of the magnet. Thus, this approach is not applicable to the domain of wireless endoscopy, as a WCE inside a patient's digestive tract would be invisible to any visual tracking system. However, the Optotrak's proven track record of accuracy when used with this maglev system makes it a natural choice for use in validation. In several of the experiments presented in section 4, the Optotrak estimation was treated as an objectively correct measurement of the magnet's position and orientation for comparison with the localization estimates generated by the magnetic flux density sensor array. The Optotrak was used in a similar capacity for calibration of the sensor array, as will be described in section 4.2.1.

3.6 MAGNETIC FLUX SENSOR ARRAY

The maglev setup used in the HRIL has previously used a visual tracking system, the Optotrak, as has previously been established. However, in order to perform experiments combining magnetic localization with magnetic levitation, a magnetic sensor array was required. This board had to be designed and built before any experiments with magnetic localization could be performed.

3.6.1 SENSOR ARRAY FINAL DESIGN

The final design for the sensor array which was eventually built and used in experiments in the HRIL was based on the sensor array comprised of sixteen magnetic flux-density sensors (also known as digital magnetic compasses) as described by Hu. et al. [82]. The sensor array was built from sixteen Honeywell® HMC5983 Digital Compasses [97]. To ease in the construction of the circuit, the HMC5983L breakout board by Drotek® [98] was used to provide SPI communication [99] to the HMC5983 integrated circuit. This sensor has a maximum effective update rate of 160 Hz and an operational range of +/- 8 Gauss. Benefits of this sensor and breakout board which led to their selection include:

1) A low cost per sensor, $6.19 USD per sensor.
2) Robust response to overloads. When the sensor is overloaded by a reading of above 8 Gauss, the overloaded axis will provide a unique reading of 4096 (raw byte value) within it read register, which indicates that an overload has occurred. Furthermore, the sensor will automatically return

to normal operation once the flux density acting on that axis returns to the normal operating range of the sensor. This was a highly valuable feature since the sensor array needs to operate in an environment in which flux density changes rapidly and some overloads are inevitable.

3) A high communication speed. SPI communication with the Drotek breakout board can occur at a clock speed of up to 8 MHz, which allowed all sixteen sensors to be read over a single communication line without SPI communication time significantly slowing the update rate of the board.
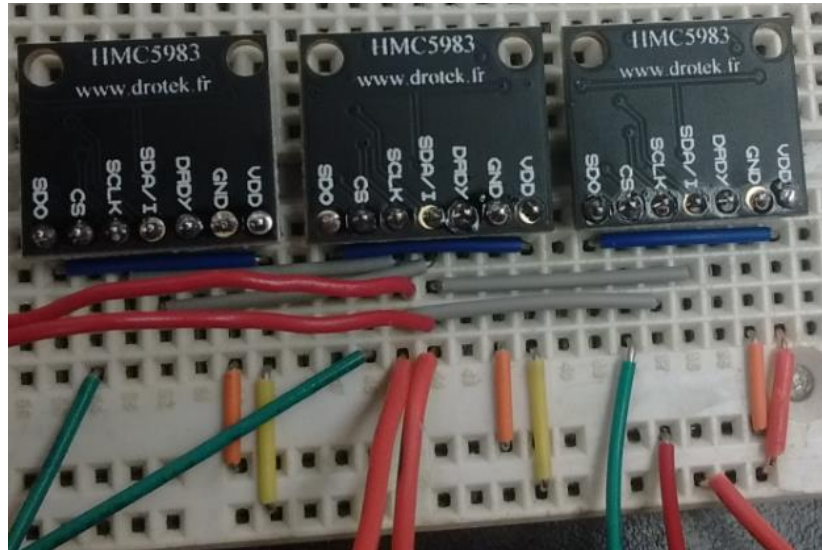
Figure 3-7: Three Drotek® breakout boards for the HMC5983 sensor, arranged on a breadboard for testing prior to the construction of the final sensor array.

The sensor array is controlled by an Arduino Due board [100], which in turn communicates with the controlling PC via a USB connection.
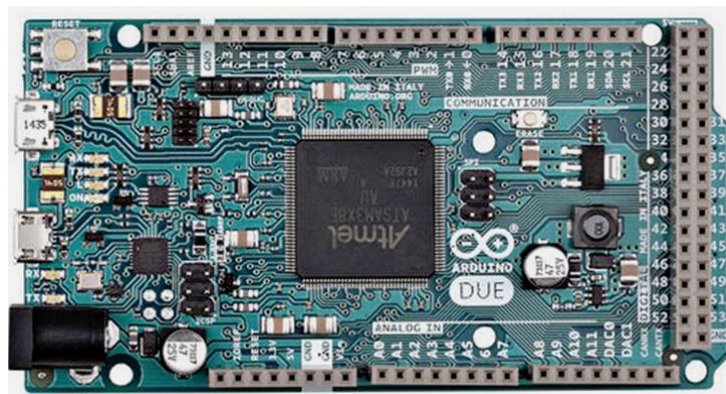
Figure 3-8: The Arduino Due board [100]

Wiring

This section describes wiring instructions for the HMC5983 sensor board which I created for this research. Each of the Drotek breakout boards can be set to communicate in either SPI mode or I2C mode. Since this project uses SPI communications, the first step is to solder the I2C/SPI jumper for SPI mode on each of the Drotek boards. Once the communications jumper has been soldered into SPI mode, the following connections are made for each of the pins on each breakout board:

Table 3-2: Connection guide for Drotek® breakout board

VDD	3.3V source, single bus for all sensors
GND	Ground, single bus for all sensors. Sensor array is grounded to a back-plate.
DRDY	Individual digital input on Arduino for each sensor.
SDA/I	Wire to Arduino MOSI, single bus for all sensors
SCLK	Wire to Arduino CLK, single bus for all sensors
CS	Individual digital output on Arduino for each sensor
SDO	Wire to Arduino MISO, single bus for all sensors.

These connections are achieved by soldering slots of seven male-to-female connectors to a prototyping board, described below, thus creating sixteen simple slots into which breakout boards can be inserted or removed. This feature proved invaluable during development, as sensors could easily be added or removed, tested individually, or swapped with other sensors to check whether a problem was with the sensor itself or with the connecting circuitry, and easily replaced in the event of a failed or broken sensor. The VDD, GND, SDA/I, SCLK, and SDO connections for each sensor connect to busses with a single connection to the Arduino Due each, as described in Table 3-2. However, the CS and DRDY pins for each sensor have unique connections, which are detailed in Table 3-3.

Table 3-3: Pin mapping (digital IO on Arduino Due)

Sensor	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CS	22	26	30	34	23	27	31	35	39	43	47	51	38	42	46	50
DRDY	24	28	32	36	25	29	33	37	41	45	49	53	40	44	48	52

The sixteen sensors are laid out in a regularly-spaced 4 x 4 grid, with each sensor separated by a distance of 38 mm in both the X and Y direction. The layout and numbering of the sensors in the array is shown in Table 3-4. A photograph showing the sensor array in this configuration can be seen in Figure 3-9.

Table 3-4: Sensor map: physical layout of sensors in array.

0	4	8	12
1	5	9	13
2	6	10	14
3	7	11	15
Arduino Due			
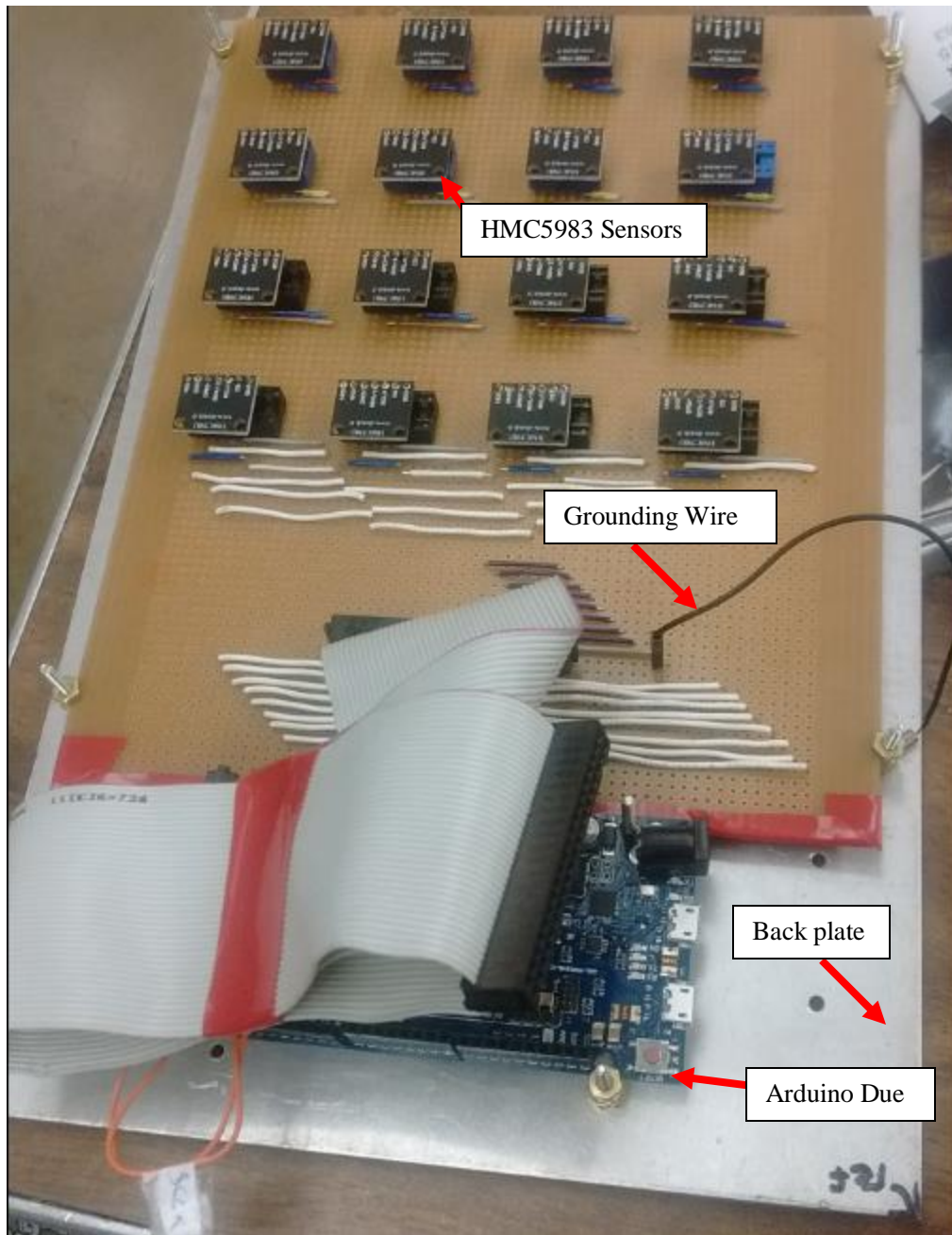
Figure 3-9: Sensor Array; final design

The sensor array was constructed using a prototyping board, designed with columns of connected conductive material running along the back of the board. Regularly spaced holes along these columns allow for the precise placement of components, with connections soldered to the back of the board as shown in Figure 3-10. Placement and soldering of all components was done by hand.
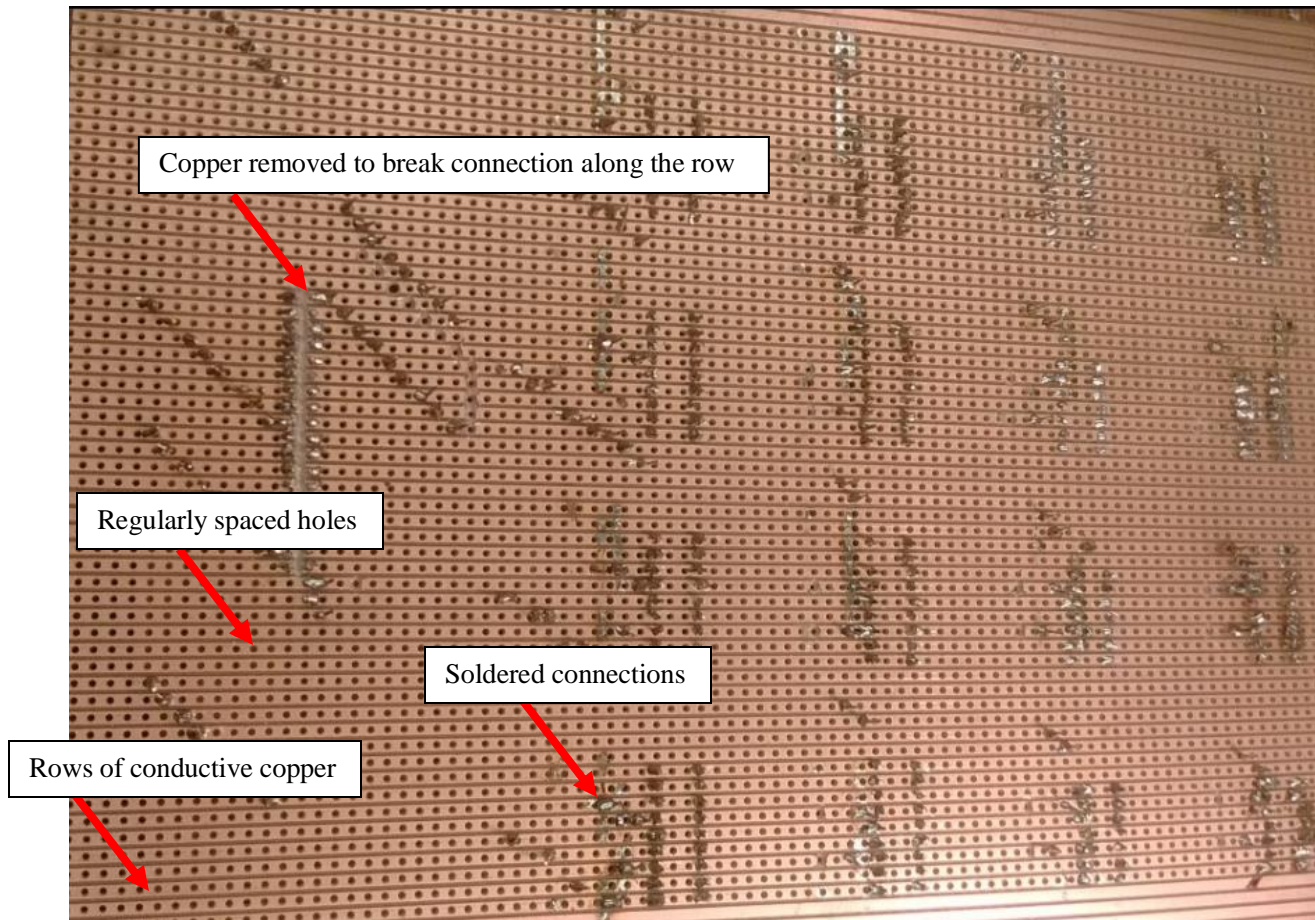
Figure 3-10: Connections soldered along the back of the prototyping board

Operation

The sensors can be run in either continuous or single-read mode. The mode register on the breakout board can be set to 0 for continuous and to 1 for single-read mode. The DRDY pin is remains high whenever the chip can be read, and is low only while data is being written to the chip. Therefore, reading a chip whenever the DRDY is high may result in duplicate reads of the same data. Waiting for the DRDY to toggle (go low, then high) guarantees that new data has been written.

In continuous mode, the sensors refresh continuously at 220hz, which is the highest possible update rate for this sensor. Unfortunately, problems with synchronization ultimately led to the decision to run the sensors only in single-read mode. Single-read mode allows for more direct control over the timing of the sensors read sequence. Setting the chip to single read mode will cause the chip to make a reading, and then to

return to "idle" mode until ordered into single-read mode again. Thus, a reliable way to get fresh, synchronized data is to order all chips into single read mode, wait for the chips to idle, and then read the data off of the chips. This can be done at ~160hz, which is in agreement with the HMC5983 spec sheet. This is the mode of operation which is used in all experiments described in this thesis and was found to be far more reliable than any attempt to use the continuous mode setting.

Data

The HMC5983 chips have 6 data registers, 2 bytes for each dimension. These are arranged as follows:

X MSB, X LSB, Z MSB, Z LSB, Y MSB, Y LSB

where MSB is "Most significant Byte" and LSB is "Least significant Byte." These two bytes store a 16-bit short integer in 2's compliment form. The range of values should be between [-2048, 2047]. A value of 4096 represents an overload error for the chip.

Arduino Program

The Arduino program is designed to interface smoothly with a user or software program. It will initialize with a greetings message which gives basic operational instructions. The program expects to receive numerical commands and will give responses over USB. The Native USB port on the Arduino Due was used in order to achieve the maximum communication speed for sending packets of data containing the sensor readings for all 48 sensor axes (three axes each, X, Y, and Z, for sixteen sensors) to the PC.

Sending a value of "0" through "15" to the Arduino will cause the Arduino to read a single sensor, corresponding to the value sent, and respond with the value of the sensor reading in a human-readable format.

Sending a "16" will cause the Arduino to begin a timing test which tests various modes of operation and displays times and speeds.

Sending a "17" will cause the Arduino to send a single 107-byte data snapshot representing all 16 sensors. The format of this data is:

The characters "START"
The values {xM xL yM yL zM zL} for all 16 sensors, in order
The characters "STOP\n"
Character 0, a null terminator

22

Where xM is the MSB of the X register, xL is the LSB of the x register, and yL, yM, zL, and zM are the same respective values for the Y and Z registers.

