Motion in Magnetic Resonance Imaging:

Sources, Solutions, & Tracker Design

A thesis submitted to McGill University in partial fulfillment of the requirements of the degree of Master's of Science in Bioelectrical Engineering

McGill

Samuel Lantela

Department of Electrical & Computer Engineering

McGill University, Montréal

July 18th, 2023

Contents

Acronyms

ADC Analog to Digital Converter. x, 30, 38, 39, 45–48, 77, 81

ECG Electrocardiogram. 5

FOV Field of View. 18

IMU Inertial Measurement Unit. ix, xiv, 29, 30, 46, 51–55, 67, 70, 84, 106, 108

LMS Least Mean Squared. 66–69

LSE Least Squares Error. 98

MCU Microcontroller Unit. x, 26, 29, 30, 35, 36, 38, 77, 81, 102, 108

MRI Magnetic Resonance Imaging. x, xiv, 1–7, 9, 12, 14, 15, 17, 19, 21, 22, 24–26, 28, 29, 31–33, 40, 41, 43, 44, 47, 48, 102, 105, 106

MSE Mean Squared Error. 61, 69, 91, 92, 107

PCB Printed Circuit Board. 83, 106

PDF Probability Density Function. xi, xii, 81–84, 86, 88–90

SNR Signal to Noise Ratio. 1, 4, 9, 11

Glossary

***k*-space** Spatial frequency domain (Fourier Domain) of images acquired in MRI. viii, 3, 5, 7, 9–13, 16, 17, 20–23, 105, 106

B-field Magnetic flux density, vector defining the density and direction of magnetic field lines in a region of space. ix, 32, 33, 40, 102

concomitant fields Magnetic fields directed in the x or y directions that invariably accompany the z-directed fields used for imaging. 14, 15

isocenter The position in the MRI scanner that experiences no gradients whatsoever, being located at the center of the magnet and bore. Acts as the point of reference around which imaging is performed. 15

pose The position and orientation comprising a total of 6 degrees of freedom; 3 position coordinates (x, y, z) and 3 orientation coordinates (pitch, roll, yaw). 8, 26, 72–74, 90, 91, 107, 108

quiescent Of an electronic circuit, sensor or amplifier output: the resting level of an output voltage or current in the absence of an input. 35, 48, 77, 78, 84

voxel The unit of MRI image resolution; a combination of the words Volume and Pixel, a 3D pixel. 10, 14, 16, 17, 19, 20

List of Figures

Abstract

Magnetic Resonance Imaging (MRI) is an invaluable medical tool in both clinical and research settings thanks to its multitude of available contrasts, being particularly useful for soft tissues. Unfortunately, patient motion during MRI can render images unusable, necessitating re-scans, slowing medical progress, and amounting to an enormous societal problem. The work presented in this thesis explores the challenge of accounting for motion in MRI and strives to provide a low-cost solution using a 6-axis Inertial Measurement Unit (IMU) in conjunction with an array of magnetic field sensors. Design of such an active electronic tracker system suitable for MRI is presented, including the safety constraints of such a device from the standpoint of magnetic forces, torques, and radio-frequency (RF) heating. State-space theory and sensor fusion resolve 6 states (3 orientation and 3 position coordinates) using a set of sensors with 9 degrees of freedom (3D accelerometer, 3D gyroscope and 3D magnetometer) for motion-tracking. The work underscores the importance of regulating the voltage supplied to the magnetic field sensors. Finally, the investigation introduces a *statistical observation* approach to Kalman filtering, demonstrating the promise of such an approach while identifying its challenges. The here-presented research implies the possibility of resolving position and orientation using only a 6-axis IMU and the *statistical observation* Kalman filter. However, further experiments, developments and calibration are required for the demonstration thereof.

Abrégé

L'imagerie à résonance magnétique (IRM) est un outil médical inestimable dans la clinique autant qu'en recherche grâce à sa multitude de contrastes disponibles, utile pour les tissus mous. Malheureusement, le mouvement du patient pendant peut rendre les images inutilisables, ralentissant les progrès médicaux et constituant un problème de société. Le travail présenté dans cette thèse explore le défi de la prise en compte du mouvement en IRM et s'efforce de fournir une solution à faible coût en utilisant une unité de mesure inertielle (UMI) à 6 axes en conjonction avec un réseau de capteurs de champ magnétique. Les méthodes de conception d'un système de suivi électronique actif pour l'IRM sont présenté, ainsi qu'une discussion sur les contraintes de sécurité d'un tel dispositif, du point de vue des forces et couples magnétiques, et d'échauffement par radiofréquence. 6 états (3 coordonnées d'orientation et 3 coordonnées de position) peuvent être déterminés à l'aide d'un ensemble de capteurs à 9 degrés de liberté (accéléromètre 3D, gyroscope 3D et magnétomètre 3D) pour le suivi de mouvement. Les travaux soulignent l'importance de réguler la tension fournie aux capteurs de champ magnétique. Enfin, l'enquête introduit une approche d'*observation statistique* du filtrage de Kalman, démontrant sa promesse tout en identifiant ses défis. La recherche présentée implique la possibilité de résoudre le mouvement en utilisant uniquement une UMI à 6 axes et le filtre de Kalman avec *observation statistique*. En revanche, d'autres expériences, développements et étalonnages seront nécessaires pour démontrer le fonctionnement.

Dedication

First and foremost, I would like to dedicate this work to my family. To my brother, Wayan, for showing me that there is no question too big or small, too simple or complex. To my grandmother, *Mummu*, who taught me that we can always dream of changing the world and that only those who try ever come close. To my dad who taught me the importance of asking questions and the importance of endeavouring for the better. To my mom who taught me that there is always a way forward and that obstacles are only suggestions. To my elder brothers, Neal and Daniel, who inspired me from a young age with their intellect and wit. To my grandmother, Françoise, who taught me the importance of the little things. Finally, to my grandfathers, Erkki and Roch, who I wish I had more time to know.

I'd also like to thank my partner, Florence, for her constant loving support, reminding me of the importance of caring for oneself and others. I'd like to thank more friends than I can name for inspiring me to think big, encouraging me to see the world from a new lens, making me laugh, humouring me, hearing some out-of-the-box ideas, and bringing me back to the box when I needed it. Thank you my friends for being my chosen family.

Finally, thank you to my research team. Thank you Reza for helping me find the right problem and guiding me toward the right solutions. Thank you Milica for being my navigator, moving me ever closer to where I want to be, even before being my supervisor; your spirit and humour are an oasis in engineering. Thank you Will for helping me with your knowledgeable mind and skilled hands; I cherished our conversations, no matter the delay in our work.

Acknowledgements

I'd like to acknowledge the McGill University Health Centre (MUHC) at the Montreal General Hospital site which allowed me to use the MRI scanner to complete my research. Specifically, thank you to the research platform manager, Paule Marcoux-Valiquette, for your supervision during numerous experiments in the MRI. Also, thank you, William Mathieu, for instructing me on how to use the MRI, supervising experiments, and for your insights into design challenges. Thank you, Dr. Reza Farivar for your role as my research co-supervisor; your knowledge of the field and your abounding ideas on possible improvements guided my focus from the very beginning. Thank you, Ourania Tsatas for your support of the BRaIN research team, myself included; we all know we can count on you.

I'd also like to thank my co-supervisor, Dr. Milica Popovic, for your continued and thorough support throughout this process. The regular check-ins and writing schedules kept everything on track and made things much more manageable. Finally, I'd like to thank McGill University as a whole for supporting me with a body of knowledge and a community of experts trying to make the world better.

I'd like to acknowledge the funding I received through my supervisors and the Natural Sciences and Engineering Research Council (NSERC) that made my research possible.

Chapter 1

Introduction and Motivation

Patient motion violates many assumptions necessary for MRI spatial encoding, causing artifacts. These artifacts manifest as ghosting, blurring, geometric distortions or seemingly degraded SNR [1], affecting the diagnosis quality, necessitating re-scans, and slowing clinical and investigative progress [3]. While there are optimized pulse sequences that mitigate motion such as a modified MP-RAGE [4], state-of-the-art motion correction techniques are prohibitively expensive and require meticulously positioned cameras and markers. The most promising techniques for maximizing MRI patient throughput are the prospective motion correction modalities [2, 5–10] because of their ability to continuously update sequence parameters based on the patient's position and orientation.

It is estimated that motion in MRI costs 115,000 USD in forgone revenue per scanner per year [11]. Furthermore, severe motion artifacts rendering MR imaging unsuccessful are so frequent that an estimated 19.8% of scans are repeated [11]. Usually, this means repeating

the sequence when motion artifacts are found by technologists, lengthening the exam time. This slows the diagnostic process, hinders progress in research, and jeopardizes lives when patients do not receive timely treatment. According to the Canadian Agency for Drugs and Technologies in Health (CADTH) there were 366 MRI units in Canada as of 2017, assuming only 1 scanner per unit. This translates to Canadian taxpayers losing over 42 million USD every year due to motion in MRI. Furthermore, the CADTH estimates that 1.86 million MRI exams occur annually in Canada, and this figure is always on the rise. Assuming that correcting motion would increase this figure by 25% (eliminating the roughly 20% rescan rate) there could be 465,000 more MRI exams per year in Canada. Even if only 1% of these exams proved to be critical to the patient's health, this would imply that 4,650 lives could be saved annually by correcting for motion artifacts. Evidently, motion in MRI remains a formidable challenge despite the decades of research aimed at correcting it.

This thesis explores the feasibility and efficacy of low-cost MRI motion tracking. The work reported aims to improve the availability of MRI motion trackers, lightening the societal burden of MRI motion artifacts. This research contributes to the field in that it is the first attempt to resolve motion using only low-cost accelerometers, gyroscopes, and magnetometers. Chapter 2 covers background information on motion in MRI and reviews the literature on motion artifact mitigation and correction solutions. Chapter 3 describes in detail the design of a low-cost MRI active motion tracker. Results are discussed in Chapter 4, while Chapter 5 offers concluding remarks and pointers for future work.

Chapter 2

Background

As we will demonstrate in this chapter, motion in MRI corrupts images by rotating and modulating an image in the spatial frequency domain, in the "k-space". k-space is the 2D or 3D spatial Fourier domain in which MRI data is acquired. The integer k represents the spatial frequency of a discrete Fourier transform. Planar imaging sequences have 2D k-spaces (representing one frequency-encoding and one phase-encoding direction), whereas, non-planar imaging sequences (i.e. not slice-selective and containing a second phase-encoding dimension as a consequence) have 3D k-spaces. k-space is a discrete Fourier transform in multiple dimensions: every point in k-space corresponds to the magnitude and phase of a spatial frequency component of an image. As we will explain in the present chapter, k-space is at the center of the MR imaging process; it explains the basis of motion artifacts, how to identify them, and how they could be corrected given the parameters of motion.

2.1 Motion Mitigation

The simplest approach to addressing motion in MRI is motion mitigation (AKA motion prevention), done in a number of ways [12]:

- Foam restraints/pads mitigate motion by increasing patient comfort and by making motion somewhat difficult. This is very simple and cost-effective.
- In pediatric MRI, the use of mock scanners that simulate the experience of being in an MRI can determine whether a child could stay relatively still during the exam.
- Unconsciousness, induced or natural, is an effective means of mitigating motion.
  - Sedation (induced) is not an option for most functional scans because it can change brain activity patterns. Sedation is also costly and carries certain risks.
  - Sleep (natural) is used most often when imaging infants, where feeding before the scan provides natural unconsciousness. However, patients may move while asleep.
- Bite bars are common practice for mitigating motion in animals and were deemed effective in the early days of MRI but are not used because of their discomfort.
- In abdominal imaging, the non-rigid and non-linear motion caused by breathing is usually addressed by scanning during a breath-hold. This limits scan times to 10-20 seconds, putting an upper bound on the quality of images (SNR and or resolution).

- Triggering and Gating are more advanced methods used in cardiac and abdominal imaging to eliminate breathing and heartbeat artifacts thanks to their periodicity. Using an ECG to record the heartbeat, for example, either repetitions can be timed to recur at the same moment in the cardiac cycle (triggering) or the data can be reordered based on the moment at which it occurred during the cardiac cycle (gating).

2.2 Motion Tracking

No single method of motion tracking has been widely adopted, as they all have their strengths and weaknesses. Currently, commercial MRI systems use 2 primary forms of motion tracking: navigators and optical tracking systems [1]. Navigators [13–16] utilize oversampling of k-space to detect and extract motion, prior to partly correcting for it; this increases scan times but has the advantage of not requiring any additional hardware. Optical tracking systems [2,7,8] have become the standard for prospective motion correction since they require minimal hardware inside the bore; passive markers, containing a distinct geometric pattern, are placed on the patient and a camera records the motion from a distance to minimize interference with the scanner. However, optical systems require precise positioning and have yet to be widely adopted in neuro-imaging, their primary application.

In research, trackers of an electronic nature [5, 6, 17, 18], called active markers, are now being used for their speed, precision, and convenience. The challenge for active markers lies in their compatibility with MRI scanners; they must meet the appropriate safety standards

set forth by the ASTM (American Society for Testing and Materials) and or the ISO (International Standards Organization) [19]. Removing most electronics from a patient when they enter the scanner is a necessary safety measure so active markers require special compliance considerations.

Of primary concern are magnetically permeable materials (such as iron or nickel, the latter being a very common coating used in electronics) which can lead to translation forces and torques [19] due to gradients in the magnetic field intensity. (Magnetically permeable materials experience forces when there is a change in magnetic field intensity over distance, they will move toward the position and orientation where the magnetic field they experience is strongest.) In particular, ASTM F2052-06e1 [20] defines the "standard test methods for measurement of magnetically induced displacement force on medical devices in the magnetic resonance environment" which ensure that the translational magnetic forces pose a negligible projectile risk. Similar considerations are made for magnetically induced torques in ASTM F2213-06 [21]. However, even trace amounts of ferromagnetic material can lead to susceptibility artifacts, where the signal is distorted or lost completely near the permeable object because of the resulting magnetic field non-uniformity. ASTM F2119–01 [22] defines a "standard test method for evaluation of MR image artifacts from passive implants" to mitigate the loss of signal near implants, its methods can be extended to tracking systems that are fixed to the patient's body (not implanted). Finally, RF heating is a concern for the use of electronics in MRI since metallic objects will experience

eddy currents in the presence of radio waves. ASTM F2182–02a [23] defines standard test procedures to tackle the RF-heating concern. Altogether, these testing standards ensure that risks are minimized while allowing active markers to be used in the MR environment.

It is worth noting that the occurrence of severe image artifacts when using active marker tracking systems can defeat the purpose of said systems. If the artifacts induced by magnetically permeable materials in the tracker degrade image quality more than the motion itself then there is evidently no use for such a tracker. Therefore, the minimization of ferromagnetic materials in active trackers is paramount for both safety and performance.

2.3 Motion Artifacts: Overview

MRI uses precise spatial variations of the magnetic field strength, called gradients or gradient fields, to encode spatial information. These gradients are pulsed in order to separate the 3 spatial dimensions. The precise timing and amplitude of these gradient pulses depend on many factors including the type of sequence (spin echo or gradient echo), the desired contrast (e.g. Proton Density, T1 weighted, T2 weighted), the specific repetition and echo times (TR and TE), the role of the gradient (slice selection, phase-encoding and frequency-encoding/readout), and the k-space sampling trajectory (e.g. cartesian, spiral, radial). Since the gradient sequence assumes an immobile patient, motion disrupts the spatial encoding process in multiple distinct ways depending on how and when the motion happens (see section 2.3.2 for more details).

2.3.1 Classification and Manifestation of Motion Artifacts

Motion can be categorized along several features, each having a unique effect on the resulting image, leading to different kinds of artifacts. Godenschweger et al. [1] describe motion by categorizing it along features of type, occurrence, pattern, and direction as shown in Fig. 2.1. These motion categories affect how the image should be reconstructed if it is possible to correct the motion at all. For example, the type of motion, either rigid or deformable, affects the kind of transformations needed to correct an MR image.

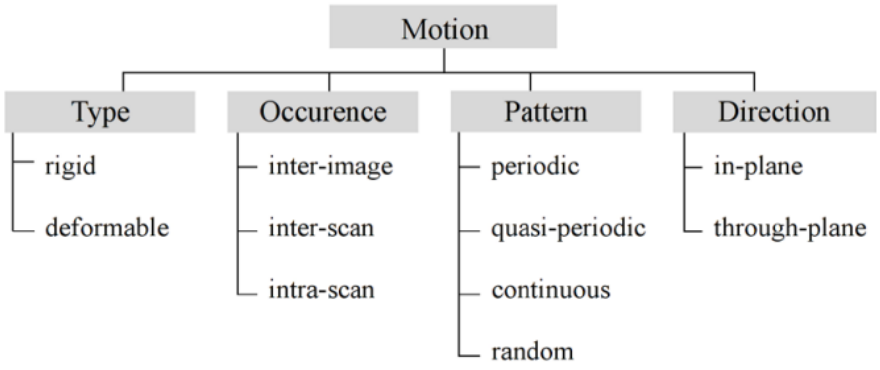

Figure 2.1: Categorization of motion along 4 features: type, occurrence, pattern, and direction. (Reproduced from Godenshweger et al. [1] with permission granted by the copyright owner IOP Publishing.)

Rigid body motion is easier to correct because it only requires 1 set of pose parameters (6 parameters: x, y, z, pitch, roll, yaw) and thus it can usually be corrected using rigid body transformations (3 translations and 3 rotations). **Deformable motion** requires at least twice the number of parameters as rigid body motion; either multiple points need to be

tracked leading to 6 parameters per point, and 6 additional transformation parameters need to be added to account for the skews that deformable motion introduces. We will focus on rigid body motion for simplicity and because it is the kind of motion most important for neuroimaging. Neuroimaging benefits from the fact that brain structures are confined to the skull. The head, skull and brain move together in a mostly rigid fashion. However, there is motion within the brain from the movements of blood and cerebrospinal fluid.

Occurrence (or timing) breaks down into inter-image, inter-scan, and intra-scan. **Inter-image motion** is the least disruptive because it happens between image acquisitions such as between the acquisition of separate slices in 2D slice-selective imaging (unfortunately, 2D imaging is often interleaved which complicates this issue). This means that inter-image motion in 2D MRI can usually be corrected by registration such that the skull (in the case of neuroimaging) is aligned along contiguous slices. In 3D imaging, inter-image motion can also be corrected through registration, where we need to align the copies of the object being imaged. (Why collect multiple 2D or 3D images? Multiple images may be collected to increase SNR through averaging). **Inter-scan motion** is motion which occurs between repetitions, i.e. between the echo of one repetition and the subsequent repetition's excitation. In 2D Fourier Transform (planar) imaging, for example, inter-scan motion is disruptive because lines of k-space acquired before and after motion will have incoherent spatial encodings. To correct for inter-scan motion, the k-space lines must be translated and rotated according to the parameters of the motion. Simple

translations could be corrected without the need to acquire more data, by phase shifting the k-space lines, but rotations must be corrected by rotating lines of k-space which can easily lead to data loss, as shown in Fig. 2.2 assuming that the motion is in-plane. Finally, **intra-scan motion** occurs between the start of excitation and the end of signal acquisition (echo) making it the most disruptive. Intra-scan motion manifests in many ways depending on when exactly the motion occurs, that is during which gradient pulses. Assuming the motion direction is in-plane, intra-scan motion disrupts the phase encoding by exposing a voxel to a different gradient intensity over time, making the phases inconsistent.

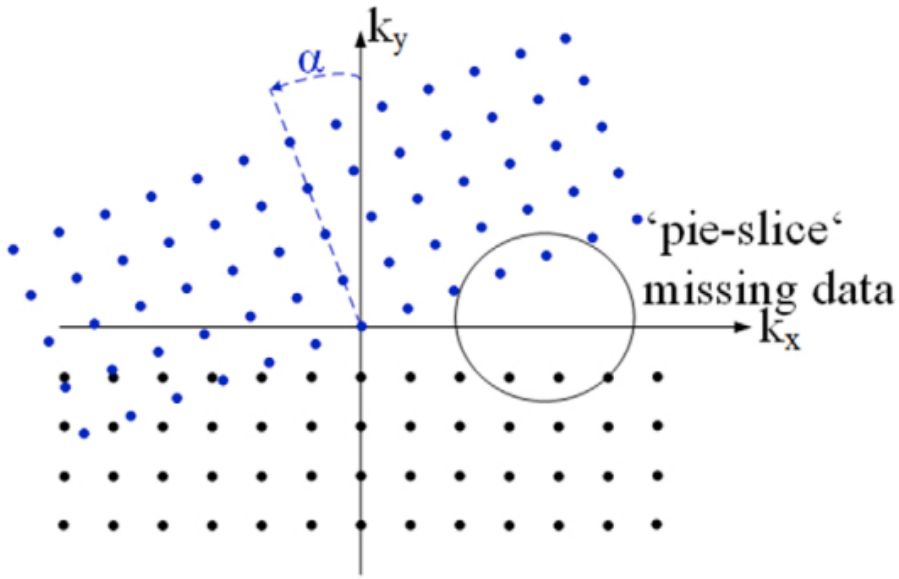

Figure 2.2: Effect of an in-plane rotation by angle α on k-space encoding (reproduced from Godenshweger et al. [1] with permission from IOP Publishing). Demonstrates how the lines of k-space acquired after the motion (blue) are rotated relative to those acquired before the motion (black) which leads to a ”pie-slice” hole in k-space sampling.

The next feature of motion is *pattern* categorized into periodic, quasi-periodic, continuous, and random, as shown in Fig. 2.1. **Periodic and quasi-periodic motion**, such as rocking or consistent nodding, creates ghosting artifacts where there are copies of the image overlaid on each other such as in Fig. 2.3 *l*); at small magnitudes, this will manifest as blurred edges and ripples along the direction perpendicular to the motion. **Continuous motion** is characterized by a constant translation or rotation through time, such as a drift in head position. This leads to a smear/blur in the direction of motion such as in 2.3 *j*) and *k*). Continuous rotations manifest similarly by smearing in the direction of rotation, being more pronounced further from the center of rotation as in 2.3 *i*). Finally, **random-pattern motion** can have characteristics similar to the other patterns. However, because it contains a stochastic mixture of motions, the artifacts add incoherently so they blur edges and degrade SNR.

The last categorization of motion is *direction*, which is broken down into in-plane and through-plane. **In-plane motion** is easier to correct because the excited volume stays inside the slice plane [12], so it preserves the slice selection of 2D imaging. Thus, in-plane motion guarantees that the k-space data can be corrected by rotations and translations within the plane of the 2D slice/image. **Through-plane motion**, either by rotation or translation, violates the assumptions made in 2D imaging. In 3D imaging, there is no concept of through-plane motion because the whole volume is excited, meaning that rotations and translations are theoretically sufficient to correct motion in 3D imaging (like in-plane motion in 2D).

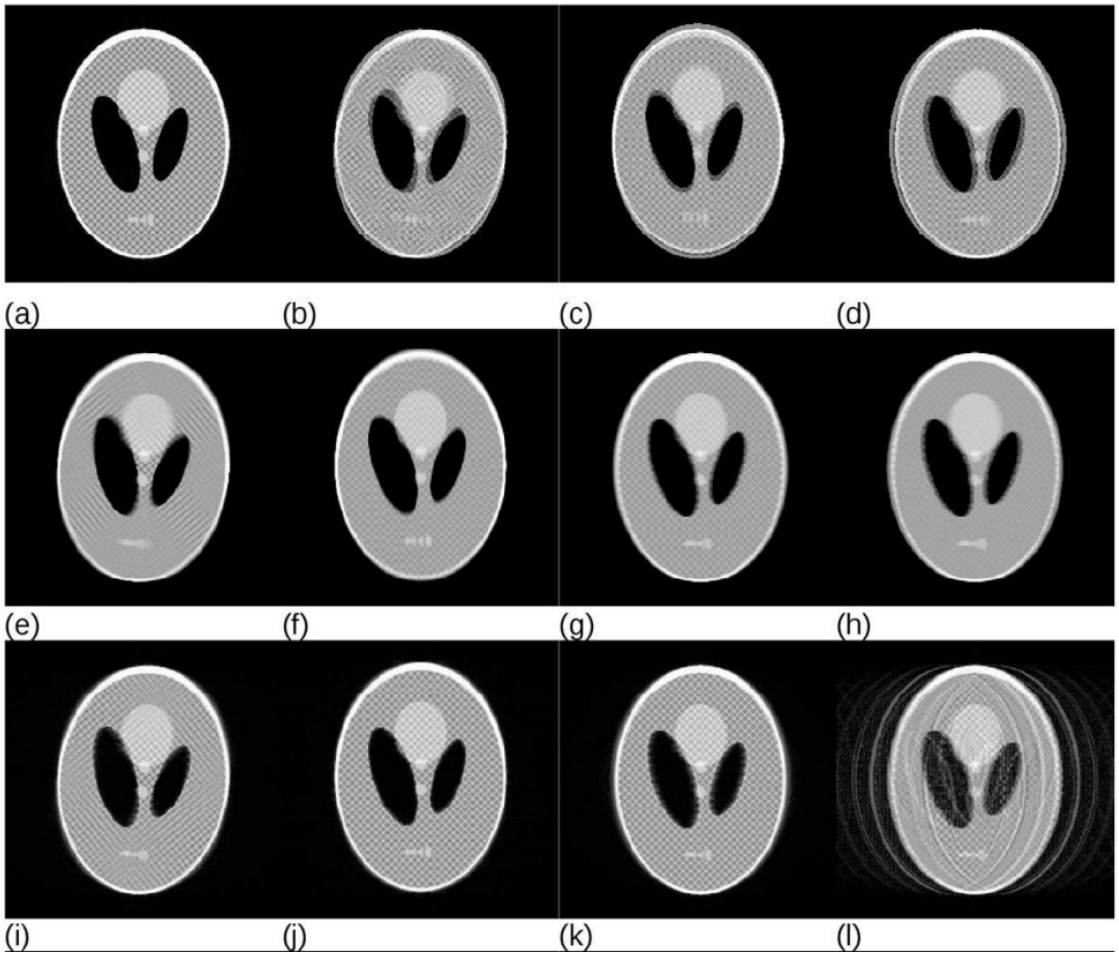

Figure 2.3: Examples of motion patterns effect on artifacts: a) Shepp Logan digital phantom, b) overlaid first and last phantom images of a continuous 10° rotation, c) overlaid first and last phantom images of a continuous vertical translation, d) overlaid first and last phantom images of a continuous horizontal translation, e) photography-equivalent with long exposure during rotation, f) photography-equivalent with long exposure during continuous vertical translation, g) photography-equivalent with long exposure during continuous horizontal translation, h) photography-equivalent with long exposure during periodic horizontal translation, i-l) simulated MRI with a linear k-space ordering of the same motion types as e-h). (Reproduced from Zaitsev et al. [12] with permission granted by the copyright owner John Wiley and Sons.)