Sending an "18" will cause the Arduino to enter "continuous data mode," in which it will output a data stream of lines formatted as above, continuously and as quickly as possible. The only way to get the Arduino out of this state once it is in it is to reset the board (either press the reset button or unplug it and plug it back in). In this mode, the sensors are placed into single-read mode and a single data snapshot is compiled as described for the "17" command. As soon as data has been read from a sensor, the sensor is placed back into single-read mode, prompting it to take a new reading. Once all sixteen sensors have finished reading new data, that data is read by the Arduino and sent to the PC and the process repeats in perpetuity, resulting in data snapshots being sent continuously at a rate of 160 Hz, each snapshot containing unique, updated data for each of the sixteen sensors.

The sensor coordinates and the board coordinates are not the same, and the data taken from each sensor must be modified by reversing the sign on the Y and Z axes in order to match the sensor coordinates to the board coordinates. In order to conform the axes of the sensors with the axes of the sensor array, the following transformations are used:

$$
\begin{aligned}
X_{board\ coordinates} &= X_{sensor\ coordinates} \\
Y_{board\ coordinates} &= -Y_{sensor\ coordinates} \\
Z_{board\ coordinates} &= -Z_{sensor\ coordinates}
\end{aligned}
\qquad \textbf{(3-3)}
$$

3.6.2   REJECTED DESIGNS

This final design of the board came about only after two other designs were attempted and rejected. To lend insight into the design process and important considerations for any future sensor boards in this style, the rejected designs will be briefly discussed here.

Original Design

The original design of the sensor array was built with a similar sensor, the HMC5883L, a predecessor of the HMC5983 [101]. Despite being substantially more expensive, this sensor was roughly equivalent to the HMC5983 in its technical capabilities. However, cost was not the primary impetus for the design of the new sensor array. In this original sensor array, breakout boards were not removable, but rather were soldered directly into the board. Problems with a single sensor could permeate throughout the entire board and invalidate the data generated. Because of the static nature of the board, there was no way to check whether problems which arose were caused by the sensors themselves or by their surrounding wiring. There was no

way to run tests on individual sensors and removing sensors from the board was an arduous process requiring solder to be removed with a soldering iron and a wick of braided copper. Once a defective or suspected to be defective sensor was removed, a new sensor would need to be soldered in. Furthermore, the sensor array did not connect to an Arduino board and its behavior was far more difficult to modify or debug.

The final design of the sensor array was designed in response to these issues. Instead of soldering in sensors directly, mountings were soldered in that allow sensors to be quickly swapped in and out, allowing sensors to be tested individually or in groups and enabling a wider range of debugging techniques, such as switching the positions of two sensors. Furthermore, the use of the Arduino enabled code to be quickly changed and a simple user interface to be implemented for even more diagnostic and debugging capabilities. This allowed different options, such as experimenting with the sensors in continuous and single-read mode and variations on the format of the data output, to be attempted with minimal or no modification to the hardware of the sensor board and only minimal changes to the software running on the Arduino.

Ultimately, now that the design has been finalized, a more static approach in which components are secured in place more precisely and a more minimalist control software is used to enable communications may be more efficient. However, the process of creating a working board required months of revisions and redesigns before all the problems were adequately understood and dealt with, and the modular approach of the final board reflects the need for experimentation, debugging, and trial-and-error associated with the design of a working sensor array.

The MMC3416PJ-B Prototyping Board

While designing the new sensor array, in addition to the HMC5983 sensor the MMC3416xPJ sensor [102] utilizing the MMC3416PJ-B prototyping (breakout) board [103] was also considered. The MMC3416xPJ sensor offered several obvious advantages over the HMC5983. It has a maximum operational range of +/- 16 Gauss, twice the range of the HMC5983's +/- 8 Gauss. Additionally, the MMC3416xPJ has a maximum refresh rate of up to 800 Hz, according to the technical specifications given for the sensor [102]. However, two problems arose when working with this sensor that caused it to be rejected in favor of the HMC5983.

The first problem to become apparent was that the MMC3416xPJ sensors did not deal with overloads robustly. With their increased operational range, overloads were less frequent but were nonetheless still inevitable. When the HMC5983 experiences an overload on one of its axes, it communicates that an overload has occurred by filling the registers for that axis with a specific byte value, and normal operation resumes once the overload has been cleared. By contrast, when an overload occurs on one of the MMC3416xPJ axes, the sensor stops working entirely and will not record any further new data. There is no clear indication that this has occurred, and although it is relatively obvious to a human observer, there is no convenient way to test for an overload digitally. The only way to restore the operation of the sensor after an overload is to re-initialize the

24

sensor. The possibility of introducing a monitoring program to attempt to accurately guess when an overhead had occurred and reset the chip accordingly proved to require substantial additional communication with the chip, which was found to introduce unacceptable overhead to the communication.

This communications overhead was so arduous because the MMC3416xPJ has an extremely slow communication speed, the second major problem which led to its rejection. The MMC3416xPJ was clearly not designed to be used in groups; and although a single sensor could update at 800 Hz, the communication from the chip could barely keep up with this update rate, with am I2C clock speed of only 400 kHz [102], in contrast to the 8MHz SPI communication of the HMC5983 [97]. At this update rate, communication speed becomes the gating factor on the update rate of the sensor array with even two sensors on the same communications bus.

As further evidence that the MMC3416xPJ was not designed to be used in groups, the HMC5983 sensor has a Chip Select (CS) pin which allows multiple sensors to be used on the same SPI data line, controlling which sensor is outputting data at any particular point in time by toggling the CS pin high or low. The MMC3416xPJ, by contrast, has no equivalent function, and had to be used in conjunction with a I2C multiplexer produced by Adafruit [104]. Switching with the multiplexer added additional time to the process of reading a group of MMC3416xPJ sensors. Ultimately, a prototype circuit was constructed on a breadboard, consisting of two Adafruit multiplexers and seven MMC3416xPJ sensors, as shown in Figure 3-11.
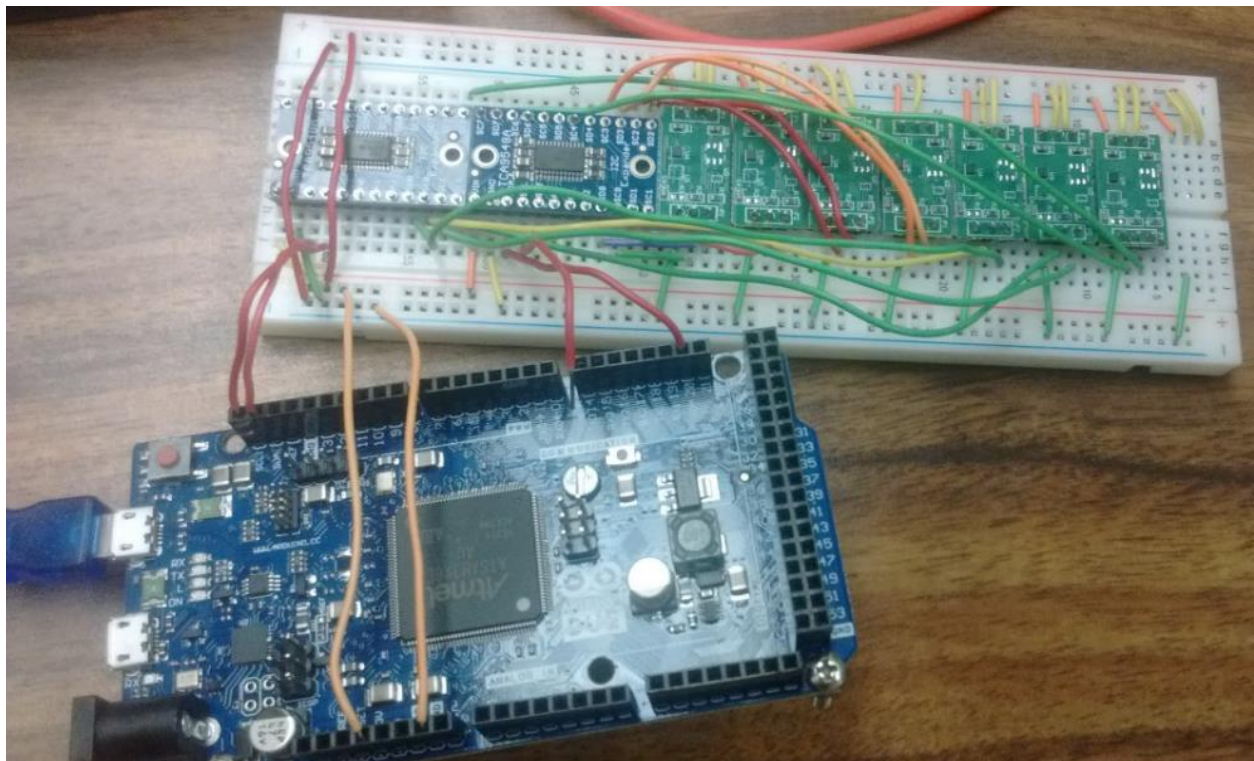
Figure 3-11: MMC3416xPJ Sensor Array Prototype

Between the slow communication speed and the delay added by the multiplexer, this seven-sensor circuit ran at an update rate of 83 Hz[1], significantly slower than the HMC5983 sensor array. Furthermore, from experimentation it seemed likely that this update rate would scale linearly, leading to an estimated final update rate of only 35 Hz. While it may have been possible to speed up this update rate with an alternative design, for example, by reserving a separate communications line for each individual sensor, this would have added considerable overhead to the electrical design. Given the complexity of using multiple MMC3416xPJ in conjunction with one another, coupled with my own lack of expertise in circuit design and a reluctance to tackle such a project, and given the sensor's poor response to overloads, it was ultimately decided that designing and building a sixteen-sensor array based on the MMC3416xPJ was not a valuable use of time, and the sensor was rejected in favor of the HMC5983.

3.7 LOCALIZATION ALGORITHMS

Once constructed, the sensor array is capable of generating sensor readings at each of its sixteen sensors and communicating those readings to software running on a PC at a rate of 160 Hz. The software which receives this data must then apply a localization algorithm to the data in order to determine the location and orientation of the magnet which generated these sensor readings.

3.7.1 LOCALIZATION WITHOUT ACTUATION

As discussed in the Prior Works section, Hu et al. have presented a non-linear numerical algorithm for finding the location of a magnet given a set of flux density sensor readings [82]–[85]. The research presented in this thesis is based on this prior work by Hu et al.

The localization algorithm uses a non-linear numerical least-squares minimization method known as the Levenberg-Marquardt optimization algorithm, as implemented in the LAPACK function *lmder1* [95]. This algorithm takes as an input a function for calculating an array of error values, and a list of parameters to that function. The Levenberg-Marquardt algorithm then attempts numerous variations on the parameters of the function, eventually converging on a set of parameters that are estimated to minimize the error produced by the error function.

In the case of the localizer, the parameters to the error function are the position and orientation of the magnet, and the error function itself works by calculating the expected value in each of the sixteen sensors for the proposed magnet position and orientation, and subtracting this value from the actual sensor readings. Variations on the position and orientation of the magnet are tried, and the error calculated for each, until the algorithm reaches an estimated position and orientation that minimizes the squared-error between expected and actual values in each of the 48 sensor axes.

---

[1] Based on data gathered on 11 August 2016 by BJ Tix

$$Error = Expected\ Sensor\ Reading - Actual\ Sensor\ Reading \qquad \text{(3-4)}$$

The expected value for each sensor is computed analytically using the magnetic dipole equation, presented in equation (3-5):

$$\mathbf{B}_l = B_{lx}\mathbf{i} + B_{ly}\mathbf{j} + B_{lz}\mathbf{k}$$
$$= \frac{\mu_r \mu_0 M_T}{4\pi}\left(\frac{3(\mathbf{H}_0 \cdot \mathbf{P})\mathbf{P}}{R^5} - \frac{\mathbf{H}_0}{R^3}\right) \quad (l=1,2,\ ...\ N) \qquad [83] \qquad \text{(3-5)}$$

In this equation, B_l is the magnetic flux density at sensor l, B_{lx}, B_{ly}, and B_{lz} are the x, y, and z components of flux density at the location of the sensor, respectively, and $\mathbf{i}, \mathbf{j}$, and $\mathbf{k}$ are unit vectors in the x, y, and z directions. μ_r represents the relative permeability of the medium (in air, $\mu_r = 1.0$), and μ_0 represents the magnetic permeability of air ($\mu_0 = 4\pi\ x\ 10^{-7}T\ m\ /\ A$). M_T is a constant representing the strength of the magnet. $\mathbf{H}_0$ is a unit vector describing the direction of the magnet, $\mathbf{P}$ is the vector position of the lth sensor relative to the center of the magnet, and R is the magnitude of $\mathbf{P}$. [83]

To summarize, the magnetic dipole equation can calculate the expected flux density at any given sensor so long as the location and orientation of the magnet and sensor are known, as well as the strength of the magnet and the magnetic permeability of the medium. The Levenberg-Marquardt equation is a numerical optimization method which attempts different possible magnet positions and orientations, checks the error of the expected values as calculated by the magnetic dipole equation when compared to the actual sensor readings, and through multiple iterations converges on an estimate of the magnet's position and orientation which minimizes the error between the actual and expected values.

3.7.2   COMBINED LEVITATION AND LOCALIZATION

The key contribution of this work is to expand the localization algorithm provided by Hu et al. to make it compatible with use with a magnetic levitation system. The algorithm as provided is not sufficient, since as soon as the maglev is activated, it will produce its own magnetic field which is not accounted for by the dipole equation, which will cause the localization algorithm to fail. Similar results are shown in [42], in which the localization of the GaussSense board failed as soon as magnetic actuation was applied.

As I will demonstrate in this work, it is possible to account for the magnetic field produced by the maglev within a localization algorithm working off of a similar principle to the algorithm already described. In order to successfully account for the magnetic field of the maglev, we require a model of the magnetic flux density generated by the maglev at each of the sensors. Since the current to each maglev coil is precisely controlled by the levitation control loop, and the magnetic forces produced by each coil are proportional to the current in the coil, the magnetic flux density for each coil can be precisely calculated based on the current in

the coil at any given point in time. While it would be possible to estimate the magnetic flux density based on the dipole equation described in equation (3-5), even greater accuracy can be achieved by using a numerical model generated by the previously discussed Radia software package [94]. The numerical model produced for a single coil contained estimates of flux density in 1mm increments radiating out from the center of the coil. Making use of the rotational symmetry of the coil, only two dimensions are required, the horizontal distance from the coil and the vertical distance from the coil. Values in the numerical model are proportional to the current running through the coil, as the field produced by the coil is proportional to the current. Values are recorded in the model as the flux density produced at each location by an applied current of 1 Amp.

The magnetic field generated by the maglev is a simple superposition of the fields generated by every coil individually. Similarly, the total magnetic flux density detected by the sensors is a superposition of the flux density generated by the maglev and the flux density generated by the permanent magnet being levitated. Using this model, it is possible to very accurately estimate the flux density produced by the maglev by summing the individual flux densities generated by each coil, according to the numerical model which has just been described. The total expected flux density sensed by each sensor can then be calculated as the sum of the flux density produced by the maglev on that sensor, and the flux density produced by the magnet, as calculated by equation (3-5).

Total Flux Density = Flux Density from Magnet + Flux Density from Maglev   **(3-6)**

By using this estimate of the total flux density at each sensor, Equation (3-4) can be solved accurately even when the maglev is running. Thus, so long as the localization software is provided with an accurate description of the state of the maglev and an accurate model of the magnetic field produced by the maglev in its current state, then the Levenberg-Marquardt least-squares optimizer can be used to converge on an accurate estimate of the position and orientation of a permanent magnet acted upon by a multi-coil magnetic levitation system.

3.8   CONTROL SOFTWARE

3.8.1   OVERVIEW

The entire system is controlled by an original software system written specifically for this research. The software was written in the C++ programming language [105] using the Eclipse integrated development environment (IDE) [106]. Older software used to control the maglev were written in C [107] without the aid of an IDE. However, the added complexity of the current software when compared to previous control software run in the HRIL justified the use of C++ and the Eclipse IDE. The full code base for this project consists of 58 code (.cpp) and header (.h) files and is too extensive to reproduce in full here. Code is available in electronic form upon request.

The software can be broken down into five primary components:

1) The main startup and initialization function.

2) One or more localizers.

3) The sensor array driver.

4) The maglev driver.

5) The graphical user interface (GUI).

The main startup function

The software is designed to be run from a Unix terminal and begins with a series of text-based prompts which determine what the program will do during this session. The user will be prompted with a series of yes/no questions which require a response of 'y' for yes and 'n' for no. Options include gathering new data for calibration, recalibrating the sensors based on existing data, or calculating the average error based on existing calibration data and calibration values. Calibration will be discussed in further detail in section 4.2.1. Other options include showing a replay of previously recorded data and running the coil calibration test, as described in section 4.2.2. Depending on the options selected by the user, the startup function will eventually reach a point where no further input is required and will begin a thread of operation for the GUI, opening a new graphical window to display real-time or replay data from the sensors and maglev depending on the options selected. The localizers, sensor array, and maglev will also be initialized at this point. The full decision tree is outlined below in

Table 3-5: Software Startup Decision Tree

ID	Prompt or Action	Input	Input	Input
0	"Do you want to run the single-coil calibration test? (y/n)"	Y:1	N:2	
1	The single coil calibration test is run as described in section 0	2		
2	"Do you want to record new calibration data? (y/n)"	Y:3	N:4	
3	New calibration data is recorded by the Optotrak as described in section 4.2.1.	4		
4	"Do you want to recalibrate based on the existing raw data? (y/n)"	Y:5	N:9	
5	"Do you want to use the Velmex data (v), the Optotrak data (o), or the maglev data (m)?"	V:6	O:7	M:8
6	Calibration is run based on data gathered from the Velmex positioning system [108] as described in section 4.2.1.	11		
7	Calibration is run based on data gathered from the Optotrak as described in section 4.2.1.	12		

8	Calibration is run based on data gathered during levitation as described in section 4.2.1.	13		
9	"Do you want to calculate the error and export to error.csv? (y/n)"	Y:10	N:14	
10	"Do you want to use the Velmex data (v), the Optotrak data (o), or the maglev data (m)?"	V:11	O:12	M:13
11	Error is calculated based on data gathered from the Velmex positioning system as described in section 4.2.1.	14		
12	Error is calculated based on data gathered from the Optotrak as described in section 4.2.1.	14		
13	Error is calculated based on data gathered during levitation as described in section 4.2.1.	14		
14	"Do you want to see a replay of the calibration data? (y/n)"	Y:15	N:18	
15	"Do you show simulated replay data? (y: simulated sensor data; n: real sensor data)"	Y:16	N:17	
16	The GUI will be initialized. A replay of the most recently recorded calibration data will be shown on the GUI, showing the motion of the magnet as recorded in the data along with the *simulated* sensor readings generated based on the recorded magnet positions.			
17	The GUI will be initialized. A replay of the most recently recorded calibration data will be shown on the GUI, showing the motion of the magnet as recorded in the data along with the *actual* sensor readings which were recorded when the calibration data was recorded.			
18	"Do you want to run the maglev in alternation mode? (y/n)"	Y:19	N:20	
19	The GUI will be initialized along with the maglev. The maglev will run in "alternation mode" as described in section 0 using the sensor array for localization.			
20	The GUI will be initialized along with the Optotrak and maglev. The maglev will run, moving the magnet along a prescribed circular path while data is recorded from both the Optotrak and the sensor array. Once the path is complete, the maglev will automatically deactivate, leaving the GUI running, showing a real-time display of the readings from the sensor array.			

The Localizer

In the context of this system, a localizer is a software object that generates an estimate of the position and orientation of a magnet. There are several possible ways to do this, the most notable three of which, in the context of this research, are;

ILocalizer

Optotrak

LM_Localizer

LM_LocalizerWithMaglev

Figure 3-12: Localizer virtual parent class inheritance tree.

1) Using the Optotrak.
2) Using the original localization algorithm designed to work in the absence of magnetic actuation.
3) Using the new localization algorithm, designed to work in conjunction with the maglev.

As shown in Figure 3-12, the localizer has been designed as implementations of a shared interface. Since C++ does not natively support an explicit "interface" mechanism, *ILocalizer* is implemented as a virtual parent class containing functions for beginning a thread of operation, running an update function in an operational loop in this thread of operation, and tracking statistics on iterations run, time since startup, speed in Hz, and other measures. The most important feature the *ILocalizer* provides is access to estimated position and orientation data about the magnet it is tracking, which can then be used by other parts of the program.

However, *ILocalizer*'s update function is empty, and it has no way to estimate the position and orientation of the magnet. That functionality is left to the child classes. The *Optotrak* child class receives data from the Optotrak hardware during its update function, and converts that data into the vector-representation of the magnet used by the rest of the program, making new data available whenever new data is received from the Optotrak. *LM_Localizer* is an implementation of the original localization algorithm as presented by Hu et al. [83], whereas *LM_LocalizerWithMaglev* implements the modified version of this algorithm intended for use with the maglev. Both of these children classes rely on receiving sensor data from an *ISensorArray* object to use as the inputs to the localization algorithm.

The Sensor Array Driver

The sensor array has been implemented in software using a similar scheme to that used by the localizer. A virtual parent class, *ISensorArray*, contains some of the common functionalities and definitions for public functions, while child class implementations determine the actual sensor data itself. As shown in Figure 3-13, there are three notable implementations of *ISensorArray*:

31
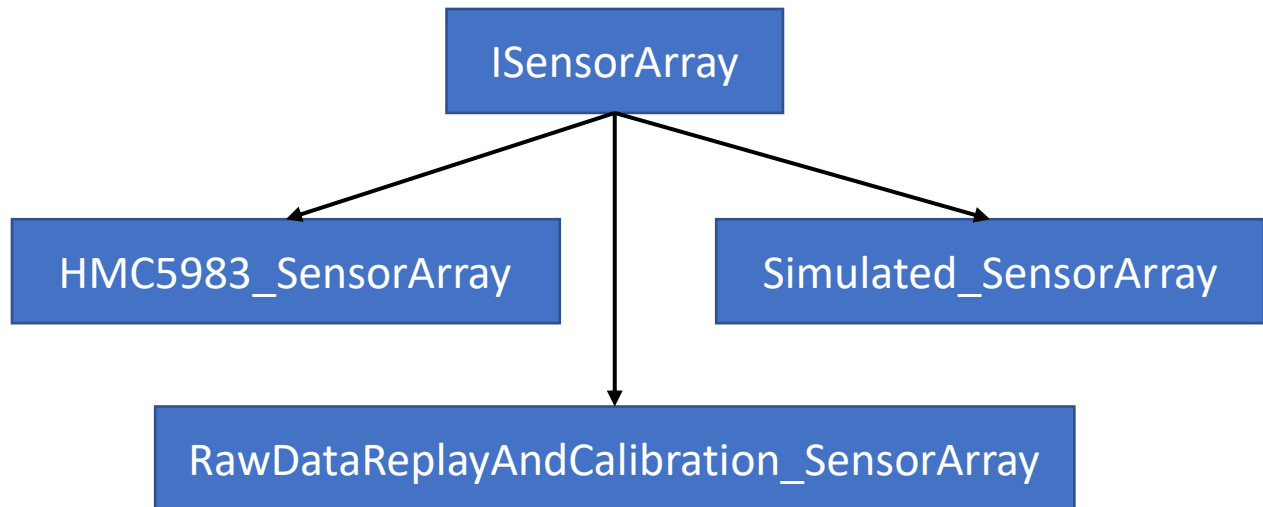
Figure 3-13: Sensor array inheritance tree

1) The *HMC5983_SensorArray* child class communicates with the sensor array Arduino board via a USB connection. During the initialization of this class, communication is established with the Arduino and the Arduino is placed into "continuous read mode" as described in section 3.6.1. Once initialized, the *HMC5983_SensorArray* receives real-time updates of the readings in each of the sixteen sensors of the sensor array.

2) The *Simulated_SensorArray* generates estimated sensor readings as described in section 3.7 if provided with a known magnet position and maglev data. To make the simulations more realistic, the *Simulated_SensorArray* has a property named *percentError*. Whenever estimated sensor readings are generated, a random error of up to *percentError* percentage of the estimated value is added or subtracted from the estimate, in order to simulate "noisy" and imperfect sensor data.

3) The *RawDataReplayAndCalibration_SensorArray* loads data recorded from previous experiments, including magnet localization data, sensor data, and maglev coil currents. The *RawDataReplayAndCalibration_SensorArray* can then simulate a replay of this saved data by incrementing through different data points at regular time intervals, thus simulating a repeat of the previously recorded sensor data.

Sensor arrays do not run their own threads of operation nor do they update on their own. Instead, they are coupled to an *ILocalizer* object which is used to control the update and timing. When an *ILocalizer* that depends on sensor input updates, it will command the *ISensorArray* object to take a new sensor reading, which may be a new reading from the physical sensor array hardware, newly generated simulated data, or simply the next recorded data step from a recorded replay, depending on the type of *ISensorArray* that is being commanded to take a new sensor reading.

The Maglev Driver

The maglev driver is based directly off of the C code which was used to run the maglev in previous HRIL experiments [48], [92]. The specifics of the maglev code's operation were described in section 3.4.3. A closed-loop PD controller is used to control the magnet's position and orientation based on input from an *ILocalizer* object. In previous experiments in the HRIL, the localizer has always been the Optotrak, but I have conducted experiments using both the Optotrak and the sensor array and localization algorithm, as will be described in detail in section 4.

The maglev driver does not run in its own thread. Instead, a publisher-subscriber model was implemented between the maglev and the *ILocalizer* object. When an *ILocalizer* with a maglev as a subscriber has a new estimate of the magnet's position and orientation, this information is sent to the maglev, triggering an update of the PD control loop using the new magnet position and orientation data as closed-loop feedback.

The Graphical User Interface

Once the main initialization function has reached the conclusion of the user input tree described in Table 3-5, a new window is opened containing a graphical user interface implemented in OpenGL [109]. This graphical user interface includes a three-dimensional representation of the values of each sensor reading and each maglev coil current value, displayed as three two-dimensional projections, the XY projection (top view), the XZ projection (front view) and the YZ projection (side view).

Within each projection, each sensor is displayed a small blue dot with a red line pointing away from its center, indicating the direction and magnitude of the magnetic flux density being reported by that sensor. If the magnitude of the flux density is small, a blue line will indicate the direction of the flux density but not the magnitude.

The maglev coils are represented by colored squares in each projection, their color varying based on the current being commanded to the coil. A faded grey square indicates that no current is being sent to the coil. The square will turn red for positive currents and blue for negative current values. Bold colors indicate a high absolute current value, while faded colors represent a low absolute current value.

The estimate of the magnet position and orientation is displayed as a colored dot representing the position of the magnet with a short line radiating outward from the dot to indicate the unit vector of the magnet's direction. It is possible to display multiple magnet estimates from different *ILocalizer* objects simultaneously, and to color-code each one to clearly indicate which magnet estimate is from which localizer.

Additionally, information about the current session including the time since the program was started, the number of iterations run, and the update rate of both the localizer and the GUI threads are displayed in the top left corner of the window. A screenshot of the GUI taken during magnetic levitation is show in Figure 3-14.

33
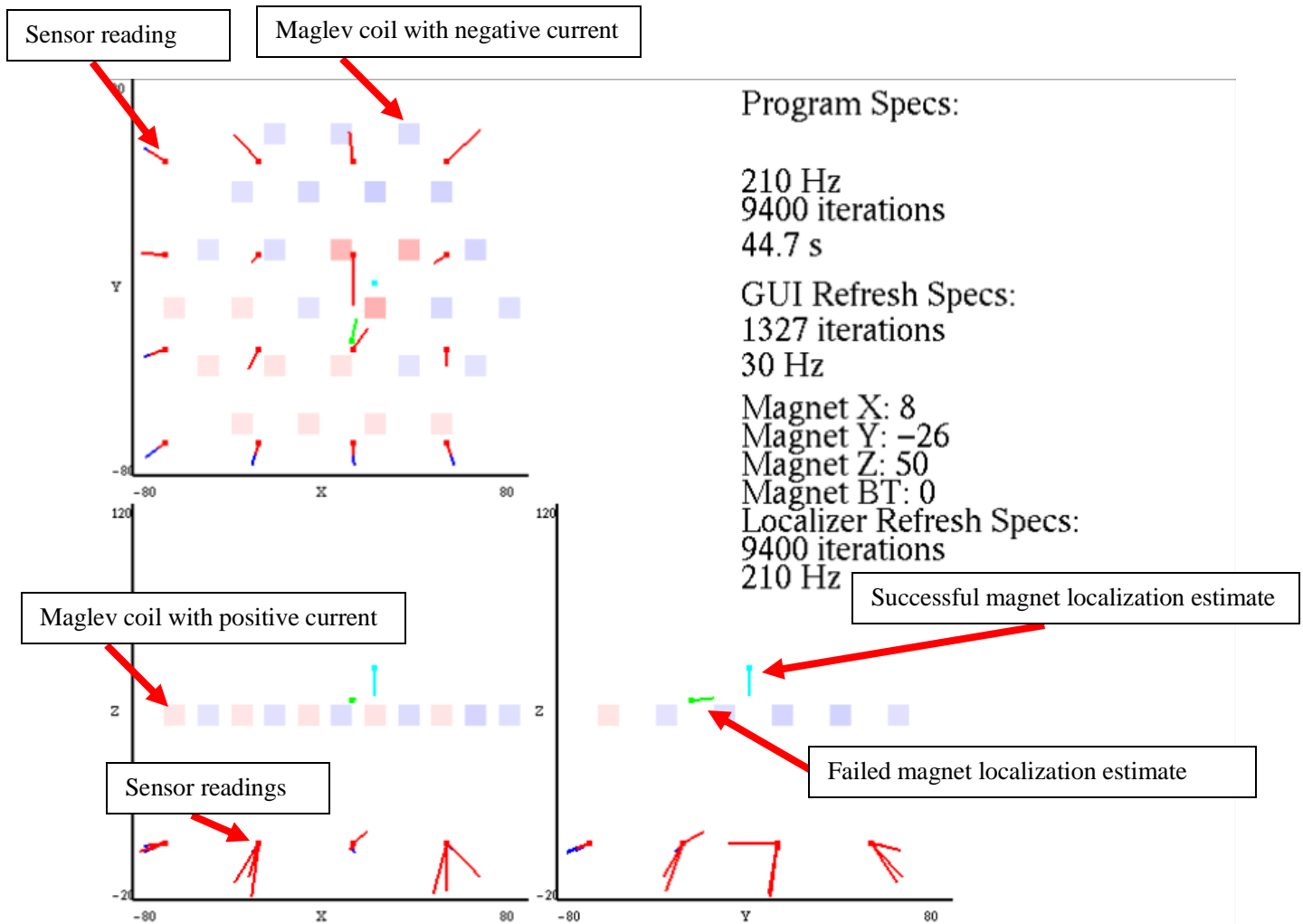
Figure 3-14: GUI Screenshot with annotations

Summary: Threads of Operation and Update Sequence

In summary, the software starts up by prompting the user through a series of text-based questions in a computer terminal. Once the parameters of the program have been set through this user-input decision tree, the program will start two or more threads of operation: one for the GUI, and one for each localizer which will be running during the session. Each localizer will update continuously as quickly as possible, but their speed is constrained by their particular means of localization. In the case of the Optotrak, the update function will be completed whenever new data is available from the Optotrak, typically at a rate of approximately 800 Hz. In the case of the localization algorithms with or without the maglev, the constraining factor is the speed of reading the sensor array and the speed of executing the Levenberg-Marquardt minimization algorithm. During each iteration of these localizers, the sensor array is read and the sensor data is used as input to the localization algorithm, and an estimate of the magnet's position and orientation is produced. Once this estimate is available, the localizer instructs the maglev to update its control loop using the new magnet estimate as closed-loop feedback.

34

4 EXPERIMENTS AND RESULTS

4.1 SIMULATED RESULTS

4.1.1 SIMULATION DESIGN AND IMPLEMENTATION

In order to validate the newly proposed localization algorithm, it made sense to begin by testing the algorithm in a simulated environment. If tested only with physical hardware, it would be extremely difficult to tell whether failures which occurred were due to a flaw in the algorithm itself, a flaw in the implementation, or a flaw in the hardware. This problem is greatly simplified if the algorithm is first validated in a simulated environment in which hardware flaws are not a concern. Furthermore, knowing that the algorithm works properly in a simulated environment is useful before attempting implementation on new hardware, in order to aid in the debugging and development process for the hardware.

The simulation software was briefly discussed in section 3.8.2, but it will be discussed in greater detail in this section. The simulated sensor array, an implementation of the *ISensorArray* virtual parent class as has already been described, operates according to the following process:

1) The magnet position is assumed to be known beforehand and must be precisely defined. In each of the simulated experiments presented in this research, this was accomplished by using recorded replays of previous physical experiments. While recording the replay, the Optotrak records the position and orientation of the magnet, and these position and orientation readings are saved and later played back into the simulated array as the known magnet positions and orientations. The maglev coil currents are also recorded and fed into the simulated sensor array as input data.

2) Once the magnet position and orientation has been established and the coil currents are known, the sensor readings expected to be generated from these conditions are calculated according to Equation (3-6) for each of the three axes of all sixteen sensors in the array.

3) A random "noise" is added to each reading. This random value may take any value between zero and an absolute value not greater than a set percentage of the expected reading, as defined by the *percentError* property of the simulated sensor array. The noise value may be positive or negative and the random values generated are evenly distributed across their allowable range.

4) The localizer, which is an instance of either *LM_Localizer* or *LM_LocalizerWithMaglev* treats the values generated by the simulated sensor array as though they were real sensor readings. Since the localizer is only seeking sensor readings from an *ISensroArray* object, its function is identical whether that sensor array is real or simulated.

5) The localizer generates an estimate of the magnet position and orientation based on the simulated sensor readings. If everything is working correctly, the estimated position and orientation should be very close to the known position and orientation which were used to generate the simulated sensor readings. Furthermore, it should be expected that greater noise values will result in greater error between the known position and the estimate.

4.1.2   LOCALIZATION WITHOUT ACTUATION

To validate the software and the implementation of the original localization algorithm, simulated experiments were performed without input from the maglev. In these experiments, the magnet was moved by hand while connected to three emitters in view of the Optotrak (see Figure 3-6 in section 3.5). Ten thousand data points were recorded, including precise measures of the magnet position and orientation, at 2ms intervals. This data was then used as the input to generate simulated sensor data for use with an *LM_Localizer* object.[2] The localization algorithm was shown to work very well, with results shown in Table 4-1. These results demonstrate what is effectively a replication of the research presented by Hu et al. in [82], which was an important step before moving forward with experiments on expansions of this algorithm.

Table 4-1: Simulation Results with Original Algorithm

Simulated Accuracy: Localization without Maglev		
	Position Error (mm)	Orientation Error (radians)
No Noise	2.64E-08	3.87E-09
5% Noise	1.13373	0.0275007
10% Noise	2.2987	0.0561892

4.1.3   COMBINED LEVITATION AND LOCALIZATION

Once the original algorithm had been validated, a similar simulated experiment was performed using data captured during magnetic levitation. In order to capture the data for this experiment, the magnet was levitated by the maglev in a pre-programmed circular pattern, varying the height (Z direction) while simultaneously moving in a horizontal circle (X and Y directions). The position and orientation of the magnet as estimated by the Optotrak were recorded, as were the commanded coil currents for all 27 maglev coils.[3] This data was then used as the input for the simulation, and the localization results are shown in Table 4-2. The localization estimate is accurate to within less than 1mm for the 0% and 5% noise cases, and increases to just over 1mm for the 10% noise case. As expected, the accuracy decreases as the noise level increases.