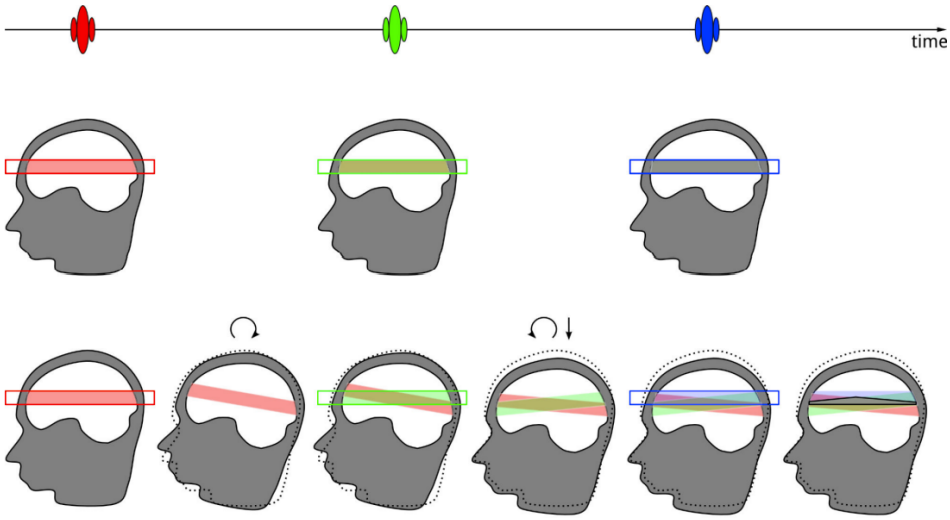

Figure 2.4: Example of through-plane inter-scan motion and how it affects the reconstruction of a 2D image/slice by exciting different volumes of tissue, the head in this case (reproduced from Zaitsev et al. [12] with permission granted by the copyright owner John Wiley and Sons). The excited slices are shown in the colour corresponding to their respective RF excitation pulse (red, green or blue). This demonstrates how k-space is sampled in different slices of the brain when the motion is through the plane of the slice, leading to inconsistent signals and potential data loss.

In the intra-scan occurrence of through-plane motion, some excited tissue moves out of the slice and unexcited tissue moves into the slice leading to a corrupted signal that is impractical to correct. Inter-scan occurrence of through-plane motion leads to the generation of undesired signals because the excited volume is different for subsequent excitations as shown in Fig. 2.4. Thus k-space reconstruction of the desired slice is disturbed; there will be signal loss within certain regions of the desired slice while the signal from neighbouring slices will be included in the current slice [1]. In the inter-image occurrence of through-plane motion,

data collected for one slice/image may actually correspond to data from other slices, and the slices may not be contiguous. For example, if a translation by one voxel width occurs in the slice-selection direction between the acquisition of two slices (inter-image through-plane translation), there will be one slice that is never imaged while another slice will be imaged twice. The distinction between in-plane and through-plane motion disappears when using 3D imaging techniques since the entire volume is excited, simplifying correction.

2.3.2 Mathematical and Physical Basis of Motion Artifacts

The Larmor equation describes the relationship between the imposed magnetic field and the frequency of nuclear precession, in general as a function of time t:

$$\vec{\omega}(x, y, z, t) = \gamma \vec{B}(x, y, z, t), \tag{2.1}$$

where $\vec{\omega}(x, y, z, t)$ is the radial frequency of precession (in radians per second or $\frac{\text{rad}}{\text{s}}$) of a nucleus in a magnetic flux density $\vec{B}(x, y, z, t)$ (in Tesla or T) at the position of the nucleus, and γ is the gyromagnetic ratio of the nucleus of interest being targeted by the MRI ($\frac{\text{rad}}{\text{s}\cdot\text{T}}$).

Generally, the Larmor equation is simplified for the purposes of MRI by recognizing that the magnetic fields used for imaging are strictly z-directed (concomitant fields are ignored).

$$\omega(x, y, z) = \omega_z(x, y, z) = \gamma B(x, y, z) = \gamma B_z(x, y, z) \tag{2.2}$$

Since gradients perform spatial encoding in MRI, we extend the Larmor equation to be a function of space, accounting for the spatial variations in the magnetic field, as follows:

$$\omega(\vec{r}) = \gamma(B_0 + \vec{G} \cdot \vec{r}); \quad \vec{G} = [G_x, G_y, G_z]^T, \tag{2.3}$$

where $\vec{r} = [x, y, z]^T$ is the vector defining the Cartesian location of the point of interest with respect to the isocenter. B_0 is the magnitude of the z-directed static magnetic field. G_x, G_y, G_z are the gradient field components altering the strength of the z-directed magnetic field in the x, y, and z directions respectively (in T/m). Far from the isocenter $\vec{r} = (0, 0, 0)$, there are concomitant fields but they only change the precession frequency by 52 ppm at 1.5 T with a gradient amplitude of 10 mT/m and a 20-cm distance from isocenter [24] so they are typically ignored for small to medium imaging volumes.

Since the MRI system will demodulate the received signal with respect to the excitation frequency $\omega_0 = \gamma B_0$, we extract the frequency shift imparted by the gradients:

$$\Delta\omega(\vec{r}) = \gamma\vec{G} \cdot \vec{r}. \tag{2.4}$$

This can be integrated over time to get the phase shift due to the accumulated gradients:

$$\Delta\phi(\vec{r}) = \gamma \int \vec{G} \cdot \vec{r}\, dt. \tag{2.5}$$

In the case of motion-free imaging, this would be sufficient to design (gradient) pulse sequences to perform spatial encoding. Spatial encoding works by manipulating the relative phase $\Delta\phi(\vec{r})$ as a function of position $\vec{r}$ to produce an echo whose analytical signal represents a specific spatial frequency component in the k-space of an image. With each repetition, the variations in the gradient pulses' intensities or durations will change dephasing between voxels and k-space can be filled line-by-line. However, the above equation implicitly assumed that position is not a function of time. To model motion, we must recognize that the phase shifts (relative to the isocenter) are essentially those of individual voxels (labelled *i,j,k* to demonstrate the voxel's imaging location). We admit that there is dephasing within a single voxel (intra-voxel dephasing) so the apparent phase of a voxel is essentially the average phase of said voxel. We must also acknowledge that the positions of these voxels are functions of time. Hence we can express the phase shift experienced by a voxel as a function of time:

$$\Delta\phi_{ijk}(t) = \gamma \int_0^t \vec{G}(t') \cdot \vec{r}_{ijk}(t')\, dt'. \tag{2.6}$$

Separating the effects of each gradient and assuming that z is the slice selection (SS) direction, y is the phase-encode (PE) direction, and x is the read-out (RO) direction. As an example, a simple 2D gradient echo sequence has the resulting phase at the end of the echo:

$$\begin{aligned}\Delta\phi_{ijk} = \gamma \int_0^{2T_{SS}} G_{SS}(t') z_{ijk}(t')\, dt' + \gamma \int_{t_{PE}^-}^{t_{PE}^+} G_{PE}(t') y_{ijk}(t')\, dt' \\ + \gamma \int_{TE-T_{RO}}^{TE+\frac{T_{RO}}{2}} G_{RO}(t') x_{ijk}(t')\, dt',\end{aligned} \tag{2.7}$$

where T_{SS} is the time for which the slice-selection gradient $G_{SS} = G_z$ is applied, this is equal to the duration of the RF excitation pulse. Since there is a slice-selection refocusing lobe, which we assume lasts T_{SS} with an intensity of $G_{SS} = -G_z/2$, the integral of the slice-selection component lasts for $2T_{SS}$. t_{PE}^{-} and t_{PE}^{+} are respectively the starting and ending times of the phase-encoding gradient $G_{PE} = G_y$. TE is the imaging sequence's echo time. T_{RO} is the duration of the read-out gradient $G_{RO} = G_x$. Since the read-out gradient has a negative lobe $G_{RO} = -G_x$ preceding it for refocusing (read-out preparation gradient), we start the last integral $\frac{T_{RO}}{2}$ before the start of the read-out at $TE - \frac{T_{RO}}{2}$.

As we can see, the net phase accumulated by each voxel is the sum of the integral of each gradient times the corresponding voxel coordinate as a function of time. In motion-free MRI, the phase accumulation of all the voxels in the slice selection and read-out directions are purposefully brought to zero at the echo time (TE) by the refocusing lobes of their respective gradients. The phase-encode gradient generates phase shifts of varying degrees at each repetition to sample k-space at the corresponding spatial frequencies. Looking at the above expression we can see how motion complicates the spatial encoding process, if not making it completely impossible. We can simplify the effect of motion by making a few reasonable assumptions:

- Motion is relatively slow compared to the echo time, eliminating the possibility of intra-scan motion. This is reasonable because common TE for MRI range from a few ms up to 100 ms. Even 100 ms would correspond to a 10 Hz oscillation, much faster

than anyone can nod their head, for example. (Granted, a fast jerking motion could approach this time scale and corrupt scans with high spatial resolutions.) This removes the time dependence for the spatial coordinates.

- Motion is either in-plane or the sequence is a 3D acquisition (i.e. without selective excitation). This is not necessarily true but accounting for through-plane motion ***retrospectively*** is exceedingly complicated because of missing k-space data. (However, prospective motion correction solves this by changing the slice plane in response to the motion.) For convenience, we will assume we are using a 3D acquisition sequence.

- Phase encoding in 3D happens simultaneously in both phase-encode directions. This is of little consequence because we assumed slow motion relative to TE.

- The tissue being imaged stays within the scanner's uniform B_0 field region, within the coils' sensitive volumes, and within the Field of View (FOV) of the imaging sequence. As long as the FOV of the imaging sequence has margins of a few centimetres around the head (for brain imaging), the magnitude of the motion should never cause the head to leave the FOV, and thus the signal will not be lost or aliased. (A tight FOV and an uncooperative patient mean that this margin may not be respected in typical clinical settings but it could be adopted at the cost of longer scan times).

- Gradients rise and fall so quickly that we can assume they are perfect step functions.

This will allow us to move the gradients outside their integrals as constants over their respective periods (T).

Under these assumptions, we can write our equation for phase accumulation in 3D MRI. Defining a new time quantity t_{RO} as the time from the start of the read-out (equals zero at the start of the read-out), after the negative preparation lobe of the read-out gradient. A voxel's accumulated phase as a function of gradient intensities and spatial coordinates is:

$$\Delta\phi_{ijk}(t_{RO}) = \gamma T_{PE}[G_{PE1}z_{ijk} + G_{PE2}y_{ijk}] + \gamma[t_{RO} - T_{RO}/2]G_{RO}x_{ijk}. \tag{2.8}$$

As will be shown in equation 2.11, the dephasing of spins along a given dimension is proportional to the spatial frequency (k) being sampled in that direction, the above expression represents how the spatial frequency in the scanner domain, where (i, j, k) have taken the place of (x, y, z), depends on the position of every voxel at the time of the scan. The equation for the signal obtained from the scan involves integrating the magnetization over the whole volume with a modulation term corresponding to the phase offset ($\Delta\phi$), as follows (note the similar structure to that of a Fourier Transform):

$$S(t) = \int_x \int_y \int_z M_{xy}(\vec{r}, 0)e^{-t/T_2(\vec{r})}e^{-i\omega_0 t}e^{-i\gamma\int_0^t \vec{G}(t')\cdot\vec{r}\,dt'}\,dz\,dy\,dx \tag{2.9}$$

$$S(t) = \int_x \int_y \int_z M_{xy}(\vec{r}, 0)e^{-t/T_2(\vec{r})}e^{-i\omega_0 t}e^{-i\Delta\phi_{ijk}(t)}\,dz\,dy\,dx, \tag{2.10}$$

where $S(t)$ is the time signal available to the scanner's receive coils (assuming a uniform sensitivity), $T_2(\vec{r})$ represents the transverse relaxation time of the tissue at the location $\vec{r}$, and $i = \sqrt{-1}$ is the imaginary unit not to be confused with the voxel indices ijk.

The signal will be demodulated by the scanner at acquisition time so we can remove the frequency modulation $e^{-i\omega_0 t}$. We can also introduce the spatial frequency coordinates:

$$k_r = \frac{\gamma}{2\pi} \int_0^t G_r(t') r_{ijk(t')}\, dt' = \frac{1}{2\pi} \Delta\phi_r \tag{2.11}$$

where r stands in for any spatial coordinate (x, y, or z). This equation shows the proportionality between the dephasing in a given dimension $\Delta\phi_r$ and the k-space coordinate k_r in that dimension. The demodulated receiver signal $\bar{S}(t)$ is thus:

$$\bar{S}(t) = \int_x \int_y \int_z M_{xy}(\vec{r}, 0) e^{-t/T_2(\vec{r})} e^{-i2\pi[k_x x_{ijk} + k_y y_{ijk} + k_z z_{ijk}]}\, dz\, dy\, dx. \tag{2.12}$$

It is vital to realize that the coordinates with subscripts (i, j, k), the coordinates of the voxels being imaged, are a 3D rotation and translation away from the actual (x, y, z) coordinates of the scanner (assuming a rigid body motion which is reasonable in neuroimaging but unreasonable in abdominal imaging). With this insight, we can use the rotation and translation properties of the Fourier Transform to infer the k-space trajectory with respect to the position and orientation during the first repetition (the origin of the desired image). For a change of position $\Delta\vec{r}$ and orientation $\Delta\vec{\theta}$, the Fourier transform will be modulated by the

translation according to the shift property and rotated in the same direction and to the same degree as the head is rotated according to the rotational symmetry of the Fourier transform. Therefore, we can write the new k-space coordinates in the head's frame of reference (k') as a function of the k-space coordinates that the scanner is trying to image (k).

$$\vec{k}' = \mathbf{R}(\Delta\vec{\theta})\vec{k} \odot \vec{s}(\Delta\vec{r}), \tag{2.13}$$

where $\odot$ denotes the element-wise multiplication,

$$\vec{s}(\Delta\vec{r}) = [e^{i2\pi k_x \Delta x}, e^{i2\pi k_y \Delta y}, e^{i2\pi k_z \Delta z}]^T \tag{2.14}$$

is the modulation vector representing the translation, $\mathbf{R}(\Delta\vec{\theta})$ is the arbitrary 3D rotation matrix (defined in Cole 2015 [25]), and $\vec{k} = [k_x, k_y, k_z]^T$ are the original spatial frequency coordinates. While the rotation term changes the coordinates of the acquired k-space data to fit the pre-motion frame of reference, the modulation term $\vec{s}(\Delta\vec{r})$ acts as a phase shift to the image data in k-space (which contains an unchanged magnitude and a modified phase).

As we can see, it is possible to account for motion in the image formation mathematics of MRI but only with ample details about the pulse sequence and with simplifying assumptions about the kind of motion (e.g. rigid, involuntary, small magnitude). In this section, we presented an example of how motion affects MRI acquisition and laid the foundation for a motion correction algorithm in 3D gradient echo imaging.

2.4 Solutions Found in the Literature

This section briefly examines solutions to motion in the MRI literature including full software solutions, and tracking system modalities (optical and active markers). The major classes of motion correction strategies are *retrospective* and *prospective*. *Retrospective* motion correction (RMC) is performed after the data has been acquired whereas *prospective* motion correction (PMC) happens during the scan itself. Prospective motion correction is more powerful since it can update the sequence in real-time, potentially enabling lossless imaging. However, PMC also requires specialized hardware and software capable of capturing motion at a high speed, relaying that information to the scanner, and adjusting gradients to acquire the correct line of k-space despite the motion of the patient.

2.4.1 Software Solutions

These techniques do not explicitly track motion but uncover it through redundancy in the acquired data or elaborate processing techniques. Motion-robust k-space sampling strategies like MOJITO, PROPELLER, Radial Trajectories, and Spiral Trajectories, seen in Fig. 2.5, have the common feature of sampling k-space with redundancy built-in. (One could point out that these techniques are not "pure software" because they change the pulse sequence, perhaps "firmware solutions" would be a more accurate label, but they are "software solutions" in the sense that they don't require any additional hardware.) Since the center of k-space represents the low-frequency information of the image, the position of

the patient can be extracted from the oversampled k-space center. This allows the data to be processed after acquisition to correct motion, making these a form of retrospective motion correction [15]. There is even a form of retrospective motion correction using Generative Adversarial Networks (deep learning) to solve the problem. [26].

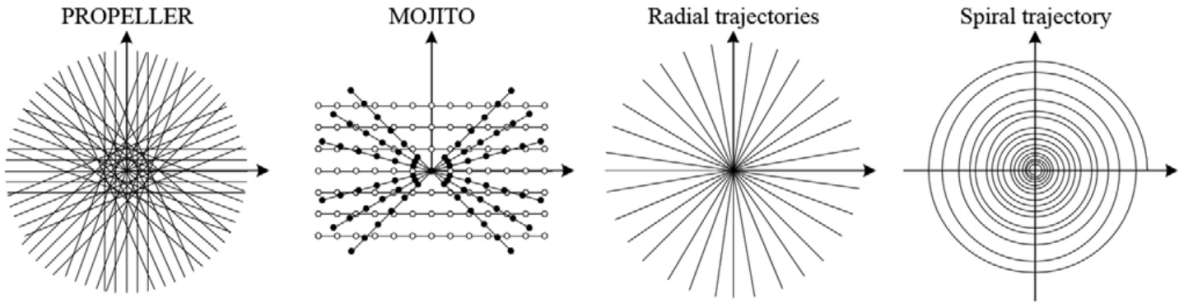

Figure 2.5: Motion-robust k-space trajectories (reproduced from Godenschweger et al. [1] with permission granted by the copyright owner IOP Publishing).

However, not all software-based approaches are retrospective. Prospective motion correction can also be done in software using, for example, 3 orthogonal 2D spiral navigator acquisitions and an extended Kalman filter [27]. These software (or firmware) techniques are testaments to the immense research interest in machine learning and other adaptive algorithms for correcting motion. The issues with software correction are:

- Redundant sampling of k-space can lengthen scan times.
- High processing power is required to achieve the more elaborate of the correction algorithms such as Generative Adversarial Networks [26].

- Navigators can affect the available contrast of the sequence by altering the time between repetitions and by reducing the longitudinal magnetization [28] because of their added excitations. Navigator sequences can be carefully designed to keep this effect to a minimum (roughly 3% contrast difference with parent sequence [28]).

2.4.2 Hardware Tracking Modalities

These techniques use some kind of device or physical arrangement (such as a camera and optical marker) to determine position and orientation at any given time. These are particularly challenging to implement given that the MRI is a hostile environment to virtually all electronics. Furthermore, the high field strength and RF power in an MRI make any metallic device a potential safety hazard. Electronic trackers are therefore built with caution to the dangers of a scanner or trackers employ optics to circumvent the hassle.

The promising research of van Niekerk et al. [5, 6, 17] featured active markers using the static magnetic field, inertial measurement units, and gradient pulse sensors to capture motion and prospectively correct it. Gholipour et al. [14] used a magnetic field sensor attached to the head to track motion for brain imaging. Accelerometer-based motion correction specifically adapted for correcting respiratory motion [18] has also been used. For now, optical motion correction appears to be the most popular in research [2, 7, 8]. Optical motion trackers are composed of a camera and a marker with an optical pattern such as the one in Fig. 2.6.

Figure 2.6: Optical tracking system for prospective motion correction from Todd et al. [2]. (Reproduced in accordance with the associated Creative Commons CC-BY Licence.)

The vast majority of hardware tracker technologies do not penetrate the clinical sphere, let alone the commercial sphere. Unfortunately, the translation from research devices to commercial products is necessary if we want to tackle the societal consequences of motion in MRI. The literature seems to point to high cost and difficult integration as the main impediments to active marker motion correction. The high cost (on the order of $10,000+) should not be such an impediment because of the relatively larger cost of a scanner (on the order of $1 million+). However, the lack of generalized motion correction solutions (e.g. applicable to both rigid and non-rigid motion) means the purchase of multiple motion correction solutions is necessary, inflating costs. If the cost of primitive motion trackers could be reduced significantly, then the cost of specialized motion tracking systems could also be reduced. Hence, we aim to design a low-cost active motion tracker, a primitive building block for future systems. The next chapter explores our design process for such a device.

Chapter 3

Methods in Design of a Low-Cost Active Tracking System

3.1 Conception of an Active Tracking System

This section introduces a low-cost active tracking system composed of an Inertial Measurement Unit (IMU: senses 3D acceleration and 3D angular velocity), a Magnetometer Array (set of Hall-Effect sensors), and an Arduino MCU with a WiFi module for communication. This active marker device records the sensor data and communicates it to an external computer in the control room for processing. Fig. 3.1 demonstrates the optimal solution that this project aims to approach: a prospective motion correction system that can relay pose estimates to the MRI host computer in real-time to update sequence parameters, accounting for head motion in neuroimaging.

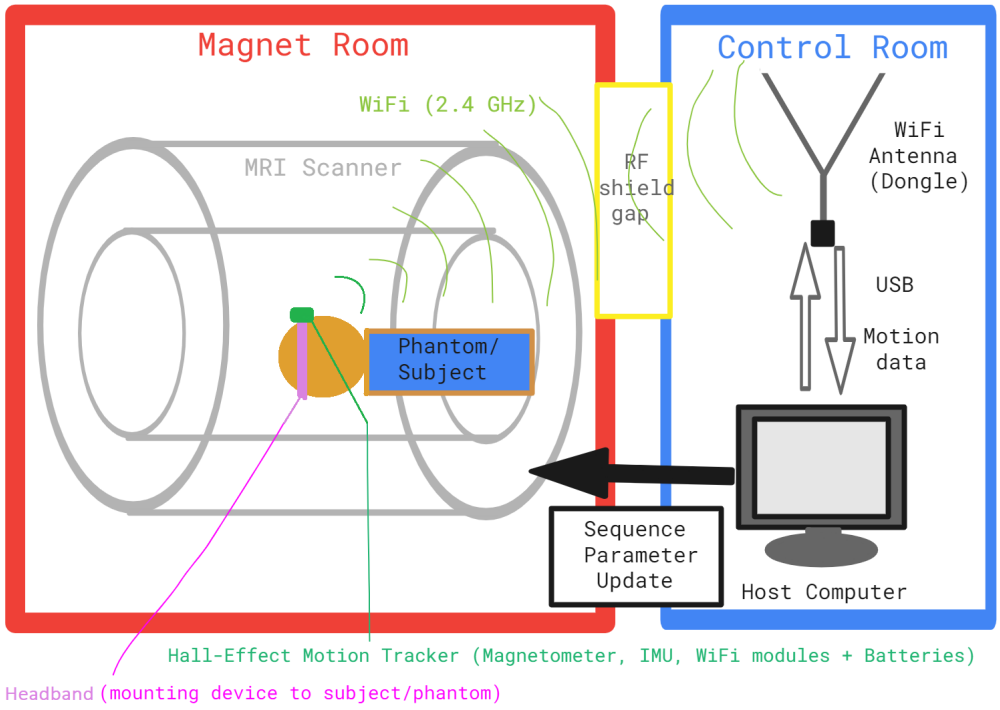

Figure 3.1: Functional system diagram describing a prospective motion correction system: motion is recorded by the tracker and relayed through WiFi to the control room where the data can be processed to allow the host computer to update sequence parameters. Since the magnet room is surrounded by RF shielding, WiFi communication is best accomplished through a gap in said shielding, in practice this is a waveguide built into the magnet room.

As mentioned in the previous section, the central challenges of active motion tracking systems that our design aims to overcome are:

- The high cost of components in active trackers. Particularly, magnetic field sensors (magnetometers) like the MV2 (starting at a unit price of roughly $900 CAD, see https://www.metrolab.com/products/magvector-mv2/ for more details), used by van

Niekerk et al. [5, 6, 17], are expensive because of the large dynamic range required to sense the main MRI B_0 field ($\pm$ 3 Tesla over a 180-degree rotation in a 3 Tesla scanner).

- The necessity of non-imaging Gradient Navigator pulses for localization such as in van Niekerk et al. [5, 6, 17]. This requires custom gradient sensing coils on the tracker and the insertion of pulses into the imaging sequence [29] which is inconvenient and can reduce the available turbo factor (slowing the imaging speed).

As a minimally viable solution, we aim to design a device capable of tracking motion accurately in the scanner for the duration of the longer exams (on the order of one hour). If we can produce this at a low cost (with components totalling less than $100) without the need for specialized sequences, this would be a large step towards a low-cost highly-integrable motion correction system. For simplicity, we will focus on building a device capable of performing motion correction retrospectively. If we succeed, then the low-cost prospective motion correction, our ultimate goal, will be within our grasp.

3.2 Hardware of the Tracking System

The first part of our solution is the construction of a device capable of accomplishing the tasks described previously: collecting data reflecting position and orientation metrics, and communicating said data wirelessly to an external computer for processing. For the collection part, we identified 2 types of sensors that deliver unique information about the device's

pose: an Inertial Measurement Unit (IMU) and a set of Hall-Effect sensors. Because of the strong magnetic fields present in an MRI, it becomes evident that any tracking modality must either be unaffected by magnetic fields or must use magnetic fields to perform their sensing. The static magnetic field's intensity, uniformity and stability in an MRI make it an ideal reference for tracking movement. Much like the use of the Earth's magnetic field in Global Positioning Systems (GPS), our magnetic field sensors will use the scanner's magnetic field as a point of reference for orientation tracking. On the other hand, the Micro-Electro-Mechanical System (MEMS) IMU is a device that can estimate the acceleration and rotational velocity independent of magnetic fields. It was reported by van Niekerk et al. [5,17] and van der Kouwe [29] that MEMS accelerometers provided much the same information as the navigator gradient sensing but the MEMS device's weak interactions with gradient fields could cause problems. However, Chen et al. [18] succeeded in making a MEMS-IMU-based respiratory motion tracker for the MR environment. Because of their common use in smartphones, IMUs are produced *en masse* at a low cost. This makes IMU-based sensors an attractive technology for our low-cost motion tracker.