---

[2] Data recorded on 24 January 2017 by BJ Tix
[3] Data recorded on 30 January 2017 by BJ Tix

Table 4-2: Simulation Results with Combined Localization and Levitation

	Simulated Accuracy: Localization with Maglev	
	Position Error (mm)	Orientation Error (radians)
No Noise	1.01E-07	3.99E-08
5% Noise	0.64406	0.0209799
10% Noise	1.28784	0.0424378

Sensitivity to Noise

As expected, the accuracy of the localization algorithm decreases as the noise (inaccuracy) of the sensors increases. This can be seen in Table 4-2, and can be shown even more dramatically in visual form in Figure 4-1, Figure 4-2, and Figure 4-3, which show visual representations of the position estimates at 0%, 5%, and 10% noise levels, respectively. The estimated can be clearly seen to spread out away from the actual value as the noise level increases. Based on these simulated results, it can be inferred that a physical system using this algorithm will be highly sensitive to any inaccuracy in the sensors or other elements of the hardware, which will lead to inaccurate sensor value estimates and thus reduce the accuracy of the localization as has been shown in this simulated experiment.
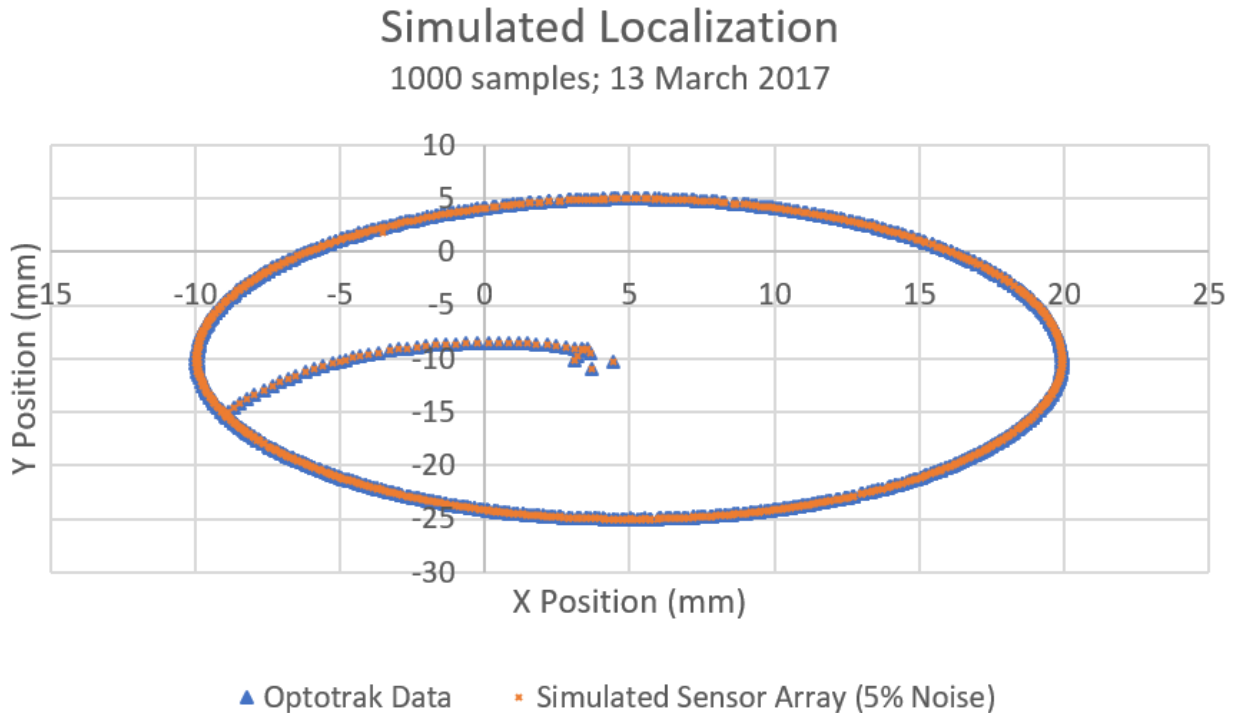
Figure 4-1: XY Projection of Simulated Localization of Position, 0% Noise

Simulated Localization with 5% random noise
1000 samples; 13 March 2017

▲ Optotrak Data   · Simulated Sensor Array (5% Noise)
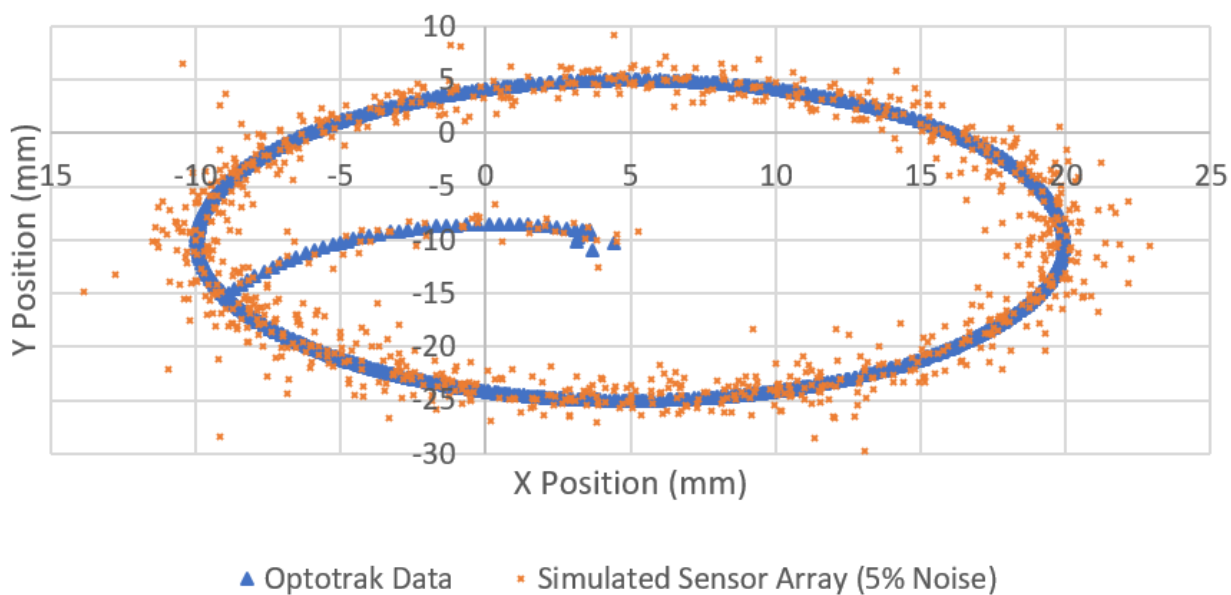
Figure 4-2: XY Projection of Simulated Localization of Position, 5% Noise

Simulated Localization with 10% random noise
1000 samples; 13 March 2017

▲ Optotrak Data   · Simulated Sensor Array (5% Noise)
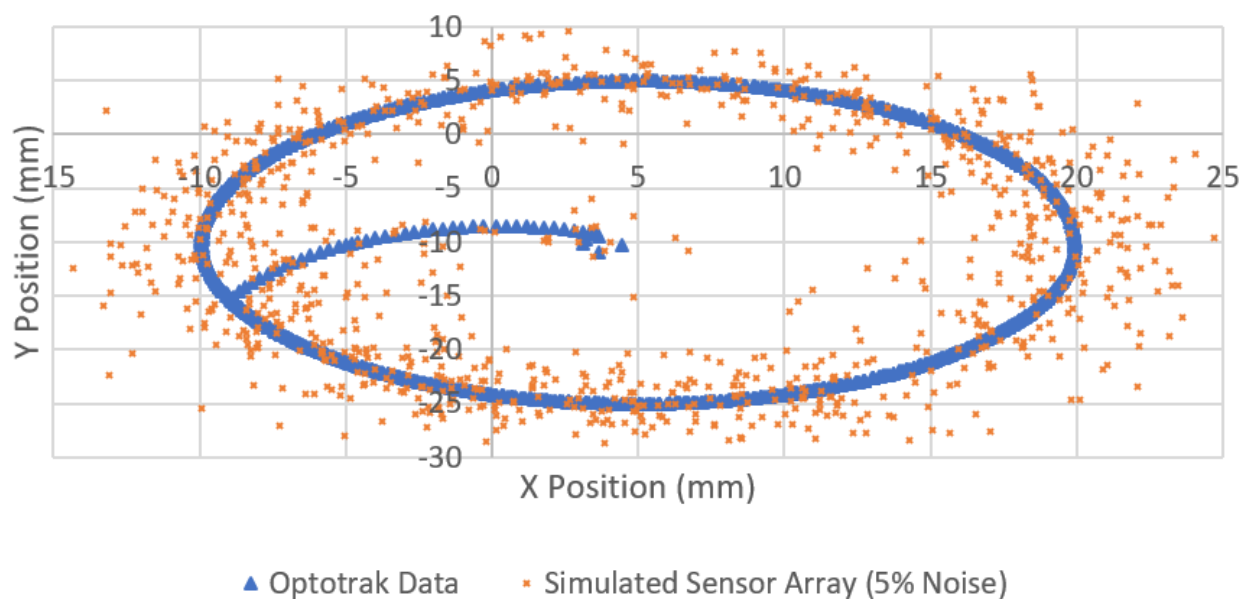
Figure 4-3: XY Projection of Simulated Localization of Position, 10% Noise

4.1.4   COMPARISON OF ACCURACY OF NEW AND ORIGINAL LOCALIZATION ALGORITHMS

One interesting result from the simulated experiments is that despite the added complexity, the localization is more accurate with the new algorithm which incorporates additional simulated magnetic field from the maglev than it is with the original algorithm, at least in a simulated environment. Table 4-3 compares the position and orientation accuracy at various noise levels between the two experiments which have just been described, in the one case taking into account the magnetic field generated by the maglev, in the other case replicating the original localization algorithm with no added magnetic field from the maglev. Figure 4-4 and Figure 4-5 show the same data in a visual bar-graph format.

Table 4-3: Comparison of Accuracy of New vs. Original Localization Algorithm

Position Error (mm)	No Maglev	With Maglev
No Noise	2.64E-08	1.01E-07
5% Noise	1.13373	0.64406
10% Noise	2.2987	1.28784
Orientation Error (radians)	No Maglev	With Maglev
No Noise	3.87E-09	3.99E-08
5% Noise	0.0275007	0.0209799
10% Noise	0.0561892	0.0424378
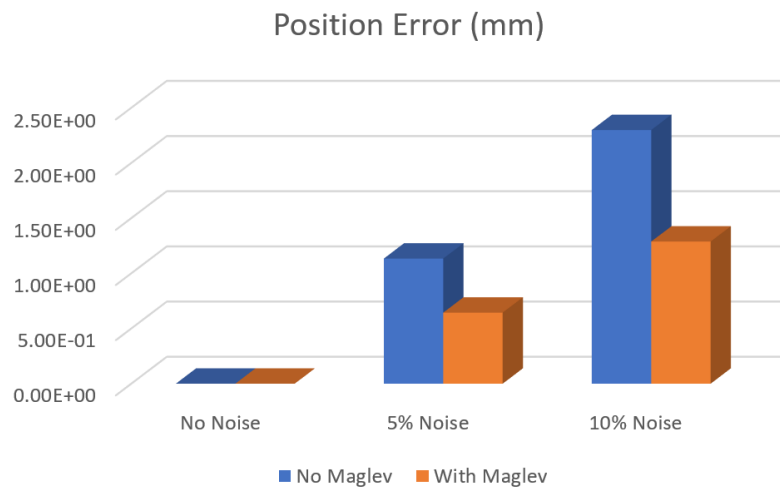
Figure 4-4: Position Accuracy of New vs. Original Localization Algorithm
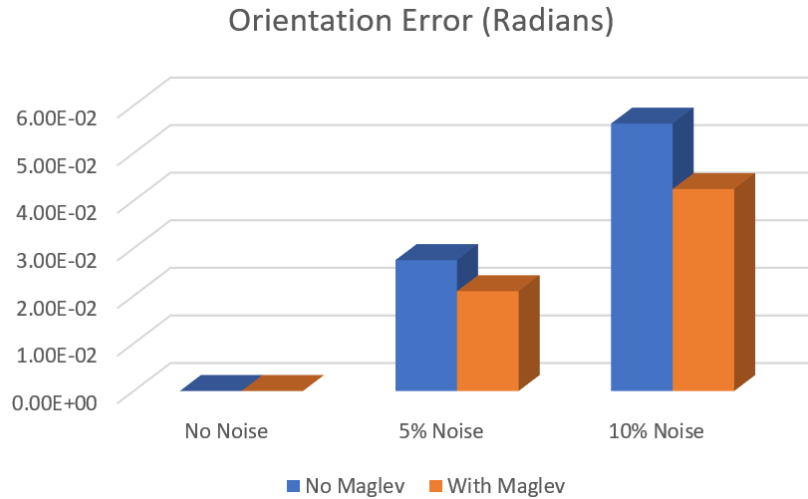
Figure 4-5: Orientation Accuracy of New vs. Original Localization Algorithm

4.1.5 UPDATE RATE

It is worth taking a moment at this point to examine the update rate for this localization algorithm. Hu et al. [82] report update rates for their algorithm of approximately 10 Hz. They identify the computation time of the algorithm as the limiting factor on the update rate of the system as a whole, and describe various techniques for improving the update speed, including techniques that sacrifice the accuracy of the localization. Since 10 Hz is insufficient for most real-time applications, such compromises between speed and accuracy are sensible. This position is echoed by other researchers, such as the work presented in [24] which focuses on improving the execution speed of a similar algorithm. However, as shown in Table 4-4, the simulated results which have just been presented were able to run the localization algorithm described by Hu et al. at over 3kHz, and even the new, more complex algorithm runs at hundreds of Hz, a speed which could most likely be further optimized now that the algorithm has been demonstrated to work in its current state. These results clearly show that with a properly structured software program, in this case the LAPACK software library [95] and appropriate PC hardware, in this case an intel i7 processor [88, p. 7], this localization algorithm can be run at speeds far exceeding what has previously been reported, at an update rate sufficient for many real-time applications.

Table 4-4: Localization Update Rate

Localization Algorithm Update Rate		
	Prior Works	**New Software**
Without Maglev	10 Hz	3081 Hz
With Maglev	-	422 Hz

As previously described, the sensor array which was built for this research consists of sixteen HMC5983 sensors arranged in a regularly-spaced 4x4 grid. In order to test whether there would be benefits to a larger sensor array, I simulated the same experiments which have already been described in sections 4.1.2 and 4.1.3 using a simulated 64-sensor array with regularly-spaced sensors placed 9mm apart in a regular 8x8 grid in a horizontal plane. This simulated sensor array covered roughly the same (virtual) physical space as the 4x4 simulation of the sixteen-sensor HMC5983 board. Interestingly, there was no significant improvement in the accuracy of the localization algorithm with a 64-sensor array when compared with the simulation of the 16-sensor array. In fact, as shown in Table 4-5, the 64-sensor array often performed worse than its 16-sensor counterpart.

Table 4-5: Simulated Experiments with a 64-Sensor Array

	Simulated Accuracy: Localization without Maglev			
	Simulated 16-Sensor Array		Simulated 64-Sensor Array	
	Position Error (mm)	Orientation Error (radians)	Position Error (mm)	Orientation Error (radians)
No Noise	2.64E-08	3.87E-09	7.83E-08	4.10E-09
5% Noise	1.13373	0.0275007	1.0629	0.0208373
10% Noise	2.2987	0.0561892	2.12987	0.0419793
	Simulated Accuracy: Localization with Maglev			
	Simulated 16-Sensor Array		Simulated 64-Sensor Array	
	Position Error (mm)	Orientation Error (radians)	Position Error (mm)	Orientation Error (radians)
No Noise	1.01E-07	3.99E-08	1.38E-07	1.03E-08
5% Noise	0.64406	0.0209799	2.93013	0.0619809
10% Noise	1.28784	0.0424378	6.4373	0.129969

4.2 E*XPERIMENTAL* R*ESULTS WITH* P*HYSICAL* H*ARDWARE*

4.2.1 L*OCALIZATION WITHOUT* A*CTUATION*

Position and Orientation Accuracy

Before testing the new localization algorithm with the physical sensor array, it was important to validate that the sensor array itself was working correctly by replicating the work of Hu et al. This was done by placing the magnet into known positions and orientations and recording the sensor readings, then checking whether or not the localization algorithm could accurately estimate the magnet's position. The initial set of experiments was conducted using the Velmex motion system [108], composed of stepper motors connected to linear screw-drive stages. By affixing a small magnet to the Velmex, the magnet's position could be precisely controlled. However, the magnet could not be rotated within this three-axis system, and so it remained completely vertical throughout the duration of the data gathering.[4] Table 4-6 shows that the initial results were

---

[4] Data gathered on 30 August 2016 by BJ Tix

very promising, with the localization algorithm achieving sub-millimeter accuracy. As shown in Figure 4-6, the largest errors occurred towards the edges of the working space, and errors were much lower when the magnet was positioned over the center of the sensor array.