3.2.1 Micro-Controller with Integrated IMU and WiFi Module

The centrepiece of our device is the Arduino RP2040 Micro Control Unit (MCU). The IMU (LSM6DSOX) is an integrated part of the MCU, using the I2C bus to communicate with the device's 32-bit processor core. The LSM6DSOX is a 6-axis IMU capable of 3-D

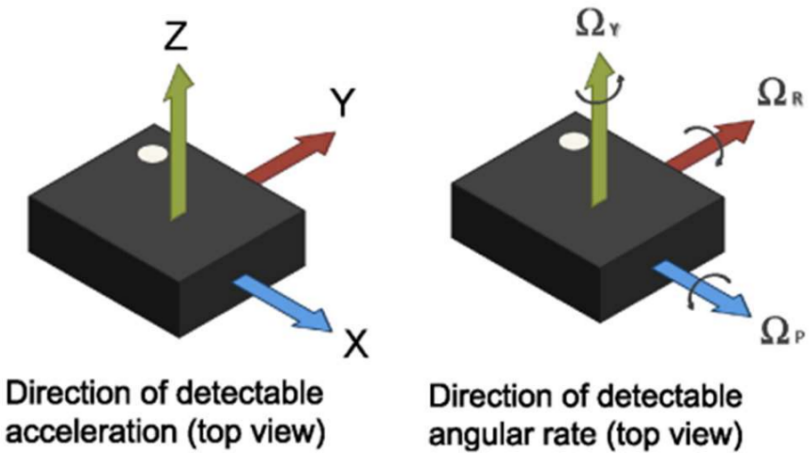

Figure 3.2: IMU detectable measurements consist of 3 linear accelerations (in gravitational constants: g's) and 3 angular velocities (in degrees per second: dps). Reproduced from the LSM6DSOX datasheet.

acceleration measurements and 3-D gyroscopic measurements (axes shown in Fig. 3.2) at programmable refresh rates and dynamic ranges. The Arduino RP2040 also contains an integrated NINA WiFi module and antenna (based on the ESP 32 MCU) that also uses an I2C bus to communicate with the device's core. Importantly, the Arduino RP2040 also has 6 available Analog to Digital Converter (ADC) channels that will be used to sample the Hall-Effect sensors. Finally, the Microcontroller has an internal Buck converter that converts DC voltages of 4-12 V to a regulated 3.3 V output used by the digital system (processor and peripherals). We will use this Buck converter because our device will be battery-powered, using a stack with an unregulated open-circuit voltage that can vary between roughly 6 and 8 V. The Buck converter is hence necessary to provide the digital systems with a regulated, nominal operating voltage of 3.3 V.

Conveniently, the Arduino RP2040 integrates several necessary components, simplifying our design process. In a future iteration, when we try to miniaturize our device, we will need to source parts separately to minimize the part count and device footprint, but during prototyping, we accept a larger device for the time and convenience it affords.

3.2.2 Hall-Effect Magnetometer Array

A particularity of our design is the use of monolithic Hall-effect sensors which are relatively low cost compared to the magnetometer MV2 used by van Niekerk et al. [5, 6, 17]. Therefore, we chose to try to design a multi-axis magnetometer specifically for this application since the state-of-the-art MV2 may not be necessary to extract the required information. Instead of using an expensive 3D magnetometer with a dynamic range of up to 10 Tesla, we are using 6 low-cost ($7 CAD each) linear magnetometers with dynamic ranges of up to 1 Tesla. As we will demonstrate, constraints on the magnitude of motion in MRI allow us to track the orientation (pitch and roll) of the head within $\pm 45^\circ$ of rotation, which is ample dynamic range since typical patients do not move deliberately during MRI but have small, involuntary movements. However, in extreme patients (e.g. Parkinson's, pediatric, or cognitively impaired) the motion may exceed this dynamic range.

The main challenge that using monolithic components elicits is that the lowest sensitivity Hall sensors on the market (MLX90251) reach a saturation flux density at $B_{sat} \approx 0.8$ T. This means that, in a 3-Tesla scanner, the sensor would be in saturation for most orientations. To

put this into tangible numbers, we need to consider the apparent flux seen by the sensors. We will call this B_n, the normal component of the main MRI magnetic field (B_0) incident on the Hall-Effect plane. For this purpose, we will explain the Hall Effect simultaneously.

3.2.2.1 Connecting Magnetic Fields to Measurable Voltages

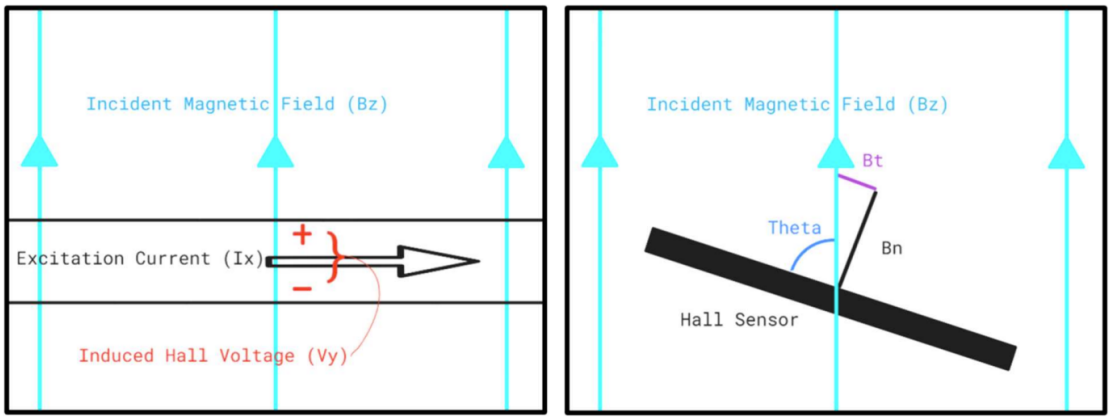

Figure 3.3: (Left) The Hall effect produces a net voltage due to Lorenz forces acting on charges travelling perpendicular to a magnetic field. (Right) The angle of the Hall sensor surface with respect to the incident field changes the induced voltage by changing the normal component of the B-field.

The Hall Effect is a phenomenon whereby charges moving in a current-carrying conductor produce a voltage in the direction perpendicular to both the current and magnetic field direction, as shown in Fig. 3.3. Starting from Lorentz force in the absence of an external electric field, the general equation of a Hall-Effect sensor is:

$$V_H = q\vec{v} \times \vec{B} = \frac{\vec{I}}{\rho T} \times \vec{B} = \frac{IB_n}{\rho T} = k_s B_n; \quad B_n = B_0 sin(\theta), \tag{3.1}$$

where k_s is the sensitivity coefficient which accounts for the free charge density ρ [C/m^3], the thickness of the current sheet T (thickness through which the B-field traverses [m]) and the current magnitude I (in Amperes [A]). The minimum sensitivity of the MLX90251 is 2.6 V/T. B_0 is the magnetic flux density, represented by the MRI's main field since the gradients can only vary the field strength by roughly 100 mT at most (assuming 200 mT/m gradients with a maximum distance from the isocenter of 0.5 meters) while the main field is 3 T.

What this means for our sensing apparatus is that the Hall sensors will saturate at a specific angle (θ_{sat}) depending on the saturation voltage of the sensor, which is a function of the sensors supply voltage (V_{DD}).

$$\theta_{sat} = sin^{-1}(B_{sat}/B_0); \quad B_{sat} = fV_{DD}/k_s, \tag{3.2}$$

where B_{sat} is the field strength that would saturate the sensor [T], which in turn is a function of the saturation fraction f [unitless or V/V] and the sensitivity k_s [V/T]. To model saturation correctly, we must first discuss our power supply and the I-V (current-voltage) characteristics of the MLX90251.

3.2.2.2 MLX90251 I-V Characteristics

The datasheet of the MLX90251 provides a graph of the I-V characteristics of the device from which we can devise an equivalent circuit model that approximates the device. Reading points off the curve, shown in Fig. 3.4, and performing a pseudoinverse regression, we find

the optimal coefficients (a and b) matching the following equation, which corresponds to a diode-resistor pair:

$$I_{DD} = aV_{DD} + b \implies I_{DD} = (V_{DD} - V_f)/R. \tag{3.3}$$

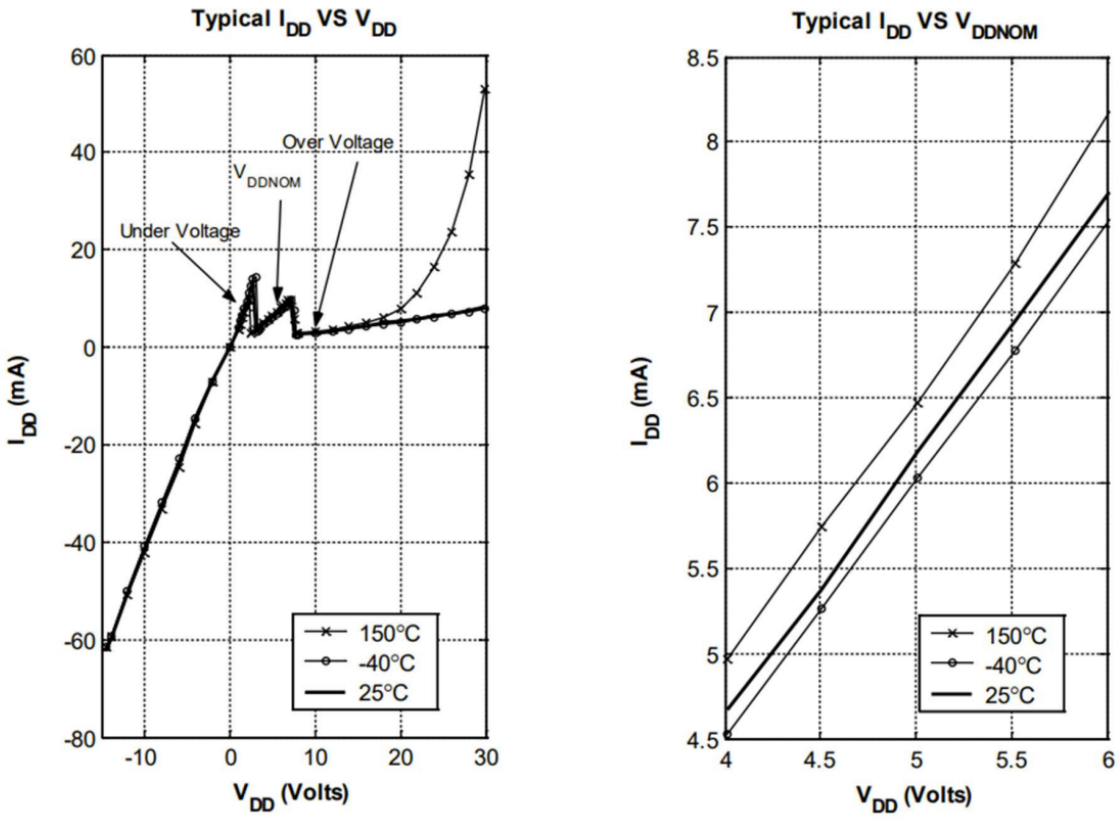

Figure 3.4: I-V characteristics of the MLX90251 Hall-effect sensor. The zoomed-in plot on the right at 25 ˚C was used as data points for the linear regressive circuit model since the datasheet states the range of operation as roughly 4.5 V to 6 V. Reproduced from the MLX90251 datasheet.

The two-parameter linear regression returned a model consisting of a series resistance and a diode, wherein $R = 665\ \Omega$ is the resistance and $V_f = 0.897$ V is the forward voltage

drop of the "diode". We were thus able to model how the sensor would respond to supply voltage fluctuations around its nominal 5 V operating voltage. We used this model to ensure the sensors can be supplied from an unregulated battery voltage. It turned out that the battery's internal resistance combined with the on-state resistance of our charge-discharge controlling relays lead to the output of our 2 × 3.7 V battery supply (V_{DD})) being roughly 5 - 5.2 V, well within the operating voltage of the Hall sensor. (More detail on the charge controlling relays can be found in the Printed Circuit Board subsection). Therefore, we simply supplied the magnetometer array with the dual battery stack.

3.2.2.3 MLX90251 Dynamic Range

As for the output of the MLX90251, the datasheet states that in the $\geq 10k\Omega$ pull-down load configuration the voltage output swing is in the 2 - 96% range of V_{DD}. Since the quiescent point of the sensor (when no field is present) is 50% V_{DD}, that leaves +46% and -48% V_{DD} for Hall-induced voltages. Taking the smaller of these two figures (46%), we can determine that the f parameter (saturation fraction) can be set to 0.46.

To determine the Hall sensors' dynamic range, we need to quantify how the source resistance (R_s from the internal battery resistance and the relay on-state resistance) and the current demands of the MCU (I_{MCU} which is drawn through the internal buck converter) will affect the voltage seen at the sensors' supply. We experimentally determined the source resistance to be 30 Ω and the current demands of the MCU to be 27 mA,

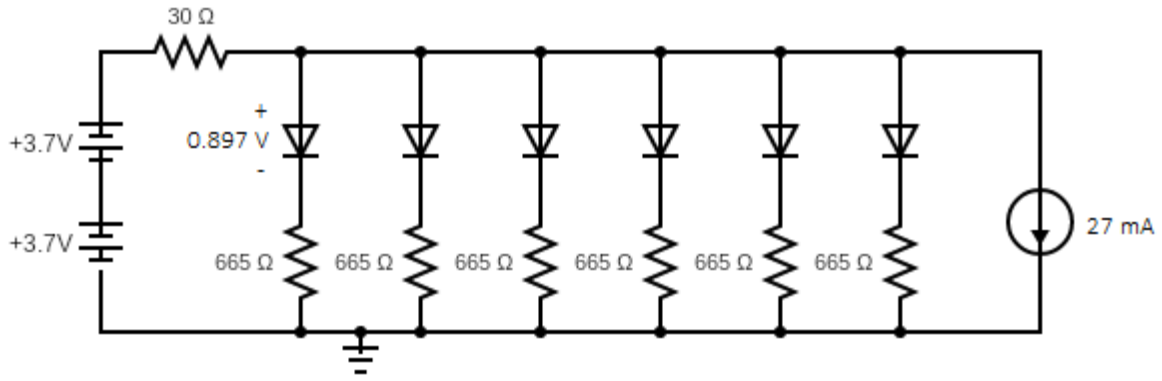

Figure 3.5: Equivalent circuit of power supply connected to magnetometer array, modelled as 6 diode-resistor pairs, and the buck converter of the MCU, modelled as a current sink.

summarized in Fig. 3.5. Combining equations 13 and 14 with the equivalent circuit shown in Fig. 3.5, we can write an equation that relates the battery voltage to the saturation angle. For 6 sensors (with IV characteristic defined by V_f and R) and 1 buck converter drawing I_{MCU} connected to a battery stack with an open circuit voltage of V_{OC} and internal combined source resistance R_s, it can be shown that the saturation angle (θ_{sat}) is:

$$\theta_{sat} = sin^{-1}\left(\frac{fV_{DD}}{k_s B_0}\right) \quad ; \quad V_{DD} = \frac{V_{OC} + 6R_s V_f/R - I_{MCU}R_s}{1 + 6R_s/R}. \tag{3.4}$$

Inserting the appropriate values ($B_0 = 3$ T, $k_s = 2.6$ V/T, $R_s = 30\ \Omega$, R = 665 Ω, $V_f = 0.897$ V, $f = 0.46$, $I_{MCU} = 27$ mA), we can approximate the sensor supply voltage V_{DD} and saturation angle θ_{sat} for various open-circuit battery stack voltages V_{OC}. Plugging in the extreme battery conditions of charge 8 V and discharge 6.8 V (± 0.6 V about the nominal

$V_{OC} = 7.4$ V), we see that $16.8° < \theta_{sat} < 20.2°$. While with the nominal 7.4V, $\theta_{sat} = 18.5°$. Therefore, we need many sensors to cover a significant area since each hall device is only in its linear range for $\pm\theta_{sat}$, yielding a nominal dynamic range of $DR = 37.0°$.

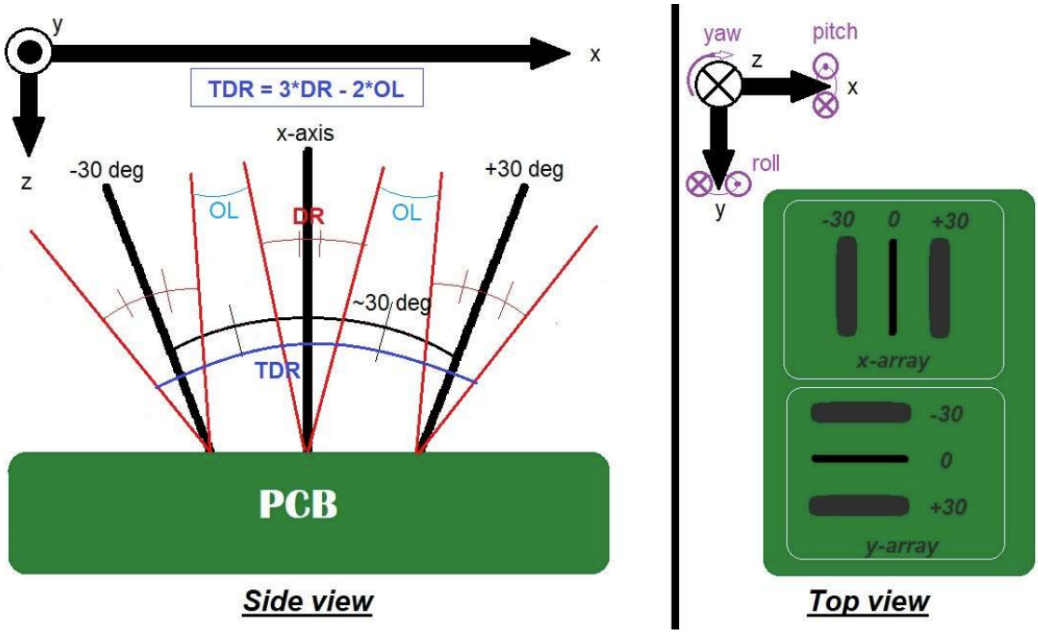

Figure 3.6: Magnetometer array diagram demonstrating how the 6 Hall sensors can be used to cover pitch and roll inside the total dynamic range (TDR), which depends on the overlap angle (OL). Note: Yaw cannot be determined by the magnetometer array since the main magnetic field is z-directed so all yaw angles look identical.

Therefore, by placing 3 sensors along the device's x-axis and 3 along the y-axis, we can cover a total dynamic range (TDR) of possible pitch and roll orientations depending on the angular offset between Hall-sensors as shown in Fig. 3.6. For simplicity, we will offset the axes of the magnetometers in steps of roughly 30° (allowing overlap OL, see Fig. 3.6),

resulting in a total dynamic range of:

$$TDR = 3DR - 2(DR - 30^\circ) = DR + 60^\circ. \tag{3.5}$$

In the end, we see that the overlap (OL) in individual magnetometer DRs reduces the overall TDR but ensures that the TDR is continuous at any above-mentioned battery voltage. We also note that there will be at least 1 sensor in each set (x- and y-arrays) which will be in saturation at any given pitch and roll. The x-array can determine the roll, while the y-array the pitch if and only if: $|\theta_{pitch}|, |\theta_{roll}| < TDR/2 = \theta_{sat} + 30^\circ$. Since the main magnetic field is oriented along the z-axis, all yaw angles look identical, thus the yaw cannot be determined from a magnetic field sensing array.

3.2.3 Printed Circuit Board

In order to interconnect all the necessary components, we designed a Printed Circuit Board (PCB), shown in Fig. 3.7, with sections accomplishing specific purposes, as follows.

- Connect the appropriate pins of the Arduino RP2040 MCU. Connecting: the 5V USB input to the charging circuit, the buck converter to the batteries for wireless operation, connect the ADC channels to the output of the Hall sensors.
- Control the charging and discharging of the batteries through relays that are triggered by the presence of the 5V USB input, as shown in Fig. 3.8.

- Prefilter the Hall sensor voltages using an RC filter to account for the sampling rate of the ADC, avoiding the aliasing of signals.

We will briefly summarize our design process for the aforementioned systems in this section.

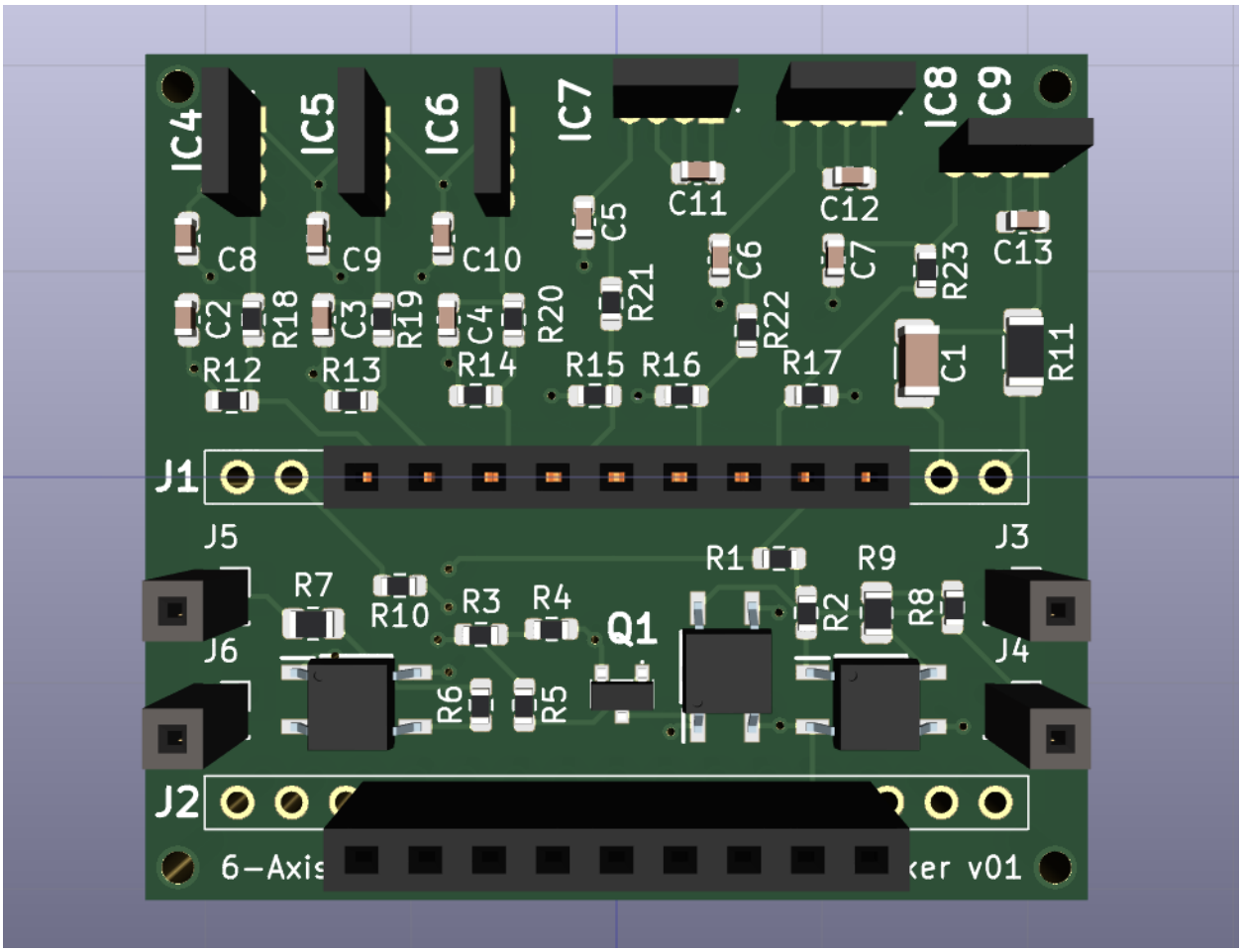

Figure 3.7: 3D view of printed circuit board virtual rendering. The dimensions of the board are roughly 4.5 cm × 4 cm × 2 mm.

3.2.3.1 MR Environment's Impact on PCB

The PCB was made with special attention to the length of the traces, a standard practice

for all PCBs. This is even more important for our application because the MRI's main magnetic field and gradients could induce voltages in the traces when the device moves. For a square trace loop (number of turns $N = 1$) with a width/length of x we can show, through Faraday's law, the magnitude of the induced voltage ϵ (for simplicity we assume a sinusoidal pattern of loop motion though this assumption is inconsequential to the magnitude):

$$\epsilon = -N\frac{d\Phi}{dt} = -B_0\frac{dA}{dt} = -B_0\frac{d}{dt}x^2 sin(\omega t + \phi_0) = -B_0\omega x^2 cos(\omega t + \phi_0), \tag{3.6}$$

where Φ is the magnetic flux through the loop [Wb], $A = x^2$ is the area contained by the loop [m^2], ω is the angular velocity of the loop caused by the motion of the head [rad/s], and ϕ_0 is the arbitrary initial phase [rad]. To get an idea of the maximum magnitude of induced voltages we will set the cosine term to 1, corresponding to the position of the greatest rate of change of the magnetic flux. The resulting induced voltage magnitude in a 3 T B-field for a loop width of 5 cm rotating at $\pi/2$ rad/s (upper bounds on x & ω) is:

$$|\epsilon_{max}| = B_0\omega x^2 = (3\text{ T})\left(\frac{\pi}{2}\text{ rad/s}\right)(0.05\text{ m})^2 = 11.8\text{ mV}. \tag{3.7}$$

As we can see, even for the largest possible trace on our device combined with a head movement faster than what we would expect from someone trying to stay still, the magnitude of induced voltages is relatively insignificant.

We must also consider the possibility of gradient-induced voltages on our PCB. The

maximum gradient slew rate of the Siemens MAGNETOM Prisma 3T MRI in our research institute is 200 $\frac{\text{T}}{\text{m}\cdot\text{s}}$. Assuming we are far (e.g. 1/4 m = 25 cm) from the isocenter, where the gradient swing is at its maximum, we can expect to see a rate of change of the magnetic field of 200(1/4) = 50 T/s corresponding to an induced voltage magnitude of:

$$|\epsilon_{grad}| = |\frac{dBA}{dt}| = A\frac{dB}{dt} = (0.05 \text{ m})^2(50 \text{ T/s}) = 125 \text{ mV}. \tag{3.8}$$

This figure is more significant than the B_0-induced voltage but is still relatively small considering that it would only affect the longest traces which are mainly power rails on the order of 5V. Truthfully, the most sensitive traces are those associated with the Hall sensors which have maximum lengths of 1.5 cm and a maximum thickness (through the PCB) of 2 mm. This corresponds to a maximum gradient-induced voltage of (0.015)(0.002)(50) = 1.5 mV. This is well below the noise floor and so can be deemed insignificant.

The final environmental concern for our PCB is RF-induced heating. The most significant part of the PCB for RF-induced heating is in the copper pours used for creating power planes and a ground plane. In our 4-layer PCB (we have a 2-layer version but it contains only 1 ground plane and is thus less susceptible to RF heating) we have 3 copper pours that occupy the whole area of the board (4.5 cm × 4 cm). The RF heating depends on the skin depth of the RF wave at the appropriate frequency, the Larmor frequency f of a proton at 3 T is 128 MHz. The skin depth of copper δ is calculated by:

$$\delta = \sqrt{\frac{1}{\pi \sigma_e \mu_m f}} = 5.8 \ \mu\text{m}, \tag{3.9}$$

where $\sigma_e = 58$ MS/m is the electrical conductivity of copper at 20°C, and $\mu_m = \mu_0 = 4\pi \times 10^{-7}$ H/m is the magnetic permeability of copper (roughly that of free space).