Table 4-6: Accuracy Experiments with the Velmex Motion System

Accuracy Experiments with the Velmex Motion System	
Average Positon Error (mm)	0.78547585
Average Orientation Error (radians)	0.02293159
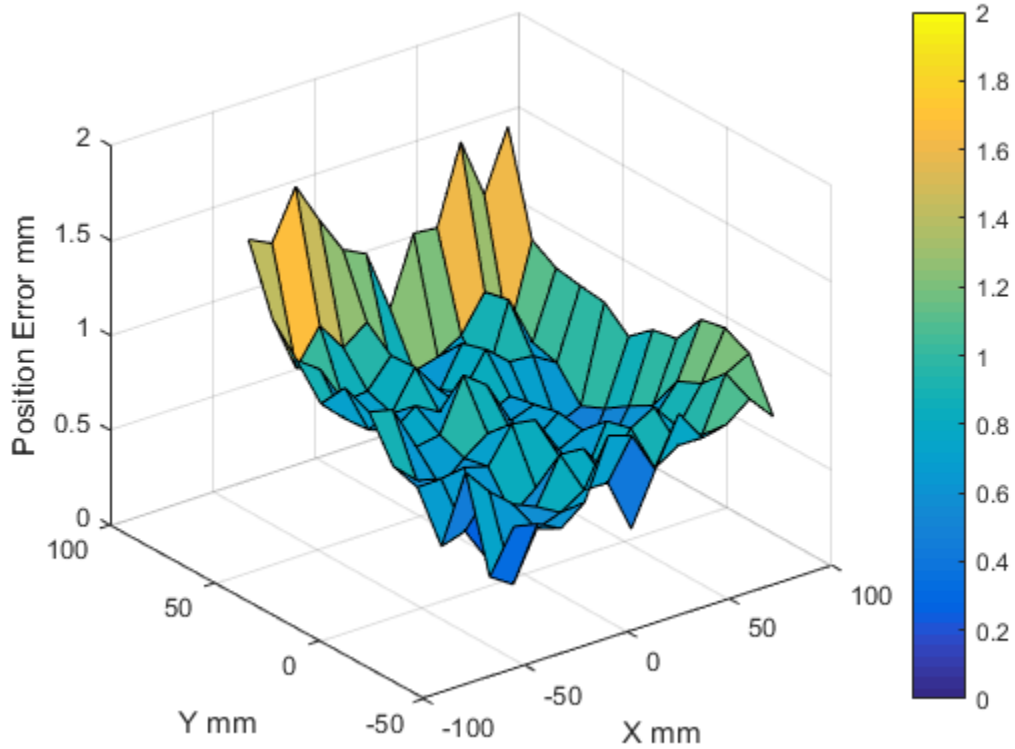
Figure 4-6: 3D Plot of Velmex Error with Respect to Magnet Position

Once the Sensor Array had demonstrated its effectiveness with the Velmex motion system, it was moved to a position underneath the maglev as shown in Figure 4-7. From this position, new data was gathered by connecting the magnet to infrared emitters and moving the magnet by hand as described in section 4.1.2. While this data was being gathered, actual sensor data was recorded as well as the position and orientation of the magnet. This dataset was considered to be more representative than the Velmex dataset of the conditions which would be faced during levitation for two reasons:

1) In this dataset, the magnet was moved in an irregular pattern, changing in both position and orientation. In the Velmex data set, the magnet moved in a very regular pattern and the magnet's orientation did not change.

2) This data was gathered with the sensor array properly positioned beneath the maglev, thus exposing it to identical physical conditions as would occur during levitation, with the sole exception that the maglev itself was not turned on while this data was being gathered.
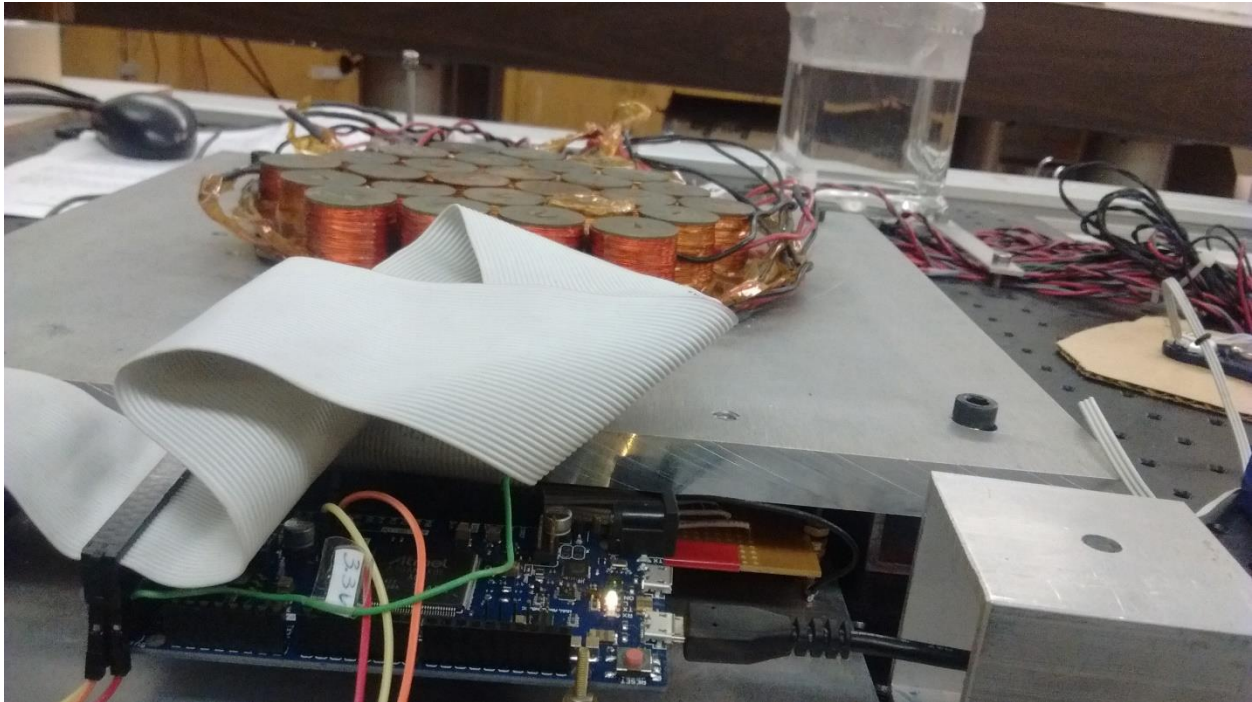
Figure 4-7: The Sensor Array Positioned Beneath the Maglev Coils

As shown in Table 4-7, the sensor array performed less accurately under these conditions than it did during the Velmex experiments, but still achieves a positional accuracy of ~1.6 mm.

Table 4-7: Accuracy Experiments with the Data Gathered by Optotrak

Position Error (mm)	1.55729
Orientation Error (radians)	0.03426

Figure 4-8 shows the XY projection of the position estimated by the localization algorithm, along with the position measured by the Optotrak. Just as with the Velmex data, the most significant errors occur towards the edges of the sensor array, in this case the far left and right sides of the array. For the sake of clarity, Figure 4-8 shows only the first 2,000 of the 10,000 data points in this dataset.

Actual and Estimated Position
No actuation; 2,000 samples; 13 March 2017

Y Position (mm)

X Position (mm)

▲ Optotrack Data   ■ Sensor Array Estimate
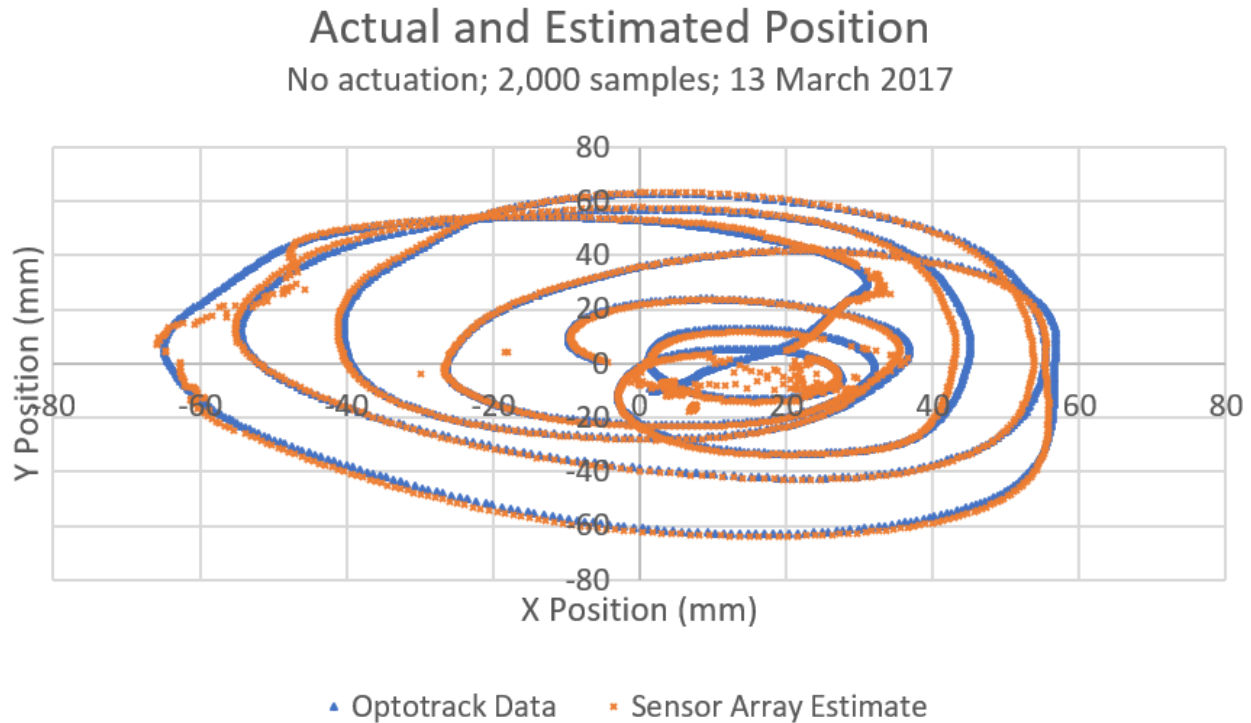
Figure 4-8: Estimated vs. Actual Position for Optotrak Gathered Data with No Maglev

Calibration Procedure

The results presented in the previous section would not have been possible without first applying a calibration procedure to modify the raw sensor readings prior to localization. Hu et al. also report the need to calibrate the sensor output, relying on three transformations of the raw sensor data [82]:

1) A position offset for each sensor axis.
2) An orientation adjustment for each sensor axis.
3) A third-order polynomial transform of the sensor data.

I chose to use the same calibration coefficients. However, instead of calibrating each sensor by hand as is shown in [82], I wrote an automatic calibration procedure based on the same Levenberg-Marquardt minimization function as is used by the localization algorithm. Using the recorded dataset, the calibration coefficients are set as the outputs for the Leveberg-Marquardt algorithm. Error is calculated by testing how closely calibrated values align with expected sensor readings, and the calibration values are optimized until the closest fit to the expected data is found. As Table 4-8 shows, calibration using this method produces dramatic improvement in the accuracy of the localization algorithm.

Table 4-8: Effects of Calibration

	Before calibration	After Calibration
Position Error (mm)	7.71957	1.55729
Orientation Error (radians)	0.124083	0.034255

Comparison to Hu et al.

The purpose of these experiments with the sensor array with no maglev interference was to validate sensor array hardware through replicating and confirming the work of Hu et al. Thus, a review of the comparisons between this work and the work of Hu et al. is appropriate before moving forward with experiments with the expanded algorithm combining localization with the use of the maglev. As shown in Table 4-9, the sensor array constructed for this research exceeds the results reported by Hu et al. in the accuracy of position estimation, orientation estimation, and update rate.

Table 4-9: Comparison of Present Work to Prior Work by Hu et al.

	Hu et al.	HRIL
Position Accuracy (mm)	3.3	1.55729
Orientation Accuracy (radians)	0.051	0.034255
Update Rate (Hz)	10	160[5]

Localization with Given Orientation

When examining the data produced from this localization experiment, a clear trend was observed that position error and orientation error were tightly coupled. In other words, position error is proportional to orientation error, when one is high, the other is high as well. This trend can be seen quite clearly in Figure 4-9. Therefore, it stands to reason that if the orientation of the magnet could be accurately and independently determined, the position accuracy of the localization algorithm would most likely improve. In [24], Natali and Simaan describe using an inertial measurement unit to determine a capsule's orientation, so the principle of establishing orientation independently prior to attempting position localization is not without precedent.

A new localization algorithm was written, *LM_LocalizaerWithGivenOrientation*, which was implemented as a child class of *ILocalizer*. *LM_LocalizaerWithGivenOrientation* is a modification of the *LM_Localizer* which has already been described. The sole difference between the two localizers is that *LM_LocalizaerWithGivenOrientation* uses orientation data from an outside source, and only attempts to solve for the position of the magnet, treating the orientation as a given. Localization was performed on the recorded replay of the experiments presented in this section, using the real recorded sensor data as input and the magnet orientation from the Optotrak as the given orientation. The results of this experiment are shown in Table 4-10.

---

[5] The localization algorithm can run much faster than 160 Hz, up to 3kHz as reported in Table 4-4. However, the physical sensor array is limited by the update rate of the HMC5983 sensor, which is 160 Hz.

As expected, the position accuracy of the localization estimates improves significantly when the orientation of the magnet is known.
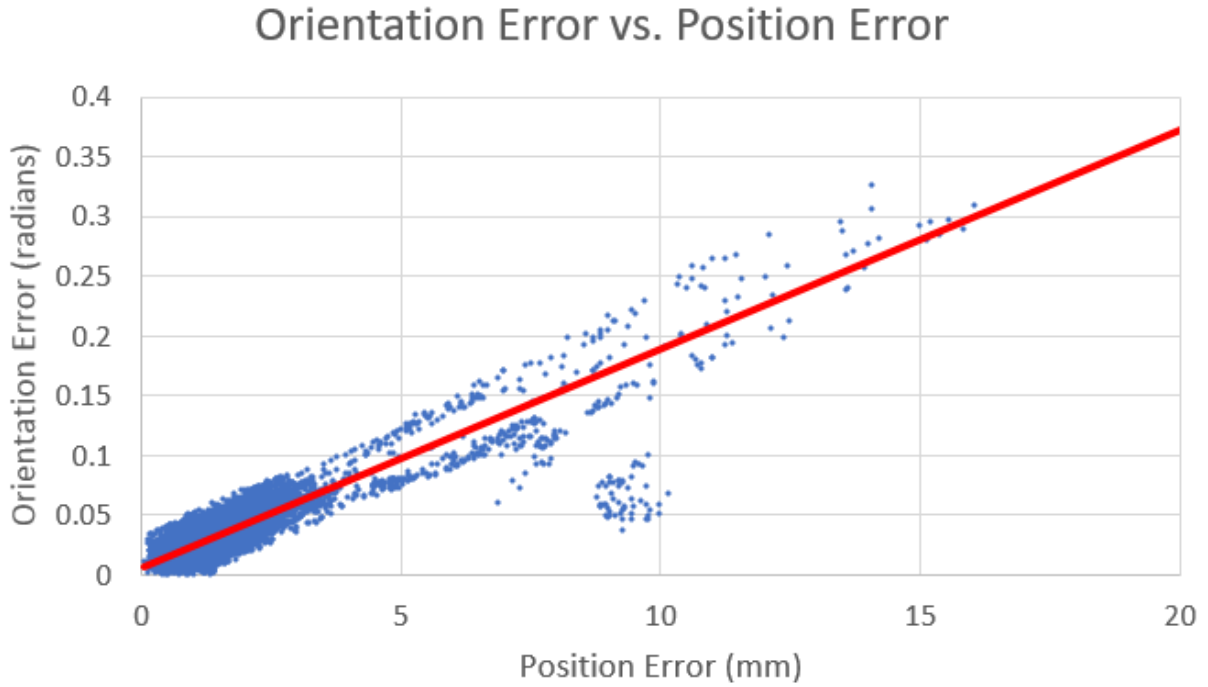
Figure 4-9: Position and Orientation Error using Localization without Actuation

Table 4-10: Accuracy of Localization with Given Orientation

	Position Error (mm)
Localization of Position and Orientation	1.55729
Localization with given Orientation	0.835472

4.2.2 COMBINED LEVITATION AND LOCALIZATION

Technical Limitations

With the prior works successfully replicated and improved upon, the sensor array proven through localization experiments without the maglev, and the localization algorithm combining localization with levitation proven to be theoretically sound through simulation, we can now turn our attention to proving the efficacy of the new algorithm through physical experiments. However, when attempting to conduct experiments which combined localization and levitation with the physical sensor array and maglev, two key technical limitations arose which prevented the use of the sensor array during ordinary levitation.

1) The HMC5983 sensors have an operational range of +/- 8 Gauss. However, in the position the sensors are located, directly beneath the maglev, the magnetic flux density was calculated to be expected to exceed 100 Gauss at its highest peaks. Flux densities exceeding an absolute value of eight Gauss would occur frequently, disrupting the operation of the sensor array.

2) The HMC5983 sensors update at a speed of 160 Hz, whereas the closed-loop PD controller for the maglev updates at over 800 Hz during levitation and 160 Hz is too slow to achieve stable levitation.

For these reasons, it was impossible to gather data using the sensor array during normal levitation.

Single and Double Coil Experiments

Although the technical limitations just described prevent the construction of a true working prototype for the time being, more limited experiments were performed in order to validate the new algorithm and provide physical evidence of whether or not the new algorithm *could* produce accurate localization, given an appropriate hardware setup. An experiment was designed according to the following specifications:

1) The coils are turned on one at a time, and slowly stepped up in increments of 0.1A from -0.8A to +0.8A. This is in contrast to the coils normal operating range of +/- 4A. Once a coil reaches +0.8A, it is reset to 0A and the cycle begins for the next coil, until all 27 coils have been stepped through the same process.

2) Sensor readings are recorded at each step as the coil currents are changed, leaving sufficient time for a full reading of the sensor array in between changes in current levels.

3) Once each coil has been cycled individually, coils are cycled in pairs through the same range of currents. Sensor readings are recorded in the same manner, this time with two coils activated simultaneously.