Since our board contains 2 copper pours of thickness 17.5 μm and 1 pour of thickness 43.2 μm we can establish that the RF power incident on its surface will be absorbed, because the combined thickness (2*17.5 + 43.2 = 78.2 μm) exceeds 13 times the skin depth. The power dissipated as heat P is proportional to the area of the pour A and the incident RF power S such that $P = SA$. Since RF power is limited by the Specific Absorption Rate (SAR) which has a typical maximum of 4 W/kg. For a large patient weighing 100 kg, this would translate to a time-averaged 400 W of absorbed RF power from P_{B_1}. Assuming that absorbed power accounts for only 40% of the transmitted power, we can estimate that the RF transmission has a total power of $P_{B_1} = 1$ kW. Since the B_1 field is circularly polarized within the transverse plane to produce a flip angle, the Poynting vector is (longitudinally) z-directed. Therefore the incident power density is the RF power divided by the circular cross-section of the cylindrical bore, as shown below. Finally, the maximum RF power deposited onto the PCB is the incident (z-directed) power density times the area of the PCB pours.

$$P_{PCB} = S_{B_1} A_{PCB} = \frac{P_{B_1}}{A_{bore}} A_{PCB} = P_{B_1} \frac{xl}{\pi r^2}, \tag{3.10}$$

where x and l are the width and length of the PCB's broad surface, and r is the radius of the MRI bore. For the Siemens MAGNETOM Prisma 3T MRI in our research institute, the bore diameter is 60 cm (so the radius is 30 cm). With the stated dimensions of our PCB (4.5 cm $\times$ 4 cm) and the RF power of 1000 W, we can estimate the rate of RF power deposition into our PCB as 6.37 W (worst case assuming the device is oriented with its broad side facing the z-direction).

The possible temperature increase of the PCB will be determined by its ability to dissipate some of that heat to the ambient environment. The amount of heat power ($Q/\Delta t$) that can be dissipated to the environment depends upon the following equation:

$$\frac{Q}{\Delta t} = kA(T_s - T_a), \tag{3.11}$$

where $k \approx 398 \frac{\mathrm{W}}{\mathrm{m \cdot K}}$ is the thermal conductivity of the copper, A is the area that can dissipate heat which is the total area of all PCB faces $A \approx 2wl$, T_s is the surface temperature, and $T_a \approx 20°C$ is the ambient temperature in the MRI. Setting this power equal to the power deposited to the PCB by the RF field, we obtain:

$$P_{PCB} = 6.37W = kA(T_s - T_a). \tag{3.12}$$

Solving for the surface temperature will give us the equilibrium temperature of the PCB copper, and thus the maximum possible temperature that can be reached during an

indefinite-length 1000 W RF transmission. We find $T_s = 24.4°C$ which is evidently a safe amount of heating, the PCB's own electronics will lead to more heating. We assume this amount of heating is safe for the patient since the temperature of the copper will remain near the ambient temperature, not nearly enough to cause burns.

3.2.3.2 Charging Control Circuitry

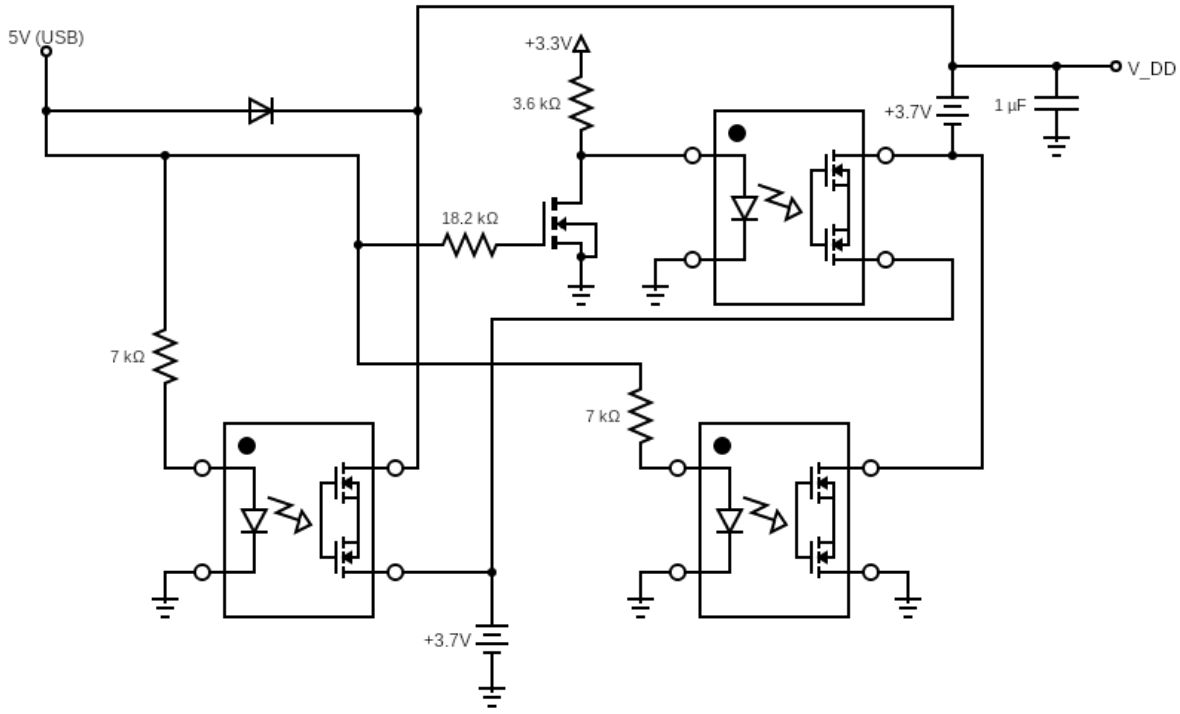

Figure 3.8: Circuit design of a solid-state relay-based charging controller for the wireless MRI motion tracker. The 3 relays are configured such that the batteries are connected in parallel when charging (when the 5V USB input is present) while the batteries are connected in series when the USB input is disconnected.

The fact that our device needs to be wireless obviated the need for a charging control circuit. Since the batteries we are using come with a battery protection circuit built-in, all we needed to do was set up a relay triggering system that would place the 2 batteries in parallel when charging, i.e when the 5V USB input was present, and in series when running on battery power. The final circuit is shown in Fig. 3.8. This enabled our Hall sensors to receive their nominal 5 V source voltage from 3.7 V batteries without a boost converter.

3.2.3.3 Hall Sensor Anti-Aliasing Filter

The final part of our PCB design is the RC filter that will band-limit the Hall voltage signals to the bandwidth of the ADC (sampling at $T_s = 9.6$ ms intervals, corresponding to $\sim$104 Hz sampling frequency). An added complication is that the Hall signals, which can reach 5V, must be voltage-divided before the ADC in order to coincide with the dynamic range of the ADC. The analog reference for the 4 ADCs on the RP2040 is 3.3 V while for the 2 ADCs on the ESP32 WiFi module it is 1.3 V. Therefore, we will use a voltage divided RC filter of the form shown in Fig. 3.9 where the ratio $R_B : R_B + R_A$ will be different depending on whether the ADC of interest is the RP2040's or the ESP32's.

The important consideration for anti-aliasing is that the time-constant $\tau = R_{eq}C$ is more than 5 times the sampling period (this is known as the 1% settling time), where R_{eq} is the equivalent resistance seen by the capacitor, i.e. the parallel combination $R_A//R_B$.

$$R_{eq}C = \frac{R_A R_B}{R_A + R_B} C \geq 5T_s \tag{3.13}$$

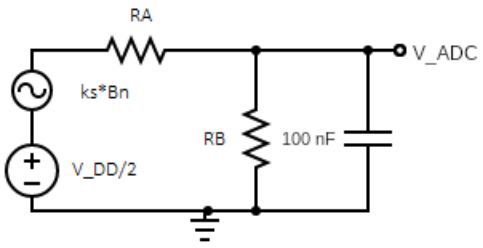

Figure 3.9: Anti-aliasing step-down filter between the Hall sensor output (left) and the ADC input (right).

Since we require non-magnetic components, we are limited in our choices and we simply found component values that met our constraints. We chose $C = 100$ nF for all ADC channels, $R_A = R_B = 1$ MΩ for the RP2040 channels, and $R_A = 4.22$ MΩ & $R_B = 820$ kΩ for the ESP32 channels. These values ensured that the voltages supplied to the ADCs were within their dynamic range while also satisfying equation 3.13.

3.3 Software of the Tracking System

The software can be segmented into 3 separable but interrelated tasks: Hall array pitch and roll estimation, IMU position and orientation estimation, and sensor fusion.

3.3.1 Hall Voltage Conversion to Pitch and Roll Estimates

The Hall sensor array described in section 3.2.2 will provide us with valuable information regarding the orientation of the tracking device. As we saw, the Hall sensors can use the main MRI field as a reference through which we can determine the pitch and roll of the device. However, the voltage output of each Hall sensor depends on the voltage rail delivered to the device which is unfortunately not regulated. Therefore, we need to pre-process the data we acquire from the Hall array before we can use its data to determine the pitch and roll.

The first step is to equalize the channels by applying a gain to each channel according to the resolution and the voltage divider circuits that were placed on the PCB, as shown in section 3.2.3. The channels connected to the ADC of the RP2040 are corrected by multiplying by a factor of (LSB: Least Significant Bit):

$$G_1 = \frac{3.3\text{V}}{2^{12}-1}\frac{1\text{M}\Omega + 1\text{M}\Omega}{1\text{M}\Omega} = 1.612 \text{ mV/LSB}, \tag{3.14}$$

considering a 12-bit resolution across a 3.3 V range. While the channels connected to the NINA WiFi module's ESP32 ADC are corrected by a factor of:

$$G_2 = \frac{1.3\text{V}}{2^{12}-1}\frac{4.22\text{M}\Omega + 820\text{k}\Omega}{820\text{k}\Omega} = 1.951 \text{ mV/LSB}, \tag{3.15}$$

considering a 12-bit resolution across a 1.3 V range.

The two gain correction factors can be stacked into a gain correction vector where the

second and third channels correspond to the ESP32's ADC.

$$\vec{G} = [G_1, G_2, G_2, G_1, G_1, G_1] \tag{3.16}$$

In this way, the voltage at the output of the Hall sensors $\vec{V}_H$ can be estimated by the element-wise multiplication ($\odot$) of the vector of Hall array ADC measurements $\bar{\vec{V}}_H$ such that:

$$\vec{V}_H = \vec{G} \odot \bar{\vec{V}}_H. \tag{3.17}$$

The next pre-processing step is to adjust for the supply voltage being fed into the Hall sensors, known as V_{DD} in section 3.2.2. As the batteries discharge there will be an exponential decay on the voltage supplied to the sensors, making V_{DD} impossible to estimate *a priori*. The value of V_{DD} can be estimated simply by assuming that at least one of the sensors in the array will be saturated at any point in time (this can be guaranteed if one sensor is tipped to face the full force of the MRI's main field). According to section 3.2.2, the sensor will saturate near $V_{max} = 0.96V_{DD}$ and the quiescent point of the sensor output is at $V_0 = V_{DD}/2$. Therefore, the Hall voltage deflections away from the quiescent point can be defined as:

$$\Delta\vec{V}_H = \vec{V}_H - V_0 \begin{bmatrix} 1 \\ ... \\ 1 \end{bmatrix} = \vec{V}_H - \frac{V_{DD}}{2} \begin{bmatrix} 1 \\ ... \\ 1 \end{bmatrix} = \vec{V}_H - \frac{max(\vec{V}_H)}{2(0.96)} \begin{bmatrix} 1 \\ ... \\ 1 \end{bmatrix}. \tag{3.18}$$

Finally, we can identify the best sensor data to use in each array (x and y) by considering that the sensor's reading will be most accurate near the center of its linear range. Therefore, for each of the arrays, we can determine the optimal voltage deflection to use according to the minimum absolute value of the voltage deflection. For the x-array:

$$\Delta V_{x_i} = min(|\Delta \vec{V}_{H_x}|) \tag{3.19}$$

where $\Delta \vec{V}_{H_x}$ is the subset of Hall sensor deflections coming from the x-axis array, and i is the index of the sensor that meets the above optimization. The equation for the y-array is equivalent, with the index j replacing i.

$$\Delta V_{y_j} = min(|\Delta \vec{V}_{H_y}|). \tag{3.20}$$

The last step which was introduced in section 3.2.2 takes the optimal Hall voltage deflection of each array and computes the pitch and roll as follows:

$$\hat{P} \equiv \hat{\theta}_{pitch} = sin^{-1}\left(\frac{\Delta V_{x_i}}{k_s B_0}\right) - \phi_i \tag{3.21}$$

$$\hat{R} \equiv \hat{\theta}_{roll} = sin^{-1}\left(\frac{\Delta V_{y_j}}{k_s B_0}\right) - \phi_j, \tag{3.22}$$

where k_s is the sensitivity of the Hall sensor (2.6 V/T), B_0 is the main field strength (3 T), and ϕ_i is the phase offset of the sensor at index i as discussed in section 3.2.2. $\hat{P}$ and $\hat{R}$

indicate an estimated value.

Using the accelerometer readings, we can also obtain an estimate of the device's yaw $\hat{Y}$. In fact, there are two equations that allow us to estimate the yaw using the estimates of pitch or roll plus the acceleration due to the gravity on one of the axes.

$$\hat{Y}_1 = sin^{-1}\left(\frac{g_x}{cos(\hat{R})}\right) \tag{3.23}$$

$$\hat{Y}_2 = sin^{-1}\left(\frac{g_y}{cos(\hat{P})}\right), \tag{3.24}$$

where g_x and g_y are the accelerations due to gravity on the device's x and y axes respectively, in "g's". Given our pseudo-stationary application, the accelerometer readings $a_x/9.81$ and $a_y/9.81$ are suitable estimates of g_x and g_y. In the end, we can combine both estimates to obtain an optimal estimate $\hat{Y}$.

$$\hat{Y} = \frac{1}{2}\left(\hat{Y}_1 + \hat{Y}_2\right) \tag{3.25}$$

3.3.2 IMU Theory and Pose Estimation

The data acquired by the accelerometer and gyroscope (IMU) are inertial measurements, i.e. they use a mass experiencing acceleration to determine readings. Linear acceleration estimates in x, y and z directions are intuitively inertial values. The gyroscope uses the centripetal acceleration resulting from rotational motion to determine angular velocity in the pitch, roll, and yaw directions.

The acceleration measurement $\vec{a}_m$ of the IMU is the sum of acceleration due to gravity $\vec{g}$, the Coriolis acceleration $\vec{a}_c$, and motion-related acceleration $\vec{a}_{net}$. The acceleration due to gravity depends, in fact, on the orientation of the device. Therefore, the gravitational term is shown as a function of the orientation vector $\vec{\theta}$ whose elements are pitch, roll, and yaw. The Coriolis acceleration component, which is the cross-product of angular and translational velocity, encapsulates the perceived acceleration experienced in a frame of reference when it rotated about an axis that is not the axis along which it is travelling. Thus, Coriolis acceleration is the result of the velocity direction changing with respect to the device's reference frame. Since our device is designed for a pseudo-stationary application, we expect that the Coriolis component of acceleration will be negligible since the velocities will be very small. Furthermore, if the device is truly stationary, then the net acceleration should also be negligible. Therefore, the accelerometer's measurement will be dominated by gravity.

$$\vec{a}_m = \vec{a}_{net_b} + \vec{g} + \vec{a}_c = \vec{a}_{net_b} + \vec{g}(\vec{\theta}) + 2\vec{\omega} \times \vec{v} \approx \vec{a}_{net_b} + \vec{g}(\vec{\theta}) \approx \vec{g}(\vec{\theta}), \tag{3.26}$$

where the subscript 'b' on the net acceleration indicates that it is in the IMU's "body" frame of reference. We will need to account for the fact that the acceleration and radial velocity sensed by the IMU are with respect to the tracker's "body" frame of reference, not the scanner's absolute frame of reference.

The definition that we use for the gravity vector $\vec{g}(\vec{\theta})$ stems from the decision to align with the scanner's reference frame. Hence, at $\vec{\theta} = \vec{0}$ (in line with the scanner's reference

frame) the gravity should be along the IMU's y-axis (later this is referred to as the v-axis to distinguish it from the global y-axis). This yields the following definition (in m/s^2):

$$\vec{g}(\vec{\theta}) = \vec{g}([P, R, Y]) = 9.81\,[cos(R)sin(Y) \;\; cos(P)cos(Y) \;\; cos(R)sin(P)]^T\,. \tag{3.27}$$

Unfortunately, the measurement that we obtain from the IMU will be contaminated with noise, will likely contain a constant offset drift term, and will likely contain a scaling error. Therefore, we could choose to write the accelerometer measurement as:

$$\hat{\vec{a}}_m = A(\vec{a}_{net_b} + \vec{g}(\vec{\theta})) + \vec{a}_{drift} + \vec{q}, \tag{3.28}$$

where $\vec{q}$ is a noise component coming from a random process which we will define later, $\vec{a}_{drift}$ is the accelerometer's fixed offset or drift, and A is the scaling constant, ideally equal to 1.

The gyroscope measures the angular velocity (of gyration) in a similarly noisy process to the accelerometer. The main difference is that the gyroscope doesn't have any "gravity analog" in the sense that there is no constant term which must be removed to get at the underlying inertial measurement. The gyroscope's measurement can be expressed simply as:

$$\hat{\vec{\omega}}_m = \vec{\omega}_b + \vec{r}, \tag{3.29}$$

where $\vec{r}$ is an additive noise (not to be confused with a position vector) and $\vec{\omega}_b$ is the true

underlying angular velocity (a vector because it contains rates of gyration in the pitch, roll and yaw directions) in the IMU's body frame of reference. We are not including any normalization constant or drift term in the gyroscope's measurement since the Hall sensor array will provide a means to correct these without explicitly defining them.

3.3.2.1 Aside: Frames of Reference and Rotation Matrices

Our system must deal with two frames of reference: the IMU's "body" frame of reference and the scanner's "global" frame of reference. The global frame of reference is the scanner's because we ultimately want to measure the motion of a patient with respect to the scanner. Furthermore, our Hall sensors will use the scanner's z-directed main field B_0 to determine pitch and roll angles. Lastly, we can choose to align our global y-axis with the direction of gravity, which is supposed to be orthogonal to B_0 (assuming that the scanner is level with respect to gravity and that the B_0 field is horizontal). The scanner's reference frame is thus fixed in space. On the other hand, the IMU and tracker have their own reference frame which depends on the orientation of the device with respect to the scanner's reference frame.

We will call the IMU's reference frame the "body" in accordance with the literature on IMU-based motion tracking [30–32] and the related Global Positioning System (GPS) theory. We will refer to this body reference frame as the UVW frame of reference to differentiate it from the scanner's proper XYZ frame of reference. The angular velocities in the body frame of reference will still have directions called pitch, roll and yaw but will be given the subscript

'b' to denote their definition within the body frame of reference.

When the IMU measures acceleration or angular velocity, it does so in whatever orientation it is in at the time of the measurement. Therefore, it may read an acceleration due to gravity of 9.81 m/s^2 along its "z-axis" (actually the w-axis) but according to the scanner's frame of reference, the gravity of 9.81 m/s^2 is on the global y-axis. This would imply that the IMU is actually rotated -90 degrees in the pitch direction (aligning the w-axis with the global y-axis). Defining a rotation matrix for this case is simple enough since it is a rotation around a single axis. However, when dealing with a rotation around an arbitrary axis, we must define a rotation matrix that takes as input the current orientation $\vec{\theta}$ and generates a matrix that rotates the XYZ coordinate frame to the UVW coordinate frame. Since it should be a proper rotation, i.e. the matrix has a norm of 1 and is full rank, the transpose of the matrix that satisfies $\vec{a}_{UVW} = \mathbf{R}(\vec{\theta})\vec{a}_{XYZ}$ will also satisfy $\vec{a}_{XYZ} = \mathbf{R}^T(\vec{\theta})\vec{a}_{UVW}$. Cole 2015 [25] proves that a rotation of magnitude (angle of the rotation) θ around an axis defined by the unit vector $\vec{u}$ is defined by the following matrix:

$$\mathbf{R}(\vec{u},\theta) = \begin{bmatrix} cos(\theta) + u_x^2(1-cos(\theta)) & u_x u_y(1-cos(\theta)) - u_z sin(\theta) & u_x u_z(1-cos(\theta)) + u_y sin(\theta) \\ u_y u_x(1-cos(\theta)) + u_z sin(\theta) & cos(\theta) + u_y^2(1-cos(\theta)) & u_y u_z(1-cos(\theta)) - u_x sin(\theta) \\ u_x u_z(1-cos(\theta)) - u_y sin(\theta) & u_y u_z(1-cos(\theta)) + u_x sin(\theta) & cos(\theta) + u_z^2(1-cos(\theta)) \end{bmatrix} \tag{3.30}$$

where u_x, u_y, u_z are the elements of the unit vector $\vec{u}$. Intuitively, we can obtain the angle θ

from the orientation vector $\vec{\theta}$ according to the Euclidean norm:

$$\theta = \|\vec{\theta}\| = \sqrt{P^2 + R^2 + Y^2}, \tag{3.31}$$

where P, R, Y represent the elements of $\vec{\theta}$: pitch, roll and yaw respectively. The axis of rotation's unit vector $\vec{u}$ is obtained by normalizing the orientation vector $\vec{\theta}$:

$$\vec{u} = \frac{\vec{\theta}}{\theta} = \frac{\vec{\theta}}{\|\vec{\theta}\|}. \tag{3.32}$$

Using this definition of the rotation matrix, we can move between the body UVW and global XYZ reference frames according to the equation:

$$\vec{a} = \mathbf{R}^T(\vec{\theta})\vec{a}_b = \mathbf{R}^T(\vec{u}, \theta)\vec{a}_b, \tag{3.33}$$

where the subscript 'b' indicates the body's UVW reference frame and no subscript indicates a vector in the global XYZ reference frame. While we show this equation for converting an acceleration vector $\vec{a}$, the same transformation applies to any vector, including the angular velocity vector $\vec{\omega}$.

3.3.2.2 Inertial Measurement-Based Tracking Dynamics

The dynamics of motion tracking from the IMU data can be expressed as a double integral

of the net acceleration to obtain the position, and a simple integral of the angular velocity to obtain the orientation. Specifically, the gyroscopic orientation estimate is:

$$\hat{\vec{\theta}}_G(t) = \int_0^t \hat{\vec{\omega}}(\tau)\, d\tau + \vec{\theta}_0, \tag{3.34}$$

where $\vec{\theta}_0$ is the initial orientation, most often set to $\vec{0}$ unless the initial orientation is known. Considering the digital nature of our tracking system and the reference frame of inertial measurements, this equation is expressed conveniently as a discrete-time difference equation:

$$\hat{\vec{\theta}}_G[n] = T_s \mathbf{R}^T(\vec{\theta}[n-1])\ \hat{\vec{\omega}}_b[n] + \vec{\theta}[n-1], \tag{3.35}$$

where T_s is the sampling period, which determines the time-step width of our digital integration, and $\mathbf{R}^T(\vec{\theta}[n-1])$ represents the rotation matrix according to the most recent knowledge of the orientation. This method of estimating orientation is obviously quite simple, however, it is plagued by the problem of drift in the measurement. In other words, the mean of the noise term $\vec{r}$ (introduced in equation 3.29) is non-zero which leads to unbounded estimates of pitch, roll and yaw from this estimate. Fortunately, the observations of orientation obtained from the Hall sensors will be enough to eliminate drift while incorporating the gyroscope's measurements into a statistically optimal estimate. (More on the sensor fusion approach can be found in section 3.3.3.)

The equation describing the accelerometer-based position estimate is:

$$\hat{\vec{p}}(t) = \int_0^t \int_0^\tau \hat{\vec{a}}_{net}(\tau')\, d\tau' d\tau + \vec{p}_0, \tag{3.36}$$

where $\vec{p}_0$ is the initial position, usually set to $\vec{0}$ since we can reference the position to the starting position of the device. $\hat{\vec{a}}_{net}$ is the estimated net acceleration of the device which, according to equation 3.28, is determined by removing the acceleration due to gravity from the measurement, with the Coriolis and drift acceleration if known. As such, we write a set of discrete-time difference equations to express our position estimate's evolution over time:

$$\hat{\vec{p}}[n] = T_s \hat{\vec{v}}[n] + \hat{\vec{p}}[n-1] \tag{3.37}$$

$$\hat{\vec{v}}[n] = T_s \hat{\vec{a}}_{net}[n] + \hat{\vec{v}}[n-1], \tag{3.38}$$

where the net acceleration estimate $\hat{\vec{a}}_{net}$ is moved into the scanner's frame of reference by a rotation matrix $\mathbf{R}(\vec{\theta}[n])$. Ignoring the drift and Coriolis acceleration terms:

$$\hat{\vec{a}}_{net}[n] = \mathbf{R}^T(\vec{\theta}[n])\{\hat{\vec{a}}_m[n] - \vec{g}(\vec{\theta}[n])\}. \tag{3.39}$$

Unfortunately, the position estimation is complicated by several factors:

- the net acceleration term depends heavily upon the quality of the orientation estimate meaning that error propagation is very likely,

- there is no external and objective observation of position that we can use to (in)validate the accelerometer's estimate like the Hall array can do for the gyroscope, and
- $\hat{\vec{a}}_m$ likely contains a drift acceleration $\vec{a}_{drift}$, i.e. a constant offset in the measurement, and a non-linearity A as shown in equation 3.28.

It will be possible to mitigate some of these problems in the following sections (sensor fusion, adaptive filtering, and calibration) but their consequences will persist.

3.3.3 Sensor Fusion for Optimal Orientation

The idea of sensor fusion is relatively simple: use statistics to combine two noisy estimates into an optimal estimate. For our purposes, this entails combining the orientation estimates obtained from the gyroscope and Hall arrays through a statistically-informed weighted average.

The gyroscopic orientation estimates $\hat{\vec{\theta}}_G[n]$ from equation 3.35 has a mean error and variance that can be measured by recording the gyroscope readings over a few minutes while the device is left perfectly stationary. If the gyroscope were perfect, it would record a speed of 0 rad/s at all times. However, in reality, the gyroscope has a non-stationary drift or bias term that leads to a non-zero mean value plus some noise variance that makes the readings fluctuate about that mean. As such, it will be necessary to quantify the mean and variance of the gyroscope readings to determine the optimal sensor fusion weight.

In theory, the Hall sensor array estimates $\hat{P}$ and $\hat{R}$ from equations 3.21 and 3.22 should not have any bias after they have been calibrated to their appropriate phase angles ϕ_i & ϕ_j, i.e. their errors are the result of noise variance alone. We can also include the estimate of yaw $\hat{Y}$ that is obtained by cross-referencing the Hall sensors' data with the accelerometer's data. However, $\hat{Y}$ will have a different noise variance from $\hat{P}$ and $\hat{R}$ because of the different methods of obtaining them. All these can be estimated by recording the Hall sensors' outputs while the device is stationary, then calculating the noise variance. The variance of the Hall voltages will translate near-linearly to the variance of the estimates since we are using the Hall sensors with the smallest absolute value of voltage deflection, according to equations 3.19 and 3.20. The small angle approximation of sine, which has a less than 1% error for angles of magnitudes less than 0.244 radians or 14.0 degrees, is $sin(\theta) = \theta$, so we can understand:

$$sin(\theta + \phi_i) = \frac{\Delta V_{H_i}}{k_s B_0} \approx \theta + \phi_i, \tag{3.40}$$

and therefore the variance of the Hall sensors determines the estimates' variance:

$$Var(\theta + \phi_i) \approx Var\left(\frac{\Delta V_{H_i}}{k_s B_0}\right). \tag{3.41}$$

Therefore, the variance of the Hall sensors' estimates for pitch and roll (in radians) is approximately equal (but upper-bounded) by the variance of V_H in volts divided by $k_s B_0 = (2.6V/T)(3T) = 7.8V$. The variance of the yaw estimate is complicated by the cascade of

operations leading to the estimate, however, we will assume that it has the same variance as the pitch and roll for simplicity.

3.3.3.1 Minimum Mean Squared Error - *A Priori* Orientation Fusion Pipeline

The optimal weight γ for sensor fusion of the form $\hat{x} = \gamma\hat{x}_1 + (1-\gamma)\hat{x}_2$ is the one that minimizes the error $e = x - \hat{x}$ for all observations. Assuming that the optimal filter is the one that Minimizes the Mean-Squared Error (MMSE), we find that the goal is to minimize the expected value of the squared error, i.e.:

$$min\ _{arg\ \gamma} \left(E\left[(x - \gamma\hat{x}_1 - (1-\gamma)\hat{x}_2)^2 \right] \right). \tag{3.42}$$

Expanding the above equation and assuming that the noise in the estimates $\hat{x}_1$ and $\hat{x}_2$ are independent of each other and independent of the true state x we get:

$$\begin{aligned} E[e^2] = \mu_x^2 + \sigma_x^2 + \gamma^2(\mu_{\hat{x}_1}^2 + \sigma_{\hat{x}_1}^2) + (1-\gamma)^2(\mu_{\hat{x}_2}^2 + \sigma_{\hat{x}_2}^2) \\ - 2\gamma\mu_x\mu_{\hat{x}_1} - 2(1-\gamma)\mu_x\mu_{\hat{x}_2} + 2\gamma(1-\gamma)\mu_{\hat{x}_1}^2\mu_{\hat{x}_2}^2, \end{aligned} \tag{3.43}$$

where μ and σ represent the mean and standard deviation of the signal in their subscript. First, we must recognize that the mean values of the state x (representing the orientation $\vec{\theta}$) and the Hall sensor's estimate $\hat{x}_1$ are both equal to zero (or specifically $\vec{0}$). However, the mean value of the gyroscopic estimate $\hat{x}_2$ is non-zero because of the sensor's bias. Then, we

can take the derivative with respect to γ to perform the optimization.

$$\frac{d}{d\gamma}E[e^2] = 2\gamma\sigma_{\hat{x}_1}^2 - 2(1-\gamma)(\mu_{\hat{x}_2}^2 + \sigma_{\hat{x}_2}^2) = 2\gamma(\sigma_{\hat{x}_1}^2 + \mu_{\hat{x}_2}^2 + \sigma_{\hat{x}_2}^2) - 2(\mu_{\hat{x}_2}^2 + \sigma_{\hat{x}_2}^2) \tag{3.44}$$

Setting the above equation equal to zero we will find the MMSE optimal weight γ is:

$$\tilde{\gamma} = \frac{\mu_{\hat{x}_2}^2 + \sigma_{\hat{x}_2}^2}{\sigma_{\hat{x}_1}^2 + \mu_{\hat{x}_2}^2 + \sigma_{\hat{x}_2}^2}. \tag{3.45}$$

We note that this is indeed a minimum since the derivative of equation 3.44 with respect to γ will be positive, meaning that the expected error is concave-up and thus the extrema is a minimum. We also note the intuitively satisfying form of this result: the weight of estimate $\hat{x}_1$ is proportional to the MSE (error power) expected from estimate $\hat{x}_2$.

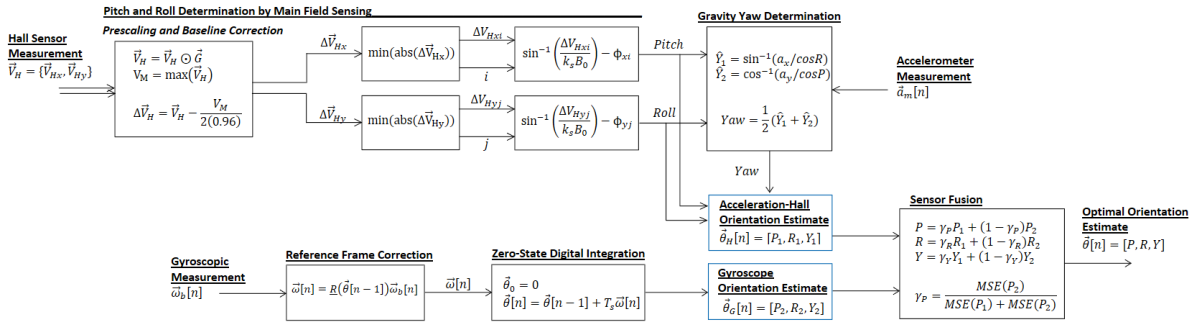

Figure 3.10: Signal pipeline for obtaining optimal orientation estimates. Summary of the Hall array algorithm, the gyroscopic algorithm, and their fusion. The optimal Mean-Squared Error (MSE) weight γ of each estimate (P, R, Y) is defined in the present section.

3.3.3.2 Kalman Filter Formulation of Orientation Fusion

An alternative to the MMSE sensor fusion discussed above is the Kalman filter. Kalman filtering is one of the most used solutions in the literature on IMU-based motion tracking and GPS [33, 34]. The Kalman filter is a state-space formulation for optimal statistical filtering; it has the benefit of remembering all the past values of inputs while using a finite memory [34]. We will briefly explain how Kalman Filtering can be applied to our situation of motion tracking.

The Kalman filter begins with a state-space model of the system whose states are being estimated. In our case, the orientation (pitch, roll, and yaw) of the system are the states being estimated, these will be encapsulated into the vector estimate $\hat{\vec{\theta}}$. Next, we define the interaction between the present states and the future states; this will be encapsulated into the state transition matrix $\mathbf{F}$ according to:

$$\hat{\vec{\theta}}[n|n-1] = \mathbf{F}\hat{\vec{\theta}}[n-1|n-1] + \mathbf{B}\vec{u}[n], \tag{3.46}$$

where the indexing $[n|k]$ refers to conditional probability, i.e. the state estimate at time n given the observations up to and including time k. Finally, we must identify the input to the system $\vec{u}[n]$ and how it will interact with the states; this is encapsulated in the input

matrix $\mathbf{B}$, which will be a function of time in our case. The resulting state space system is:

$$\hat{\vec{\theta}}[n|n-1] = \mathbb{I}_3\hat{\vec{\theta}}[n-1|n-1] + T_s\mathbf{R}^T(\vec{\theta}[n-1])\ \hat{\vec{\omega}}_b[n] + \vec{\nu}[n], \tag{3.47}$$

where $\mathbb{I}_3$ is the 3x3 identity matrix, $\hat{\vec{\omega}}_b[n]$ is the gyroscope's measurement/estimate of the angular velocity in the IMU/body reference frame, $\vec{\nu}$ is the stochastic disturbance to the system which is not explicitly used in the prediction but is used in the statistical model, and $\mathbf{R}^T(\vec{\theta}[n-1])$ is the rotation that converts vectors from the body reference frame to the global one. Note the interchangeability between the above equation and equation 3.35. The above equation is referred to as the prediction phase of the Kalman filtering process: estimating the next state from the previous one given the measurable input signal $\vec{u}[n] = \hat{\vec{\omega}}_b[n]$.

The next step in the Kalman filter is the observation of the states. The observation process is defined by an output matrix $\mathbf{H}$ according to $\hat{\vec{z}}[n] = \mathbf{H}\hat{\vec{\theta}}[n|n-1]$. In our case, the actual observation is performed by the Hall sensor array, returning an estimate of pitch, roll, and yaw $\hat{\vec{\theta}}_H[n]$. Therefore, the observation matrix $\mathbf{H}$ is also an identity matrix:

$$\hat{\vec{z}}[n] = \hat{\vec{\theta}}_G[n] = \mathbb{I}_3\hat{\vec{\theta}}[n|n-1] + \vec{v}[n], \tag{3.48}$$

where $\hat{\vec{\theta}}_G[n]$ is the predicted observation according to the information obtained by the gyroscope and state-space system, and $\vec{v}$ is the measurement noise that we explore in the statistical update sequence. This predicted observation can then be compared to the actual

observation $\vec{z} = \hat{\vec{\theta}}_H[n]$ of the orientation coming from the Hall array and accelerometer. Note that the observation $\vec{z}$ has no hat because it is the true ***observed*** state but not necessarily the true state since it is contaminated with noise. The error between the predicted observation and the true state observation is therefore stated as follows:

$$\vec{e}[n] = \vec{z}[n] - \hat{\vec{z}}[n] = \hat{\vec{\theta}}_H[n] - \hat{\vec{\theta}}_G[n]. \tag{3.49}$$

This step is known as the update phase of the Kalman filter. We will define statistical matrices to update our state estimate according to the above error.

The initial covariance of the state estimates $\mathbf{P}_{0|0} = cov(\hat{\vec{\theta}}[0])$ can be set to a matrix of zeros if the initial state is known [34]. In our application, we will initialize our device in the $\vec{\theta}[0] = \vec{0}$ position with near absolute certainty so that we can make $\mathbf{P}_{0|0} = \mathbb{O}_3$, a 3x3 matrix of zeros. The second necessary initialization is the covariance of the system disturbances $\mathbf{Q} = cov(\vec{\nu})$, this is effectively the same as the covariance of the accelerometer noise $\vec{q}$ in equation 3.28, thus we can estimate it from a stationary accelerometer recording. The final initialization is the covariance of the measurement noise $\mathbf{R} = cov(\vec{v})$ which can be estimated using the samples of the Hall voltages during a stationary period of time. The same logic as that used in equation 3.41 applies here. We expect that $\mathbf{R}$ will be a diagonal matrix, i.e. the noise in the pitch, roll and yaw estimates are independent of each other. For the best results, both the $\mathbf{Q}$ and $\mathbf{R}$ matrices should be determined empirically through experiments,

using the covariance of $\hat{\vec{\theta}}_G[n]$ and $\hat{\vec{\theta}}_H[n]$ respectively during a stationary period of time.

After initialization, we can perform the statistical analysis as follows:

$$\mathbf{P}_{n|n-1} = \mathbf{F}\mathbf{P}_{n-1|n-1}\mathbf{F}^T + \mathbf{Q} \tag{3.50}$$

$$\mathbf{S}_n = \mathbf{H}\mathbf{P}_{n|n-1}\mathbf{H}^T + \mathbf{R} \tag{3.51}$$

$$\mathbf{K}_n = \mathbf{P}_{n|n-1}\mathbf{H}^T\mathbf{S}_n^{-1} \tag{3.52}$$

$$\mathbf{P}_{n|n} = (\mathbb{I} - \mathbf{K}_n\mathbf{H})\mathbf{P}_{n|n-1}, \tag{3.53}$$

where the subscripts in $\mathbf{P}_{n|n-1}$ indicate, for example, the covariance of the state estimate at time n given the observations up to time $n-1$. The new variables $\mathbf{S}_n$ and $\mathbf{K}_n$ are the covariance of the measurements and the Kalman gain respectively, at time n. Using the statistically optimal Kalman gain we can update our state estimate according to [34]:

$$\hat{\vec{\theta}}[n|n] = \hat{\vec{\theta}}[n|n-1] + \mathbf{K}_n\vec{e}[n]. \tag{3.54}$$

In this way, the Kalman filter uses the statistical properties of the state space and its measurements to construct an optimal estimate of the state encompassing all of the sensor readings up to and including the latest measurements. We note that the behaviour of the filter will depend heavily upon the accuracy of the noise covariance matrices $\mathbf{Q}$ and $\mathbf{R}$.

Therefore, special attention should be paid to determining the best possible estimates of these matrices, and caution should be exercised.

3.3.4 Adaptive Filtering for Optimal Position Prediction

As alluded to in equation 3.26, the acceleration that we expect to see in a pseudo-stationary application such as ours (one where the motion is very small and heavily constrained) is almost exclusively the acceleration due to gravity. Because of this feature, we can implement an adaptive filter that will shape the input accelerometer signals to make them fit the expected gravity given the orientation estimate. First, we will demonstrate a Least Mean Squared (LMS) filter formulation. Then, we will demonstrate how a Kalman filter could be extended to this purpose.

3.3.4.1 Gravity-Residual Least Mean Squares Adaptive Positioning Filter

The general formulation of a Least Mean Squared (LMS) Filter is:

$$min_{\ arg\ \tilde{h}[n]}(E[e^2]). \tag{3.55}$$

In other words, minimize the expected value of the squared error by finding the optimal filter impulse response $\tilde{h}[n]$. The error e is defined as the difference between the desired output signal and the estimated signal. In our case, the desired output signal is the gravity vector $\vec{g}(\vec{\theta}[n])$ which is assumed to be accurate despite the fact that it depends upon the

orientation $\vec{\theta}[n]$ which is itself an estimate, as discussed in the previous section. The estimated signal is the filtered accelerometer measurement. Because the LMS filter is intended for uni-dimensional signals, i.e. a set of values over time, we must split our LMS problem into the UVW spatial dimensions, in the IMU's reference frame. Hence, we will really be implementing 3 separate LMS algorithms in parallel, one for each dimension. Therefore, the error of the u-axis is expressed as follows:

$$e_u[n] = g_u(\vec{\theta}[n]) - \sum_{m=0}^{M-1} \tilde{h}_u[m] a_u[n-m], \tag{3.56}$$

where M is the order of the adaptive LMS filter, and the subscript u represents the first dimension of the IMU accelerometer, effectively the IMU's local x-axis. We note that the same equation applies to the v-axis and w-axis of the IMU. For simplicity, we may want to express the above equation's convolution in vector form, such that the impulse response and accelerometer readings are vectors. The resulting equation is:

$$e_u[n] = g_u(\vec{\theta}[n]) \; - \; \tilde{\vec{h}}_u^T \; \vec{a}_u[n], \tag{3.57}$$

where the impulse response vector of the u-axis filter is

$$\tilde{\vec{h}}_u = \left[\tilde{h}_u[0] \; ... \; \tilde{h}_u[M-1]\right]^T, \tag{3.58}$$

and the u-axis acceleration measurement vector is

$$\vec{a}_u[n] = [a_u[n] \ ... \ a_u[n-M-1]]^T \ . \tag{3.59}$$

The LMS algorithm uses the error e and learning rate α to descend the gradient toward the optimal filter weights according to the following equation [34]:

$$\tilde{\vec{h}}'_u = \tilde{\vec{h}}_u + \alpha \ e_u[n] \ \vec{a}_u[n]. \tag{3.60}$$

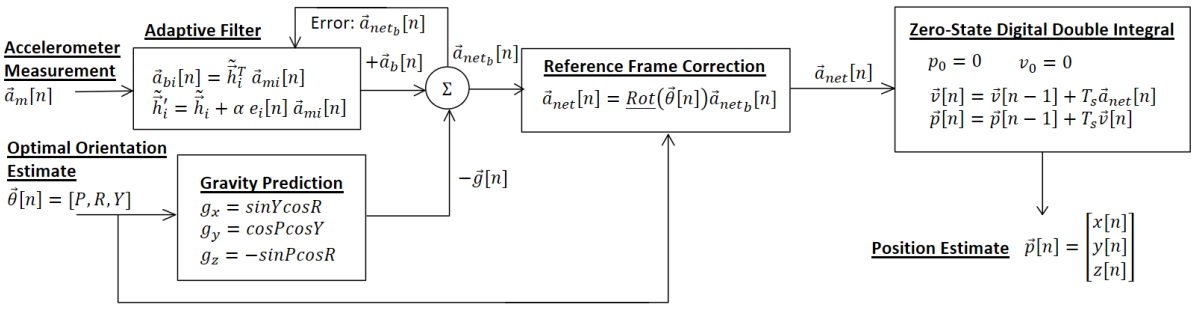

Figure 3.11: Signal processing pipeline summarizing the adaptive position estimation. The adaptive filters (one for each axis, identified by an arbitrary index 'i') applied to the accelerometer measurements use the net body-frame acceleration as the error signal since the expected value of the net acceleration is zero.

The learning rate α is a hyper-parameter to this adaptive filter. Making α larger will make the filter more responsive to changes in the underlying system dynamics, but making it too large will lead to instability. If α is too small, the LMS filter may not converge to the

optimal weights within the necessary time horizon.

The central benefit of filtering the acceleration input signal with an adaptive LMS filter is the empirical grounding of the algorithm. Filtering in this fashion will shape the frequency content of the acceleration measurements according to an empirical set of observations, instead of using a heuristic (for example low-pass of an arbitrary cutoff frequency) as is often the case.

Since the expected value of the acceleration is equal to the gravity vector, we expect this LMS filter to deliver a reasonable estimate of the true acceleration. In theory, the LMS filter should eliminate drift because its objective is to drive the net acceleration toward zero. Furthermore, the LMS filter could eliminate a scaling error by adjusting the sum of the filter impulse response terms to something approaching A^{-1} from equation 3.28. That is, we expect:

$$\sum_{m=0}^{M-1} \tilde{h}[m] = A^{-1}. \tag{3.61}$$

3.3.4.2 Kalman Filter for Position Estimation

Analogous to how we could perform orientation sensor fusion using an *a priori* minimum MSE algorithm or a Kalman filter in section 3.3.3, we can use a Kalman filter to estimate position instead of the gravity-residual adaptive filter method. Here, we will describe the state-space model necessary to construct a position-estimating Kalman filter in our pseudo-stationary application. Similar to equations 3.37 to 3.39, position tracking dynamics are

described by the following state-space system, like the form in equation 3.46:

$$\begin{bmatrix} \vec{p}[n] \\ \vec{v}[n] \end{bmatrix} = \begin{bmatrix} \mathbb{I}_3 & T_s\mathbb{I}_3 \\ \mathbb{O}_3 & \mathbb{I}_3 \end{bmatrix} \begin{bmatrix} \vec{p}[n-1] \\ \vec{v}[n-1] \end{bmatrix} + \begin{bmatrix} \frac{T_s^2}{2}\mathbf{R}^T(\vec{\theta}[n]) \\ T_s\mathbf{R}^T(\vec{\theta}[n]) \end{bmatrix} \vec{a}_{net_b}, \tag{3.62}$$

where $\vec{a}_{net_b} = \hat{\vec{a}}_m[n] - \vec{g}(\vec{\theta}[n])$ is the net acceleration in the IMU's frame of reference, or the accelerometer reading minus the expected gravity. $\vec{p} = [x, y, z]^T$ represents the position of the device, while $\vec{v}$ represents its velocity. The above equation could be expressed as the prediction phase of the Kalman filter by replacing indices: n with the conditional index $n|n-1$ and $n-1$ with the conditional index $n-1|n-1$. We could also add the disturbance/noise term like $\vec{\nu}$ in equation 3.47, the covariance of which is needed for the Kalman algorithm.

In truth, we don't have any means to measure/observe this system because the only sensor that contains information pertaining to translational motion is the accelerometer, which is already the input. Since we have no way of observing either the velocity or position, we can only provide the Kalman filter with an approximation of the state. Considering what we know about our application, we posit that we could provide the Kalman filter with an "observation" that indicates no motion, i.e. the velocity and position are zero in all dimensions. This may seem like it would only lead to the Kalman filter ignoring the acceleration. However, much like the gravity-residual adaptive filter method, this technique exploits our statistical knowledge of the underlying process. We know that the device is unlikely to move significantly from its starting position, therefore, the mean value of position

and velocity is a zero vector $\vec{0}$. We suspect that this approach will mitigate drift by steering the Kalman filter toward the no-movement state. The predicted observation of the system $\hat{\vec{z}}[n]$ is therefore:

$$\hat{\vec{z}}[n] = \begin{bmatrix} \hat{\vec{p}}[n|n-1] \\ \hat{\vec{v}}[n|n-1] \end{bmatrix} = \mathbf{H} \begin{bmatrix} \vec{p}[n|n-1] \\ \vec{v}[n|n-1] \end{bmatrix}; \quad \mathbf{H} = \mathbb{I}_6, \tag{3.63}$$

where $\mathbf{H}$ is the observation matrix. We note that there is an observation noise associated with the above equation, similar to $\vec{v}$ in equation 3.48, that makes our observation inaccurate. In our case, we know that our pseudo-observation of the state is inaccurate. However, we know that it is close to the mean value of the state because the patient would be staying relatively still in the MRI. This *statistical observation* (or *pseudo observation*) approach forces us to consider the "observation noise" as the variance from the zero states.

Other than the changes to the dynamic system and observation system matrices $\mathbf{F}, \mathbf{B}, \mathbf{H}$, the Kalman filter algorithm runs exactly as described in section 3.3.3.

3.3.5 Calibration and Sensor Characterization

The device calibration is relatively simple: place the tracker in a known orientation in the scanner bore and record measurements from all sensors (accelerometers, gyroscopes, and magnetometers) while the device is stationary, repeat for at least 3 orientations. This procedure will primarily serve to find the precise angle of the Hall sensors relative to the PCB surface, which are shown as 30° intervals in Fig. 3.6. Another purpose of this

procedure is to determine the statistics of our measurements (e.g. mean, variance, covariance, error...) which are used in the sensor fusion algorithms discussed in section 3.3.3. This sensor characterization will be done for the accelerometer, gyroscope, and Hall sensors during a stationary time interval (effectively obtaining the sensors' zero-input responses). Finally, calibration could provide a reasonable estimate of the optimal adaptive filter impulse response $\tilde{h}[m]$, however, the non-stationary nature of the accelerometer noise prohibits this approach.

3.4 Testing Methods

Testing of our device centred around a 3D-printed mould with known position and orientation waypoints as shown in Fig. 3.12. Having this kind of test mould removed the significant experimental error that would have occurred if we used a ruler, protractor, and grid paper to record positions and orientations. In this way, the 3D CAD design guaranteed a certain precision in the *ground truth* to which we will compare our predictions.

Since we could not know the tracker's exact pose at all times, we had known positions and orientations for a set of waypoints that lasted a predetermined amount of time. We synchronized the periods of motion (when the pose is unknown) and the periods of rest (when the pose is known) using a built-in LED indicator on our tracker. A red LED indicated that the device should be stationary at the waypoint, while a blue LED indicates that the device can move from one waypoint to the next. In this fashion, we tested the tracker's prediction

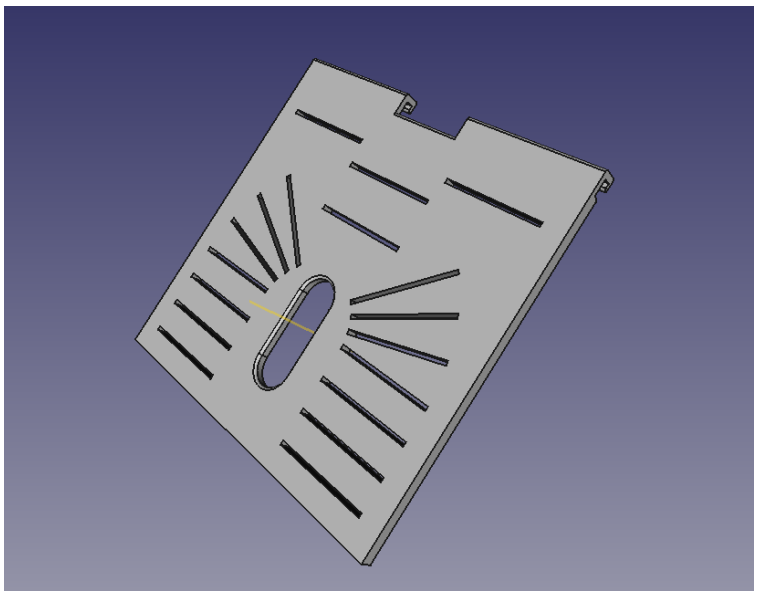

Figure 3.12: Rendering of the 3D-printed testing mould made in FreeCAD. The slots are sized to fit the tracker PCB snugly and spaced by known distances & known angles to evaluate performance accurately. The hooks at the top were sized to fit into an existing Siemens head coil holder such that it is fixed to the scanner through the head coil.

accuracy by comparing the known waypoint pose to the predicted pose during the periods of rest. The testing procedure is summarized as follows (see the accompanying Fig. 3.13):

1. place the tracker at a predetermined waypoint (the $t_0 - t_1$ pose)

2. trigger the sensor data collection, the red LED should turn on

3. leave the tracker at the waypoint until the LED turns from red to blue

4. move the tracker to the next predetermined waypoint ($t_2 - t_3$) pose, the LED should be blue during the ($t_1 - t_2$) transition period

5. leave the tracker at the waypoint until the LED turns from red to blue

6. repeat steps 4 & 5 as desired to test the tracker's performance as time goes on

Once the data was collected according to the process defined above, we computed the pose estimates over time according to the algorithms described in the previous sections (specifically figures 3.10 and 3.11). Comparing the pose estimates during the stationary periods to the known waypoint poses, we determined the tracker's mean-squared error.

Finally, we tested the severity of magnetic susceptibility artifacts created by our device through the imaging of a spherical phantom. Ultimately, a motion tracker used for artifact correction is only worthwhile if it does not produce severe artifacts of its own.

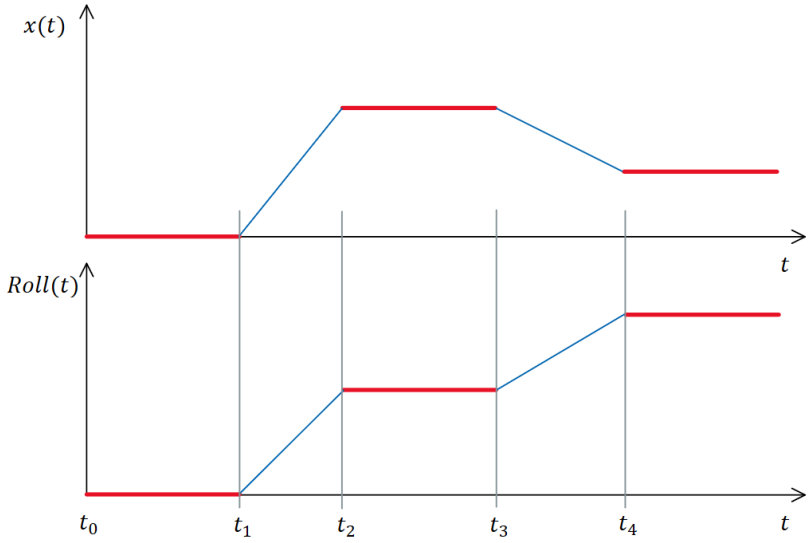

Figure 3.13: Plot illustrating our ground truths for an arbitrary motion pattern across 3 waypoints. While we only show x and *Roll* for simplicity, these plots of the known pose are extended to all 6 degrees of freedom. The red sections in the plot are known positions and orientations, while the red LED is on, to which we will compare our predicted pose. The blue sections are transition regions where the pose is not known, while the blue LED is on, therefore no comparison is made with the estimates.