4) This experiment is repeated twice, once with no permanent magnet present, to validate that the model of the expected field generated by the maglev is accurate, and once with a permanent magnet present, to test localization.

5) With the magnet present, the *LM_LocalizerWithMaglev* localizer is run to produce an estimate of the magnet's position and orientation.

6) The recorded data for this test is exported to a spreadsheet. Expected, calibrated, and raw values are compared to one another and average relative error between calibrated and expected sensor values is computed. The estimate of the magnet's position is plotted against the actual position of the magnet.

Figure 4-10 and Figure 4-11 show the results of running this experiment with and without the magnet present for the Z-Axis of sensor #6. The Z-Axis of sensor 6 was chosen as representative because sensor 6 is one of the four central sensors and thus will read higher magnetic flux density on average from any given coil than an outer edge sensor would. Furthermore, the Z-Axis was chosen because the sensors are placed directly beneath the coils, resulting in most of the magnetic flux occurring in the Z (vertical) direction.
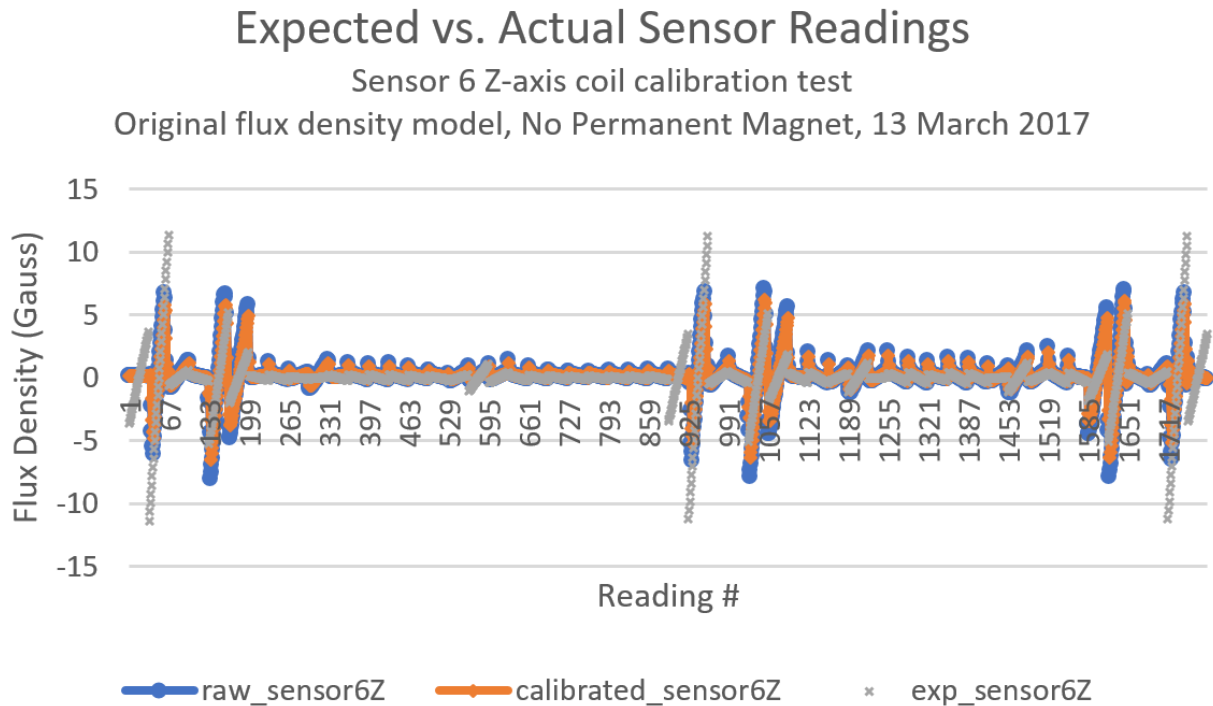
Figure 4-10: Coil Model Validation with No Magnet; Z-Axis of Sensor #6

As can be seen in Figure 4-10 and Figure 4-11, the estimated values for flux density appear to be correct in principle, as they follow the correct general shape and pattern of the recorded data.[6] However, the magnitudes of the expected and recorded data differ significantly. With no magnet present, the relative error (computed as the total error divided by the actual sensor reading) across all 48 sensor axes (3 axes each for sixteen sensors) is 68%. With the magnet present the relative error improves, but is still quite high at 20% average relative error.

---

[6] Data gathered 13 March 2017 by BJ Tix

Expected vs. Actual Sensor Readings
Sensor 6 Z-axis coil calibration test
Original flux density model, Magnet present, 13 March 2017
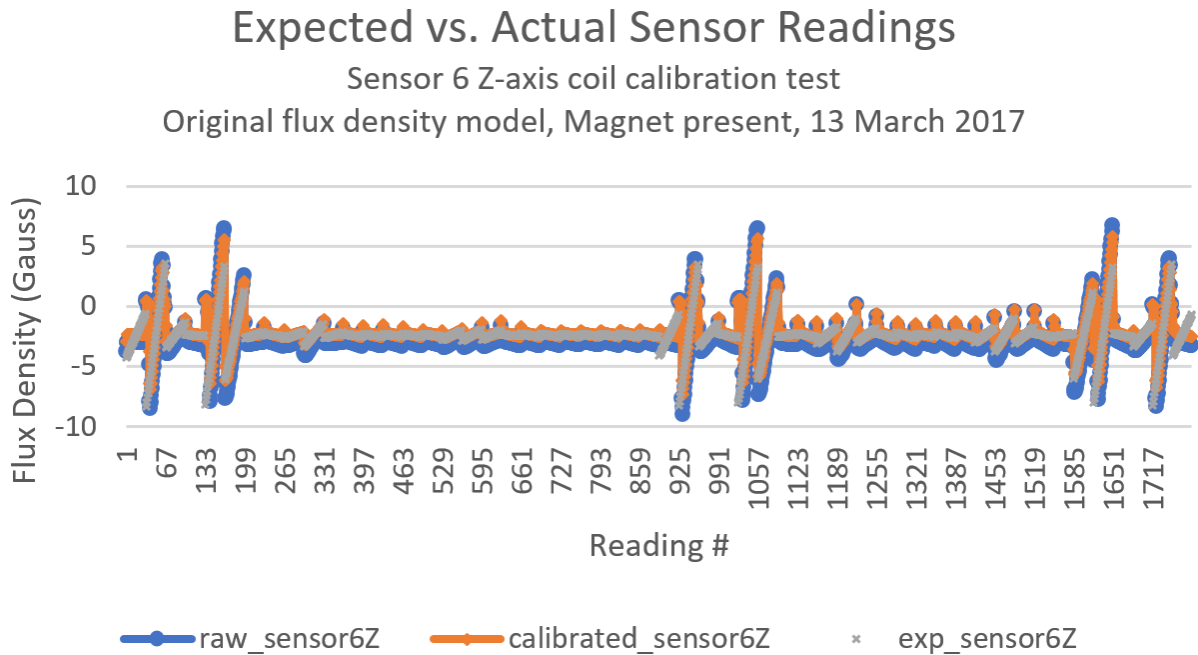
Figure 4-11: Coil Model Validation with Magnet Present; Z-Axis of Sensor #6

A New Numerical Model of Coil Fields

In response to the high relative error of the experiments just described, an alternative approach was attempted. The single-coil portion of the sequence was run, and sensor values for each of the sixteen sensors were recorded. This recorded data was used to form a new numerical model of the expected flux density at each coil. In this model, rather than estimating the flux density at arbitrary regularly spaced points in space for each coil, the effect of each coil on each sensor was estimated directly based on the recorded data of the sensor readings for each coil at different current levels. The hope was that by measuring these values directly, a more accurate numerical model would be achieved which bypassed any errors relating to discontinuities between the theoretical field model and the physical hardware. Unfortunately, this method failed to improve the agreement between expected and measured results, resulting in an average relative error of 48%.[7] Figure 4-12 shows the results of this experiment for the Z-Axis of senor 6, and Table 4-11 summarizes the results.

Table 4-11: Relative Error of Coil Numerical Models

	Relative Error without Magnet	Relative Error with Magnet
Original Numerical Model	68%	20%
New Numerical Model	77%	48%

---

[7] Data gathered 13 March 2017 by BJ Tix

Expected vs. Actual Sensor Readings
Sensor 6 Z-axis coil calibration test
New flux density model, Magnet present, 13 March 2017
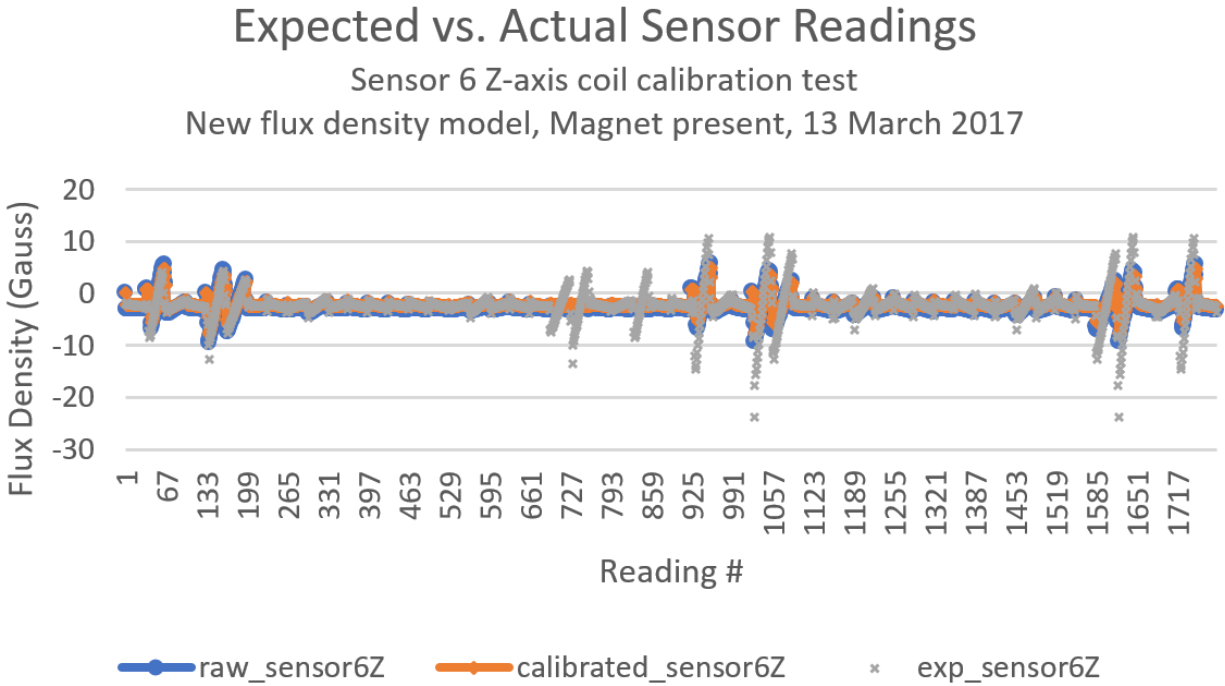
Figure 4-12: Coil Model Validation with Magnet Present; Z-Axis of Sensor #6; New Numerical Model

Attempts at Localization

Despite the high error values, I ran the new localization algorithm on the data captured during these experiments. The best results come from the experiment with the lowest relative error in the sensor estimates, using the original numerical model for coil flux as described in section 3.7.2. The average position error was 59.52 mm, far too large an error to be useful as input to the closed-loop PD feedback control used for levitation. Figure 4-13 and Figure 4-14 show the estimated positions generated by *LM_LocalizerWithMaglev* during the coil cycling procedure. During this experiment, the magnet was stationary, so the Optotrak estimate of the magnet position is represented in both figures as a single blue dot.

Figure 4-14 shows a zoomed-in view of the same data as Figure 4-13, with the farthest outliers excluded to give a clearer view of the majority of the data. The average estimate of the magnet's position in the X and Y dimensions is within 1 cm, as is reported in Table 4-12. However, the estimates are inconsistent, ranging across nearly 300 mm if all outliers are considered. The relatively accurate average estimate surrounded by imprecise outliers can be clearly seen in Figure 4-14.

Table 4-12: Average Estimate of Magnet Position

	X	Y	Z
Localization (mm)	-2.16731	-34.2593	76.46502
Optotrak (mm)	2.324622	-43.0838	100.7578
Error (mm)	4.491932	-8.82444	24.2928

Magnet Position Esimates

Original flux density model, best results
13 March 2017
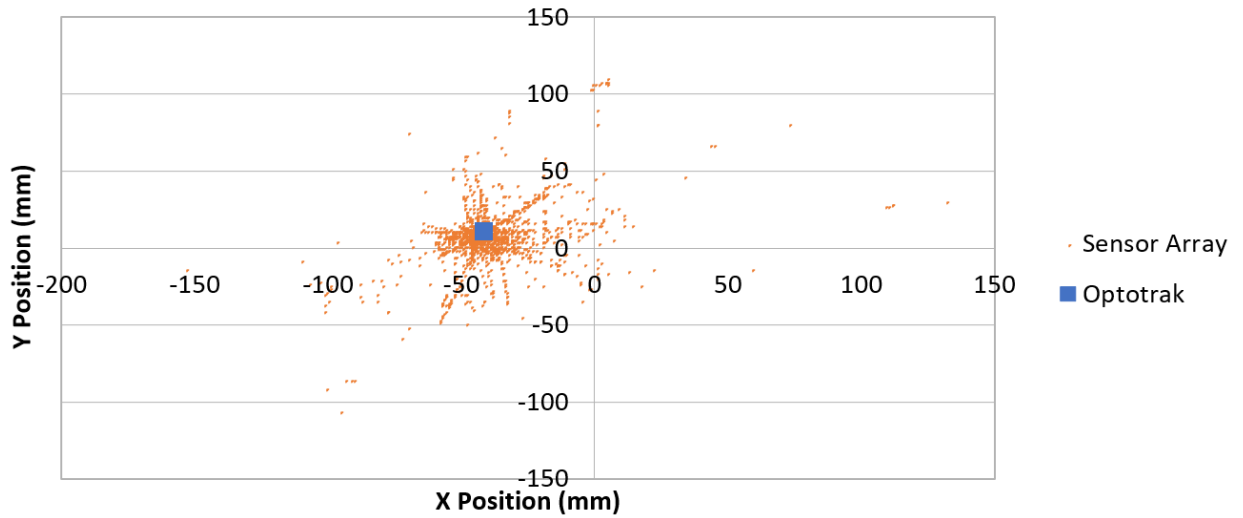
Figure 4-14: Magnet Localization Estimates During Coil Model Validation Experiments

Magnet Position Esimates

Original flux density model, best results, excluding outliers
13 March 2017
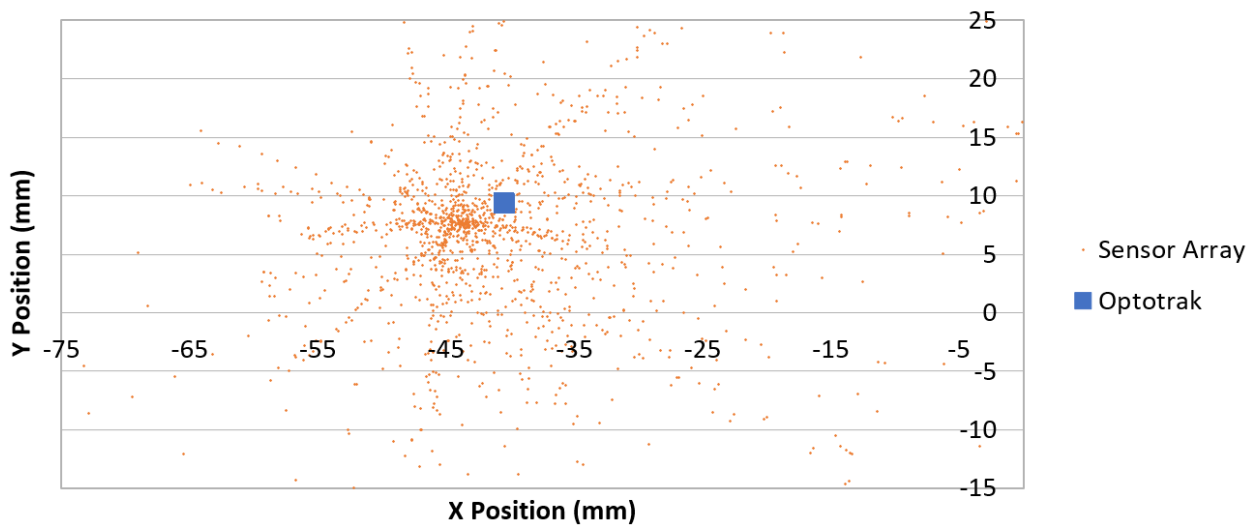
Figure 4-13: Magnet Localization Estimates During Coil Model Validation Experiments; Outliers Excluded

51

5 DISCUSSION AND CONCLUSIONS

5.1 VIABILITY OF THE PROPOSED LOCALIZATION ALGORITHM

Ultimately, this work failed to produce a viable working prototype demonstrating combined localization and levitation. However, I do not believe that the localization algorithm itself is at fault. Although full validation of the proposed algorithm will not be complete until a working prototype is demonstrated, I believe that the existing work is sufficient to demonstrate that this localization algorithm shows great promise, and would be likely to succeed given a more precise hardware setup, as will be discussed in depth throughout the remainder of this thesis. The proposed localization algorithm has been shown to achieve a high degree of accuracy in simulated experiments presented in section 4.1. Those same experiments showed that the localization algorithm is sensitive to noisy or inaccurate sensor readings, with even 10% error substantially reducing the accuracy of localization. Furthermore, the random error introduced during experimentation is always centered around the correct sensor reading, and thus real sensor errors, which may not be so conveniently distributed, should be expected to have an even greater impact on localization accuracy. Given these results, it is not surprising that localization fails to produce accurate estimates in a sensor array with an average relative error of 20%, as shown in section 4.2.2.