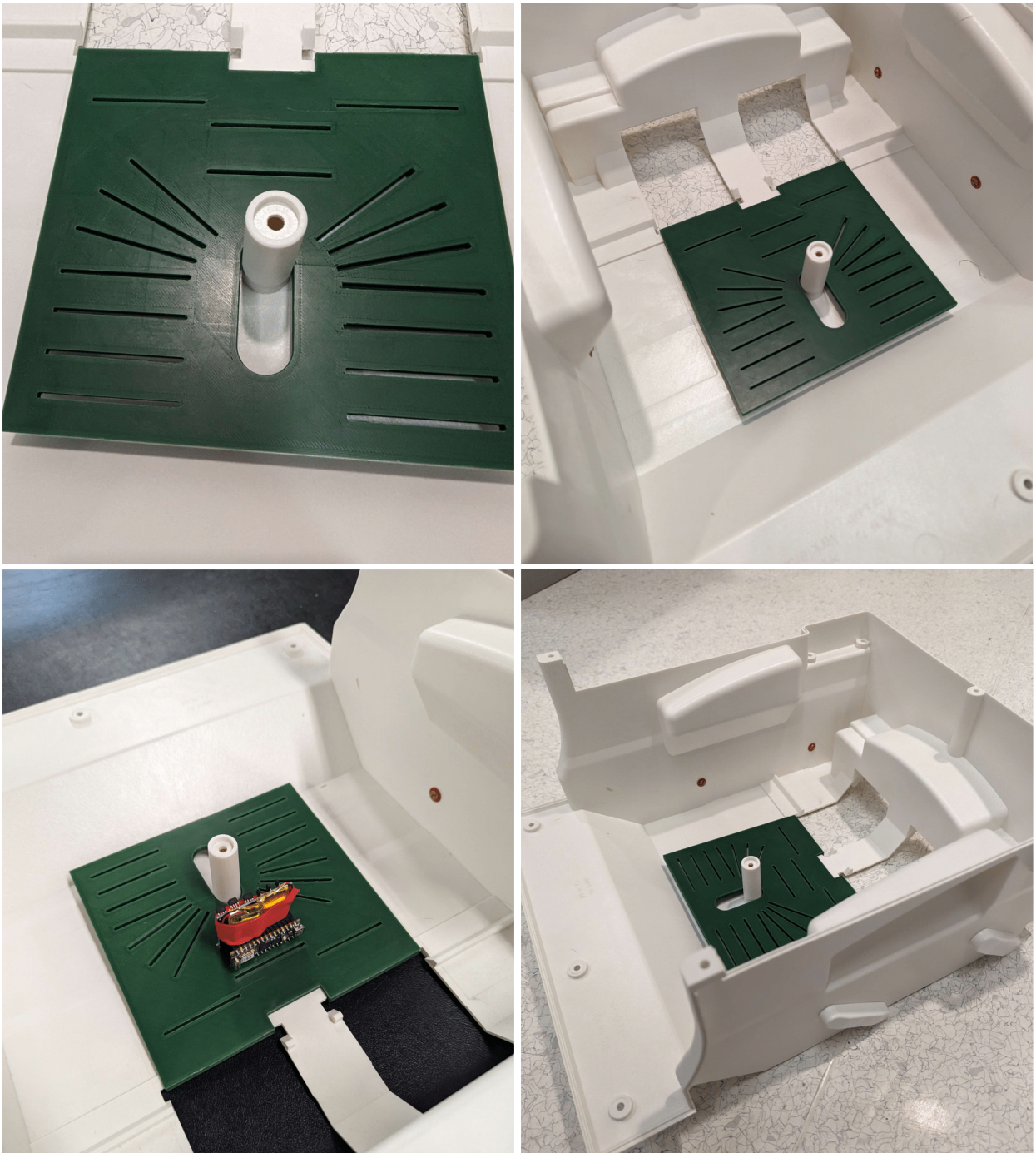

Figure 3.14: Pictures of the actual 3D-printed testing kit when attached to the Siemens head coil housing (with the actual coil removed). The bottom left shows the tracker inserted in the back-center slot, which we used as the starting point of our motion experiments.

Chapter 4

Results and Discussion

In this chapter, we will discuss the results of our previously explained design. We will start by showing the results of the Hall sensors, discussing how their readings were influenced by other systems on the PCB, and proposing hypotheses for the observed data. Second, we will examine the gyroscope data and determine the statistical parameters of its measurements. Third, we will examine the accelerometer data and determine its statistical parameters. Fourth, we will show and discuss the results of our signal processing pipelines, evaluating their performance. We will show how the Kalman filter can perform without any actual state observations, what we call *statistical* or *pseudo observations*, though the performance will be hindered. Finally, we will explain why our design creates magnetic susceptibility artifacts in the scanner and how a future iteration may avoid these.

4.1 Hall Sensor Array Data

The measurements obtained by our MCU's ADCs show a Hall voltage reading that decays over time as the batteries discharge. We had planned to account for this by scaling the measurements with the value of a saturated sensor, as described in equation 3.18 but, as shown in Fig. 4.1, the Hall array outputs were unexpectedly flat. When inside the scanner and moved to different angles, we expected the Hall sensors to output a wide range of voltages, 0-5 V.

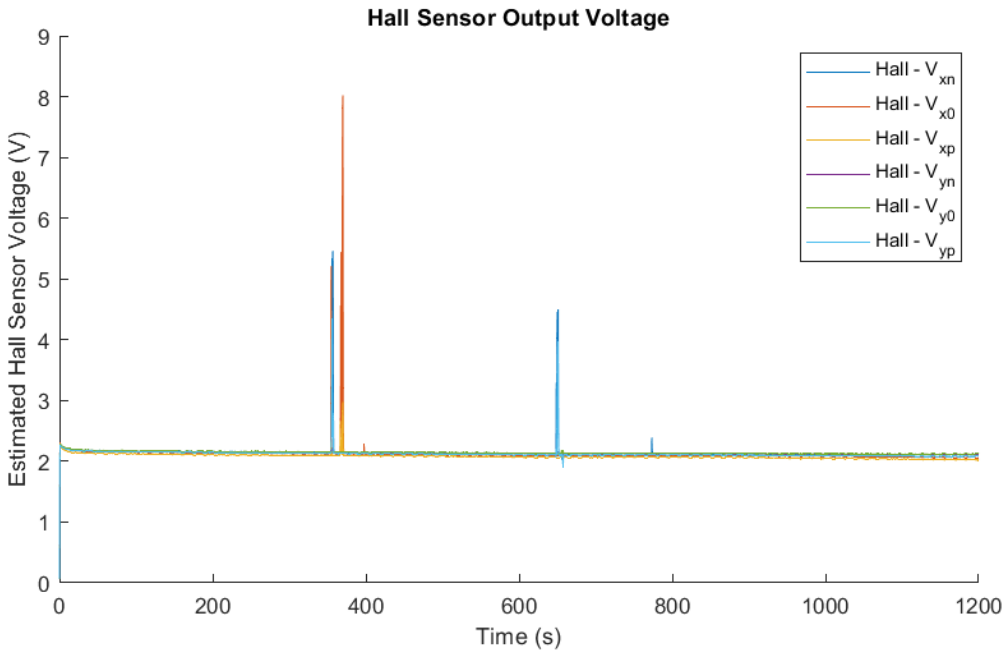

Figure 4.1: Raw Hall voltage signals, estimated according to equation 3.17. Note the appearance of spikes in the sensed voltages, these will be discussed in the present section.

Unfortunately, the sensor outputs stayed near their quiescent point for almost the entire

duration of every experiment. As shown clearly in Fig. 4.2, there were instances where a sensor would output a large voltage fluctuation from the quiescent point but these events were sporadic and did not correlate significantly with the tracker orientation, nor could these events be reliably reproduced. We suspect that the spikes are caused by a combination of the sensors momentarily responding to the field and an induced voltage caused by movement in the magnetic field. Otherwise, it is impossible to obtain such readings since our battery pack only has a nominal open circuit voltage of 7.4 V.

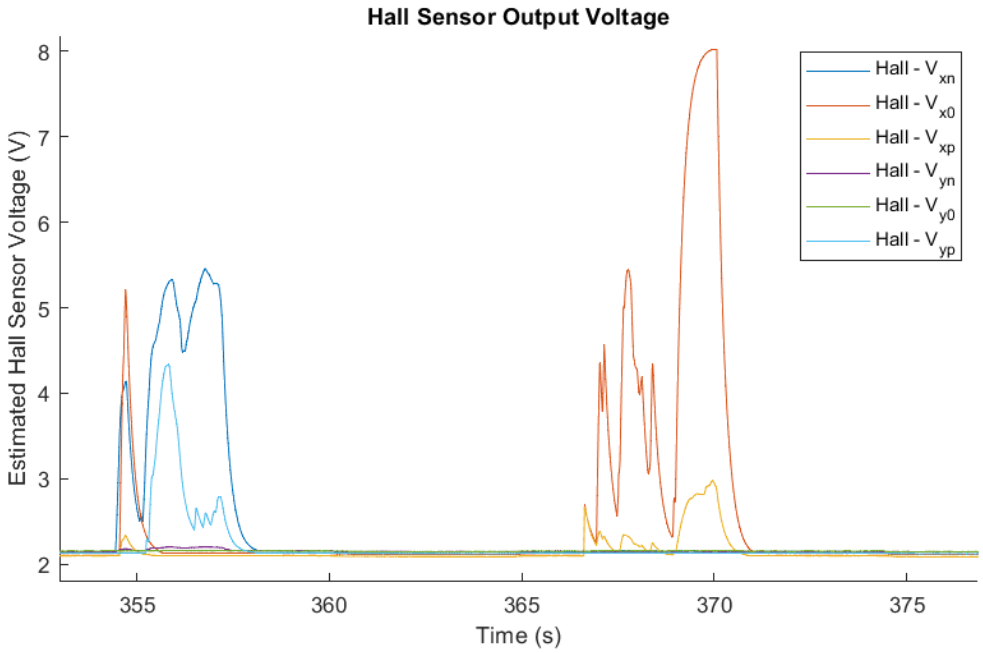

Figure 4.2: Zoomed-in figure of the Hall voltages seen in Fig. 4.1. Note the unexpected magnitude of the red waveform, as the Hall sensors can only output a voltage up to their power supply rail, 7.4 V absolute maximum in our case. The voltage overshoot is anticipated to have resulted, in part, from induced voltages from motion in the magnetic field.

Without suitable data from the Hall array, we are unable to perform the math outlined

in section 3.2.2 to resolve the pitch and roll of the device. This makes it nearly impossible to test and evaluate the subsequent stages of our signal-processing pipeline.

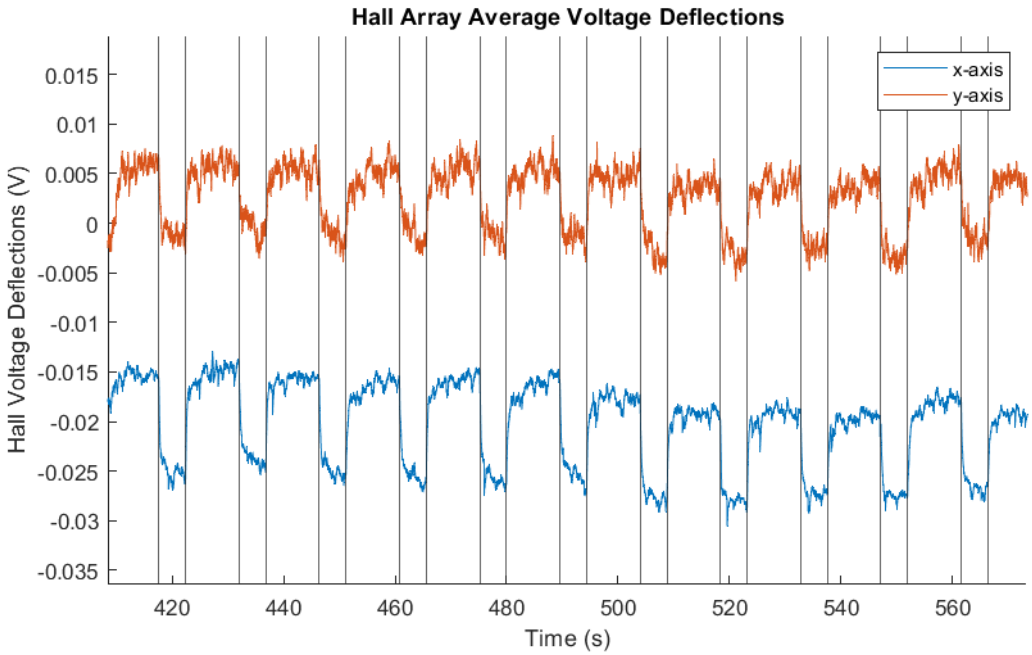

Figure 4.3: Estimated Hall voltage deflections with exponential smoothing filter during moving and resting intervals. The short intervals are resting periods of 4.8 s (500 samples at 9.6 ms) and the long intervals are moving periods of 9.6 s (1000 samples at 9.6 ms).

When we noticed how insignificant the Hall Voltage fluctuations were, we aligned all the x-axis and y-axis sensors to perform array averaging, with the aim to uncover some pattern in the Hall voltage signals. Then, we applied a smoothing filter to the x-axis and y-axis Hall voltages for denoising before removing the signals' trendline using linear regression with first- and zero-order terms. The resulting waveforms overlaid with vertical lines indicating the moving and resting periods are in shown in Fig. 3.18. We note that there were fluctuations

that correlated strongly with the onset of the moving and resting periods, the waveform tended high during the moving periods and low during the resting periods.

Seeing the uncanny correlation of the Hall voltages with the onset of the red LED and blue LED (resting and moving) periods, we realized that this correlation with the moving and resting periods was simply an artifact of our unregulated power supply fluctuating as a result of changing current demand. Upon examining the Arduino RP2040 datasheet we found that the red LED and blue LED have the same series resistance of 330 Ω connecting them to the Arduino's regulated 3.3 V power supply. However, red LEDs and blue LEDs require different voltage levels (forward voltages) to emit their respective frequencies of light. Since blue light is more energetic, the blue LED will have a larger forward voltage drop than the red LED. We confirmed this by looking at the RGB diode's datasheet (the SMLP34RGB) which states that the red LED's forward voltage is 2.1 V while the blue LED's is 3.0 V. This means that the red LED will draw more current than the blue, thus lowering the power supply's terminal voltage when the red LED is on, leading to the low signal amplitude at the output of the Hall sensors. We confirmed this hypothesis by replacing the red LED period with the green LED whose forward voltage is 3.1 V (much closer to the blue LED's). That experiment showed no voltage fluctuations correlating with the moving and resting periods. Therefore, we speculate that all the other observed variance in the Hall voltages are a product of noise and bear no relation to the orientation of the device as we had hoped.

Our main explanation for the unexpected Hall voltage waveforms is that the unregulated

power supply's voltage dropped below the Hall sensors' cut-off voltage of 4.5 V (according to the MLX90251 datasheet). Our experimental analysis of the current demand shown in Fig. 3.5 was based on observations of the tracker running without the ADCs sampling the voltages coming from the Hall sensors. As such, the current demand of the MCU was greater than the 27 mA we anticipated. Admittedly, this was a critical oversight. Since the current demand of the MCU was greater, the voltage supplied to the Hall sensors (the power supply's terminal voltage) fell from 5 V to around 4 V. We confirmed this hypothesis with a multimeter, finding that the unregulated power supply was responsible for disabling the Hall sensors by bringing their supply below their under-voltage threshold of 4.5 V. In future iterations, it will be paramount to regulate the Hall sensor supply voltage. We have not used a voltage regulator (in this work, a boost converter) to avoid adding inductors to our device since they would create magnetic fields and likely contain iron or nickel.

Despite our lack of concrete data for the Hall sensor arrays, we will report on the statistics of the Hall voltage signals to inform the techniques such as sensor fusion in section 3.3.3 which require second-order statistics. Fig. 4.4 shows the estimated PDF. The best fit is a normal distribution, however, the Hall voltages seem to move steadily toward lower values as the batteries discharge, shown in Fig. 4.5, skewing the distribution. This also demonstrates the importance of powering the Hall sensors with a regulated power supply. We expect many sources of noise contributing to the variance of these signals, hence we are considering the central limit theorem in our assertion that the noise of the Hall voltages will approximate a

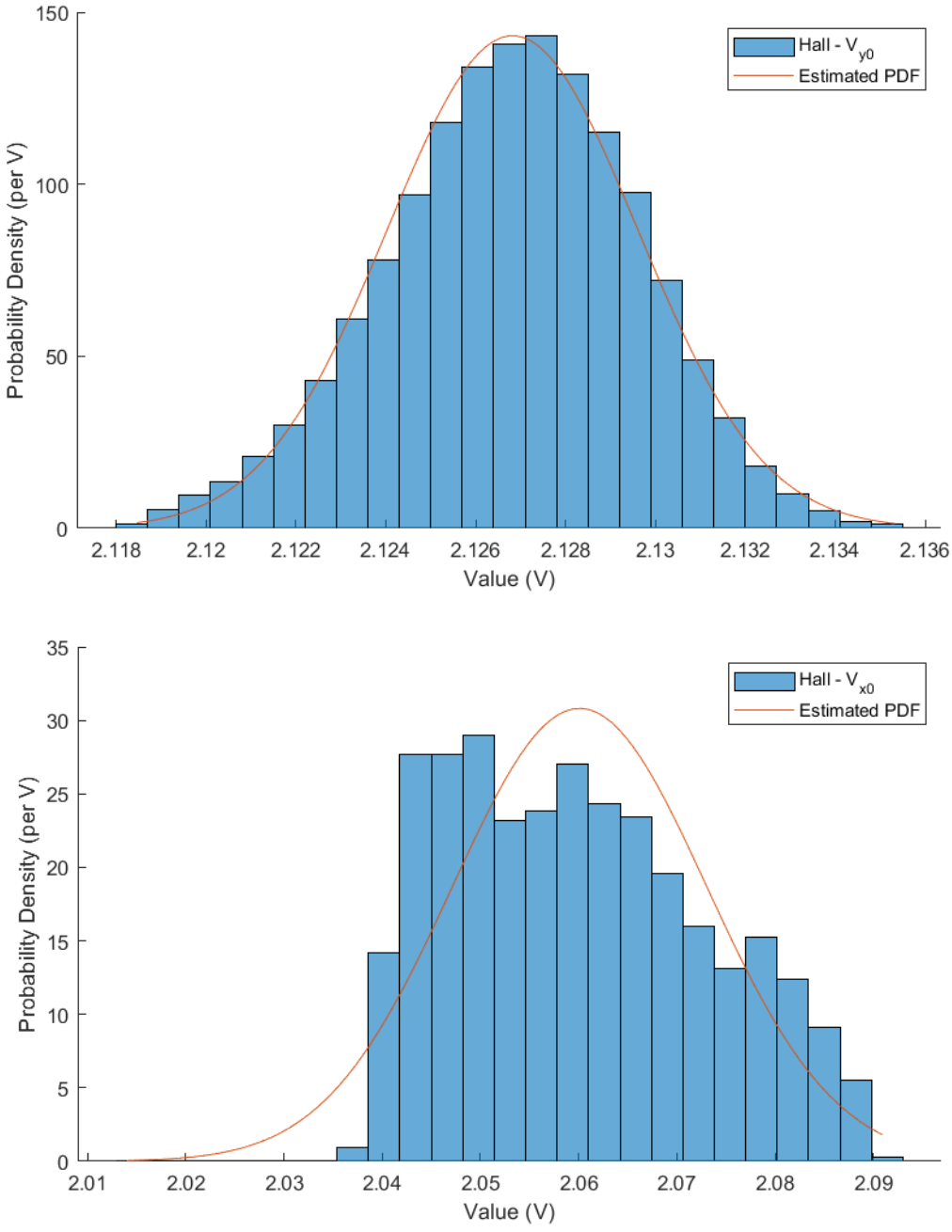

Figure 4.4: Estimated Probability Density Functions (PDF) of the filtered Hall sensor voltages, described approximately by a normal distribution. We note that this estimated PDF fits the x-axis data (bottom graph) relatively poorly while fitting the y-axis data (top graph) nicely.

normal distribution. The standard deviation of the Hall voltages is necessary for estimating the MSE of Hall-voltage-inferred orientation estimates. Since the data is relatively noisy, we applied an exponential smoothing filter with a time-constant of 192 ms, the standard deviation came to under 10 mV for the x-array and 1 mV for the y-array.

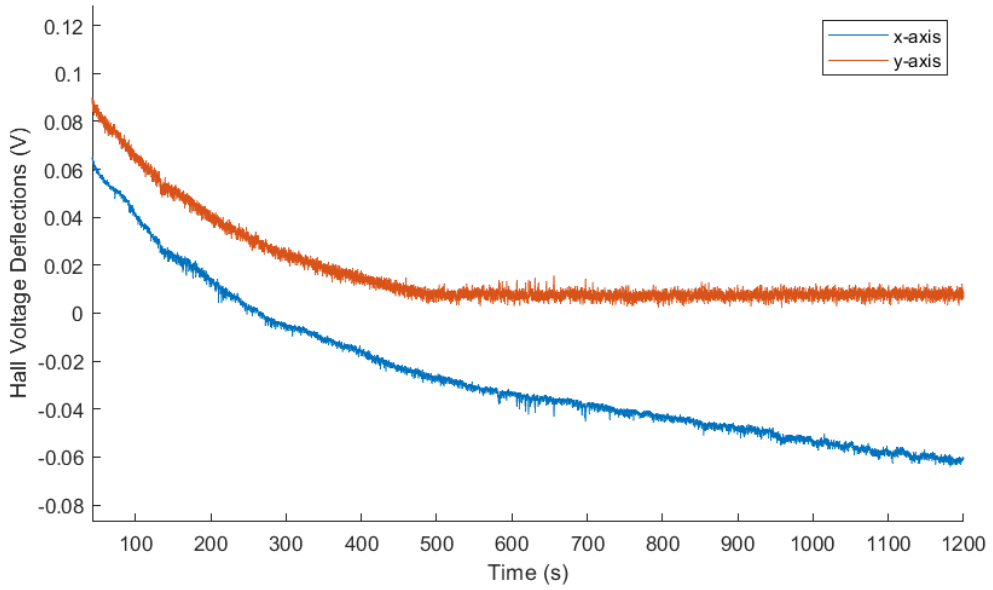

Figure 4.5: Plot of the Hall sensor voltage deflections as the batteries discharge. We note the decrease in the signal as time goes on, this would inevitably lead to a distorted PDF. We note that the decline in voltage is more pronounced in the x-axis array.

The reason for the different behaviour of the Hall voltages across the x-axis and y-axis arrays seems to stem from a relatively poor power supply distribution on our PCB. In short, the Hall sensors' power supply was closer to the y-axis sensors, making their voltages overall larger and less susceptible to voltage regulation issues. We show the trend in Fig. 4.5 to demonstrate why the x-axis PDF in Fig. 4.4 departs significantly from a normal distribution.

We opted to only include data after the 400-second mark in order to avoid skewing the PDF. We note that, because of the voltage regulation issue, the more accurate estimate of noise in the Hall-effect measurements comes from the y-axis PDF, i.e. a 1 mV standard deviation.

4.2 Gyroscope Data

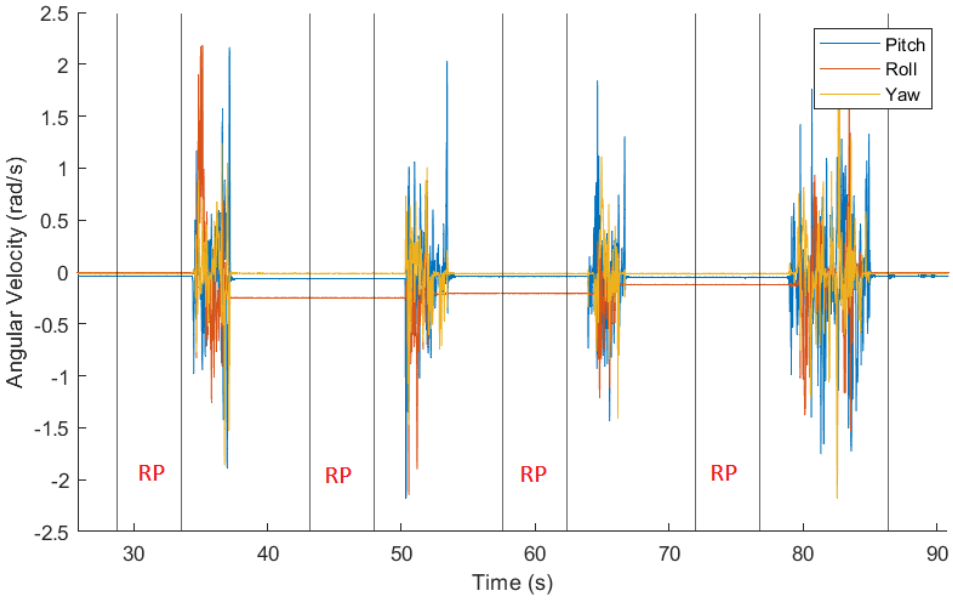

Figure 4.6: Example of the gyroscope readings during a test of the sort described in section 3.4 and Fig. 3.13. The labels "RP" indicate the resting period.

The gyroscope (a subsystem of the LSM6DSOX IMU) provided us with three-dimensional radial velocity measurements. We can see from the time-domain plot in Fig. 4.6 that the gyroscope readings are relatively still during the periods of rest. However, they often have non-zero quiescent radial velocity readings during these supposedly resting periods. These

non-zero readings are a form of drift but they are not necessarily constant over time, meaning that the gyroscope drift is a non-stationary process. A constant offset of this sort is non-stationary when its value changes over time, making it tremendously difficult to correct.

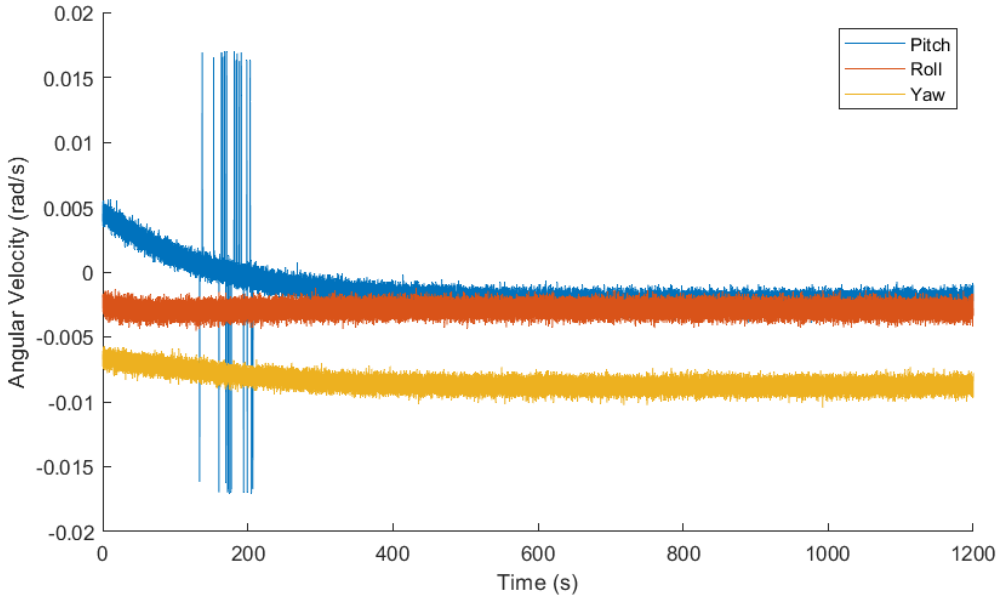

Figure 4.7: Example of the gyroscope readings during a 20-min stationary rest condition test. Note the non-zero mean values (bias) and the moving trendlines of the measurements.

We extracted the parameters of the gyroscope noise/error by recording their measurements under a stationary rest condition, see Fig. 4.7, which is effectively the zero-input response of the gyroscope system. We note that the readings do not have zero-mean values, a further indication of their drift. Furthermore, the pitch readings in particular seem to oscillate around an exponentially decaying trendline settling around the 400-second mark, a further indication that the drift is not a stationary random variable.

There is a high amplitude fluctuation around the 200-second mark, we suspect this was caused by a stray vibration disrupting the measurements. Therefore, we computed the statistical characteristics of the gyroscope from the steady-state portion of the Fig. 4.7 signals. The steady-state portion was considered as the 400-second mark onward since the exponential transient is complete by the 400-second mark and there are no stray vibrations thereafter. By performing our statistical analysis on this region we are avoiding the skewing that an exponential decay would introduce to our histogram. Furthermore, the stray vibration would lead to outliers being included in the statistical analysis.

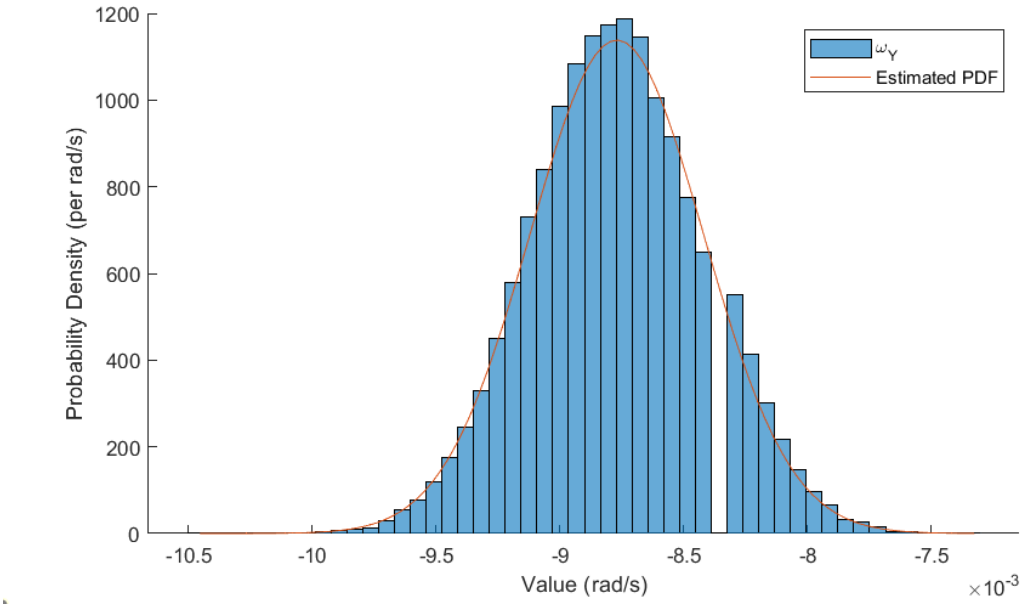

Figure 4.8: Example of the normally distributed PDF estimate of the gyroscope measurement noise. We note the Gaussian curve fit.

The statistical properties of the gyroscope readings are summarized by the estimated PDF

plot in Fig. 4.8. The gyroscope measurement noise is defined very well by a Gaussian/Normal distribution. The root-mean-squared average radial velocity of all channels, having a non-zero value due to drift, is approximately 5.45 mrad/s. Then, the average standard deviation of the gyroscope noise is approximately 0.397 mrad/s. These can be combined to form the mean-squared error according to the formula $MSE = \mu^2 + \sigma^2 = 29.9\ \mu\text{rad}^2/\text{s}^2$. Therefore, the RMS error is 5.47 mrad/s, significantly larger than the noise value of 1.31 mrad/s in the datasheet of the LSM6DSOX but within the same order of magnitude. We suspect the reason for this discrepancy is the imperfect experimental condition, namely the presence of mechanical and acoustic vibrations in the environment would logically increase the variance of the measurements. Whereas the datasheet figure was likely obtained in conditions where most vibrations were eliminated, such as on a gyroscopically stable plate.

4.3 Accelerometer Data

The accelerometer (a subsystem of the LSM6DSOX) provided us with 3-dimensional acceleration measurements. We can see from the time-domain plot in Fig. 4.9 that the accelerometer readings are relatively still during the periods of rest. However, they occasionally present a non-zero value relative to the global mean because of a differing orientation changing the extent to which each axis experiences gravity.

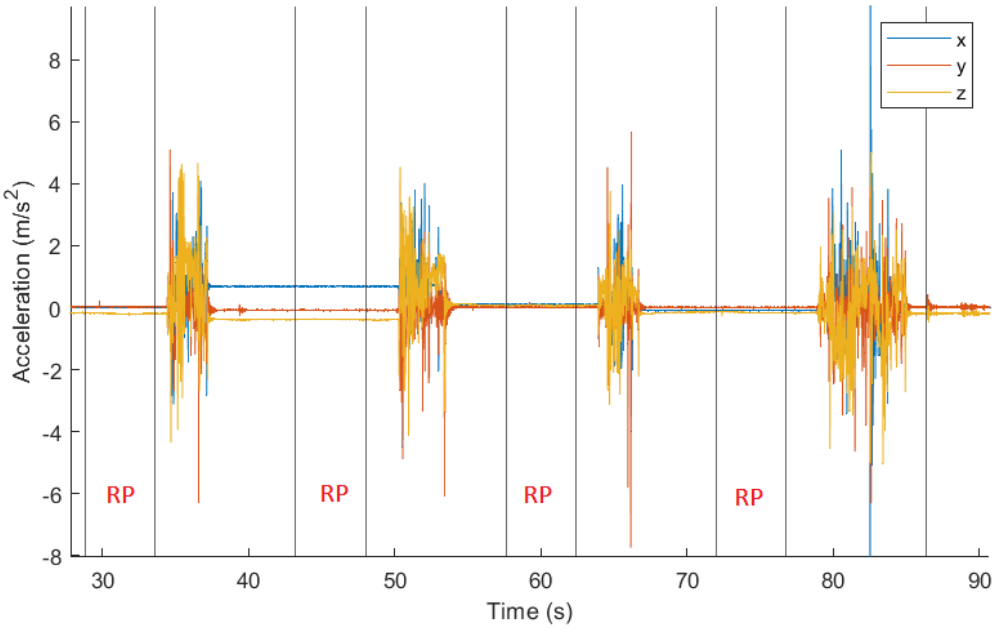

Figure 4.9: Example of the accelerometer readings during a test of the sort described in section 3.4 and Fig. 3.13. The labels "RP" indicate the resting period.

As with the gyroscope, we extracted the parameters of the accelerometer noise by recording their measurements while at rest, see Fig. 4.10. We removed the mean value of the signals, which means that any constant drift is not visible. However, we can see that the readings oscillate around transient mean values, demonstrating that the drift is a non-stationary random variable. There is a high amplitude fluctuation around the 200-second mark, corroborating the gyroscope's measurements of a stray vibration. As with the gyroscope, we computed the statistical characteristics of the accelerometer from the steady-state portion of the Fig. 4.10 (after the 400-second mark) signals to avoid skewing our PDFs.

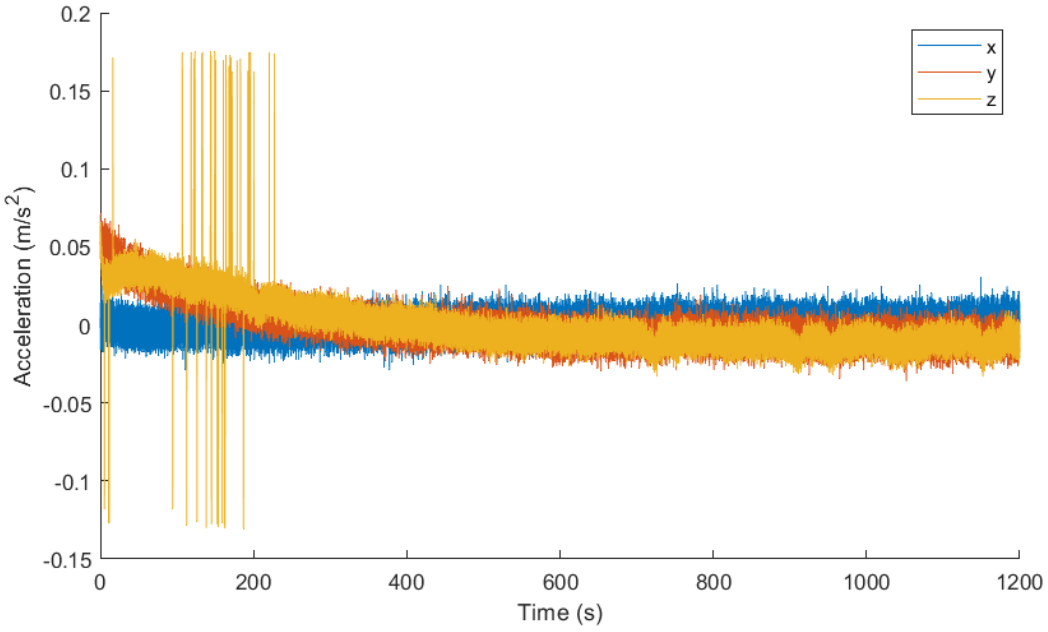

Figure 4.10: Example of the accelerometer readings minus their means during a 20 min stationary rest condition test. Note the decaying trendline of the y and z measurements.

The statistical properties of the accelerometer readings are summarized by the estimated PDF plot in Fig. 4.11. The accelerometer measurement noise is defined very well by a t-distribution. The average standard deviation of the accelerometer noise is approximately 6.50 mm/s^2. This value is slightly larger than the noise value of 4.95 mm/s^2 obtained from the datasheet of the LSM6DSOX (computed for the 104 Hz sampling frequency, 52 Hz bandwidth, ± 2 g, i.e. ± 19.62 m/s^2 dynamic range settings we used). As with the gyroscope, our experimental estimate of the noise is likely overestimating the figure because there are vibrations in the environment. The datasheet's noise figure should represent the noise of measurements in the absence of external vibrations, which we could not guarantee.

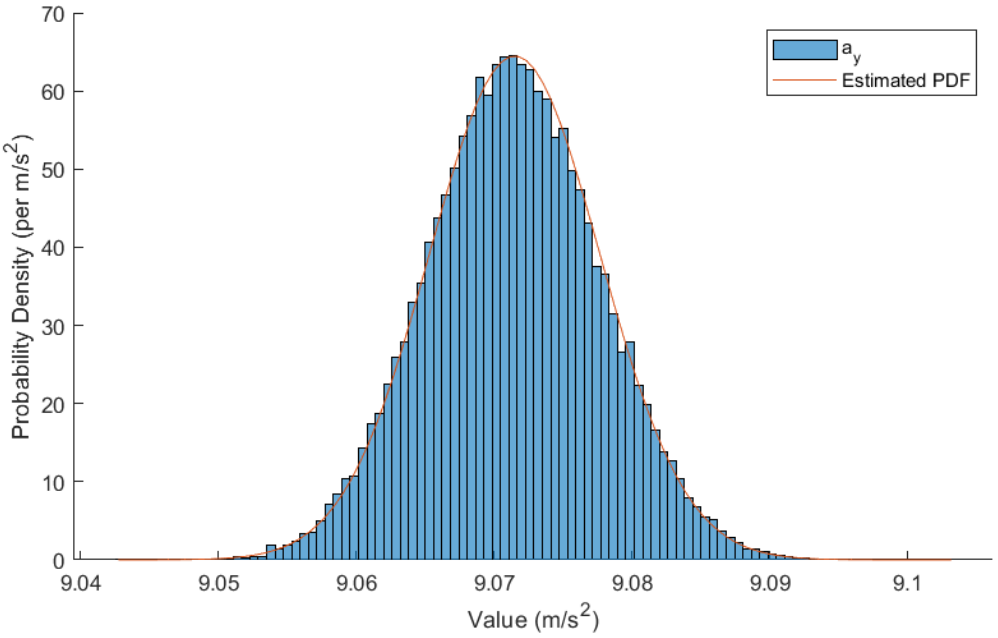

Figure 4.11: Estimate of the t-distributed PDF of the accelerometer measurement noise.

4.4 Signal Processing Pipelines

In this section, we will report and discuss the performance of our signal-processing pipelines where applicable. The first 2 subsections will focus on the orientation prediction task, while the last 2 subsections will be on the position prediction task. As discussed in section 4.1, the Hall-effect array could not be used because of problems maintaining the power supply from the battery pack. Therefore, our results in this section will be hampered by the lack of reliable orientation data. We will demonstrate how the gyroscope and accelerometer data's 6 degrees of freedom can theoretically be used for pose estimating

(which has 6 degrees of freedom) but they lack enough information to reliably resolve the pose of the device. This is an example of the importance of redundancy in data: with only the minimum necessary data, one cannot accurately predict the states of a system unless the sensors are near-perfect.

4.4.1 *A Priori* Orientation Sensor Fusion

Unfortunately, due to the lack of reliable Hall sensor data, we cannot properly evaluate the performance of the *a priori* sensor fusion algorithm. We could only test the performance on a set of synthetic Hall-array orientation estimates which would reflect the ground truth of orientation plus an additive noise component. In this setup, the noise of the Hall estimates of orientation θ_H follows from the variance determined through the statistical analysis in section 4.1 combined with equation 3.41:

$$Var(\theta_H) = MSE(\theta_H) \approx \left(\frac{1 \text{ mV}}{7.8 \text{ V}}\right)^2 = 16.4 \text{ nrad}^2 \tag{4.1}$$

Then we can use the gyroscope readings' MSE from section 4.2 to determine the θ_G MSE:

$$MSE(\theta_G) = T_s^2 MSE(\omega_G) = (9.6 \text{ ms})^2 (29.9 \ \mu\text{rad}^2/\text{s}^2) = 2.76 \text{ nrad}^2 \tag{4.2}$$

This provides us with the *a priori* optimal sensor fusion weight estimate:

$$\tilde{\gamma} = \frac{MSE(\theta_G)}{MSE(\theta_G) + MSE(\theta_H)} = \frac{2.76}{2.76 + 16.4} = 0.144 \tag{4.3}$$

Since the MSE of the gyroscope's prediction is smaller than the Hall sensors', the sensor fusion favours the gyroscope's estimates. However, the performance of this algorithm cannot be truly evaluated without Hall sensor data. Therefore, we will forgo the evaluation of this algorithm, as it would be a purely contrived example.

4.4.2 Orientation Kalman Filter

This section will demonstrate the performance of the Kalman Filter for determining orientations in a trial of the sort laid out in section 3.4 (moving the tracker through a set of known waypoints). Despite the lack of reliable Hall sensor data, we can apply the Kalman filter algorithm to the gyroscope's data. Instead of providing true observations of the system states, we will use the *statistical observation* scheme analogous to the one outlined in section 3.3.4.2. In this way, we will drive the system to the no-movement orientation defined by the initial orientation that is determined in the calibration phase (the first $\tilde{1}5$ s when the device is stationary and the accelerometer can estimate the orientation through the measured gravity). The initial orientation θ_0 is the observation passed to the Kalman filter at every step of the algorithm. The result is an estimate of the orientation change shown in Fig. 4.12.

We estimated the value of the $\mathbf{Q}$ matrix (system disturbance covariance) from the statistical analysis in section 4.2 just like in equation 4.2. This includes the mean value in addition to the variance because the mean also degrades the estimates. In theory, the stochastic signals of the Kalman filter model are supposed to be zero-mean, so we will expect some bias in our estimates. However, the mean value of gyroscope readings (the drift) is non-stationary, therefore even the mean value can be viewed as a variance present at a much lower frequency. As such, we estimate that $\mathbf{Q} = 2.76 \text{ nrad}^2 \ \mathbb{I}_3$.

The optimal observation noise matrix $\mathbf{R}$ was determined through trial and error because there is no appropriate metric to use. We are not actually observing the state but driving the algorithm toward the no-movement state. This means that we cannot really ascribe any empirical value to the variances. Therefore, all we can do is posit a reasonable variance and refine it through trial and error. It turns out that a good estimate of the observation variance is $\mathbf{R} = (\frac{3\pi}{100})^2 \text{ rad}^2 \ \mathbb{I}_3$. In other words, the observation noise that yielded reasonable orientation waveforms was of a 0.36-degree ($\frac{3\pi}{100}$ rad) standard deviation.

The resulting orientation waveforms are shown in Fig. 4.12. We note that the Kalman filter succeeded in estimating near-constant orientation during the resting intervals. Removing the drift in gyroscopic measurements and correctly identifying the resting intervals is a satisfying result considering that we did not provide any state observations. To our satisfaction, the Kalman filter still differentiated the different roll angles that we introduced with our testing mould.

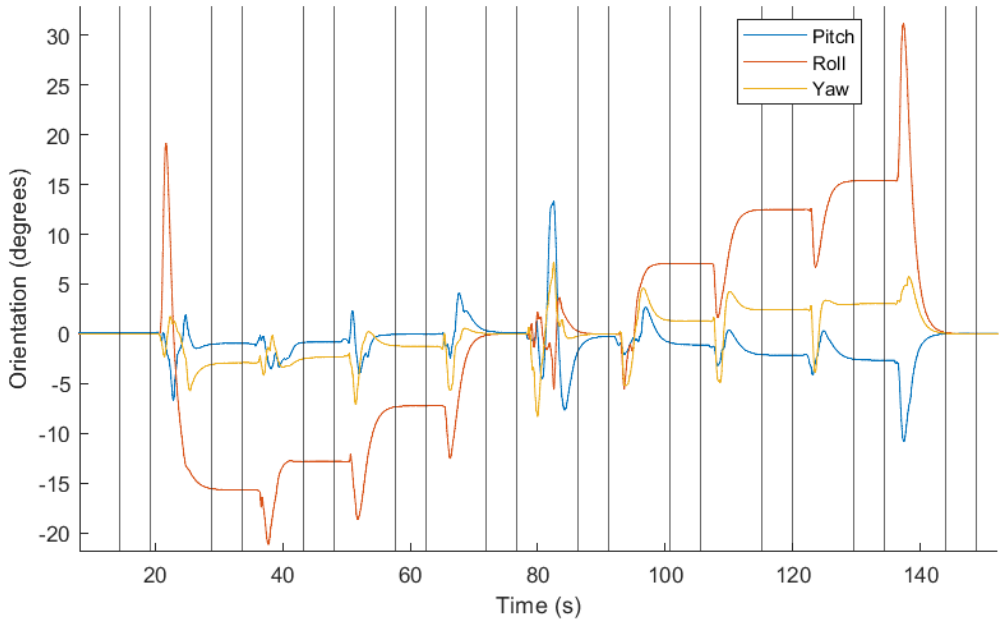

Figure 4.12: Example of the Kalman Orientation Estimation Filter during a test run of the sort described in section 3.4: the tracker is moved through waypoints of varying position and orientation (mainly roll). Note that the short vertical line intervals are the rest periods.

Unfortunately, the angles predicted by our Kalman filter are not accurate. As shown in Fig. 4.13, the estimates fall short of the actual roll angle systematically. We note that the θ_0 angle is 0 degrees but the estimate settles at roughly 0.5 degrees, demonstrating a slight bias. We also note that the degree of underestimation is more severe for the larger angles. Specifically, the estimate on the first step is roughly 57% of the actual angle, roughly 49% on the second step, and finally, 40% on the third step. This shows a nonlinear response to angle changes which could potentially be corrected by applying the inverse nonlinear operation to the estimates. However, the filter would obviously perform best if it were

given real observations from a functional Hall array for example. The reason for the systematic underestimation seems to stem from the *pseudo observation* we are performing. As we consistently reinforced the no-movement observation to the Kalman filter algorithm, it reduced the state estimate to find the statistical mid-point between the *pseudo observation* we provided and the gyroscope's correct estimate of movement. Since the Kalman filter fuses the gyroscope-generated prediction with the zero-movement *pseudo observation*, the result is a state estimate that is increasingly attenuated at larger orientation changes, as the "observation" becomes more and more of an underestimate.

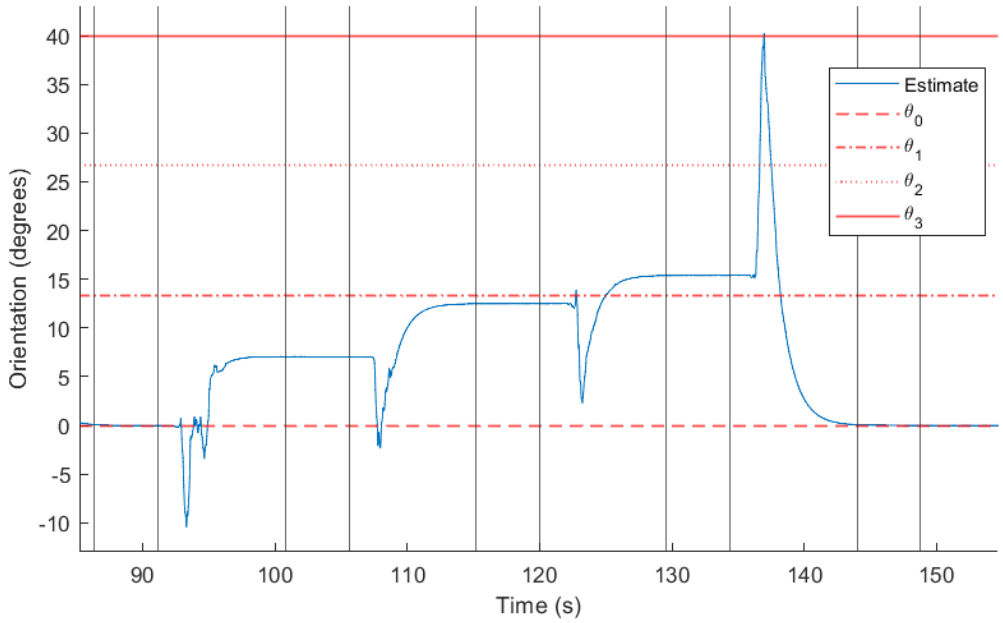

Figure 4.13: Kalman filter's roll estimate error during a test run of the sort described in section 3.4. The red lines are the actual orientations that the staircase-shaped blue Roll estimate curve should approach during the rest periods. Note that the estimate falls short of the real value by a larger and larger proportion as the angle increases from θ_0 to θ_3.

Overall, we see that the Kalman filter is a highly effective strategy for orientation estimates. Unfortunately, we were not able to provide the appropriate input data, in the form of orientation observations, for the Kalman filter to realize its full potential.

4.4.3 Gravity-Residual Adaptive Position Filter

The main reliable way we can test this approach without the Hall array data is to see how it performs during a stationary period of time. The average accelerometer reading provides the ground truth for the orientation in this experiment, given that the device is stationary so all other accelerations are negligible. Under this rest condition, we can evaluate the gravity-residual algorithm's performance using the final position estimate. The final position estimate is a reflection of the drift according to $\vec{p}(t) = \int_0^t \int_0^{t'} \vec{a}_{drift} dt'' dt' = \frac{1}{2} t^2 \vec{a}_{drift}$, therefore:

$$\vec{a}_{drift} = \frac{2}{t^2}\vec{p}(t) = \frac{2\vec{p}_f}{N^2 T_s^2} \tag{4.4}$$

where t is the time, $\vec{p}_f$ is the final position, N is the number of samples in the record length, and T_s is the sampling period. This assumes that the drift is constant for the duration of the recording; alternatively, this is the mean value of the drift over the record length.

If the gravity-residual filter works properly, it should eliminate drift and result in a final position very close to the starting position. Unfortunately, as we can see in Fig. 4.14, the final position after 2 minutes (120 seconds) is on the order of 20 meters in the x and y dimension.

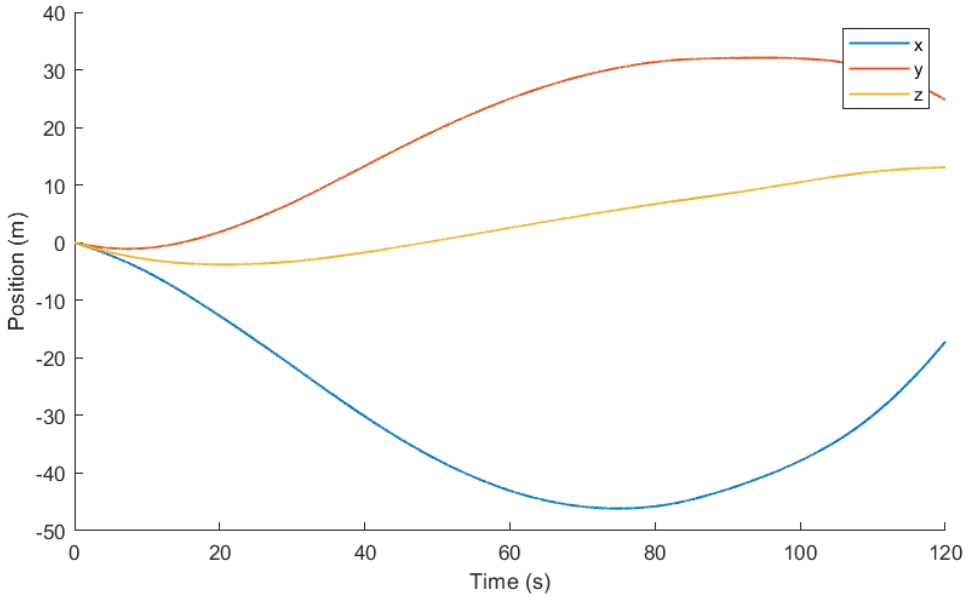

Figure 4.14: Example plot of the gravity-residual adaptive filter run for a 2-minute stationary test. The filter used was 64 samples long with a learning rate α of $2 \cdot 10^{-4}$. Note the extremely large x-position estimate, an indication that the filter was not effective at removing drift.