5.2 POSSIBLE EXPLANATIONS FOR FAILURE OF THE NEW NUMERICAL MODEL

One of the most surprising experimental results presented in this work was the failure of the new numerical model presented in section 4.2.2 to improve upon the results of the original model. The original numerical model used to estimate the magnetic flux density produced by each coil was based on a physical simulation of the coil using the Radia software package, as described in section 3.7.2. This model was based on the assumption that all the coils were identical and conformed to precise physical properties, which is one possible source of error. Furthermore, this model relied on knowing the precise location of the sensor relative to each coil, which was another potential source of error if there were any inaccuracies in the measurement or calibration of sensor position.

The new numerical model had neither of these drawbacks. Rather it directly estimated the flux density read by each sensor in response to changes in coil currents based on previous direct measurements from the same sensors. One possible explanation for the failure of this model is that the sensors themselves are not entirely reliable, producing readings which are not identical between experiments despite physical conditions being extremely similar if not identical. Using previously measured data as the basis for the numerical model introduces the possibility of erroneous sensor readings being incorporated into the numerical model and thus the estimate of all future sensor readings. However, if the sensors do not produce the same erroneous readings in response to the same magnetic flux density during repeated experiments, then these erroneous data points will make the estimate less accurate, rather than more accurate. Furthermore, although the magnetic field produced by the maglev is a superposition of all of the individual fields produced by each coil, the purely

empirical numerical model estimates of the flux on each coil may not be commutable in the same way, leading to additional errors in estimation when the flux density from two or more coils are added together to produce an estimate of the total flux density produced at a particular sensor by the maglev as a whole. Ultimately, the numerical model based on the physics of the coils proved to be more robust and reliable than the purely empirical model.

5.3 THE NEED FOR MORE PRECISE HARDWARE

It is my belief, based on the results and conclusions already presented, that in order to produce a working physical prototype which demonstrates combined localization and levitation using the algorithm proposed by this research, a more precise hardware setup is needed. Ultimately, the algorithm has been shown to work in theory, and the physical setup used in these experiments has been shown to have significant uncertainties and inaccuracies that result in significant errors in the estimation of the maglev's magnetic field. These hardware limitations will need to be addressed if future work is to produce physical validation of these already successful simulated results. Recommendations for the construction of a new system which addresses the shortcomings of the current hardware are presented in section 6.

6 LIMITATIONS AND FUTURE WORK

6.1 SENSOR HARDWARE

The HMC5983 sensor, as described in section 3.6.1, is ultimately insufficient for use in combination with a maglev system, necessitating the more limited and constrained experiments presented in this work. No commercially available sensor was found with the capabilities required for this research, making the HMC5983 a logical choice for initial investigations. However, now that the problem space has been more thoroughly explored, future research would benefit greatly from designing or commissioning the design of a magnetic flux density sensor capable of sensing magnetic flux density in the range of +/- 100 Gauss or more, and capable of updating at 1kHz or faster. Ultimately, such a sensor will be required if a working prototype which fulfils the original vision of this research is to be achieved.

6.2 FRAME AND MOUNTING

The sensor array described in section 3.6.1 was constructed on a prototyping board, each component was soldered by hand, and sensors are inserted by hand into slots which allow sensors to be quickly added or removed from the array. Furthermore, the prototyping board itself is mounted by hand to its aluminum back-plate, and the back-plate and sensor array are then seated by hand under the maglev coils as shown in Figure 6-1. Each of these processes is inaccurate to at least some degree, introducing small errors in the position and orientation of each sensor at each step of the construction and placement of the sensor array.

Greater precision could be achieved by fabricating a new sensor array as a printed circuit board with a high degree of accuracy in the placement of each of the sensors. Furthermore, a precision-machined back-plate could be designed and fabricated to provide a frame for the printed sensor array, removing another degree of imprecision from the process. Finally, rather than being slid by hand under the maglev coils relying on visual cues from imprecise reference markers, a precision-machined frame with guide rails and a back stop could ensure that the sensor array is located in a known location which is precisely the same each time it is placed under the maglev coils. The calibration procedure described in section 4.2.1 is intended to account for the small position and orientation errors that are introduced by the inherent imprecision of the process, but no calibration technique is perfect and I believe that superior results could be achieved with precisely machined hardware.
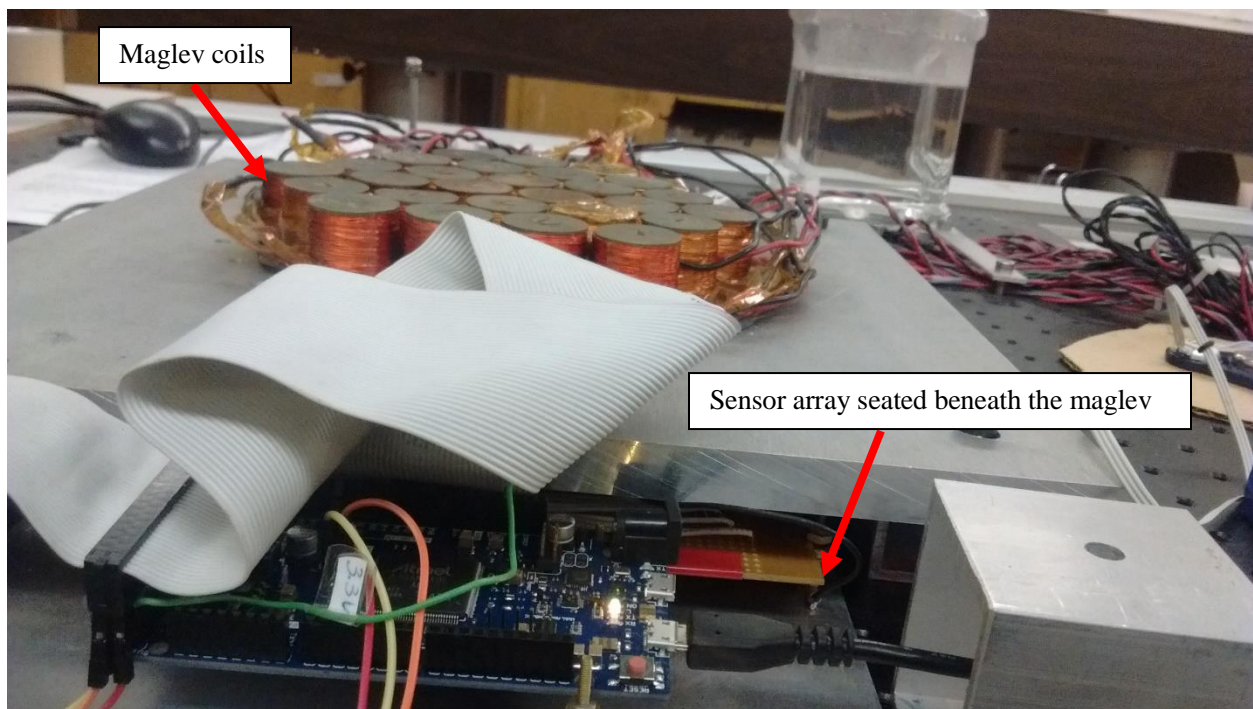
Figure 6-1: Sensor Array Seated Under Maglev Coils

6.3 MAGLEV COILS

Since the experiments shown in section 4.2.2 demonstrate significant disagreement between the expected and measured values of the magnetic flux density produced by the maglev, it is worth noting at this point that each of the coils in the maglev are wound by hand and may not precisely and identically match either each-other or the physical model which describes their expected behavior. Since previous experiments with the maglev have relied on a closed-loop PD controller, this control system may have masked any inaccuracies that were introduced by coils which did not behave exactly as expected. Further investigation should be carried out to determine how precisely the magnetic field produced by the maglev coils matches the physical model used to estimate that field.

6.4 ADDITION OF AN INERTIAL MEASUREMENT UNIT (IMU)

As described in section 4.2.1, independently determining the orientation of the magnet using the Optotrak resulted in greatly improved positional accuracy. Therefore, it is expected that if an accurate inertial measurement unit (IMU) were used to determine the orientation of the magnet and communicate this information to the localization algorithm, similar improvements in positional accuracy could be expected from the localization estimate. Therefore, future research which seeks to use the proposed algorithm to combine localization with magnetic levitation should strongly consider incorporating an accurate IMU into the capsule.

7   SUMMARY OF ORIGINAL CONTRIBUTIONS

In summary, the key contributions of this work have been to:

1) Replicate and validate the work of Hu et al. with regards to magnet localization through the use of a sensor array comprised of sixteen magnetic flux density sensors [82]. This contribution was described in section 4.2.1.

2) Improve upon the work of Hu et al. by:

   a. Demonstrating substantially faster update rates for the execution of the localization algorithm, which was a key area of concern for several prior works [24], [83]–[85]. This contribution is described in section 4.1.5.

   b. Demonstrating the efficacy of a novel automatic calibration procedure, described in section 4.2.1.

   c. Demonstrating that further improvements to the accuracy of the position estimate can be made through the incorporation of an IMU or other means for determining the orientation of the magnet, as described in section 4.2.1

3) Present a novel localization algorithm which incorporates information from an array of magnetic coils to effectively combine magnetic levitation and localization, described in section 3.7.2.

4) Demonstrate that the proposed algorithm is theoretically sound through simulated experiments, presented in section 4.1.3.

5) Demonstrate that the proposed algorithm is sensitive to sensor inaccuracies and therefore requires a precise hardware setup in order to produce accurate localization estimates, as described in section 4.1.3.

6) Show through various physical experiments that the hardware presently available in the HRIL lacks adequate precision for the proposed algorithm to produce accurate localization estimates, presented in section 4.2.2.

7) Outline the necessary hardware improvements which will lead to a more precise physical system capable of simultaneous localization and levitation, presented in section 6.

REFERENCES

[1] M.Q.-H Meng, Tao Mei, and Jiexin Pu, "Wireless robotic capsule endoscopy: state-of-the-art and challenges," *Proc. Fifth World Congr. Intell. Control Autom.*, pp. 5561–5565, Jun. 2004.

[2] A. Brett Slatkin, Joel Burdick, and Warren Grundfest, "The Development of a Robotic Endoscope," *Exp. Robot. IV*, pp. 161–169, 1997.

[3] S.J. Phee, W.S. Ng, I.M. Chen, F. Seow-Choen, and B.L. Davies, "Locomotion and Steering Aspects in Automoation of Colonoscopy," *IEEE Eng. Med. Biol.*, vol. 16, no. 6, pp. 85–96, 1997.

[4] C. Alexander Mosse, Timothy N. Mills, Mark N. Appleyard, Srinathan S. Kadirkamanathan, and C. Paul Swain, "Electrical stimulation for propelling endoscopes," *Gastrointest. Endosc.*, vol. 54, no. 1, pp. 79–83, Jul. 2001.

[5] Lois Ohee, Dino Accoto, Arianna Menciassi, Cesare Stefanini, Maria Chiara Carrozza, and Paolo Dario, "Analysis and development of locomotion devices for the gastrointestinal tract," *IEEE Trans. Biomed. Eng.*, vol. 49, no. 6, pp. 613–616, Jun. 2002.

[6] Byungkyu Kim, Younkoo Jeong, Hyun young Lim, Jong-Oh Park, Arianna Menciassi, and Paolo Dario, "Functional colonoscope robot system," *IEEE Int. Conf. Robot. Autom. Proc.*, vol. 1, pp. 1092–1097, 2003.

[7] Irwan Kassim, Wan S. Ng, Gong Feng, and Soo J. Phee, "Review of locomotion techniques for robotic colonoscopy," *IEEE Int. Conf. Robot. Automoation Proc.*, vol. 1, pp. 1086–1091, 2003.

[8] Paul Swain, Gavriel Iddan, Gavriel Meron, and Arkady Glukhovsky, "Wireless capsule endoscopy of the small bowel: development, testing, and first human trials," *Proc. SPIE-- Int. Soc. Opt. Eng.*, vol. 4158, no. 1, pp. 19–23, 2001.

[9] Douglas G. Alder and Christopher J. Gostout, "Wireless Capsule Endoscopy," *Hosp. Physician*, vol. 39, no. 5, pp. 14–22, May 2003.

[10] Trung Duc Than, Gursel Alici, Hao Zhou, and Weihua Li, "A review of localization systems for robotic endoscopic capsules," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 9, pp. 2387–2399, 2012.

[11] Amy Wang *et al.*, "Technology Status Evaluation Report: wireless Capsule Endoscopy," *Gastrointest. Endosc.*, vol. 78, no. 6, pp. 805–815, 2013.

[12] "How PillCam SB works during a capsule endoscopy procedure." [Online]. Available: http://ssshare.it/czcY. [Accessed: 22-Feb-2017].

[13] D. O. Faigel and D. R. Cave, *Capsule endoscopy*. Philadelphia: Saunders Elsevier, 2008.

[14] J L Toennies, G Tortota, M Simi, P Valdstri, and R J Webster III, "Swallowable Medical Devices for Diagnosis and Surgery: the State of the Art," *J. Mech. Eng. Sci.*, vol. 224, no. 7, pp. 1397–1414, 2010.

[15] E. Morita *et al.*, "In vivo trial of a driving system for a self-propelling capsule endoscope using a magnetic field (with video)," *Gastrointest. Endosc.*, vol. 72, no. 4, pp. 836–840, 2010.

[16] J. Rey *et al.*, "Feasibility of stomach exploration with a guided capsule endoscope," *Endoscopy*, vol. 42, no. 07, pp. 541–545, 2010.

[17] J.-F. Rey *et al.*, "First blinded nonrandomized comparative study of gastric examination with a magnetically guided capsule endoscope and standard videoendoscope," *Gastrointest. Endosc.*, 2012.

[18] T. Rupprecht, C. Rupprecht, S. Mühldorfer, M. Vieth, and M. Zapke, "The physical basics of magnetic-guided capsule endoscopy of the stomach and results of a feasibility study in the porcine stomach," *Endoscopy*, vol. 44, no. 4, pp. 437–8, 2012.

[19] Jutta Keller *et al.*, "Remote magnetic control of a wireless capsule endoscope in the esophagus is safe and feasible: results of a randomized, clinical trial in healthy volunteers," *Gastrointest. Endosc.*, vol. 72, no. 5, pp. 941–946, Nov. 2010.

[20] P. Swain *et al.*, "Remote magnetic manipulation of a wireless capsule endoscope in the esophagus and stomach of humans (with videos)," *Gastrointest. Endosc.*, vol. 71, no. 7, pp. 1290–1293, 2010.

[21] Imadadur Rahman, Mathieu Pioche, Chan Sup Shim, in Kyung Sung, Jean-Chrisophe Saurin, and Praful Patel, "Magnet Assisted Capsule Endoscopy (MACE) in the Upper GI Tract Is Feasible: First Human Series Using the Novel Mirocam-Navi System," *Gastrointest. Endosc.*, vol. 79, no. 5, p. AB122, 2014.

[22]   "MiroCam Navi uses the magnetic force to control the movement of capsule endoscope." [Online]. Available: http://www.reyyanmedical.com/index.php?goster=mirocam-navi-endoscope-capsul-camera&dil=en. [Accessed: 22-Feb-2017].

[23]   Gastone Ciuti, Pietro Valdastri, Arianna Menciassi, and Paulo Dario, "Robotic magnetic steering and locomotion of capsule endoscope for diagnostic and surgical endoluminal procedures," *Robotica*, vol. 28, pp. 199–207, 2010.

[24]   Christian Di Natali and Nabil Simaan, "Jacobian-Based Iterative Method for Magnetic Localization in Robotic Capsule Endoscopy," *IEEE Trans. Robot.*, vol. 32, no. 2, pp. 327–338, Apr. 2016.

[25]   I. Mohammed, B. Sharif, and J. Neasham, "Design and implementation of a magnetic levitation control system for robotically actuated capsule endoscopes," in *Robotic and Sensors Environments (ROSE), 2012 IEEE International Symposium on*, 2012, pp. 7–12.

[26]   I. Aoki, A. Uchiyama, K. Arai, K. Ishiyama, and S. Yabukami, "Detecting system of position and posture of capsule medical device," US7751866 B2, 06-Jul-2010.

[27]   T. Honda, K. I. Arai, and K. Ishiyama, "Micro swimming mechanisms propelled by external magnetic fields," *IEEE Trans. Magn.*, vol. 32, no. 5, pp. 5085–5087, Sep. 1996.

[28]   Yamazaki A, "Spiral-type micro-machine for medical applications," *J. Micromechatronics*, vol. 2, no. 1, pp. 77–86, 2002.

[29]   Atsushi Chiba *et al.*, "Magnetic Actuator for a Capsule Endoscope Navigation System," *J. Magn.*, pp. 89–92, 2007.

[30]   D. Chen, H. Chao, W. Lei, and M.Q.-H Meng, "The Force Model of Wireless Active Actuation for Capsule Endpscope in the GI Tract," 2007, pp. 93–98.

[31]   K. Ishiyama, M. Sendoh, A. Yamazaki, and K. I. Arai, "Swimming micro-machine driven by magnetic torque," *Sens. Actuators Phys.*, vol. 91, no. 1, pp. 141–144, 2001.

[32]   M. Sendoh, K. Ishiyama, and K.-I. Arai, "Fabrication of magnetic actuator for use in a capsule endoscope," *Magn. IEEE Trans. On*, vol. 39, no. 5, pp. 3232–3234, 2003.

[33]   A. Chiba, M. Sendoh, K. Ishiyama, and K. I. Arai, "Basic Characteristics of a Magnetic Actuator for Capsule Endoscope," *Trans. Jpn. Soc. Med. Biol. Eng.*, vol. 42, no. 4, pp. 313–317, 2004.