We also note that the x-position estimate swings more severely to nearly 50 meters from the starting position during the record length. However, we can see that the algorithm works to some extent because the position is not a monotonically-increasing quadratic with respect to time, as it would be if no drift correction algorithm were applied. The fact that the x and y position moves away from the starting point and then comes back towards it could be a manifestation of the non-stationary drift, i.e. the drift changes with time.

We elected to use a learning rate α of $2 \cdot 10^{-4}$ with the 64-sample adaptive filter because it was the maximum learning rate that did not lead to instability. Instability was detected

whenever the position estimates grew unbounded, leading to position estimates near the maximum double floating-point value of 10^{308} or became "NaN" ("Not a Number"). We note that the maximum learning rate is inversely proportional to the number of samples. Therefore, longer adaptive filters require lower learning rates to avoid instability.

Overall, the performance of the gravity-residual adaptive filter is unsatisfactory since a stationary test should be where it performs best. Yet, our experiments show that the gravity-residual adaptive filter is unable to eliminate drift even in this ideal scenario. An alternative approach could be to formulate this LSE filter as an offline/non-causal filter, effectively determining the filter weights through a linear regression (or preferably a pseudoinverse) using the whole record length. The performance might be improved if the filter weights can take into account all of the observations in this way. However, such an offline/non-causal (non-real-time) algorithm would limit the tracker to strictly retrospective motion correction.

4.4.4 Position Kalman Filter

Much like the Orientation Kalman Filter in section 4.4.2, we can devise a Kalman filter to estimate the position as a function of time. Again, instead of providing true observations of the system states, we will use the *statistical observation* scheme outlined in section 3.3.4.2. This will drive the system to the no-movement state defined by a vector of zeros.

We estimated the value of the $\mathbf{Q}$ matrix (system disturbance covariance) from the statistical analysis in section 4.3. This does not include the mean value because the mean

error of the accelerometer (the drift) is mixed in with the acceleration due to gravity and thus can't be reliably estimated. Hence, we estimate that the variance of the system disturbances is dominated by the variance of accelerometer noise $Var(a) = (6.50 \text{ mm/s}^2)^2$. We can then compute the velocity and position variances as $Var(v) = T_s^2 Var(a)$ and $Var(p) = (0.5T_s^2)^2 Var(a)$. Therefore, the disturbances' covariance matrix is approximately:

$$\mathbf{Q} = (6.50 \text{ mm/s}^2)^2 \begin{bmatrix} \frac{1}{4}T_s^4 \mathbb{I}_3 & \mathbb{O}_3 \\ \mathbb{O}_3 & T_s^2 \mathbb{I}_3 \end{bmatrix} \tag{4.5}$$

The optimal observation noise matrix $\mathbf{R}$ was determined through trial and error, again, because there is no appropriate metric to use. We are driving the algorithm toward the no-movement state. This means that we cannot really ascribe any empirical value to the variances. However, there is a structure to be followed in the selecting of values for $\mathbf{R}$. Similar to the disturbance covariance matrix, the observation matrix should be of the form:

$$\mathbf{R} = \sigma_o^2 \begin{bmatrix} T_s^2 \mathbb{I}_3 & \mathbb{O}_3 \\ \mathbb{O}_3 & \mathbb{I}_3 \end{bmatrix} \tag{4.6}$$

where σ_o represents the standard deviation of the observation noise. The matrix structure involving the appropriate time-scale factor T_s^2 ensures that the variance of the position is scaled appropriately relative to the velocity. To get a realistic waveform, we found an observation standard deviation of $\sigma_o = 5$ cm/s was appropriate.

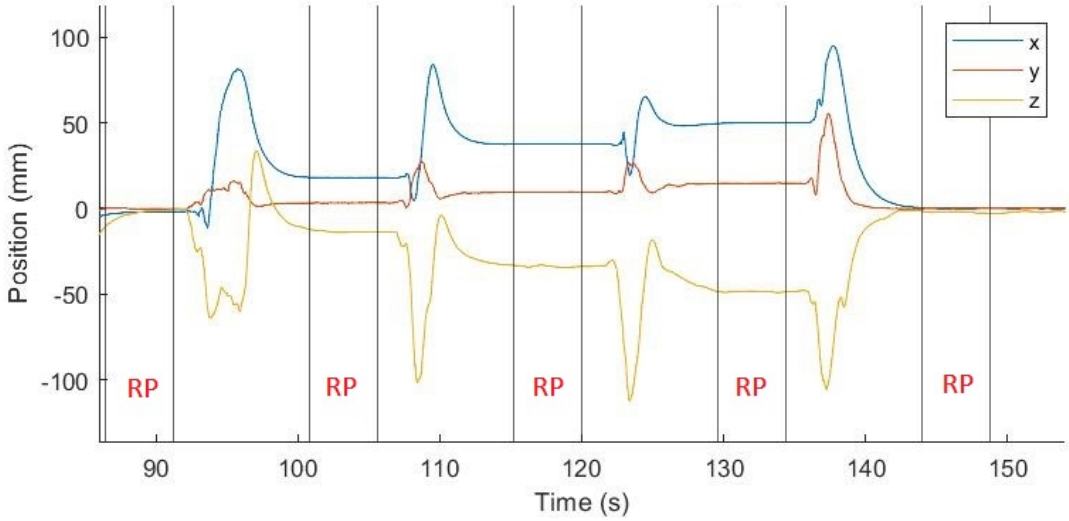

Figure 4.15: Example of the Kalman Position Estimation Filter during a test run of the sort described in section 3.4: the tracker is moved through waypoints of varying position and orientation. "RP" indicates the resting periods.

The resulting position waveforms are shown in Fig. 4.15. We note that the Kalman filter succeeded in estimating near-constant positions during the resting intervals. Removing the drift from acceleration measurements and correctly identifying the resting intervals is a satisfying result considering that we did not provide any state observations, just like in the orientation estimation section. Unfortunately, it seems that the Kalman filter did not differentiate between the positions correctly. In fact, the Kalman filter inverted the scale in a non-trivial way. The positions of a large change in the z-direction were shown to be close, while the positions that were close were shown to be far, see Fig. 4.16.

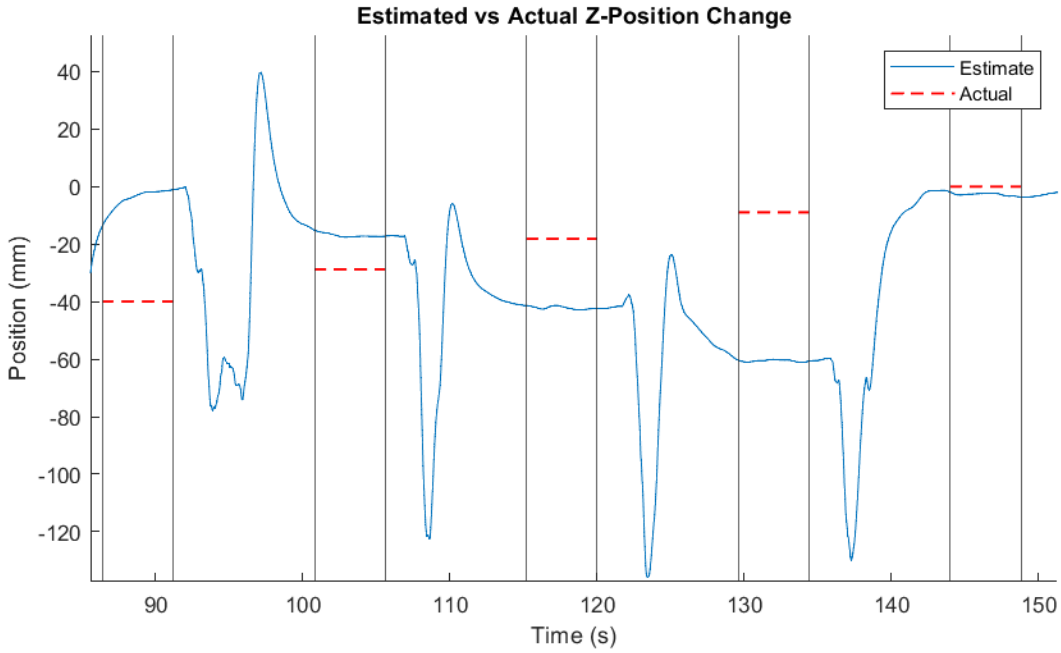

Figure 4.16: Kalman filter's z-position estimate relative to the starting position during a test run of the sort described in section 3.4. The red lines are the actual positions that the staircase-shaped estimate curve should approach during the rest periods (RP). Note that the actual z-positions in chronological order are: -40.0, -28.7, -18.2, -8.9, and 0.0 millimetres.

As shown in Fig. 4.16, the estimates do not correspond to the actual positions. The staircase-shaped curve should be ascending (from left to right) but it is descending. It is unclear why the shape of the curve would be inverted in such a way, it may be partially caused by errors in the orientation estimates as shown in figure 4.13. We suspect that there may be an error with respect to the reference frame, which might explain such an inversion of distances. However, no errors were found in the MatLab script after a thorough examination.

Overall, we see that the Kalman filter is an effective strategy for drift correction. We

were not able to demonstrate the Kalman filter as a functional approach to position tracking. These results may stem, in part, from the *pseudo-observation* scheme we used; however, the errors of the orientation Kalman filter were not so severe under the same conditions. The poor performance may be the result of a sub-optimal application of the Kalman filter to the position-tracking problem and error propagation from the orientation estimates.

4.5 Device-Induced Susceptibility Artifacts

It is important to discuss the impact that placing an electronic device in the MRI scanner can have on image quality. Particularly, placing a ferromagnetic material changes the static B_0 field strength across inside the sensitive volume of the scanner, manifesting as a susceptibility artifact. The local strength of the B_0 field is changed because the presence of a ferromagnetic material warps the direction of the field by presenting it with a low-reluctance path. According to equation 2.1, the fluctuations in B-field caused by magnetic materials will change the resonance frequency of the protons in proximity to the device. This change in resonance frequency leads to a disruption of spatial encoding and even a loss of signal in that region.

In our case, it was difficult to avoid nickel-coating on the contacts of some surface mount device (SMD) electrical components, most of our components were nickel free but some had trace amounts. Furthermore, a couple of inductors on the RP2040 MCU, used for power supply regulation, contained ferrite cores. The ferrite core inductors were desoldered and

replaced with non-ferromagnetic inductors. Finally and most importantly, the micro USB charging port of the device (a part of the RP2040) contains a nickel charge plate. We tried to find a replacement for this part but it seems that nickel charge plates are ubiquitous for USB connectors. All of these factors combine to create susceptibility artifacts of the sort shown in Fig. 4.17.

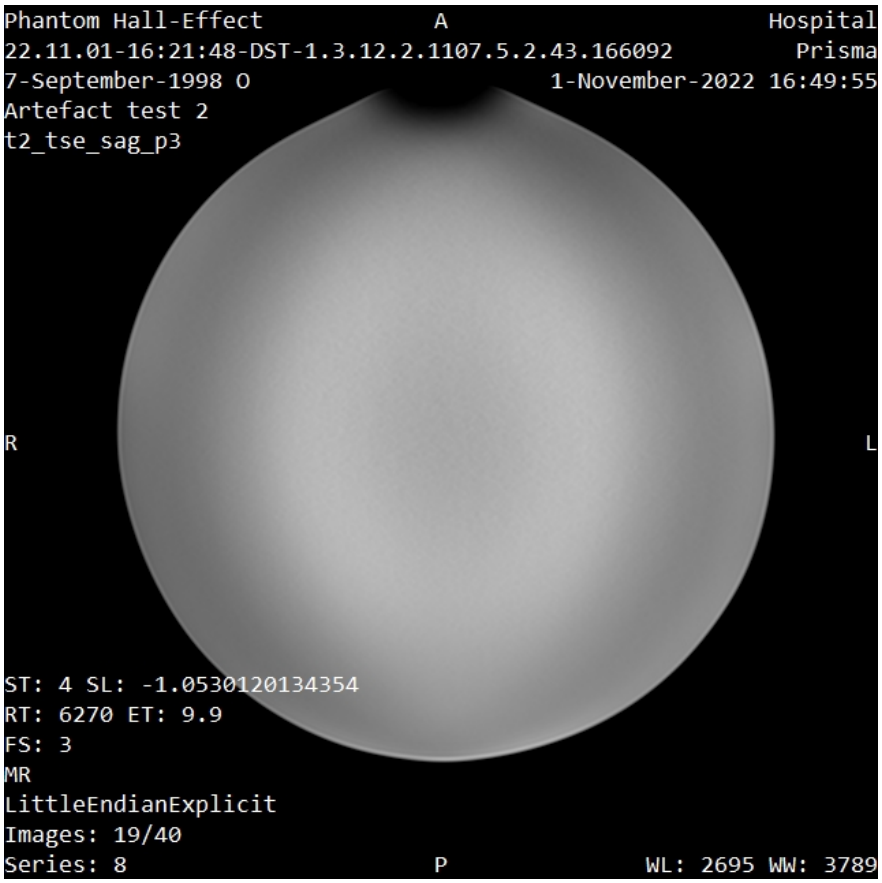

Figure 4.17: T2-weighted turbo spin-echo sequence image of a spherical phantom (composed of $NiSO_4 \times 6H_2O$ dissolved in distilled water to a 1.25:1000 weight-ratio) when the tracker is placed on its "forehead". Note the warping of the phantom near the top of the image; the phantom no longer appears smooth and the MR signal is lost near the device.

As we can see from Fig. 4.17, a future iteration of this device would require the replacement of the USB charging port with a non-ferromagnetic alternative. We note that other components may contribute to the susceptibility artifact but attraction tests with a permanent magnet demonstrated that the USB connector was by far the largest source of ferromagnetism. If the connector were replaced, we need to consider how to charge the device and how to program the device. Charging can be easily replicated, but programming through the serial connection may prove more challenging.

Chapter 5

Conclusions and Future Work

The work presented in this thesis is motivated by the importance of motion in Magnetic Resonance Imaging as a societal and economic problem. We estimated a 57 million CAD (42 million USD) annual cost associated with the elimination of 465,000 scans because of motion artifacts in MRI, and the annual number of scans is increasing. We highlighted that most solutions are too expensive or cumbersome to incite wide adoption and indicated that a low-cost active-tracker method has the most potential to address this challenge.

The background introduced the artifacts in Magnetic Resonance Imaging (MRI), the mathematical basis of motion artifacts in MRI and the various manifestations of motion in an image. We explained how motion (rotation and translation) in, non-selective, 3D MRI changes the k-space trajectory by means of rotating and modulating k-space with respect to the scanner's frame of reference. We discussed the loss of selective excitation when motion

is through-plane in 2D MRI. We explained how k-space over-sampling software techniques provide motion mitigation, their primary disadvantage being increased scan times. Finally, we discussed the tracking solutions present in the literature from passive/optical trackers to active electronic marker systems.

In response to the challenge, this thesis outlined methods in the design of a low-cost active MRI motion-tracking system composed of an Inertial Measurement Unit (IMU) and a set of Hall-effect magnetometers. We defined the system-level picture of our problem: capture motion and wirelessly relay its parameters in real time to a computer in the control room. We used a microcontroller with built-in Wi-Fi and Inertial Measurement Units, plus a Printed Circuit Board (PCB) for connecting the batteries and Hall-effect sensors. We explained the required precautions entrained by the MR environment, including the forces on ferromagnetic materials and the RF heating risks with conductors. We presented the signal processing used to determine the position and orientation of the device from first principles. To this end, we explained our use of *a priori* sensor fusion, Kalman filtering, and adaptive-residual filters. We introduced the notion of *statistical observation* which allowed us to use a Kalman filter in the absence of a state observation measurement. Finally, we presented our testing methods, including a 3D-printed mould with slots fitted to our tracker's dimensions.

The results of the presented investigations can be summarized as follows. We found that powering the Hall effect sensors from an unregulated supply is not acceptable, leading

to an absence of reliable magnetic field data. This oversight proved to be a critical mistake as it hindered our tracker from performing as intended. Initially, we felt that using a boost converter to power our sensors was an unacceptable option since it would likely introduce stray magnetic fields to the MRI by way of an inductive element, not to mention that most inductors contain at least trace amounts of ferromagnetic materials. In conclusion, using a boost converter is the most reasonable way to ensure the proper functioning of our Hall sensors. We analyzed the statistical characteristics of all our sensors by recording their zero-input responses (i.e. when the device was not moving). We found the related Mean Squared Error (MSE) of our sensor's state estimates and used them to define our signal processing parameters. The gravity-residual adaptive filter was not effective in eliminating drift from our pose estimates. However, we demonstrated that a Kalman filter with *statistical observations* (as defined in section 3.3.4.2) was effective in eliminating drift from our pose estimates. Unfortunately, the estimates produced by a Kalman filter with *statistical observations* followed a non-linear relationship with respect to the actual states. In short, the Kalman filter with *statistical observations* eliminated drift completely, and provided an orientation estimate that appears to be useful in the recovery of the true orientation but provided position estimates that could not be resolved to the ground truth.

The first step in the future work that could follow that of this thesis begins with an implementation of a regulated power supply (boost converter from the battery's range to 5 V) for the Hall sensors. This would enable us to use our system as intended. However,

given the promising results from the *statistical observation* Kalman filter, we could envision a system that does not require any magnetic field sensors and only uses an IMU to track pose. This direction would facilitate the miniaturization of the device since it would only require a Microcontroller Unit, an Inertial Measurement Unit, a Wi-Fi module, and a battery. However, such a direction requires the resolution of the non-linear relationship between orientation estimates and ground truths shown in section 4.4.2. Furthermore, future work includes determining the complex relationship between position estimates and ground truths shown in section 4.4.4. We suspect that the inclusion of functioning Hall sensors would improve the performance of the system since they would provide much-needed corroboration of orientation estimates, improving the orientation estimate accuracy. With better orientation estimates, we also expect better position estimates since the latter depends on the former to remove acceleration due to gravity from the measurements. No matter the direction of future work, it will be necessary to remove all ferromagnetic materials from the tracker to eliminate the susceptibility artifacts, specifically replacing the USB connector of any MCU with a non-ferromagnetic alternative. This will involve a custom charging and programming port because virtually all USB connectors contain a nickel charge plate. Finally, we suggest that the ideal solution would not only contain the Hall sensors to observe the orientation but would include a fourth sensor capable of observing the position. In this way, the Kalman filters for orientation and position could be used as intended by their prediction-observation workflow.

Bibliography

[1] F. Godenschweger, U. Kägebein, D. Stucht, U. Yarach, A. Sciarra, R. Yakupov, F. Lüsebrink, P. Schulze, and O. Speck, "Motion correction in MRI of the brain," *Physics in Medicine and Biology*, vol. 61, pp. R32–R56, feb 2016.

[2] N. Todd, O. Josephs, M. F. Callaghan, A. Lutti, and N. Weiskopf, "Prospective motion correction of 3d echo-planar imaging data for functional MRI using optical tracking," *NeuroImage*, vol. 113, pp. 1–12, 2015.

[3] I. Havsteen, A. Ohlhues, K. H. Madsen, J. D. Nybing, H. Christensen, and A. Christensen, "Are movement artifacts in magnetic resonance imaging a real problem?—a narrative review," *Frontiers in Neurology*, vol. 8, 2017.

[4] J. Wang, L. He, H. Zheng, and Z.-L. Lu, "Optimizing the magnetization-prepared rapid gradient-echo (mp-rage) sequence," *PloS one*, vol. 9, p. e96899, 05 2014.

[5] A. van Niekerk, E. Meintjes, and A. van der Kouwe, "A wireless radio frequency triggered acquisition device (wrad) for self-synchronised measurements of the rate of change of the

MRI gradient vector field for motion tracking," *IEEE Transactions on Medical Imaging*, vol. 38, no. 7, pp. 1610–1621, 2019.

[6] A. van Niekerk, J. Berglund, T. Sprenger, O. Norbeck, E. Avventi, H. Rydén, and S. Skare, "Control of a wireless sensor using the pulse sequence for prospective motion correction in brain MRI," *Magnetic Resonance in Medicine*, vol. 87, no. 2, pp. 1046–1061, 2022.

[7] D. Stucht, K. Danishad, P. Schulze, F. Godenschweger, M. Zaitsev, and O. Speck, "Highest resolution in vivo human brain MRI using prospective motion correction," *PloS one*, vol. 10, p. e0133921, 08 2015.

[8] M. Zaitsev, C. Dold, G. Sakas, J. Hennig, and O. Speck, "Magnetic resonance imaging of freely moving objects: prospective real-time motion correction using an external optical motion tracking system," *NeuroImage*, vol. 31, no. 3, pp. 1038–1050, 2006.

[9] M. Zaitsev, B. Akin, P. LeVan, and B. R. Knowles, "Prospective motion correction in functional MRI," *NeuroImage*, vol. 154, pp. 33–42, 2017. Cleaning up the fMRI time series: Mitigating noise with advanced acquisition and correction strategies.

[10] D. C. Hoinkiss, P. Erhard, N.-J. Breutigam, F. von Samson-Himmelstjerna, M. Günther, and D. A. Porter, "Prospective motion correction in functional MRI using simultaneous multislice imaging and multislice-to-volume image registration," *NeuroImage*, vol. 200, pp. 159–173, 2019.

[11] J. B. Andre, B. W. Bresnahan, M. Mossa-Basha, M. N. Hoff, C. P. Smith, Y. Anzai, and W. A. Cohen, "Toward quantifying the prevalence, severity, and cost associated with patient motion during clinical mr examinations," *Journal of the American College of Radiology*, vol. 12, no. 7, pp. 689–695, 2015.

[12] M. Zaitsev, J. Maclaren, and M. Herbst, "Motion artefacts in MRI: a complex problem with many partial solutions," *Journal of magnetic resonance imaging : JMRI*, vol. 42, 01 2015.

[13] M. Louis Lauzon and B. K. Rutt, "Generalized k-space analysis and correction of motion effects in mr imaging," *Magnetic Resonance in Medicine*, vol. 30, no. 4, pp. 438–446, 1993.

[14] A. Gholipour, M. Polak, A. van der Kouwe, E. Nevo, and S. K. Warfield, "Motion-robust MRI through real-time motion tracking and retrospective super-resolution volume reconstruction," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 5722–5725, 2011.

[15] G. Vaillant, C. Prieto, C. Kolbitsch, G. P. Penney, and T. Schaeffter, "Retrospective rigid motion correction in k-space for segmented radial MRI," *IEEE Transactions on Medical Imaging*, vol. 33, pp. 1–10, 2014.

[16] J. A. Derbyshire, G. A. Wright, R. M. Henkelman, and R. S. Hinks, "Dynamic scan-plane tracking using MR position monitoring," *Journal of Magnetic Resonance Imaging*, vol. 8, no. 4, pp. 924–932, 1998.

[17] A. van Niekerk, A. van der Kouwe, and E. Meintjes, "A method for measuring orientation within a magnetic resonance imaging scanner using gravity and the static magnetic field (vectorient)," *IEEE Transactions on Medical Imaging*, vol. 36, no. 5, pp. 1129–1139, 2017.

[18] B. Chen, N. Weber, F. Odille, C. Large-Dessale, A. Delmas, L. Bonnemains, and J. Felblinger, "Design and validation of a novel MR-compatible sensor for respiratory motion modeling and correction," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 1, pp. 123–133, 2017.

[19] T. O. Woods, "Standards for medical devices in MRI: Present and future," *Journal of Magnetic Resonance Imaging*, vol. 26, no. 5, pp. 1186–1189, 2007.

[20] A. S. for Testing and Materials, "ASTM F2052-06e1 standard test method for measurement of magnetically induced displacement force on medical devices in the magnetic resonance environment." https://www.astm.org/f2052-06e01.html.

[21] A. S. for Testing and Materials, "ASTM F2213–06 standard test method for measurement of magnetically induced torque on medical devices in the magnetic resonance environment." https://www.astm.org/f2213-06.html.

[22] A. S. for Testing and Materials, "ASTM F2119-01 standard test method for evaluation of MR image artifacts from passive implants." https://www.astm.org/f2119-01.html.

[23] A. S. for Testing and Materials, "ASTM F2182-02 standard test method for measurement of radio frequency induced heating near passive implants during magnetic resonance imaging." https://www.astm.org/f2182-02.html.

[24] M. A. Bernstein, X. J. Zhou, J. A. Polzin, K. F. King, A. Ganin, N. J. Pelc, and G. H. Glover, "Concomitant gradient terms in phase contrast mr: Analysis and correction," *Magnetic Resonance in Medicine*, vol. 39, no. 2, pp. 300–308, 1998.

[25] I. R. Cole, "Modelling CPV," 6 2015.

[26] M. Usman, S. Latif, M. Asim, B.-D. Lee, and J. Qadir, "Retrospective motion correction in multishot MRI using generative adversarial network," *Scientific Reports*, vol. 10, mar 2020.

[27] N. White, C. Roddey, A. Shankaranarayanan, E. Han, D. Rettmann, J. Santos, J. Kuperman, and A. Dale, "Promo: Real-time prospective motion correction in MRI using image-based tracking," *Magnetic Resonance in Medicine*, vol. 63, no. 1, pp. 91–105, 2010.

[28] M. Tisdall, A. Hess, M. Reuter, E. Meintjes, B. Fischl, and A. Kouwe, "Volumetric navigators for prospective motion correction and selective reacquisition in

neuroanatomical mri," *Magnetic resonance in medicine : official journal of the Society of Magnetic Resonance in Medicine / Society of Magnetic Resonance in Medicine*, vol. 68, pp. 389–99, 12 2011.

[29] A. van der Kouwe, "Chapter 4 - motion artifacts and correction in neuro mri," in *Advanced Neuro MR Techniques and Applications* (I.-Y. Choi and P. Jezzard, eds.), vol. 4 of *Advances in Magnetic Resonance Technology and Applications*, pp. 53–68, Academic Press, 2021.

[30] J. Torres, B. O'Flynn, P. Angove, F. Murphy, and C. O. Mathuna, "Motion tracking algorithms for inertial measurement," ICST, 6 2007.

[31] A. Sabatini, C. Martelloni, S. Scapellato, and F. Cavallo, "Assessment of walking features from foot inertial sensing," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 3, pp. 486–494, 2005.

[32] X. Yun, E. R. Bachmann, H. Moore, and J. Calusdian, "Self-contained position tracking of human movement using small inertial/magnetic sensor modules," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 2526–2533, 2007.

[33] Q. Fan, H. Zhang, Y. Sun, Y. Zhu, X. Zhuang, J. Jia, and P. Zhang, "An optimal enhanced kalman filter for a zupt-aided pedestrian positioning coupling model," *Sensors*, vol. 18, p. 1404, 05 2018.

[34] D. G. Manolakis, V. K. Ingle, and S. M. Kogon, *Statistical and adaptive signal processing : spectral estimation, signal modeling, adaptive filtering, and array processing.* Artech House signal processing library, Boston, Mass.: Artech House, 2005.