[34]   A. Chiba, M. Sendoh, K. Ishiyama, and K. I. Arai, "Moving of a Magnetic Actuator for a Capsule Endoscope in the Intestine of a Pig," *J. Magn. Soc. Jpn.*, vol. 29, no. 3, pp. 343–346, 2005.

[35]   C. Quaglia, E. Buselli, R. J. W. III, P. Valdastri, A. Menciassi, and P. Dario, "An endoscopic capsule robot: a meso-scale engineering case study," *J. Micromechanics Microengineering*, vol. 19, no. 10, p. 105007, 2009.

[36]   S. Sudo, S. Segawa, and T. Honda, "Magnetic Swimming Mechanism in a Viscous Liquid," *J. Intell. Mater. Syst. Struct.*, vol. 17, no. 8–9, pp. 729–736, 2006.

[37]   L. Zhang, J. J. Abbott, L. Dong, B. E. Kratochvil, D. Bell, and B. J. Nelson, "Artificial bacterial flagella: Fabrication and magnetic control," *Appl. Phys. Lett.*, vol. 94, no. 6, 2009.

[38]   Hannes Bleuler, "A Survey of magnetic Levitation and Magnetic Bearing Types," *JSME Int. J.*, vol. 35, no. 3, pp. 335–342, 1992.

[39]   Ahmed El Hajjaji and M Ouladsine, "Modelling and Nonlinear Control of Magnetic Levitation Systems," *IEEE Trans. Ind. Electron.*, vol. 48, no. 4, pp. 831–838, Aug. 2001.

[40]   Michel Dussaux, "Status of the Industrial Applications of the Active magnetic Bearings Technology," *Proc. 2nd Int. Symp. Magn. Bear.*, pp. 33–38, 1990.

[41]   P. Berkelman and M. Dzadovsky, "Extending the motion ranges of magnetic levitation for haptic interaction," in *World Haptics 2009 - Third Joint EuroHaptics conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 2009, pp. 517–522.

[42]   Peter Berkelman and BJ Avery, "Co-located Graphic and Haptic Display using an Array of Electromagnetic Coils," presented at the IEEE Haptics Symposium, Philadelphia, Pensylvania, 09-Apr-2016.

[43]   P. J. Berkelman and R. L. Hollis, "Lorentz Magnetic Levitation for Haptic Interaction: Device Design, Performance, and Integration with Physical Simulations," *Int. J. Robot. Res.*, vol. 19, no. 7, pp. 644–667, Jul. 2000.

[44] P. J. Berkelman, R. L. Hollis, and S. E. Salcudean, "Interacting with virtual environments using a magnetic levitation haptic interface," in *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, 1995, vol. 1, pp. 117–122 vol.1.

[45] Cheng-Chi Tai, Ming-Kun Chen, and Ching Hsing Luo, "Study of Magnetic Levitation Technique for Use in a Wireless Capsule Endoscope," *J. Med. Biol. Eng.*, vol. 27, no. 2, pp. 57–64, Feb. 2007.

[46] Michael P. Kummer, Jake J Abbott, Bradley E. Kratochvil, Ruedi Borer, Ali Sengul, and Bradley J. Nelson, "OctoMag: An Electromagnetic System for 5-DOF Wireless Micromanipulation," *IEEE Trans. Robot.*, vol. 26, no. 6, pp. 1006–1017, 2010.

[47] Muneaki Miyasaka and Peter Berkelman, "Magnetic Levitation with Unlimited Omnidirectional Rotation Range," *Mechatronics*, vol. 24, no. 3, 2014.

[48] Peter Berkelman and Michael Dzadovsky, "Magnetic Levitation Over Large Translation and Rotation Ranges in All Directions," *IEEEASME Trans. Mechatron.*, vol. 18, no. 1, pp. 44–52, Feb. 2013.

[49] F. Ullrich *et al.*, "Mobility experiments with microrobots for minimally invasive intraocular surgery," *Invest. Ophthalmol. Vis. Sci.*, vol. 54, no. 4, pp. 2853–63, 2013.

[50] K. Berk Yesin, K. Vollmers, and B. J. Nelson, "Modeling and control of untethered biomicrorobots in a fluidic environment using electromagnetic fields," *Int. J. Robot. Res.*, vol. 25, no. 5 6, p. 527, 2006.

[51] P. Berkelman and M. Dzadovsky, "Novel Design, Characterization, and Control Method for Large Motion Range Magnetic Levitation," *IEEE Magn. Lett.*, vol. 1, pp. 0500104–0500104, 2010.

[52] W. D. M. Jr, W. D. Meadow, and P. J. Berkelman, "Adaptive magnetic levitation apparatus and method," US7224252 B2, 29-May-2007.

[53] "Optotrak Certus Brochure." Northern Digital Inc (NDI), 2015.

[54] Jill Schmidt, Devin R. Berg, and Heidi-Lynn Ploeg, "Precision, repeatabliity and accuracy of Optotrak optical motion tracking system," *Int. J. Exp. Comput. Biomech.*, vol. 1, no. 1, pp. 114–127, 2009.

[55] H. Keller *et al.*, "Method for navigation and control of a magnetically guided capsule endoscope in the human stomach," in *2012 4th IEEE RAS EMBS International Conference on Biomedical Robotics and Biomechatronics (BioRob)*, 2012, pp. 859–865.

[56] F. Carpi, S. Galbiati, and A. Carpi, "Controlled Navigation of Endoscopic Capsules: Concept and Preliminary Experimental Investigations," *Biomed. Eng. IEEE Trans. On*, vol. 54, no. 11, pp. 2028–2036, 2007.

[57] M. Pourhomayoun, M. Fowler, and Z. Jin, "A novel method for medical implant in-body localization," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2012, pp. 5757–5760.

[58] Olympus America Inc, 350 Corporate Parkway, PO Box 610, Center Valley, PA 18034, "Olympus ScopeGuide Promotional Materials." 2014.

[59] JS Bladen, AP Anderson, GD Bell, B Rameh, B Evans, and DJ Heatley, "Non-radiological technique for three-dimensional imaging of endoscopes.," *The Lancet*, vol. 341, no. 8847, pp. 719–722, Mar. 1993.

[60] SG Shah, JC Brooker, CB Williams, C Thapar, and BP Saunders, "Effect of magnetic endoscope imaging on colonoscopy performance: a randomised controlled trial.," *The Lancet*, vol. 356, no. 9243, pp. 1718–1722, Nov. 2000.

[61] Imadadur Rahman, Praful Patel, Philip Boger, Mike Thomson, and Nadeem Ahmad Afzal, "Utilisation of magnets to enhance gastrointestinal endoscopy," *World J. Gastrointest. Endosc.*, vol. 7, no. 19, pp. 1306–1310, Dec. 2015.

[62] J. Hou *et al.*, "Design and Implementation of a High Resolution Localization System for In-Vivo Capsule Endoscopy," in *Dependable, Autonomic and Secure Computing, 2009. DASC '09. Eighth IEEE International Conference on*, 2009, pp. 209–214.

[63] L. Zhang, Y. Zhu, T. Mo, J. Hou, and H. Hu, "Design of 3D Positioning Algorithm Based on RFID Receiver Array for In Vivo Micro-Robot," in *Dependable, Autonomic and Secure Computing, 2009. DASC '09. Eighth IEEE International Conference on*, 2009, pp. 749–753.

[64] L. Zhang, Y. Zhu, T. Mo, J. Hou, and G. Rong, "Design and Implementation of 3D Positioning Algorithms Based on RF Signal Radiation Patterns for In Vivo Micro-robot," in *Body Sensor Networks (BSN), 2010 International Conference on*, 2010, pp. 255–260.

[65] Doron Fischer *et al.*, "Localization of the wireless capsule endoscope in its passage through the GI tract," *Gastrointest. Endosc.*, vol. 53, no. 5, p. AB126, 2001.

[66] Doron Fischer, Reuven Shreiber, Daphna Levi, and Rami Eliakim, "Capsule endoscopy: The localization system," *Gastrointest. Endosc. Clin. N. Am.*, vol. 14, pp. 25–31, 2004.

[67] Lujia Wang, Li Liu, Chao Hu, and Max Q.-H. Meng, "A Novel RF-based Propagation Model with Tissue Absorption for Localization of the GI Tract," *32nd Annu. Int. Conf. IEEE EMBS*, pp. 654–657, Sep. 2010.

[68] R. Kuth, J. Reinschke, and R. Rockelein, "Method for determining the position and orientation of an endoscopy capsule guided through an examination object by using a navigating magnetic field generated by means of a navigation device," US20070038063 A1, 15-Feb-2007.

[69] F. Carpi and C. Pappone, "Magnetic Maneuvering of Endoscopic Capsules by Means of a Robotic Navigation System," *Biomed. Eng. IEEE Trans. On*, vol. 56, no. 5, pp. 1482–1490, 2009.

[70] B. Laulicht, N. J. Gidmark, A. Tripathi, and E. Mathiowitz, "Localization of magnetic pills," *Proc. Natl. Acad. Sci.*, vol. 108, no. 6, pp. 2252–2257, Feb. 2011.

[71] S. Tognarelli *et al.*, "Magnetic propulsion and ultrasound tracking of endovascular devices," *J. Robot. Surg.*, vol. 6, no. 1, pp. 5–12, Mar. 2012.

[72] M. Salerno *et al.*, "A discrete-time localization method for capsule endoscopy based on on-board magnetic sensing," *Meas. Sci. Technol.*, vol. 23, no. 1, p. 015701, 2012.

[73] J. Hummel *et al.*, "Evaluation of a new electromagnetic tracking system using a standardized assessment protocol," *Phys. Med. Biol.*, vol. 51, no. 10, p. N205, 2006.

[74] H. Zhang *et al.*, "Electromagnetic tracking for abdominal interventions in computer aided surgery," *Comput. Aided Surg.*, vol. 11, no. 3, pp. 127–136, Jan. 2006.

[75] "NDI Aurora System," *NDI Aurora System*. [Online]. Available: http://www.ndigital.com/medical/products/aurora/. [Accessed: 15-Feb-2017].

[76] "MINDFLUX - Ascension microBIRD." [Online]. Available: http://www.mindflux.com.au/products/ascension/microbird.html. [Accessed: 15-Feb-2017].

[77] Rong-Hao Liang, Kai-Yin Cheng, Chao-Huai Su, Chien-Ting Weng, Bing-Yu Chen, and De-Nian Yang, "GaussSense: attachable stylus sensing using magnetic sensor grid," *Proc. 25th Annu. ACM Symp. User Interface Softw. Technol.*, pp. 319–326, Oct. 2012.

[78] R.-H. Liang, H.-C. Kuo, L. Chan, D.-N. Yang, and B.-Y. Chen, "GaussStones: Shielded Magnetic Tangibles for Multi-token Interactions on Portable Displays," in *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology*, New York, NY, USA, 2014, pp. 365–372.

[79] R.-H. Liang *et al.*, "GaussBricks: Magnetic Building Blocks for Constructive Tangible Interactions on Portable Displays," in *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*, New York, NY, USA, 2014, pp. 3153–3162.

[80] Rong-Hao Liang *et al.*, "GaussBits: Magnetic Tangible Bits for Portable and Occlusion-Free Near-Surface Interactions," *ACM SIGCHI Conf. Hum. Factors Comput. Syst.*, 2013.

[81] N. M. Prakash and F. A. Spelman, "Localization of a magnetic marker for GI motility studies: an in vitro feasibility study," in *Proceedings of the 19th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 1997*, 1997, vol. 6, pp. 2394–2397 vol.6.

[82] Chao Hu, Max Q.-H. Meng, Mrinal Mandal, and Xiaona Wang, "3-axis Magnetic Sensor Array System for Tracking Magnet's Position and Orientation," *Proc. 6th World Congr. Intell. Control Autom.*, pp. 5304–5308, Jun. 2006.

[83] Chao Hu, Mao Li, Shuang Song, Wan'an Yang, Rui Zhang, and Max Q.-H. Meng, "A Cubic 3-Axis Magnetic Sensor Array for Wirelessly Tracking Magnet Position and Orientation," *IEE Sens. J.*, vol. 10, no. 5, pp. 903–913, May 2010.

[84] Wan'an Yang, Chao Hu, Max Q.-H. Meng, Shuang Song, and Houde Dai, "A Six-Dimensional Magnetic Localization Algorithm for a Rectangular Magnet Objective Based on a Particle Swarm Optimizer," *IEEE Trans. Magn.*, vol. 45, no. 8, pp. 3092–3099, Aug. 2009.

[85] Chao Hu, Max Q.-H. Meng, and Mrinal Mandal, "Efficient magnetic localization and orientation technique for capsule endoscopy," *Int. J. Inf. Acquis.*, vol. 02, no. 01, pp. 23–36, Mar. 2005.

[86]   Peter Berkelman, "Human-Robot Interaction Laboratory," *Human-Robot Interaction Laboratory*. [Online]. Available: http://me.hawaii.edu/HRIL/. [Accessed: 21-Feb-2017].

[87]   "Ubuntu," *The leading operating system for PCs, tablets, phones, IoT devices, servers and the cloud | Ubuntu*. [Online]. Available: https://www.ubuntu.com/. [Accessed: 21-Feb-2017].

[88]   "Intel® Core™ i7-2600 Processor (8M Cache, up to 3.80 GHz) Product Specifications," *Intel® ARK (Product Specs)*. [Online]. Available: http://ark.intel.com/products/52213/Intel-Core-i7-2600-Processor-8M-Cache-up-to-3_80-GHz. [Accessed: 18-Mar-2017].

[89]   "Welcome to Gallium's documentation! — Gallium 0.4 documentation." [Online]. Available: http://gallium.readthedocs.io/en/latest/index.html. [Accessed: 18-Mar-2017].

[90]   United Electronic Industries, Inc., "PD2-AO-32/16 32-Channel PCI Card for Analog Output Data Acquisition," Technical Specifications, Feb. 2006.

[91]   "Copley Controls - Servo Drives - Servo Amplifiers for Brushless andBrush Motors." [Online]. Available: http://copleycontrols.com/Motion/Products/index.html. [Accessed: 18-Mar-2017].

[92]   Sahba Aghajani Pedram, "HAPTIC TEXTURE RENDERING AND PERCEPTION USING COIL ARRAY MAGNETIC LEVITATION HAPTIC INTERFACE: EFFECTS OF TORQUE FEEDBACK AND PROBE TYPE ON PERCIEVED ROUGHNESS," University of Hawai'i at Manoa, 2016.

[93]   "Wolfram Mathematica: Modern Technical Computing." [Online]. Available: https://www.wolfram.com/mathematica/. [Accessed: 02-Mar-2017].

[94]   European Synchrotron Radiation Facility., "Radia." [Online]. Available: http://www.esrf.eu/Accelerators/Groups/InsertionDevices/Software/Radia. [Accessed: 02-Mar-2017].

[95]   Anderson, E. and Bai, Z. and Bischof, C. and, Blackford, S. and Demmel, J. and Dongarra, J. and, Du Croz, J. and Greenbaum, A. and Hammarling, S. and, and McKenney, A. and Sorensen, D., *LAPACK User's Guide*, 3rd ed. Philadelphia, PA: Society for Industrial and Applied Mathematics, 1999.

[96]   "LAPACK — Linear Algebra PACKage." [Online]. Available: http://www.netlib.org/lapack/. [Accessed: 02-Mar-2017].

[97]   Honeywell, "Datasheet - HMC5983 3-Axis Compass IC Advanced Information."

[98]   "www.drotek.fr - Magnetometer sensor - HMC5983L," *drotek*. [Online]. Available: https://drotek.com/shop/en/drotek-parts/270-hmc5983-magnetometer-compass-sensor-board.html. [Accessed: 18-Mar-2017].

[99]   "SPI Block Guide," Motorola, Inc., v0.3.06, Feb. 2003.

[100]   "Arduino Due Board," *Arduino Project Hub*. [Online]. Available: https://www.arduino.cc/en/Main/arduinoBoardDue. [Accessed: 18-Mar-2017].

[101]   "Datasheet - HMC5883L 3-Axis Digital Compass IC." [Online]. Available: https://cdn-shop.adafruit.com/datasheets/HMC5883L_3-Axis_Digital_Compass_IC.pdf. [Accessed: 18-Mar-2017].

[102]   "Magnetic Sensor - MMC3416xPJ," MEMSIC, Inc, Rev C, Oct. 2013.

[103]   "MMC3416PJ-B User Guide," MEMSIC, Inc, Rev A.

[104]   "Overview | Adafruit TCA9548A 1-to-8 I2C Multiplexer Breakout | Adafruit Learning System." [Online]. Available: https://learn.adafruit.com/adafruit-tca9548a-1-to-8-i2c-multiplexer-breakout/overview?gclid=CjwKEAjwtbPGBRDhoLaqn6HknWsSJABR-o5sNUnxMMdEeSv7g9UDHwJ11mzCad26pTHO-3y2_sDStxoCMjjw_wcB. [Accessed: 18-Mar-2017].

[105]   "Standard C++." [Online]. Available: https://isocpp.org/. [Accessed: 19-Mar-2017].

[106]   Eclipse Foundation, Inc, "Eclipse - The Eclipse Foundation open source community website." [Online]. Available: https://eclipse.org/. [Accessed: 19-Mar-2017].

[107]   B. W. Kernighan and D. M. Ritchie, *The C Programming Language*, 2 edition. Englewood Cliffs, N.J: Prentice Hall, 1988.

[108]   "Velmex Inc - Positioning Systems for Science and Industry." [Online]. Available: http://www.velmex.com/. [Accessed: 19-Mar-2017].

[109]   "OpenGL - The Industry Standard for High Performance Graphics." [Online]. Available: https://www.opengl.org/. [Accessed: 19-Mar-2017].